ABSTRACT. Efficient and robust iterative solvers for strong anisotropic elliptic equations are very challenging. In this paper a block preconditioning method is introduced to solve the linear algebraic systems of a class of micro-macro asymptotic-preserving (MMAP) scheme. MMAP method was developed by Degond et al. in 2012 where the discrete matrix has a $2 \times 2$ block structure. By the approximate Schur complement a series of block preconditioners are constructed. We first analyze a natural approximate Schur complement that is the coefficient matrix of the original non-AP discretization. However it tends to be singular for very small anisotropic parameters. We then improve it by using more suitable approximation for boundary rows of the exact Schur complement. With these block preconditioners, preconditioned GMRES iterative method is developed to solve the discrete equations. Several numerical tests show that block preconditioning methods can be a robust strategy with respect to grid refinement and the anisotropic strengths.

CONTENTS

1. Introduction 2
2. Review of the Micro-Macro Asymptotic Preserving schemes 3
3. Block preconditioning methods 5
   3.1. Approximate Schur complement 6
   3.2. A further modification 7
4. Numerical results 7
   4.1. Setup 8
   4.2. Aligned case 8
   4.3. Non-aligned case 12
   4.4. Without inflow boundary condition 14
5. Conclusion and perspectives 16
Acknowledgements 17
Appendix A. Finite difference discretizations 17
   A.1. Aligned case 17
   A.2. Non-aligned case 18
Appendix B. The property of the approximate Schur complement $S_4$ 20

2010 Mathematics Subject Classification. Primary: 68Q25; 68R10; Secondary: 68U05.
Key words and phrases. Block preconditioning methods; Asymptotic preserving scheme; Anisotropic elliptic problems; Schur complement; Finite difference method.
1. INTRODUCTION

In this paper, the following anisotropic elliptic equation is considered:

\[
\begin{cases}
-\Delta_1 \phi^\varepsilon - \frac{1}{\varepsilon} \Delta |\phi^\varepsilon| = f^\varepsilon, & \text{in } \Omega, \\
\phi^\varepsilon = 0, & \text{on } \Gamma_D, \\
\nabla_1 \phi^\varepsilon \cdot \mathbf{n} + \frac{1}{\varepsilon} \nabla_1 |\phi^\varepsilon| \cdot \mathbf{n} = 0, & \text{on } \Gamma_N,
\end{cases}
\]

where \( \Omega \) is an open set in \( \mathbb{R}^2 \) or \( \mathbb{R}^3 \), its boundary is defined by \( \partial \Omega = \bar{\Omega} \setminus \Omega \), \( \mathbf{n} \) is exterior normal vector on the boundary \( \partial \Omega \). The parameter \( \varepsilon \) is reciprocal of anisotropic strength, and it can be very small, \( \varepsilon > 0 \). Let \( \mathbf{b} \) be a magnetic field, then we can divide the boundary \( \partial \Omega \) into two parts, \( \partial \Omega = \Gamma_D \cup \Gamma_N \), with \( \Gamma_D = \{ x \in \partial \Omega \mid \mathbf{b} \cdot \mathbf{n} = 0 \} \) and \( \Gamma_N = \{ x \in \partial \Omega \mid \mathbf{b} \cdot \mathbf{n} \neq 0 \} \).

Moreover, we define parallel or perpendicular differential operators as follows:

\[
\begin{align*}
\nabla_1 \phi^\varepsilon &= \mathbf{b} \otimes \mathbf{b} \nabla \phi^\varepsilon, \\
\nabla_1 |\phi^\varepsilon| &= (I - \mathbf{b} \otimes \mathbf{b}) \nabla |\phi^\varepsilon|, \\
\Delta_1 \phi^\varepsilon &= \nabla \cdot \nabla_1 \phi^\varepsilon, \\
\Delta_1 |\phi^\varepsilon| &= \nabla \cdot \nabla_1 |\phi^\varepsilon|.
\end{align*}
\]

where \( I \) is the identity matrix. This anisotropic elliptic equation (1) has important applications in the physics of magnetized plasma, such as confinement fusion [13], plasma of ionosphere [1], plasma propulsion [12], etc.

The problem (1) is well-posed for any \( \varepsilon > 0 \). Indeed let \( \Omega \) be a convex polygon and \( f^\varepsilon \in L^2(\Omega) \), then we have a unique solution \( \phi^\varepsilon \in H^2_D(\Omega) \) of the problem (1), with

\[
H^2_D(\Omega) = \{ v \in H^2(\Omega) : v|_{\Gamma_D} = 0 \}.
\]

Formally, if \( \varepsilon \) goes to 0, we get a degenerate problem:

\[
\begin{cases}
-\Delta_1 \phi^0 = 0, & \text{in } \Omega, \\
\phi^0 = 0, & \text{on } \Gamma_D, \\
\nabla_1 \phi^0 \cdot \mathbf{n} = 0, & \text{on } \Gamma_N.
\end{cases}
\]

The problem (2) is ill-posed due to loss of uniqueness of solution, since any function belongs to set

\[
H^2_{D,||}(\Omega) = \{ v \in H^2(\Omega) : v|_{\Gamma_D} = 0, \nabla \parallel v = 0 \text{ in } \Omega \}
\]

is solution of (2). In other words, the solution of (2) is in kernel of differential operator \( \nabla_\parallel \).

We then define a limit problem

\[
\begin{cases}
-\Delta_1 \tilde{\phi} = f^0, & \text{in } \Omega, \\
\tilde{\phi} = 0, & \text{on } \Gamma_D, \\
\nabla_1 \tilde{\phi} \cdot \mathbf{n} = 0, & \text{on } \Gamma_N, \\
\nabla_\parallel \tilde{\phi} = 0, & \text{in } \Omega.
\end{cases}
\]

This limit problem (3) has unique solution \( \tilde{\phi} \in H^2_{D,||} \) thanks to the constraint \( \nabla_\parallel \tilde{\phi} = 0 \) in \( \Omega \).

Solving directly the anisotropic elliptic equation with classic numerical methods may involve severe numerical pollution [21]. One possible way to alleviate such pollution of anisotropic elliptic or parabolic problems is to use high order methods, such as [10] for fourth-order finite volume method.
and [9] for high order finite element method. Moreover, Chacón et al. in [3] proposed an asymptotic-preserving semi-Lagrangian algorithm for time-dependent anisotropic heat conduction equations. In the last decade, another class of asymptotic preserving (AP) schemes were developed for the problem (1). Those AP schemes, on the one hand, are equivalent to the problem (1) when $\varepsilon > 0$; on the other hand, they coincide with the limit problem (3) as $\varepsilon \to 0$. Therefore, the AP schemes are well-defined for all values of $\varepsilon$. The first AP scheme was given for aligned case ($b = (0, 1)^T$) in [4]. Some extensions were made for aligned case, namely Duality Based methods [2, 19]. Other methods focused on general non-aligned case, such as Micro-Macro AP (MMAP) scheme [5], field line integration method [17, 18], two field iterated method [6, 20], AP scheme based on first-order system method [11]. In most of these methods, the key point is to introduce an auxiliary variable, then we have an AP system. To solve the AP system numerically, one must solve a linear algebraic system which has more degree of freedoms than original systems, where direct solvers may not be practical in large scale simulation. Thus the main purpose of this paper is to develop efficient and robust iterative methods.

