Science & Technology Facilities Council

Preprint

RAL-P-2019-002

A scalable approximate inverse block preconditioner for an incompressible magnetohydrodynamics model problem

M Wathen, C Greif

April 2019

Submitted for publication in SIAM Journal on Scientific Computing

RAL Library
STFC Rutherford Appleton Laboratory
R61
Harwell Oxford
Didcot
OX11 0QX

Tel: +44(0)1235 445384
Fax: +44(0)1235 446403
email: libraryral@stfc.ac.uk

ISSN 1361- 4762

A Scalable Approximate Inverse Block Preconditioner for an Incompressible Magnetohydrodynamics Model Problem

Michael Wathen* and Chen Greif†

Abstract. We introduce a new approximate inverse preconditioner for a mixed finite element discretization of an incompressible magnetohydrodynamics (MHD) model problem. The derivation exploits the nullity of the discrete curl-curl operator in the Maxwell subproblem. We show that the inverse of the coefficient matrix contains zero blocks, and we use discretization considerations to further sparsify and obtain a practical preconditioner. We demonstrate the viability of our approach with a set of numerical experiments.

Key words. incompressible magnetohydrodynamics, saddle-point linear systems, null space, preconditioners, approximate inverse, Krylov subspace methods

AMS subject classifications. 65F08, 65F10, 65F15, 65F50, 65N22

1. Introduction.

Incompressible magnetohydrodynamics (MHD) describes the flow of an electrically conductive fluid in the presents of a magnetic field [2, 7, 11, 17]. Given a sufficiently smooth domain Ω, consider the steady-state incompressible MHD model [2, Ch. 2]:

$$-\nu\,\Delta \boldsymbol{u} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} + \nabla p - \kappa\,(\nabla\times\boldsymbol{b})\times\boldsymbol{b} = \boldsymbol{f} \qquad \text{in } \Omega, \tag{1.1a}$$

$$\nabla\cdot\boldsymbol{u} = 0 \qquad \text{in } \Omega, \tag{1.1b}$$

$$\kappa\nu_m\,\nabla\times(\nabla\times\boldsymbol{b}) + \nabla r - \kappa\,\nabla\times(\boldsymbol{u}\times\boldsymbol{b}) = \boldsymbol{g} \qquad \text{in } \Omega, \tag{1.1c}$$

$$\nabla\cdot\boldsymbol{b} = 0 \qquad \text{in } \Omega. \tag{1.1d}$$

Here $\boldsymbol{u}$ is the velocity, p is the hydrodynamic pressure, $\boldsymbol{b}$ is a magnetic field, and the Lagrange multiplier associated with the divergence constraint on the magnetic field is denoted by r. The vector functions $\boldsymbol{f}$ and $\boldsymbol{g}$ represent external forcing terms. The three dimensionless parameters that characterize this model are: the hydrodynamic viscosity ν, the magnetic viscosity ν_m, and the coupling number κ.

To complete the model, we consider the inhomogeneous Dirichlet boundary conditions:

$$\boldsymbol{u} = \boldsymbol{u}_0 \qquad \text{on } \partial\Omega, \tag{1.2a}$$

$$\boldsymbol{n}\times\boldsymbol{b} = \boldsymbol{n}\times\boldsymbol{b}_0 \qquad \text{on } \partial\Omega, \tag{1.2b}$$

$$r = r_0 \qquad \text{on } \partial\Omega, \tag{1.2c}$$

with $\boldsymbol{n}$ being the unit outward normal on $\partial\Omega$ and $\boldsymbol{u}_0$, $\boldsymbol{b}_0$ and r_0 being the functions defined on the boundary.

*STFC Rutherford Appleton Laboratory, Chilton, Didcot, Oxfordshire OX11 0QX, michael.wathen@stfc.ac.uk.

†Department of Computer Science, The University of British Columbia, Vancouver, BC, V6T 1Z4, Canada, greif@cs.ubc.ca. The work of this author was supported in part by an NSERC Discovery Grant.

We will consider two nonlinear iteration schemes – Picard iteration and Newton's method. Using the notation

$$\begin{aligned} \boldsymbol{u}_{k+1} &= \boldsymbol{u}_k + \delta\boldsymbol{u}, \quad p_{k+1} = p_k + \delta p, \\ \boldsymbol{b}_{k+1} &= \boldsymbol{b}_k + \delta\boldsymbol{b}, \quad r_{k+1} = r_k + \delta r, \end{aligned}$$

where $*_{k+1}$ is the $k+1^{\text{st}}$ iteration of either the Picard or Newton's iteration, the linearized system can be written as:

$$\begin{aligned} -\nu\Delta\delta\boldsymbol{u} + (\boldsymbol{u}_k\cdot\nabla)\delta\boldsymbol{u} + \alpha(\delta\boldsymbol{u}\cdot\nabla)\boldsymbol{u}_k + \nabla\delta p - \kappa(\nabla\times\delta\boldsymbol{b})\times\boldsymbol{b}_k - \alpha\kappa(\nabla\times\boldsymbol{b}_k)\times\delta\boldsymbol{b} &= \boldsymbol{r}_u, \\ \nabla\cdot\delta\boldsymbol{u} &= r_p, \\ \kappa\nu_m\nabla\times(\nabla\times\delta\boldsymbol{b}) + \nabla r - \kappa\nabla\times(\delta\boldsymbol{u}\times\boldsymbol{b}_k) - \kappa\alpha\nabla\times(\boldsymbol{u}_k\times\delta\boldsymbol{b}) &= \boldsymbol{r}_b, \\ \nabla\cdot\delta\boldsymbol{b} &= r_r, \end{aligned} \tag{1.3}$$