In this paper we propose to use GMRES method [15] to solve the linear algebraic systems. To improve the robustness and efficiency of the iterative solver, the key ingredient is efficient preconditioning. The coefficient matrix generated from the MMAP scheme used in this paper has a typical $2 \times 2$ block structure. Motivated by the block preconditioning methods from Navier-Stokes equations in [7], where the key point is approximate Schur complement, we attempt to devise efficient approximate Schur complement of the MMAP system in this paper. We firstly give a natural approximation from the continuous operator level. However this natural approximate Schur complement may be ill conditioned if $\varepsilon$ is very small. We then replace its boundary rows by the ones of exact Schur complement. It is surprising that the GMRES method converges only in 2 iterations in our numerical examples. Therefore we find the core point is how to approximate the boundary rows of exact Schur complement. Based on this fact, we present some improvements in Section 3.2. This paper is organized as follows: In Section 2, the Micro-Macro Asymptotic Preserving (MMAP) schemes introduced in [5] is briefly revised. In Section 3, several block preconditioning methods for the MMAP schemes are described. In Section 4, a number of numerical comparison of those block preconditioning methods are given, including the aligned case and non-aligned case. The preconditioned methods for the MMAP scheme without inflow boundary condition is also discussed here. Finally, we finish this paper with some conclusion and perspectives.

2. Review of the Micro-Macro Asymptotic Preserving schemes

In this part, we briefly revise the Micro-Macro Asymptotic Preserving (MMAP) schemes introduced in [5]. For clarity, let us omit the subscript $\varepsilon$. Moreover, let us focus on an anisotropic elliptic equation defined in a unit square, e.g. $\Omega = [0, 1] \times [0, 1]$. Moreover, let us denote $\Omega = (0, 1) \times (0, 1)$, $\Gamma_D = \{0, 1\} \times [0, 1]$ and $\Gamma_N = [0, 1] \times \{0, 1\}$. Thus the anisotropic elliptic equation is given as follows

$$\begin{cases} -\Delta_\perp \phi - \frac{1}{\varepsilon} \Delta_\parallel \phi = f, & \text{in } \Omega, \\ \phi = 0, & \text{on } \Gamma_D, \\ n \cdot \nabla_\perp \phi + \frac{1}{\varepsilon} n \cdot \nabla_\parallel \phi = 0, & \text{on } \Gamma_N. \end{cases}$$

(4)

The key point of the MMAP scheme in [5] is to introduce an auxiliary function $q$, with the following ansatz

$$\nabla_\parallel \phi = \varepsilon \nabla_\parallel q.$$ 

(5)
Taking another time of parallel derivative on (5), we have
\[
\Delta_{\parallel} \phi = \varepsilon \Delta_{\parallel} q, \quad \text{in } \Omega.
\]

Injecting (6) into the first equation of (4), we get
\[
-\Delta_{\bot} \phi - \Delta_{\parallel} q = f, \quad \text{in } \Omega.
\]

Similarly, putting the ansatz (5) into the boundary conditions in (4), we can obtain the corresponding boundary conditions. Finally, the MMAP scheme is written as
\[
\begin{align*}
-\Delta_{\bot} \phi - \Delta_{\parallel} q &= f, \quad \text{in } \Omega, \\
\Delta_{\parallel} \phi - \varepsilon \Delta_{\parallel} q &= 0, \quad \text{in } \Omega, \\
\phi &= 0, \quad \text{on } \Gamma_D, \\
n \cdot \nabla_{\bot} \phi + n \cdot \nabla_{\parallel} q &= 0, \quad \text{on } \Gamma_N, \\
n \cdot \nabla_{\parallel} \phi - \varepsilon n \cdot \nabla_{\parallel} q &= 0, \quad \text{on } \Gamma_N.
\end{align*}
\]

Now, let us give two remarks on the MMAP scheme (7). The first one is that the MMAP scheme has a unique solution $\phi$ for any $\varepsilon \geq 0$ provided that the parallel derivative of $\phi$ is vanishing on the ends of the magnetic field lines. Indeed, provided that $\varepsilon > 0$, the MMAP scheme (7) is equivalent to the anisotropic elliptic equation (4), which is well-posed for $\varepsilon > 0$ (see [5]), thus we have unique solution $\phi$. Formally letting $\varepsilon \to 0$ in (7), we have
\[
\Delta_{\parallel} \phi = 0 \quad \text{in } \Omega.
\]

Integrating (8) along magnetic field line and thanks to $\nabla_{\parallel} \phi$ vanishing at one end of magnetic field line, we find that
\[
\nabla_{\parallel} \phi = 0 \quad \text{in } \tilde{\Omega},
\]

which means that $\phi$ is constant along magnetic field line, thus $\phi \equiv \tilde{\phi}$, where we denote $\tilde{\phi}$ as the average of $\phi$ along a magnetic field line. At last, integrating the first equation of (7) along a magnetic field line, we obtain a unique solution $\tilde{\phi}$, which is also the unique solution for the MMAP scheme (7). Notice that the constraint the parallel derivative of $\phi$ vanishing on the ends of the magnetic field lines is equivalent to say that the magnetic field lines cross the boundary $\Gamma_N$.

The second remark is that the MMAP scheme (7) does not determine a unique $q$. Indeed, if $q$ satisfies (7), then $q + q_0$ also satisfies (7) for any $q_0$ in the kernel of the operator $\nabla_{\parallel}$. To determine a unique $q$, in [5], the inflow boundary condition was added on one end of magnetic field lines (denoted as $\Gamma_{\text{in}} \subset \Gamma_N$ where $b \cdot n < 0$), that is $q = 0$ on $\Gamma_{\text{in}}$. Moreover, notice that along a magnetic field line, one boundary condition of $n \cdot \nabla_{\parallel} \phi - \varepsilon n \cdot \nabla_{\parallel} q = 0$ on $\Gamma_N$ is tedious, since one can easily verify this fact by integrating the second equation of (7) along a magnetic field line. As a result, on one end of magnetic field lines, the boundary condition $n \cdot \nabla_{\parallel} \phi - \varepsilon n \cdot \nabla_{\parallel} q = 0$ can be substituted by the inflow boundary condition $q = 0$. 
3. Block preconditioning methods

In this part, we will introduce block preconditioning methods for discrete MMAP scheme of Section 2. Let us reformulate the MMAP scheme as follows

\[
\begin{align*}
-\Delta q - \Delta_1 \phi &= f, \quad (x, z) \in (0, 1) \times (0, 1), \\
q &= 0, \quad (x, z) \in \{0, 1\} \times [0, 1], \\
q &= 0, \quad (x, z) \in [0, 1] \times \{0\}, \\
\mathbf{n} \cdot \nabla q + \mathbf{n} \cdot \nabla \phi &= 0, \quad (x, z) \in [0, 1] \times \{1\},
\end{align*}
\]

And

\[
\begin{align*}
\varepsilon \Delta q - \Delta_1 \phi &= 0, \quad (x, z) \in (0, 1) \times (0, 1), \\
\phi &= 0, \quad (x, z) \in \{0, 1\} \times [0, 1], \\
\mathbf{n} \cdot \nabla q + \mathbf{n} \cdot \nabla \phi &= 0, \quad (x, z) \in [0, 1] \times \{0\}, \\
\varepsilon \mathbf{n} \cdot \nabla q - \mathbf{n} \cdot \nabla \phi &= 0, \quad (x, z) \in [0, 1] \times \{1\}.
\end{align*}
\]

In the sequel, two cases will be considered:

- Aligned case: the magnetic field line aligned with z-axis, i.e. \( \mathbf{b} = (0, 1)^T \).
- Non-aligned case: the magnetic field line does not aligned with Cartesian coordinates.

In Appendix A, the finite difference methods for the MMAP scheme (9)-(10) are given for both aligned case and non-aligned case. In any case, the linear system takes a matrix form as

\[
\begin{pmatrix} A_3 & B \\ A_2 & A_1 \end{pmatrix} \begin{pmatrix} Q \\ \Phi \end{pmatrix} = \begin{pmatrix} F \\ 0 \end{pmatrix}.
\]

This block matrix can be factorized as

\[
\mathbf{A} = \begin{pmatrix} A_3 & B \\ A_2 & A_1 \end{pmatrix} = \begin{pmatrix} A_3 & 0 \\ A_2 & A_1 - A_2 A_3^{-1} B \end{pmatrix} \begin{pmatrix} I & A_3^{-1} B \\ 0 & I \end{pmatrix}.
\]

The so called Schur complement is

\[
E = A_1 - A_2 A_3^{-1} B.
\]

Denote the lower block triangular matrix as follows

\[
\mathbf{P} = \begin{pmatrix} A_3 & 0 \\ A_2 & E \end{pmatrix}.
\]

Using \( \mathbf{P} \) as preconditioner, typical Krylov subspace methods such as GMRES [15] will converge in two iterations [14] [16] for any grid size and any value of \( \varepsilon \). However, the price to pay is to invert the matrix \( A_3 \), which is too costly when discretization mesh size \( N \) is large enough.

Now the central question is how to find a suitable approximation \( S \) of the exact Schur complement \( E = A_1 - A_2 A_3^{-1} B \) with a cheap cost. More precisely, let us consider the following block preconditioner

\[
\mathbf{P}_S = \begin{pmatrix} A_3 & 0 \\ A_2 & S \end{pmatrix}.
\]
its inverse is

\[ P^{-1}_S = \begin{pmatrix} A_3^{-1} & 0 \\ -S^{-1}A_2A_3^{-1} & S^{-1} \end{pmatrix}. \]

Now applying the preconditioner to \( A \), we get

\[ P^{-1}_S A = \begin{pmatrix} I & A_3^{-1}B \\ 0 & S^{-1}E \end{pmatrix}, \]

which means the eigenvalues of \( P^{-1}_S A \) are 1 and \( \lambda \), where \( \lambda \) is eigenvalue of \( S^{-1}E \). If we could construct a suitable approximation for the exact Schur complement \( E \), then the eigenvalues of \( P^{-1}_S A \) will be clustered tightly. As a result one can expect that the Krylov iterative methods have a rapid convergence.

3.1. Approximate Schur complement. In this subsection we give the basic ideas to develop the approximate Schur complement for the MMAP system matrix \( A \) in (11). We express the coefficient matrix of (11) in continuous operator form

\[ \begin{pmatrix} A_3 & B \\ A_2 & A_1 \end{pmatrix} \rightarrow \begin{pmatrix} -\Delta_1 & -\Delta_1 \\ \varepsilon\Delta_1 & -\Delta_1 \end{pmatrix}. \]

Formally, one can verify that the underlying continuous operator for the exact Schur complement \( E = A_1 - A_2A_3^{-1}B \) is

\[ -\Delta_1 - \varepsilon\Delta_1(-\Delta_1)^{-1}(\Delta_1) = -\Delta_1 - \varepsilon\Delta_1, \]

which is precisely the original PDE operator for variable \( \phi \) in (4). Thus a natural approximation matrix for \( E \) is the discrete coefficient matrix for (13) with zero boundary condition on \( \Gamma_D \) and zero Neumann boundary condition on \( \Gamma_N \). We denote such natural approximate Schur complement as \( S_1 \). However when \( \varepsilon \rightarrow 0 \), due to zero parallel derivative \( n \cdot \nabla \phi \), one can deduce that \( S_1 \) tends to be singular. The performance of such block preconditioner with \( S_1 \) will deteriorate. For this assertion, one can see numerical results of aligned case in Table 1.

Now we want to give a new approximation from the algebraic level. Let us denote the diagonal part of \( A_3 \) by \( D \), which may offer some useful spectral information of \( A_3 \). Hence, substituting \( A_3 \) by \( D \), we obtain a new approximation for \( E \)

\[ S_2 = A_1 - A_2D^{-1}B. \]

The matrix \( S_2 \) contains the spectral information of \( A_3 \) in the algebraic level and will show extra advantage when \( \varepsilon \) becomes very small as indicated by our numerical results in section 4. We remark that the proposed two approximate Schur complement \( S_1 \) and \( S_2 \) can be easily constructed in any space dimension and be conveniently accessed in parallel computing environment.

Remark 1. The natural approximate Schur complement \( S_1 \) needs more explanation here. Firstly it performs rather well for large \( \varepsilon \). Moreover, in our numerical experiments we find that the iteration numbers of GMRES using \( S_1 \) is also stable in non-aligned case. But this is not in aligned case. In non-aligned case, we numerically observe that the condition number of \( S_1 \) scales as follows

\[ \text{cond}(S_1) \sim \frac{1}{\varepsilon + \beta^2 h^2} \frac{1}{h^2}. \]
where $\beta$ is a parameter of magnetic field in (17). Therefore if $\beta$ is not very small and the grid size $h$ is not very fine, this “bad approximation” of non-AP discretization can be viewed as a good preconditioning for the MMAP discretization.

3.2. A further modification. Above we have constructed two approximate Schur complement. One is motivated from the continuous PDE level and the other mainly from the algebraic level. In this subsection we combine the two ideas to give further modification for the approximation.

Recall that the operator for $A_1$ is $-\Delta_1$ and for $B$ is $-\Delta_1$, so we can also use matrix $A_1 + \epsilon B$ to approximate operator (13). Unfortunately, on the one hand, the matrix $A_1$ may have zero rows on $[0,1] \times \{0\}$, since $\mathbf{n} \cdot \mathbf{b}^T (\mathbf{b}^T$ is orthogonal to $\mathbf{b})$ may vanish on $[0,1] \times \{0\}$, and on the other hand, matrix $B$ has no boundary conditions on $[0,1] \times \{0\}$. This fact means that $A_1 + \epsilon B$ may be singular due to null rows on bottom boundary $[0,1] \times \{0\}$. To remedy this, we replace such null rows in $A_1 + \epsilon B$ by the corresponding rows in $S_2$. Then we obtain a third approximate Schur complement $S_3$.

In order to assess the influence of the approximation of the boundary rows in $S_3$, we construct an extra expense approximation $S_4$ using exact inverse of $A_3$. In detail we replace null boundary rows of $A_1 + \epsilon B$ on $[0,1] \times \{0\}$ using the corresponding rows of the exact Schur complement $A_1 - A_2 A_3^{-1} B$ instead of the ones of $S_2$. Here $S_4$ is just used for performance comparison and we do not recommend this use in practice.

We also propose two additional approximate Schur complements. The first one, denoted as $S_5$, is a combination of $S_1$ and $S_3$, that is the rows corresponding to $[0,1] \times \{0\}$ of $S_5$ are simply sum of the rows of $S_1$ and $S_3$. From this, we expect to incorporate the both advantages of $S_1$ and $S_3$. The second one is only valid for aligned case with the discretization given in A.1, where we expect to construct a symmetric positive definite approximate Schur complement. For this, we impose a Robin boundary condition as follows

$$\phi - h \partial_z \phi = 0, \quad (x,z) \in (0,1) \times \{0\}.$$

The corresponding discretization is

$$2\phi_{i,1} - \phi_{i,2} = 0, \quad i \in [2,n_x - 1].$$

Let us denote this approximate Schur complement by $S_6$.

Now let us give a short summarize. We first construct three approximate Schur complements $S_\alpha, \alpha = 1, 2, 3$ from different point of view. We also use an expense $S_4$ for performance comparison. Finally, we give two additional ones $S_5$ and $S_6$ for improved performance.

Remark 2. In the appendix, we prove that for aligned case the exact Schur complement $E$ is precisely $S_4$. Our numerical tests also seem to indicate this because the GMRES solver converges in only 2 steps for both aligned case and non-aligned case and for any grid size and anisotropic parameters. This fact tells us that if we can find good approximation of boundary rows of exact Schur complement then we can find good approximate Schur complement.