where

$$\begin{aligned} R_u &= \boldsymbol{f} - \left[-\nu\,\Delta\boldsymbol{u}_k + (\boldsymbol{u}_k\cdot\nabla)\boldsymbol{u}_k + \nabla p_k - \kappa\,(\nabla\times\boldsymbol{b}_k)\times\boldsymbol{b}_k\right], \\ R_p &= -\nabla\cdot\boldsymbol{u}_k, \\ R_b &= \boldsymbol{g} - \left[\kappa\nu_m\,\nabla\times(\nabla\times\boldsymbol{b}_k) + \nabla r - \kappa\,\nabla\times(\boldsymbol{u}_k\times\boldsymbol{b}_k)\right], \\ R_r &= -\nabla\cdot\boldsymbol{b}_k, \end{aligned}$$

and

$$\alpha = \begin{cases} 0 & \text{for Picard} \\ 1 & \text{for Newton.} \end{cases} \tag{1.4}$$

We consider a finite element discretization of the MHD model (1.1)–(1.2) where the hydrodynamic unknowns ($\boldsymbol{u}$ and p) are discretized with any stable mixed finite elements and the magnetic and multiplier unknowns are discretized through a mixed edge and nodal element pair. Using the same formulation as [26], we use Taylor-Hood elements [23] for $(\boldsymbol{u}, p)$ and the lowest order Nédélec [19] pair for $(\boldsymbol{b}, r)$. This choice of mixed finite elements avoids the necessity to stabilize the fluid/pressure variables and provides conforming $H(\text{curl})$ elements for the magnetic variables. Thus, it provides us with a robust and well understood discretization to test the robustness of our new preconditioning techniques. Upon discretization, we obtain the following linear system:

$$\begin{pmatrix} F + \alpha F_{\text{NT}} & B^T & C^T + \alpha C_{\text{NT}}^T & 0 \\ B & 0 & 0 & 0 \\ -C & 0 & M + \alpha M_{\text{NT}} & D^T \\ 0 & 0 & D & 0 \end{pmatrix} \begin{pmatrix} \delta u \\ \delta p \\ \delta b \\ \delta r \end{pmatrix} = \begin{pmatrix} r_u \\ r_p \\ r_b \\ r_r \end{pmatrix}, \tag{1.5}$$

with

$$\begin{aligned} r_u &= f - Fu_k - C^T b_k - B^T p_k, \\ r_p &= -Bu_k, \\ r_b &= g - Mu_k + Cb_k - D^T r_k, \\ r_r &= -Db_k, \end{aligned}$$

where F is a discrete convection-diffusion operator; B is a fluid divergence operator; M is the curl-curl operator; D is the magnetic divergence operator and C represents the coupling terms. We use subscript NT to denote Newton linearization; see (1.3) for the continuous continuous quantities. The dimensions are defined as follows:

$$\dim(\delta u) = n_u, \quad \dim(\delta p) = m_u, \quad \dim(\delta b) = n_b, \quad \text{and} \quad \dim(\delta r) = m_b.$$

System (1.5) needs to be solved repeatedly, with changing right-hand-sides, throughout the nonlinear iteration.

In recent years, the interest in the development of block preconditioning methods for the MHD model has increased; see [1, 6, 15, 20, 21, 24–26]. Those papers propose ways to approximate the block Schur complements that arise for the specific formulations and factorizations they consider and show good scalable iterations with respect to mesh refinement. So far, fully scalable iterations for large three-dimensional problems with high coupling numbers have not been fully developed. In this paper, we build upon and utilize the approximations and techniques used in the class of block triangular preconditioners [20, 21, 26], aiming to develop a scalable block approximate inverse preconditioner.

The remainder of the paper is structured as follows. In Section 2, we derive a new formula for the inverse and show that the (exact) inverse has a few zero blocks. In Section 3, we approximate Schur complements that appear in the formula by sparse operators, and derive a new inverse approximation formula. In Section 4, we use the approximate Schur complement to form a block triangular preconditioner. Section 5 presents numerical experiments that demonstrate the viability and effectiveness of this preconditioning approach. Finally, we offer some concluding remarks in Section 6.

2. A new formula for the inverse of the MHD coefficient matrix.

The null spaces of $M + \alpha M_{\mathrm{NT}}$ and $C^T + \alpha C_{\mathrm{NT}}^T$ are different for $\alpha = 1$ but are the same for $\alpha = 0$, therefore to preserve the null space properties of the block system, we derive the new inverse formula for the Picard iteration, $\alpha = 0$. The derivation of an inverse formula for the $\alpha = 1$ case is significantly more complex due to the fact that the null spaces of $M + \alpha M_{\mathrm{NT}}$ and $C^T + \alpha C_{\mathrm{NT}}^T$ are different. However, in our numerical experiments (Section 5), we show that the same approach derived for the Picard case works very well for the Newton nonlinear iteration scheme. Let us denote by $\mathcal{K}$ the coefficient matrix in the MHD model (1.5) and write it as:

$$\mathcal{K} = \begin{pmatrix} \mathcal{K}_{\mathrm{NS}} & \mathcal{K}_{\mathrm{C}}^T \\ -\mathcal{K}_{\mathrm{C}} & \mathcal{K}_{\mathrm{M}} \end{pmatrix},$$

where $\mathcal{K}_{\mathrm{NS}}$ is the Navier-Stokes subproblem, $\mathcal{K}_{\mathrm{C}}$ is the block for the coupling and $\mathcal{K}_{\mathrm{M}}$ is the Maxwell subproblem:

$$\mathcal{K}_{\mathrm{NS}} = \begin{pmatrix} F & B^T \\ B & 0 \end{pmatrix}, \quad \mathcal{K}_{\mathrm{M}} = \begin{pmatrix} M & D^T \\ D & 0 \end{pmatrix},$$

$$\mathcal{K}_{\mathrm{C}}^T = \begin{pmatrix} C^T & 0 \\ 0 & 0 \end{pmatrix} \quad \text{and} \quad \mathcal{K}_{\mathrm{C}} = \begin{pmatrix} C & 0 \\ 0 & 0 \end{pmatrix}.$$

Then, by [5, Equation (3.4)], the inverse is given by

$$\mathcal{K}^{-1} = \begin{pmatrix} \mathcal{K}_{\rm NS}^{-1} + \mathcal{K}_{\rm NS}^{-1}\mathcal{K}_{\rm C}^{T}\mathcal{S}^{-1}\mathcal{K}_{\rm C}\mathcal{K}_{\rm NS}{}^{-1} & -\mathcal{K}_{\rm NS}^{-1}\mathcal{K}_{\rm C}^{T}\mathcal{S}^{-1} \\ -\mathcal{S}^{-1}\mathcal{K}_{\rm C}\mathcal{K}_{\rm NS}{}^{-1} & \mathcal{S}^{-1} \end{pmatrix}, \tag{2.1}$$

where $\mathcal{S}$ denotes the Schur complement,

$$\mathcal{S} \;=\; \mathcal{K}_{\rm M} + \mathcal{K}_{\rm C}\mathcal{K}_{\rm NS}^{-1}\mathcal{K}_{\rm C}^{T}. \tag{2.2}$$

The inverses $\mathcal{K}_{\rm NS}^{-1}$ and $\mathcal{S}^{-1}$ appear multiple times in (2.1), and we now derive explicit formulas that further reveal their block structure. Notably, using results that have appeared in [9], we show that $\mathcal{S}^{-1}$ has a zero (2,2) block, and can be expressed in terms of a free matrix parameter.

Theorem 2.1. *The inverse of the block Schur complement, $\mathcal{S}$, can be written as:*

$$\mathcal{S}^{-1} = \begin{pmatrix} M_F^{-1}(I - D^T W^{-1} G^T) & GW^{-1} \\ W^{-1}G^T & 0 \end{pmatrix}, \tag{2.3}$$

where W is a (free) symmetric positive definite matrix,

$$\mathcal{K}_{\rm NS}^{-1} = \begin{pmatrix} K_1 & K_2 \\ K_3 & K_4 \end{pmatrix}, \quad M_F = M + D^T W^{-1} D + C K_1 C^T, \quad \textit{and} \quad G = M_F^{-1} D^T.$$

Proof. Writing out all the matrices involved in formula (2.2) for $\mathcal{S}$, we have

$$\mathcal{S} = \begin{pmatrix} M + CK_1C^T & D^T \\ D & 0 \end{pmatrix}.$$

Since the discrete gradient operator is the null space of M and C^T, we have

$$\dim(\text{null}(M + CK_1C^T)) = m_b.$$

Thus, the (1,1) block of the Schur complement has the maximum nullity which still allows a nonsingular saddle point system. Therefore, using [9, equation (3.6)], the inverse of the Schur complement is given by (2.3). ∎

Using the inverse formula (2.1) and (2.3) together gives

$$\mathcal{K}^{-1} = \begin{pmatrix} K_1 - K_1ZK_1 & K_2 - K_1ZK_2 & -K_1C^TM_F^{-1}H & 0 \\ K_3 - K_3ZK_1 & K_4 - K_3ZK_2 & -K_3C^TM_F^{-1}H & 0 \\ M_F^{-1}CK_1 & M_F^{-1}CK_2 & M_F^{-1}H & GW^{-1} \\ 0 & 0 & W^{-1}G^T & 0 \end{pmatrix}, \tag{2.4}$$

where $Z = C^T M_F^{-1} C$ and $H = I - D^T W^{-1} G^T$. Here we would like to draw attention to the zero blocks present in the inverse (2.4). In general, the structure of the inverse of a sparse matrix is dense. However, due to the specific null space properties (similarly to [9]), we have shown that the inverse of the MHD coefficient matrix has five zero blocks.

3. A new approximate inverse-based preconditioner. In this section, we aim to sparsify the inverse formula in (2.4) by exploiting the null-space properties and the magnitude of matrix entries.