4. Numerical results

This section is devoted to show performance of the proposed block preconditioner of Section 3. We notice that to apply $P_S$, as a preconditioner, in each step of GMRES iteration, a following solve of
triangular block linear system is needed

\[
\begin{pmatrix}
A_3 & 0 \\
A_2 & S_\alpha
\end{pmatrix}
\begin{pmatrix}
e_1 \\
e_2
\end{pmatrix} =
\begin{pmatrix}
r_1 \\
r_2
\end{pmatrix},
\]

which is equivalent to solve two linear systems successively as

\[
\begin{align*}
A_3 e_1 &= r_1, \\
S_\alpha e_2 &= r_2 - A_2 e_1.
\end{align*}
\]  

4.1. Setup. In this section, we consider a fixed magnetic field \([5, 6]\)

\[
B = \begin{pmatrix}
\pi \beta (x^2 - x) \sin(\pi z) \\
\beta (2x - 1) \cos(\pi z) + \pi
\end{pmatrix}.
\]

Thus we have

\[
b = \frac{B}{|B|}.
\]

The manufactured solution \(\phi\) is given by

\[
\phi(x, z) = \sin(m(\pi x + \beta (x^2 - x) \cos(\pi z))) + \epsilon \cos(2\pi z) \sin(\pi x).
\]

Below, following parameters are used, \(m = 4, \beta = 0\) or \(2, \epsilon \in [10^{-20}, 1]\). The magnetic field lines defined in \([17]\) are illustrated in Figure 1, where \(\beta = 0\) corresponds to aligned case and \(\beta = 2\) for non-aligned case. In the sequel, without special explanation, the generalized minimal residual algorithm (GMRES) \([15]\) method with no restart will be used as iterative solver for solving the linear system \([11]\). The relative tolerance is set to be 1.0e-6, and the maximum iterative number is 100. Moreover, all the resolution of the MMAP scheme have reach desired precision, thus we will not show errors analysis in this section.

4.2. Aligned case. In this part, we consider aligned case, where discretization is described in Appendix A.1.

Let us first show graphically the eigenvalues of different matrices, which is reported in Figure 2. We observe that the eigenvalues of the unpreconditioned matrix \(A\) have both large distribution in real axis and image axis, and this similar phenomena is valid for all values of \(\epsilon\). On the contrary, for preconditioned matrix \(P S^1 A\), we expect eigenvalues are tightly close to positive real axis and are as clustered as possible. Ideally, we hope these eigenvalues tend to 1. For the case with \(S_1\),
we see the extent of eigenvalues is proportional to $\frac{1}{\varepsilon}$. This is due to the fact that there is no cross derivatives in aligned case, thus the discretizations of $\Delta_1$ and $\Delta_1$ can be made separately, i.e. $\Delta_1$ is discretized aligned with $z$-axis and the one of $\Delta_1$ is aligned with $x$-axis. The eigenvalues of $S_1$ are real and the maximum range can be estimated as 19

$$\max(\text{eig}(S_1)) \sim \frac{1}{\varepsilon h^2}.$$  

For the case with $S_2$, we see the eigenvalues already becomes more clustered than the case with $S_1$, especially for small $\varepsilon$, e.g. $\varepsilon < 10^{-6}$. However, for $\varepsilon = 1$, the eigenvalues are not tightly clustered to the positive real axis, and this phenomena can be observed more significantly on refined mesh. This is because when $\varepsilon$ is big, the solutions of original systems are close to solutions of MMAP scheme. In such case $S_1$ ia a very good approximation of $E$. However this is not the case with very small $\varepsilon$.

For the case with $S_3$, $\alpha = 3, 5, 6$, we see that the eigenvalues are tightly close to the positive real axis for all values of $\varepsilon$. More precisely, we find that for $\varepsilon = 1$, the eigenvalues for cases with $S_3$ and $S_6$ are very similar, and they are better than the case with $S_3$. However, for small $\varepsilon$, the eigenvalues for cases with $S_3$ and $S_5$ become more similar, and they are better than the case with $S_6$. As a result, the case with $S_5$ seems to be the best in the comparison shown in Figure 2. Notice that in Figure 2, the eigenvalues of the case with $S_4$ are not shown, it is because $S_4$ is nothing but the exact Schur complement $E$ (see the detail in Appendix B), thus the eigenvalues of this case are all equal to 1.

Now let us solve the linear system with GMRES solver. At each iteration we need a resolution of the linear systems (15)-(16). This resolution is divided into two steps: the first step is to factorize the matrices $A_3$ and $S_5$, then the second step is to solve (15)-(16) thanks to the factorized matrices obtained in the first step. Due to the finite difference discretization in Appendix A.1, we know that the matrices $A_3$ and $S_6$ are symmetric positive definite (SPD), we thus factorize them by the Cholesky decomposition. For the other $S_\alpha, \alpha = 1, \ldots, 5$, we just apply the LU decomposition.

The iteration numbers of the GMRES method are reported in Table 1 where we compare among different mesh sizes and anisotropy strength. Overall, the results we got are consistent with the previous analysis of eigenvalues. For the case with $S_1$, at most 3 iterations are needed for convergence of the GMRES method when $\varepsilon \geq 10^{-10}$. However when $\varepsilon$ is vanishing, the GMRES method does not converge anymore. For the case with $S_2$, the preconditioner works well for small $\varepsilon$, but the convergence rate is quite slow for large $\varepsilon$. For the case with $S_3$, it combines the advantages of cases with $S_1$ and $S_2$. We observe that the iterations do not increase when refining mesh and the iterations decrease for vanishing $\varepsilon$. For the case with $S_4$, just two iterations are needed since $S_4$ is exact Schur complement. For the case with $S_5$, we obtain exact table as for the case with $S_3$. Finally, for the case with $S_6$, the results are very similar as the one for the case with $S_3$ or $S_5$, except for $\varepsilon = 10^{-6}$ and $N = 256$, where the GMRES solver converges in 6 iterations.

The computational time for solving preconditioned linear system is reported in Figure 3. Firstly, we see the matrix factorization takes most of computational time, especially for very fine mesh, factorization time is much more than that for GMRES resolution. Secondly, Cholesky decomposition takes less than half of computational time of LU decomposition. This agrees with well-known results, that is complexity of Cholesky decomposition is $O(\frac{1}{3}n^3)$ ($n$ is the size of matrix), while the one of LU decomposition is $O(\frac{2}{3}n^3)$. Therefore, we conclude that the resolution for the case with approximate Schur complement $S_6$ is more efficient than the cases with the other approximate Schur complements $S_\alpha, \alpha = 1, 2, 3, 5$ for the aligned case. Note that finite difference discretization
**Figure 2.** Aligned case. Eigenvalues for the original system $A$ and the preconditioned system $P^{-1}_S A$. Top row: unpreconditioned eigenvalues (red $\bullet$'s); second row: $S_1$ (green $\bullet$'s); third row: $S_2$ (blue $\bullet$'s); third row: $S_3, S_5, S_6$ (yellow $\star$'s, black $+$'s, magenta $\times$'s respectively). $N$ stands for mesh size.

**Figure 3.** Aligned case. The computational time for solving preconditioned linear system. The presented time is rescaled by the maximum value from each group of comparison. $S_\alpha, \alpha=3, 5, 6$ stands for different approximate Schur complement. Each bar is decomposed into two portions: bottom one is time of matrix factorizations and top one is for resolution time. $N$ stands for mesh size.
Table 1. Iteration numbers of GMRES method with approximate Schur complements $S_\alpha$, $\alpha = 1, \ldots, 6$.