3.1. A block-sparse approximation of the Schur complement. Recall that K_1 is the (1, 1) block matrix of the inverse of the Navier-Stokes subproblem. Thus forming it is very computationally costly and will yield a dense matrix. Therefore, computing CK_1C^T is a major bottleneck within M_F, which is a integral part of Schur complement, $\mathcal{S}$. Defining

$$M_F = M_W + CK_1C^T$$

where $M_W = M + D^T W^{-1} D$, then using a generalization of the Sherman-Morrison-Woodbury theorem, we can re-write M_F as

$$M_F^{-1} = M_W^{-1} - M_W^{-1}CK_1(K_1 - K_1C^T M_W^{-1} CK_1)^{-1} K_1 C^T M_W^{-1}. \tag{3.1}$$

From [9, Proposition 3.6], $M_W^{-1}D^T$ is defined to be the null space of M. Since the null spaces of M and C^T coincide, it follows that $C^T M_W^{-1} D^T = 0$ as well. Thus, using this null space property and (3.1) for G in (2.3) we obtain

$$\begin{aligned} G &= \left(M_W^{-1} - M_W^{-1}CK_1(K_1 - K_1C^T M_W^{-1} CK_1)^{-1} K_1 C^T M_W^{-1}\right) D^T, \\ &= M_W^{-1}D^T - M_W^{-1}CK_1(K_1 - K_1C^T M_W^{-1} CK_1)^{-1} K_1 C^T M_W^{-1} D^T, \\ &= M_W^{-1}D^T. \end{aligned}$$

In order to approximate M_F^{-1} we again use the Sherman-Morrison-Woodbury formula and note that the approximation order for C, M_W and K_1 are $\mathcal{O}(h^{-1})$, $\mathcal{O}(h^2)$ and $\mathcal{O}(h^{-2})$, respectively. Therefore it can be seen that:

$$M_W^{-1} \approx \mathcal{O}(h^2) \quad \text{and} \quad M_W^{-1}CK_1(K_1 - K_1C^T M_W^{-1} CK_1)^{-1} K_1 C^T M_W^{-1} \approx \mathcal{O}(h^4).$$

Thus, we use the approximation

$$M_F^{-1} \approx M_W^{-1}.$$

The final step is to decide what the arbitrary W should be. In [9], the authors define $W = DG$, where we recall that $MG = 0$ and D is the magnetic divergence operator. The null space of M is the discrete gradient operator, and in [12, Proposition 2.2] the authors showed that selecting W as the scalar Laplacian yields mesh-independent spectral bounds and a scalable preconditioner. For the approximation of $D^T W^{-1} D$, the authors in [12] showed that the vector mass matrix, X, is spectrally equivalent to $D^T W^{-1} D$. Thus, the approximation we select is:

$$\mathcal{S}^{-1} \approx \hat{\mathcal{S}}^{-1} = \begin{pmatrix} M_X^{-1}(I - D^T W^{-1} \hat{G}^T) & \hat{G} W^{-1} \\ W^{-1}\hat{G}^T & 0 \end{pmatrix},$$

where $M_X = M + X$ and $\hat{G} = M_X^{-1} D^T$.

From the inverse formula in (2.1) and the multiplication of the inverse with the coefficient matrix, it is possible to observe that some block operations in $\mathcal{S}^{-1}\mathcal{K}_{\mathrm{C}}$ and $\mathcal{S}^{-1}\mathcal{K}_{\mathrm{M}}$ appear multiple times. Therefore there are multiplications of the leading block of the Schur complement with either M or C. Since $G^T M = 0$ and $G^T C = 0$, the simplified inverse Schur complement becomes:

$$\mathcal{S}^{-1}_{\text{approx}} = \begin{pmatrix} M_X^{-1} & \hat{G}W^{-1} \\ W^{-1}\hat{G}^T & 0 \end{pmatrix}. \tag{3.2}$$

3.2. A practical preconditioner. In a similar fashion to the reduction of the simplification of M_F to M_W using the approximate magnitude of the matrix entries, we observe that

$$K_i Z K_j \gtrsim \mathcal{O}(h^3)$$

for any $i, j = 1, 2, 3, 4$. Hence, removing these terms from the block 2-by-2 leading block matrix and substituting the approximate Schur complement (3.2) forms the first step for the approximation of (2.4) as:

$$\tilde{\mathcal{P}}_1^{-1} = \begin{pmatrix} K_1 & K_2 & -K_1 C^T M_X^{-1} & 0 \\ K_3 & K_4 & -K_3 C^T M_X^{-1} & 0 \\ M_X^{-1} C K_1 & M_X^{-1} C K_2 & M_X^{-1} & \hat{G}W^{-1} \\ 0 & 0 & W^{-1}\hat{G}^T & 0 \end{pmatrix}. \tag{3.3}$$

The final step to approximate (3.3) is to consider what approximation to use for the inverse of the Navier-Stokes system $\mathcal{K}_{\mathrm{NS}}$. Numerically it appears to be beneficial to consider the exact inverse of the Navier-Stokes coefficient matrix. For this we return to the exact inverse formula of a block matrix in (2.1). Applying this to the Navier-Stokes system gives the following precise expression for the inverse

$$\mathcal{K}_{\mathrm{NS}}^{-1} = \begin{pmatrix} F^{-1} - F^{-1}B^T S_{\mathrm{NS}}^{-1} B F^{-1} & F^{-1}B^T S_{\mathrm{NS}}^{-1} \\ S_{\mathrm{NS}}^{-1} B F^{-1} & -S_{\mathrm{NS}}^{-1} \end{pmatrix}. \tag{3.4}$$

In practice, we use the pressure-convection diffusion (PCD) approximation developed in [8]. Substituting (3.4) into the expression for $\tilde{\mathcal{P}}_1^{-1}$ in (3.3) gives

$$\tilde{\mathcal{P}}_1^{-1} = \begin{pmatrix} N & F^{-1}B^T S_{\mathrm{NS}}^{-1} & -NC^T M_X^{-1} & 0 \\ S_{\mathrm{NS}}^{-1} B F^{-1} & -S_{\mathrm{NS}}^{-1} & -S_{\mathrm{NS}}^{-1} B F^{-1} C^T M_X^{-1} & 0 \\ M_X^{-1} C N & M_X^{-1} C F^{-1} B^T S_{\mathrm{NS}}^{-1} & M_X^{-1} & \hat{G}W^{-1} \\ 0 & 0 & W^{-1}\hat{G}^T & 0 \end{pmatrix}, \tag{3.5}$$

where

$$N = F^{-1} - F^{-1}B^T S_{\mathrm{NS}}^{-1} B F^{-1}.$$

As with the approximation of M_F^{-1} in Section 3, we consider the approximate orders of the individual blocks of (3.5). Removing the $\mathcal{O}(h^3)$ terms in the (1,3) and (3,1) blocks of (3.5) yields the approximation:

$$\mathcal{P}_1^{-1} = \begin{pmatrix} N & F^{-1}B^T S_{\mathrm{NS}}^{-1} & 0 & 0 \\ S_{\mathrm{NS}}^{-1} B F^{-1} & -S_{\mathrm{NS}}^{-1} & -S_{\mathrm{NS}}^{-1} B F^{-1} C^T M_X^{-1} & 0 \\ 0 & M_X^{-1} C F^{-1} B^T S_{\mathrm{NS}}^{-1} & M_X^{-1} & \hat{G}W^{-1} \\ 0 & 0 & W^{-1}\hat{G}^T & 0 \end{pmatrix}. \tag{3.6}$$

3.3. Spectral analysis. For the eigenvalue analysis, we use $\tilde{\mathcal{P}}_1^{-1}$ in (3.5) since the analysis is significantly simplified. Practically, the preconditioner $\mathcal{P}_1^{-1}$ exhibits good results and the eigenvalue analysis for $\tilde{\mathcal{P}}_1^{-1}\mathcal{K}$ provides a theoretical justification for the use of these preconditioners. We therefore use the preconditioner $\mathcal{P}_1^{-1}$ for the numerical results in Section 5.

Due to the fact that the preconditioner is in inverse form the generalized eigenvalue problem considered is:

$$\mathcal{K}v = \lambda\tilde{\mathcal{P}}_1 v \quad \Longrightarrow \quad \tilde{\mathcal{P}}_1^{-1}\mathcal{K}v = \lambda v.$$

Thus, $\tilde{\mathcal{P}}_1^{-1}\mathcal{K}$ is given by:

$$\tilde{\mathcal{P}}_1^{-1}\mathcal{K} = \begin{pmatrix} I_{\mathrm{u}} + NC^T M_W^{-1} C & 0 & NC^T(I - M_W^{-1}M) & 0 \\ S_{\mathrm{NS}}^{-1} BF^{-1} C^T M_W^{-1} C & I_{\mathrm{p}} & S_{\mathrm{NS}}^{-1} BF^{-1} C^T (I - M_W^{-1} M) & 0 \\ 0 & 0 & M_W^{-1}(M + CNC^T) & \hat{G} \\ 0 & 0 & 0 & I_{\mathrm{r}} \end{pmatrix}. \tag{3.7}$$

where

$$M_W = M + DW^{-1}D^T.$$

Let us introduce a few identities utilizing the null space properties of C^T and M_W. Proposition 3.1 is particularly useful to simplify (3.7).

Proposition 3.1. *The following relations hold:*

$$\begin{aligned} &\text{(i)} \quad C^T(I - M_W^{-1}M) = 0 \\ &\text{(ii)} \quad C^T(I + M_W^{-1}M) = 2C^T \end{aligned}$$

Proof. Since C^T and M_W have the same null space (space of discrete gradients), thus by using the Helmholtz decomposition

$$b = d + \nabla\,\phi,$$

we obtain

$$C^T(I - M_W^{-1}M)b = C^T(I - M_W^{-1}M)d,$$

where $d \notin \mathrm{Null}(M)$. Since $M_W^{-1}M$ is zero on the null space of M and an identity on the range space of M, then

$$(I - M_W^{-1}M) \in \mathrm{Null}(M).$$

Therefore, identity (i) holds due to the fact that the null spaces of C^T and M are the same. Using similar arguments, identity (ii) can also be shown to be true. ∎

Using Proposition 3.1 simplifies (3.7) to:

$$\tilde{\mathcal{P}}_1^{-1}\mathcal{K} = \left(\begin{array}{cc|cc} I_{\mathrm{u}} + NC^T M_W^{-1} C & 0 & 0 & 0 \\ S_{\mathrm{NS}}^{-1} BF^{-1} C^T M_W^{-1} C & I_{\mathrm{p}} & 0 & 0 \\ \hline 0 & 0 & M_W^{-1}(M + CNC^T) & \hat{G} \\ 0 & 0 & 0 & I_{\mathrm{r}} \end{array}\right), \tag{3.8}$$

where we recall that $G = M_W^{-1}D^T$ and $N = F^{-1} - F^{-1}B^T S_{\mathrm{NS}}^{-1} BF^{-1}$.