| Case with $S_1$ | $\epsilon$ | $10^{-1}$ | $10^{-2}$ | $10^{-6}$ | $10^{-10}$ | $10^{-20}$ |
|-----------------|------------|-----------|-----------|-----------|-----------|-----------|
| $N = 32$        | 3          | 3         | 3         | 3         | 1         | $\times$ |
| $N = 64$        | 3          | 3         | 3         | 3         | 2         | $\times$ |
| $N = 128$       | 3          | 3         | 3         | 3         | 2         | $\times$ |
| $N = 256$       | 3          | 3         | 3         | 3         | 2         | $\times$ |

| Case with $S_2$ | $\epsilon$ | $10^{-1}$ | $10^{-2}$ | $10^{-6}$ | $10^{-10}$ | $10^{-20}$ |
|-----------------|------------|-----------|-----------|-----------|-----------|-----------|
| $N = 32$        | 14         | 6         | 5         | 2         | 2         | 2         |
| $N = 64$        | 18         | 7         | 5         | 2         | 2         | 2         |
| $N = 128$       | 33         | 11        | 4         | 2         | 2         | 2         |
| $N = 256$       | 67         | 11        | 4         | 2         | 2         | 2         |

| Case with $S_3$ | $\epsilon$ | $10^{-1}$ | $10^{-2}$ | $10^{-6}$ | $10^{-10}$ | $10^{-20}$ |
|-----------------|------------|-----------|-----------|-----------|-----------|-----------|
| $N = 32$        | 4          | 4         | 4         | 3         | 2         | 2         |
| $N = 64$        | 4          | 4         | 4         | 3         | 2         | 2         |
| $N = 128$       | 4          | 4         | 4         | 3         | 2         | 2         |
| $N = 256$       | 4          | 4         | 4         | 4         | 2         | 2         |

| Case with $S_4$ | $\epsilon$ | $10^{-1}$ | $10^{-2}$ | $10^{-6}$ | $10^{-10}$ | $10^{-20}$ |
|-----------------|------------|-----------|-----------|-----------|-----------|-----------|
| $N = 32$        | 2          | 2         | 2         | 2         | 2         | 2         |
| $N = 64$        | 2          | 2         | 2         | 2         | 2         | 2         |
| $N = 128$       | 2          | 2         | 2         | 2         | 2         | 2         |
| $N = 256$       | 2          | 2         | 2         | 2         | 2         | 2         |

| Case with $S_5$ | $\epsilon$ | $10^{-1}$ | $10^{-2}$ | $10^{-6}$ | $10^{-10}$ | $10^{-20}$ |
|-----------------|------------|-----------|-----------|-----------|-----------|-----------|
| $N = 32$        | 4          | 4         | 4         | 3         | 2         | 2         |
| $N = 64$        | 4          | 4         | 4         | 3         | 2         | 2         |
| $N = 128$       | 4          | 4         | 4         | 3         | 2         | 2         |
| $N = 256$       | 4          | 4         | 4         | 4         | 2         | 2         |

| Case with $S_6$ | $\epsilon$ | $10^{-1}$ | $10^{-2}$ | $10^{-6}$ | $10^{-10}$ | $10^{-20}$ |
|-----------------|------------|-----------|-----------|-----------|-----------|-----------|
| $N = 32$        | 4          | 4         | 4         | 3         | 2         | 2         |
| $N = 64$        | 4          | 4         | 4         | 3         | 2         | 2         |
| $N = 128$       | 4          | 4         | 4         | 2         | 2         | 2         |
| $N = 256$       | 4          | 4         | 4         | 6         | 2         | 2         |
given in Appendix [A.2] can also deal with aligned case just by setting $\beta = 0$. We find that for the case with $S_\alpha, \alpha = 1, \ldots, 5$, we almost obtain the same results as presented in this subsection. However, since in this case $S_0$ is no longer symmetric, we can not benefit the advantages of SPD.

4.3. Non-aligned case. For the non-aligned case, we use the finite difference discretization given in Appendix [A.2]. Again, let us start with the graphics of the eigenvalues distribution of different matrices, which is reported in Figure 4. We obtain very similar graphics for the matrices $A$ and $P^{-1}_S A$ (in the cases with $S_\alpha, \alpha = 2, 3, 5$) as we have in aligned case. One exception is for the case with $S_1$, where we find that the range of eigenvalues on the real axis is not proportional to $\frac{1}{\epsilon}$, instead it becomes a constant for small $\epsilon$, e.g. $\epsilon = 10^{-10}$ or $10^{-20}$. A very similar phenomena can also be observed from the condition number of $S_1$, which is reported in Figure 5. From Figure 5(a), we see that the condition number of $S_1$ tends to be constant when $\epsilon < 10^{-6}$ for the non-aligned case, while it is proportional to $\frac{1}{\epsilon}$ for the aligned case. Moreover, from Figure 5(b), when varying $\beta$, we see that for large $\epsilon$ the condition number is a constant and for small $\epsilon$ the condition number is proportional to $\frac{1}{\epsilon^2}$. From Figure 5(c), when varying mesh number $N$, we see that for large $\epsilon$ the condition number is proportional to $N^2$ and for small $\epsilon$ the condition number is proportional to $N^4$. Therefore, we make a conjecture of the estimation of the condition number of $S_1$ as

\begin{equation}
\text{cond}(S_1) \approx \frac{1}{\epsilon} + \frac{1}{\epsilon^2}.
\end{equation}

As a result, for small $\epsilon$, the condition number of $S_1$ is proportional to $\frac{1}{\epsilon}$ for fixed $\beta > 0$, thus the condition number is better than that of the aligned case. This explains that the range of real part of eigenvalues of $P^{-1}_S A$ does not explode for the non-aligned case when vanishing $\epsilon$ and the good performance even with small $\epsilon$. Finally, from numerical computation, we find that $S_1$ is almost equal to the exact Schur complement $E$, thus the corresponding eigenvalues are very tightly clustered to 1. This fact indicates that the difference between the exact Schur complement $E$ and $A_1 + \epsilon B$ is just the added rows corresponding to the boundary $\{0, 1\} \times \{0\}$, therefore the more accurate we approximate well these boundary rows, the better performance of the block preconditioner $P_S$ we can expect.

The numbers of the GMRES iteration are summarized in Table 2. Remarkably for the case with $S_1$, we see the GMRES iterations almost do not increase when refining meshes and they increase slightly when $\epsilon$ vanishing. It seems that the case with $S_1$ preforms the best comparing to the cases with $S_\alpha, \alpha = 2, 3, 5$. This phenomena is very different from the aligned case, especially when $\epsilon = 10^{-20}$, where the GMRES method does not converge. This ”robustness” seems to verify our conjecture (20) from a different point of view. For the case with $S_2$, we see again the GMRES method does not work well for large $\epsilon$. For $S_3$, the GMRES iteration are almost independent of the choice of $\epsilon$. More importantly, it does not deteriorate again for vanishing $\beta$. For the case with $S_4$, the GMRES method converges in two iterations which indicates that $S_4$ is almost equal to the exact Schur complement $E$. Finally, the case of $S_5$ combines the advantages of the cases with $S_1$ and $S_3$. Thus we conclude that approximate Schur complement $S_1$ and $S_5$ are the preferred ones for non-aligned case.

Figure 6 presents relative residual history of GMRES iteration. We see a clear tendency of convergence of GMRES method with different approximate Schur complements. For the current case with $\beta = 2$, we find $S_1$ performs the best. However, from the conjecture (20), the condition number of $S_1$ deteriorates when vanishing $\beta$, thus the convergence of the GMRES method may also be perturbed. For instance, by setting $\beta = 0.1$, the GMRES method does not converge when $N = 256$. 
The case with $S_5$ performs less well than the one with $S_1$, but it is better than the case with $S_3$, especially for large $\varepsilon$.

The computational time for solving preconditioned linear system is reported in Figure 7. Again, we factorize first the matrices $A_3$ and $S_3$, then we apply the GMRES method. The first observation is that, very different from the aligned case, for the non-aligned case most computational time comes
from the GMRES solver. For large $\varepsilon$, the cases with $S_1$ and $S_5$ are more efficient than the one with $S_3$. But for small $\varepsilon$, the efficiency of all choices is almost the same. Therefore we recommend to use the preconditioner with the approximate Schur complement $S_5$.

4.4. **Without inflow boundary condition.** In Section 2 the introduced MMAP scheme provides a unique couple of $(\phi, q)$, where the key point is to impose the inflow condition for $q$. However, we notice that the auxiliary $q$ does not involve directly in computation of $\phi$, instead the parallel derivative $\nabla \parallel \hat{q}$ is used in the MMAP scheme. This similar situation also appears in the solution of
saddle point problem from incompressible Navier-Stokes equations (see [7] and references therein), where the matrix of NS equations is singular because pressure variable is not unique even though its gradient is unique.