Theorem 3.2. *The matrix $\tilde{\mathcal{P}}_1^{-1}\mathcal{K}$ has an eigenvalue $\lambda = 1$ of algebraic multiplicity at least $n_u - n_b + 3m_b + m_u$. The corresponding eigenvectors $\{v_i\}_{i=1}^{n_u-n_b+3m_b+m_u}$ are given by*

$$v_i = (u_i,\, p_i,\, b_i,\, r_i),$$

where $u_i \in Null(C)$, $b_i \in Null(M)$ and p_i and r_i are free.

Proof. Since (3.8) is block diagonal where the two 2-by-2 blocks are also triangular, then generalized eigenvalue problem

$$\tilde{\mathcal{P}}_1^{-1}\mathcal{K}v = \lambda v,$$

where $v = (u,\, p,\, b,\, r)$ can be written as:

$$\lambda u = (I_u + NC^T M_W^{-1} C)u, \tag{3.9a}$$

$$\lambda p = p, \tag{3.9b}$$

$$\lambda b = M_W^{-1}(M + CNC^T)b, \tag{3.9c}$$

$$\lambda r = r. \tag{3.9d}$$

Consider $\lambda = 1$, then trivially (3.9b) and (3.9d) are automatically satisfied. Taking $u \in \text{Null}(C)$ and $b \in \text{Null}(M + CNC^T)$, (3.9a) and (3.9c) also hold. Since the null space of M and C^T are the same we choose $b \in \text{Null}(M)$. ∎

Figure 1(a) shows the preconditioned eigenvalues for $\mathcal{P}_1^{-1}\mathcal{K}$ using the PCD approximation developed in [8] for the fluid Schur complement. The approximation is based on

$$S_{\text{NS}} = BF^{-1}B^T \approx A_p F_p^{-1} Q_p, \tag{3.10}$$

where the matrix A_p is the pressure Laplacian, F_p is the pressure convection-diffusion operator and Q_p is the pressure mass matrix. We approximate the magnetic Schur complement using the vector mass matrix:

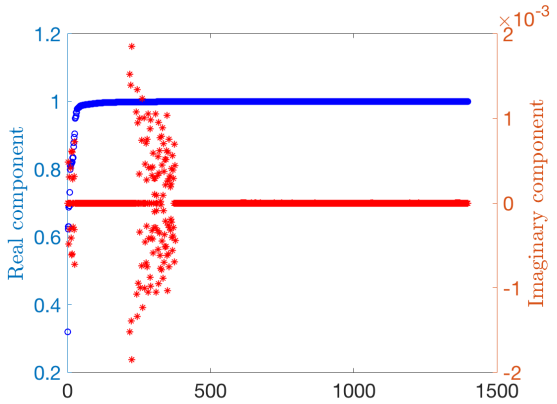
$$M + D^T W^{-1} D \approx M + X.$$

Here we see that the clustering around the eigenvalue $\lambda = 1$ is very strong.

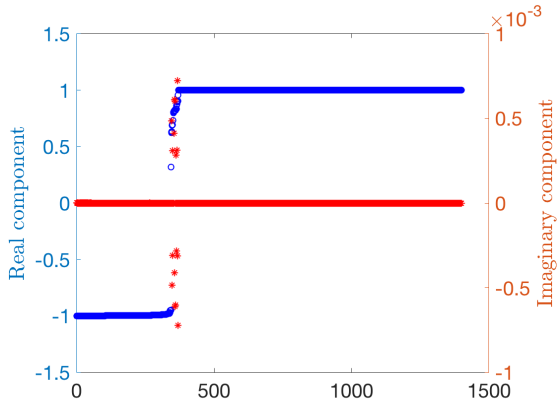
4. A block triangular preconditioner. For comparison against the approximate inverse preconditioner, we derive a block triangular preconditioner that utilizes the approximate block Schur complement inverse in (3.2) and compare the performance of the two preconditioners with respect to both timing and iteration results. Again we will consider the simpler Picard case, $\alpha = 0$, but in practice use the Newton nonlinear scheme. We follow the well known setting of [14, 18] for experimental comparison. Let us define $\tilde{\mathcal{P}}_2$ as:

$$\tilde{\mathcal{P}}_2 = \begin{pmatrix} \mathcal{K}_{\text{NS}} & \mathcal{K}_{\text{C}} \\ 0 & -\mathcal{S} \end{pmatrix}. \tag{4.1}$$

From [14, 18] the preconditioned matrix, $\tilde{\mathcal{P}}_2^{-1}\mathcal{K}$, has precisely two eigenvalues ± 1 and is diagonalizable. We would therefore expect an appropriate Krylov subspace solver to converge within two iterations in exact precision.



(a) Real (blue) and imaginary (red) parts of eigenvalues of preconditioned matrix $\mathcal{P}_1^{-1}\mathcal{K}$.



(b) Real (blue) and imaginary (red) parts of eigenvalues of preconditioned matrix $\mathcal{P}_2^{-1}\mathcal{K}$.

Figure 1: Preconditioned eigenvalue plots for (a) the approximate inverse preconditioner in (3.6) and (b) the block triangular preconditioner in (4.3) using the smooth solution given (5.2). The dimensions of these matrices are 1399×1399.

The direct solve for the Navier-Stokes system is too costly, so we approximate $\mathcal{K}_{\text{NS}}$ with the Schur complement system:

$$\mathcal{P}_{\text{NS}} = \begin{pmatrix} F & B^T \\ 0 & -S_{\text{NS}} \end{pmatrix}, \tag{4.2}$$

where $S_{\text{NS}} = BF^{-1}B^T$ is the fluid Schur complement. Using (4.2), we obtain the more practical preconditioner

$$\mathcal{P}_2 = \begin{pmatrix} \mathcal{P}_{\text{NS}} & \mathcal{K}_{\text{C}} \\ 0 & -\mathcal{S} \end{pmatrix}. \tag{4.3}$$

Theorem 4.1. *The matrix $\mathcal{P}_2^{-1}\mathcal{K}$ has an eigenvalue $\lambda = 1$ of algebraic multiplicity at least n_u, and an eigenvalue $\lambda = -1$ of algebraic multiplicity at least n_b. The corresponding (known) eigenvectors are given as follows:*

$\lambda = 1$: with eigenvectors $\{v_i\}_{i=1}^{n_b-m_b}$ and $\{v_j\}_{j=n_b-m_b+1}^{n_u}$, as follows:

$$v_i = (u_i, -S^{-1}Bu_i, b_i, 0) \quad \text{and} \quad v_j = (u_j, -S^{-1}Bu_j, 0, 0),$$

where $b_i \in null(D) \neq 0$, $Cu_i = (2M + CK_1C^T)b_i$ and $u_j \in null(C)$.

$\lambda = -1$: with eigenvectors $\{v_i\}_{i=1}^{n_b-m_b}$ and $\{v_j\}_{j=n_b-m_b+1}^{n_b}$, as follows:

$$v_i = (u_i, 0, b_i, r_i) \quad \text{and} \quad v_j = (0, 0, b_j, r_j), \tag{4.4}$$

where $u_i \in null(B) \neq 0$, $Fu_i + C^Tb_i = 0$, $b_j \in null(M)$, r_i and r_j free.

Proof. The corresponding eigenvalue problem is

$$\begin{pmatrix} F & B^T & C^T & 0 \\ B & 0 & 0 & 0 \\ -C & 0 & M & D^T \\ 0 & 0 & D & 0 \end{pmatrix} \begin{pmatrix} u \\ p \\ b \\ r \end{pmatrix} = \lambda \begin{pmatrix} F & B^T & C^T & 0 \\ 0 & -S_{\mathrm{NS}} & 0 & 0 \\ 0 & 0 & -(M+K_C) & -D^T \\ 0 & 0 & -D & 0 \end{pmatrix} \begin{pmatrix} u \\ p \\ b \\ r \end{pmatrix},$$

where $K_C = CK_1C^T$. The four block rows of the generalized eigenvalue problem can be written as

$$(1-\lambda)(Fu + B^T p + C^T b) = 0, \tag{4.5a}$$

$$Bu = -\lambda S_{\mathrm{NS}}\, p, \tag{4.5b}$$

$$(1+\lambda)(Mb + D^T r) + \lambda C K_1 C^T b - Cu = 0, \tag{4.5c}$$

$$(1+\lambda)Db = 0. \tag{4.5d}$$

We split the eigenvalue analysis into two parts: $\lambda = 1$ and $\lambda = -1$.

$\lambda = 1$**:**

Equation (4.5a) is automatically satisfied. Equation (4.5b) simplifies to:

$$p = -S_{\mathrm{NS}}^{-1} B u.$$

From (4.5d) we have $Db = 0$, hence, $b \in \mathrm{null}(D)$. Let us take $r = 0$, then (4.5c) yields

$$Cu = (2M + CK_1C^T)b. \tag{4.6}$$

- *Case 1:* Consider $b = 0$, then we have that $Cu = 0$. Hence, u must be in the null space of C. Since

$$\dim(\mathrm{null}(C)) \;=\; n_u - n_b + m_b,$$

this accounts for $n_u - n_b + m_b$ such eigenvectors.

- *Case 2:* Consider $0 \neq b \in \mathrm{null}(D)$, then $Cu = (2M + CK_1C^T)b$. Since, rank of C and $(2M + CK_1C^T)$ is $n_b - m_b$, then the condition (4.6) has at least $n_b - m_b$ linearly independent eigenvectors.

Therefore $\lambda = 1$ is an eigenvalue with algebraic multiplicity at least n_u.

$\lambda = -1$**:**

Equation (4.5d) is satisfied, hence, r is free. Simplifying (4.5c) obtains

$$CK_1C^T b + Cu = 0. \tag{4.7}$$

Let us take $u \in \mathrm{null}(B)$, then $p = 0$ and the condition for b is

$$Fu + C^T b = 0. \tag{4.8}$$

Under the condition that $u \in \mathrm{null}(B)$, (4.8) satisfies the equality (4.7).

- *Case 1:* Consider $u = 0$, then for (4.8) to hold $C^T b = 0$. Therefore, we take $b \in \mathrm{null}(C^T)$. Since, the null space of C^T is made up of discrete gradients then

$$\dim(\mathrm{null}(C^T)) \;=\; m_b.$$

Matrix	Implementation method
Q_p	single AMG V-cycle
A_p	single AMG V-cycle
$\hat{F}$	Preconditioned AMG GMRES with tolerance 1e-2
$M + X$	AMG method developed in [13] with tolerance 1e-2
W	single AMG V-cycle

Table 1: Solution method for block systems associated with the preconditioners

- *Case 2:* Consider $u \in \text{null}(B)$ and $u \neq 0$, then from (4.8) we have $u = -F^{-1}C^T b$. Since the rank of C^T is $n_b - m_b$ and F is full rank, then there are only $n_b - m_b$ such linearly independent b's that determine u. Hence, for this case we obtain at least $n_b - m_b$ such eigenvectors.