In preconditioned GMRES iteration, one mainly needs to apply matrix vector multiplication and a inverse of the preconditioner. Thus in principle GMRES method can still work even if the matrix is singular provided the preconditioner is applicable. In the solving of NS equations [7], GMRES can obtain right solutions for the velocity. In this paper $q$ is just a auxiliary variable, thus the inflow condition can be dropped if GMRES iteration can generate the correct solutions for $\phi$. In another word, the inflow condition is not necessary to be introduced for determining a unique $\phi$ when using GMRES. But when using direct solvers we can not drop the inflow condition.

Figure 6. Convergence history of GMRES iteration for the non-aligned case.

Figure 7. Non-aligned case. The computational time for solving preconditioned linear system. The presented time is rescaled by the maximum value from each group of comparison. $\tilde{S}_\alpha$, $\alpha = 1, 3, 5$ stands for different approximate Schur complement. Each bar is decomposed into two portions: bottom one is time of matrix factorizations and top one is for resolution time. $N$ stands for mesh size.
In this subsection, we substitute the inflow condition in (9) by \( n \cdot \nabla q + n \cdot \nabla \phi = 0, (x, z) \in [0, 1] \times \{0\} \) and solve

\[
\begin{align*}
-\Delta q - \Delta \phi &= f, \quad (x, z) \in (0, 1) \times (0, 1), \\
q &= 0, \quad (x, z) \in \{0, 1\} \times [0, 1], \\
n \cdot \nabla q + n \cdot \nabla \phi &= 0, \quad (x, z) \in [0, 1] \times \{0, 1\}.
\end{align*}
\]

(21)

Now the MMAP scheme becomes (21) and (10), and its corresponding discretization matrix (denoted again by \( A \)) is no longer invertible, because the discretization matrix of the operator \(-\Delta q\) of (21) (denoted by \( \hat{A}_3 \)) is now not invertible. We write the new linear system by

\[
\begin{pmatrix}
\hat{A}_3 & B \\
A_2 & A_1
\end{pmatrix}
\begin{pmatrix}
\hat{Q} \\
\Phi
\end{pmatrix} =
\begin{pmatrix}
F \\
0
\end{pmatrix}.
\]

(22)

Remember that we need the preconditioner to be applicable. To make the sub-blocks of the preconditioner invertible, we use \( A_3 \) to compute the approximate Schur complement. Namely we use \( A_3 \) as an approximation of \( \hat{A}_3 \), so that the preconditioners discussed in Section 3 remain the same. In short we use inflow conditions in the preconditioning but not in the computation of coefficient matrix \( A \).

Here, we only consider the approximate Schur complement \( S_5 \), which is shown to be the most robust choice in previous subsections. The iteration number of the GMRES method is reported in Table 3. For aligned case, Table 3(a) is the same as in Table 1(e). While for non-aligned case, Table 3(b) is also similar as in Table 2(e). This show that even though the matrix of linear system (22) is not invertible, the proposed preconditioned GMRES algorithm still works well.

**Table 3.** Iteration number of the GMRES method for the scheme without inflow boundary condition. The approximate Schur complement \( S_5 \) is used.

| (a) Aligned case | \( \varepsilon \) | \( 1 \times 10^{-1} \) | \( 10^{-2} \) | \( 10^{-6} \) | \( 10^{-10} \) | \( 10^{-20} \) |
|------------------|----------------|-----------------|----------|----------|----------|----------|
| \( N = 32 \)     | 4              | 4               | 3        | 2        | 2        |
| \( N = 64 \)     | 4              | 4               | 4        | 3        | 2        | 2        |
| \( N = 128 \)    | 4              | 4               | 4        | 3        | 2        | 2        |
| \( N = 256 \)    | 4              | 4               | 4        | 4        | 2        | 2        |

| (b) Non-aligned case | \( \varepsilon \) | \( 1 \times 10^{-1} \) | \( 10^{-2} \) | \( 10^{-6} \) | \( 10^{-10} \) | \( 10^{-20} \) |
|---------------------|----------------|-----------------|----------|----------|----------|----------|
| \( N = 32 \)        | 6              | 10              | 13       | 12       | 12       |
| \( N = 64 \)        | 6              | 12              | 16       | 15       | 14       | 14       |
| \( N = 128 \)       | 6              | 12              | 20       | 18       | 17       | 17       |
| \( N = 256 \)       | 6              | 12              | 25       | 19       | 19       |

5. Conclusion and perspectives

In this paper, we introduce block preconditioning methods for the MMAP method. We propose a series of block preconditioners based on the approximate Schur complement techniques. We perform a number of numerical tests for both the aligned case and non-aligned case. Numerical results show
that on the one hand, the proposed block preconditioners are robust with grid refinement; on the other hand, they can work well for all anisotropic strengths.

The MMAP method works for the cases where the border of magnetic field lines cross the boundaries of computational domain. For the case with closed magnetic field lines, we still need to explore other asymptotic-preserving methods. For instance, in [11], an asymptotic-preserving scheme based on first order system (APFOS) was introduced. Therefore, to develop a suitable block preconditioning method for APFOS scheme can be an extension of present work. Moreover the implementation of finite element method and better approximation of the boundary rows of exact Schur complement is under consideration.

ACKNOWLEDGEMENTS

This work has been supported by Heilongjiang Provincial Natural Science Foundation of China (LH2019A013) and the Fundamental Research Funds for the Central Universities. Chang Yang acknowledges support by a public grant from the "Laboratoire d’Excellence Centre International de Mathématiques et d’Informatique" (LabEx CIMI) overseen by the French National Research Agency (ANR) as part of the "Investissements d’Avenir" program (reference ANR-11-LABX-0040) in the frame of the PROMETEUS project (PROspect of nOvel nuMerical modEls for elecTric propulsion and low tEmperatUre plaSmas) and from the FrFCM (Fédération de recherche pour la Fusion par Confinement Magnétique) in the frame of the NEMESIA project (Numerical mEthods for Macroscopic models of magnEtized plaSmas and related anIsotropic equAtions). Lingxiao Li has also been supported by National Natural Science Foundation of China under Grant 11901042.

APPENDIX A. FINITE DIFFERENCE DISCRETIZATIONS

In this part, let us give the finite difference discretizations for the MMAP scheme introduced in Section 2.

A.1. Aligned case. In the aligned case, we consider the magnetic field aligned with z-axis, i.e. \( \mathbf{b} = (0,1)^T \). For the convenience of construction of matrices, we write the MMAP scheme of Section 2 as follows

\[
\begin{align*}
-\partial^2_{zz} q - \partial^2_{x\phi} &= f, \quad (x,z) \in (0,1) \times (0,1), \\
q &= 0, \quad (x,z) \in \{0,1\} \times [0,1], \\
q &= 0, \quad (x,z) \in [0,1] \times \{0\}, \\
\partial_z q &= 0, \quad (x,z) \in [0,1] \times \{1\},
\end{align*}
\]

and

\[
\begin{align*}
\varepsilon \partial^2_{zz} q - \partial^2_{x\phi} &= 0, \quad (x,z) \in (0,1) \times (0,1), \\
\phi &= 0, \quad (x,z) \in \{0,1\} \times [0,1], \\
\partial_z q &= 0, \quad (x,z) \in [0,1] \times \{0\}, \\
\partial_z \phi &= 0, \quad (x,z) \in [0,1] \times \{1\}.
\end{align*}
\]
The mesh is defined by
\[
\begin{align*}
x_i &= (i-1)\Delta x, \quad i \in \{1, \ldots, n_x\}, \\
z_j &= (j-3/2)\Delta z, \quad j \in \{1, \ldots, n_z\},
\end{align*}
\]
where \(\Delta x = \frac{1}{n_x-1}\) and \(\Delta z = \frac{1}{n_z-2}\). Furthermore, we take \(n_x = N\) and \(n_z = N+1\), so that \(h = \Delta x = \Delta z\). Then a second order finite difference discretization for (23) and (24) can be directly determined as
\[
\begin{align*}
-\phi_{i-1,j} + 2\phi_{i,j} - \phi_{i+1,j} &= 0, \quad (i, j) \in \{2, n_x - 1\} \times \{2, n_z - 1\}, \\
\phi_{1,j} &= \phi_{n_x,j} = 0, \quad j \in \{1, n_z\}, \\
q_{i,1} &= 0, \quad i \in \{2, n_x - 1\}, \\
-q_{i,n_z-1} + q_{i,n_z} &= 0, \quad i \in \{2, n_x - 1\},
\end{align*}
\]
and
\[
\begin{align*}
\varepsilon(q_{i,j-1} - 2q_{i,j} + q_{i,j+1}) + \\
(-\phi_{i-1,j} + 2\phi_{i,j} - \phi_{i+1,j}) &= 0, \quad (i, j) \in \{2, n_x - 1\} \times \{2, n_z - 1\}, \\
\phi_{1,j} &= \phi_{n_x,j} = 0, \quad j \in \{1, n_z\}, \\
-\phi_{i,n_z-1} + \phi_{i,n_z} &= 0, \quad i \in \{2, n_x - 1\}, \\
q_{i,1} - q_{i,2} &= 0, \quad i \in \{2, n_x - 1\},
\end{align*}
\]
where \(\phi_{i,j}\) and \(q_{i,j}\) are numerical approximations of \(\phi(x_i, z_j)\) and \(q(x_i, z_j)\) respectively, and \(f_{i,j} = f(x_i, z_j)\). Now, to construct the linear system to (25)-(26), we define a lexicographical, that is to count first in \(z\) direction then in \(x\) direction. Denote \(\text{iline}\) the index of the linear system, we then define a function ‘lexico’ to convert from the couple \((i, j)\) to \(\text{iline}\) as
\[
\text{iline} = \text{lexico}(i, j) = j + (i-1)n_z.
\]
So the linear system corresponding to (25)-(26) is given by
\[
\begin{pmatrix} A_3 & B \\
A_2 & A_1 \end{pmatrix} \begin{pmatrix} Q \\
\Phi \end{pmatrix} = \begin{pmatrix} F \\
0 \end{pmatrix}.
\]
At last, in order to get a more symmetrical matrix forms, we make some algebraic operations, which yield in the first equation of (25) as
\[
\begin{align*}
-\phi_{i-1,j} + 2\phi_{i,j} - \phi_{i+1,j} &\rightarrow 2\phi_{i,j} - \phi_{i+1,j}, \quad (i, j) \in \{2\} \times \{2, n_x - 1\}, \\
-\phi_{i-1,j} + 2\phi_{i,j} - \phi_{i+1,j} &\rightarrow -\phi_{i-1,j} + 2\phi_{i,j}, \quad (i, j) \in \{n_x - 1\} \times \{2, n_z - 1\}, \\
-q_{i,j-1} + 2q_{i,j} - q_{i,j+1} &\rightarrow 2q_{i,j} - q_{i,j+1}, \quad (i, j) \in \{2, n_x - 1\} \times \{2\}.
\end{align*}
\]
Thanks to these algebraic operations, the matrices \(A_3\) and \(B\) become symmetric.

### A.2. Non-aligned case
Now let us concentrate discretization for the general MMAP scheme (9)-(10). Again, this time we will consider a second order finite difference method.
Firstly, the mesh is defined different from the previous subsection, since we have to discretize cross derivative on all interior grids. For this, the mesh is defined by

\[
\begin{align*}
  x_i &= (i-1)\Delta x, \ i \in \{1, \ldots, n_x\}, \\
  z_j &= (j-1)\Delta z, \ j \in \{1, \ldots, n_z\},
\end{align*}
\]

where \(\Delta x = \frac{1}{n_x-1}\) and \(\Delta z = \frac{1}{n_z-1}\). Again, we take \(n_x = n_z = N\), so that \(h = \Delta x = \Delta z\). Secondly, we notice that

\[
\Delta \phi = \Delta |\phi| + \Delta \bot \phi,
\]

its corresponding discretization on interior grids is

\[
\Delta_h \phi_{i,j} = \Delta^h \phi_{i,j} + \Delta^h \phi_{i,j}, \quad (i, j) \in \{2, \ldots, n_x-1\} \times \{2, \ldots, n_z-1\}.
\]

Using the usual central discretization, \(\Delta \phi\) can be approximated as

\[
h^2 \Delta_h \phi_{i,j} = \phi_{i+1,j} + \phi_{i-1,j} + \phi_{i,j+1} + \phi_{i,j-1} - 4\phi_{i,j}.
\]

Then, let us denote \(b \otimes b\) by

\[
b \otimes b = \begin{pmatrix}
  b^{11} & b^{12} \\
  b^{21} & b^{22}
\end{pmatrix},
\]

thus we can recast \(\Delta |\phi|\) as

\[
\Delta |\phi| = \partial_x G + \partial_z H,
\]

where \(G = b^{11}\partial_x \phi + b^{12}\partial_z \phi\), \(H = b^{21}\partial_x \phi + b^{22}\partial_z \phi\). Similarly, we get approximate \(\Delta \bot \phi\) as

\[
h^2 \Delta^h \phi_{i,j} = (G_{i+1/2,j} - G_{i-1/2,j}) + (H^h_{j,i+1/2} - H^h_{j,i-1/2}),
\]

where

\[
G^h_{i+1/2,j} = \frac{1}{4} b^{11}_{i+1/2,j} (\phi_{i+1,j} - \phi_{i,j}) + \frac{1}{4} b^{12}_{i+1/2,j} (\phi_{i+1,j+1} + \phi_{i+1,j+1} - \phi_{i+1,j-1} - \phi_{i+1,j-1}),
\]

\[
H^h_{i,j+1/2} = \frac{1}{4} b^{21}_{i,j+1/2} (\phi_{i+1,j+1} + \phi_{i+1,j+1} - \phi_{i+1,j} - \phi_{i+1,j}) + \frac{1}{4} b^{22}_{i,j+1/2} (\phi_{i+1,j} - \phi_{i+1,j}.
\]

At last, we obtain immediately approximated \(\Delta |\phi|\) by

\[
\Delta^h |\phi|_{i,j} = \Delta^h \phi_{i,j} - \Delta^h \phi_{i,j}.
\]

Thanks to these discrete operators, the sub-blocks \(A_1, A_2, A_3\) and \(B\) from the linear system \([11]\) correspond to \(-\Delta^h \phi\), \(\varepsilon \Delta^h q\), \(-\Delta^h q\) and \(-\Delta^h \phi\) respectively on interior grids. Finally, let us consider discretization for boundary conditions. From \([9]-[10]\), we see that it is necessary to approximate \(n \cdot \nabla_\perp \phi\) and \(n \cdot \nabla_\parallel q\) on \([0, 1] \times \{0, 1\}\). We only present the approximation on boundary \([0, 1] \times \{0\}\), the one on boundary \([0, 1] \times \{1\}\) is similar. Notice that the normal direction on boundary \([0, 1] \times \{0\}\) is \(n = (0, 1)^T\), we thus has

\[
\begin{align*}
  n \cdot \nabla_\perp \phi &= b^{11} \partial_z \phi - b^{12} \partial_x \phi, \\
  n \cdot \nabla_\parallel q &= b^{21} \partial_x q + b^{22} \partial_z q.
\end{align*}
\]

Now applying center discretizations, we get a second approximation for above differential operations

\[
\begin{align*}
  h n \cdot \nabla^h_\perp \phi_{i,1} &= \frac{1}{2} b^{11}_{i,1} (-3\phi_{i,1} + 4\phi_{i,2} - \phi_{i,3}) - \frac{1}{2} b^{12}_{i,1} (\phi_{i+1,1} - \phi_{i-1,1}), \\
  h n \cdot \nabla^h_\parallel q_{i,1} &= \frac{1}{2} b^{21}_{i,1} (q_{i+1,1} - q_{i-1,1}) + \frac{1}{2} b^{22}_{i,1} (-3q_{i,1} + 4q_{i,2} - q_{i,3}).
\end{align*}
\]
At last, substituting \( \mathbf{n} \cdot \nabla \phi \) and \( \mathbf{n} \cdot \nabla q \) in equations (9)-(10) by the above approximation, we obtain desired second order finite difference schemes.

**Appendix B. The property of the approximate Schur complement \( S_4 \)**

In this part, for aligned case we will show the approximate Schur complement \( S_4 \) of Section 3 is precisely the exact one \( E \). We will consider the finite difference discretization described in Appendix A.1, thus the relevant matrices are given as follows:

\[
A_1 = \begin{pmatrix}
I & a_1 \\
& \ddots & \ddots & \ddots \\
& & a_1 & I
\end{pmatrix}, \quad a_1 = \begin{pmatrix}
0 & -1 & 2 & -1 \\
& \ddots & \ddots & \ddots \\
& & -1 & 2 & -1 \\
& & & -1 & 1
\end{pmatrix}
\]

\[
A_2 = \begin{pmatrix}
0 & a_2 \\
& \ddots & \ddots & \ddots \\
& & a_2 & 0
\end{pmatrix}, \quad a_2 = \begin{pmatrix}
1 & -1 & \varepsilon & -2\varepsilon & \varepsilon \\
& \ddots & \ddots & \ddots & \ddots \\
& & \varepsilon & -2\varepsilon & \varepsilon
\end{pmatrix}
\]

\[
A_3 = \begin{pmatrix}
I & a_3 \\
& \ddots & \ddots & \ddots \\
& & a_3 & I
\end{pmatrix}, \quad a_3 = \begin{pmatrix}
1 & 2 & -1 \\
& \ddots & \ddots & \ddots \\
& & -1 & 2 & -1 \\
& & & -1 & 1
\end{pmatrix}
\]

\[
B = \begin{pmatrix}
I & -b_0 & 2b_0 & -b_0 \\
& \ddots & \ddots & \ddots & \ddots \\
& & -b_0 & 2b_0 & -b_0 \\
& & & \ddots & \ddots \\
& & & & I
\end{pmatrix}, \quad b_0 = \begin{pmatrix}
0 & 1 \\
& \ddots & \ddots \\
& & 1
\end{pmatrix}
\]

Thanks to the results on tridiagonal matrices [8], we find

**Lemma B.1.** The inverse of the matrix \( A_3 \) is as follows

\[
A_3^{-1} = \begin{pmatrix}
1 & 0 \\
0 & C
\end{pmatrix}, \quad C = \left[ \min\{i,j\}\right]_{i,j=1,...,N-1}
\]

**Corollary B.2.** From Lemma B.1, we immediately have

\[-A_2A_3^{-1}B = \begin{pmatrix}
0 & 1 & \cdots & 1 & 0 \\
-b_1 & 2b_1 & -b_1 & \ddots & \ddots \\
& \ddots & \ddots & \ddots & \ddots \\
& & -b_1 & 2b_1 & -b_1 \\
& & & & 0
\end{pmatrix}, \quad b_1 = \begin{pmatrix}
0 & 1 \\
& \ddots & \ddots \\
& & \varepsilon
\end{pmatrix}
\]
Finally, from Corollary B.2 we deduce a consequence of the approximate Schur complement $S_4$ of Section 3.

**Proposition B.3.** The approximate Schur complement $S_4$ is equal to the exact Schur complement $E$, i.e.

$$S_4 = A_1 - A_2 A_3^{-1} B.$$

We remark that if we substitute the first row of the matrix $b_1$ by null row, then the matrix $-A_2 A_3^{-1} B$ becomes $\varepsilon B$. That is to say the only difference between the exact Schur complement $E$ and $A_1 + \varepsilon B$ comes from these rows, which correspond to numerical approximation on the boundary $[0, 1] \times \{0\}$. This fact for aligned case tells us that if we construct a better approximation of these rows, we can obtain a better approximate Schur complement, and then the preconditioned Krylov solver converges more rapidly.

**References**

[1] C. Besse, P. Degond, F. Deluzet, J. Claudel, G. Gallice, and C. Tessieras. A model hierarchy for ionospheric plasma modeling. *Mathematical Models & Methods in Applied Sciences*, 14(03):393–415, 2004.

[2] C. Besse, F. Deluzet, C. Negulescu, and C. Yang. Efficient numerical methods for strongly anisotropic elliptic equations. *Journal of Scientific Computing*, 55(1):231–254, 2013.

[3] L. Chacón, D. Del-Castillo-Negrete, and C. D. Hauck. An asymptotic preserving semi-lagrangian algorithm for the time-dependent anisotropic heat transport equation. *Journal of Computational Physics*, 272:719–746, 2014.

[4] P. Degond, F. Deluzet, and C. Negulescu. An asymptotic preserving scheme for strongly anisotropic elliptic problems. *Multiscale Modeling & Simulation*, 8(2):645–666, 2009.

[5] P. Degond, A. Lozinski, J. Narski, and C. Negulescu. An asymptotic-preserving method for highly anisotropic elliptic equations based on a micro-macro decomposition. *Journal of Computational Physics*, 231(7):2724–2740, 2012.

[6] F. Deluzet and J. Narski. A two field iterated asymptotic-preserving method for highly anisotropic elliptic equations. *Multiscale Modeling & Simulation*, pages 434–459, 2019.

[7] H. Elman, D. D. Silvester, and A. Wathen. *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics (Second Edition)*. New York: Oxford University Press, 2014.

[8] C. Fonseca. On the eigenvalues of some tridiagonal matrices. *Journal of Computational & Applied Mathematics*, 200(1):283–286, 2007.

[9] N. Krasheninnikova and L. Chacón. Fourth order discretization of anisotropic heat conduction operator. In *APS Division of Plasma Physics Meeting Abstracts*, volume 50 of APS Meeting Abstracts, page BP6.040, Nov. 2008.

[10] L. Li and C. Yang. Asymptotic preserving scheme for anisotropic elliptic equations with deep neural network. *Preprint*.

[11] E. T. Meier. *Modeling plasmas with strong anisotropy, neutral fluid effects, and open boundaries*. PhD thesis, University of Washington, 2011.

[12] J. D. Meiss. *Plasma confinement*. Dover Publications, 2003.

[13] M. F. Murphy, G. H. Golub, and A. J. Wathen. A note on preconditioning for indefinite linear systems. *SIAM J. Sci. Comput.*, 21(6):1969–1972, 2000.

[14] Y. Saad and M. H. Schultz. Gmres: A generalized minimal residual algorithm for solving non-symmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7(3):856–869, 1986.

[15] D. Silvester, H. Elman, D. Kay, and A. Wathen. Efficient preconditioning of the linearized navier-stokes equations for incompressible flow. *Journal of Computational and Applied Mathematics*, 128(12):261–279, 2001.

[16] M. Tang and Y. Wang. An asymptotic preserving method for strongly anisotropic diffusion equations based on field line integration. *Journal of Computational Physics*, 330:735–748, 2017.

[17] Y. Wang, W. Ying, and M. Tang. Uniformly convergent scheme for strongly anisotropic diffusion equations with closed field lines. *SIAM Journal on Scientific Computing*, pages B1253–B1276, 2018.
[19] C. Yang, J. Claustre, and F. Deluzet. Iterative solvers for elliptic problems with arbitrary anisotropy strengths. *Multiscale Modeling & Simulation*, 16(4):1795–1823, 2018.

[20] C. Yang, F. Deluzet, and J. Narski. On the numerical resolution of anisotropic equations with high order differential operators arising in plasma physics. *Journal of Computational Physics*, 386:502–523, 2019.

[21] C. Yang, F. Deluzet, and J. Narski. Preserving the accuracy of numerical methods discretizing anisotropic elliptic problems. Preprint.

*Email address: lilingxiao@lsec.cc.ac.cn*

*Email address: yangchang@hit.edu.cn*