Therefore $\lambda = -1$ is an eigenvalue with algebraic multiplicity at least n_b. ■

An approximation of the fluid Schur complement, S_{NS}, is needed to create a practical preconditioner. Again we use the PCD approximation (3.10).

$\mathcal{P}_2$ is defined in (4.3), but in practice we use the approximation for the inverse of the Schur complement (3.2). The eigenvalues of the preconditioned matrix $\mathcal{P}_2^{-1}\mathcal{K}$ are represented in Figure 1(b). As with the eigenvalues for the approximate inverse preconditioned matrix we see a small degradation of the eigenvalue clusters. However, we still see strong clustering around 1 and -1.

5. Numerical experiments.

In this section, we present several 3D numerical results to illustrate the the performance and scalability of our preconditioning techniques.

Experimental setup. We use `FEniCS` [16], a finite element software package, to create the matrix system, and `PETSc` [3, 4] and `HYPRE` [10] to solve the resulting system.

Similarly to [21], we set the nonlinear stopping tolerance to 1e-4 and the linear solve tolerance as 1e-3. For all experiments we use FGMRES [22] and the Newton nonlinear iteration scheme , $\alpha = 1$ (see (1.4)). This means that for every solve associated with F and multiplication associated with C^T we replace them with $\hat{F} = F + F_{\text{NT}}$ and $\hat{C}^T = C^T + C_{\text{NT}}^T$, respectively. Table 1 details the methods which we use to solve the systems associated with the block preconditioner.

We use the notation:

- ℓ is the mesh level, DoF is the total degrees of freedom, time is the average solve time;
- it_{NL} is the number of nonlinear/Newton iterations to solve;
- it_{O} is the average number of linear/FGMRES iterations;
- it_{MX} is the average of CG/Auxiliary Space iterations to solve $M + X$;
- it_{F} is the average of GMRES iterations to solve $\hat{F}$.

Adding an A or B superscript to time or it_* denotes whether we use the approximate inverse or block preconditioner, respectively

ℓ	DoFs	time$^{\mathrm{A}}$	$it^{\mathrm{A}}_{\mathrm{NL}}$	$it^{\mathrm{A}}_{\mathrm{O}}$	$it^{\mathrm{A}}_{\mathrm{MX}}$	$it^{\mathrm{A}}_{\mathrm{F}}$	time$^{\mathrm{B}}$	$it^{\mathrm{B}}_{\mathrm{NL}}$	$it^{\mathrm{B}}_{\mathrm{O}}$	$it^{\mathrm{B}}_{\mathrm{MX}}$	$it^{\mathrm{B}}_{\mathrm{F}}$
1	14,012	1.2	3	10.0	1.7	1.6	0.8	4	21.5	2.0	2.0
2	28,436	2.8	3	10.0	1.8	1.6	1.9	4	21.5	2.0	2.0
3	64,697	11.4	3	9.7	1.9	1.8	6.9	4	21.0	2.0	2.0
4	245,276	34.6	4	11.0	2.1	1.9	13.7	3	18.3	2.0	2.0
5	937,715	255.2	4	10.5	2.2	1.9	135.9	3	19.0	2.5	2.0
6	5,057,636	1979.2	3	9.7	2.7	2.1	2273.5	3	22.3	3.0	2.5

Table 2: 3D Cavity Driven using both the approximate inverse and block triangular preconditioner with parameters $\kappa = 1$, $\nu = 1$, $\nu_m = 1$ and $\mathrm{Ha} = 1$.

5.1. 3D Cavity driven flow. The first example we consider is the classic lid driven cavity problem [8]. The problem is driven by the following Dirichlet boundary conditions:

$$\begin{aligned} &\mathbf{u} = (1,0,0) \quad \text{on} \quad z = 1, \\ &\mathbf{u} = (0,0,0) \quad \text{on} \quad x = \pm 1,\ y = \pm 1,\ z = -1, \\ &\mathbf{n} \times \mathbf{b} = \mathbf{n} \times \mathbf{b}_N \quad \text{on} \quad \partial\Omega, \\ &r = 0 \quad \text{on} \quad \partial\Omega, \end{aligned} \tag{5.1}$$

where $\mathbf{b}_N = (-1,0,0)$.

Scalability results. Along with the nondimensional parameters (ν, ν_m and κ) described in Section 1, we introduce the Hartmann number, which is equal to the ratio of electromagnetic and viscous forces. It is defined as

$$\mathrm{Ha} = \sqrt{\frac{\kappa}{\nu\nu_m}}.$$

The larger the Hartmann number is, the stronger the coupling between the electromagnetics and hydrodynamics variables, and thus, the more challenging the setting is for the numerical solution method. Tables 2 and 3 show computational time and iteration counts using the approximate inverse and block triangular preconditioners. We can see that the approximate inverse preconditioner exhibits near-perfect scaling with respect to the FGMRES iterations. On the other hand, the FGMRES iterations for the block triangular preconditioner increase each mesh level for the harder problem, $\mathrm{Ha} = \sqrt{1000}$. For the easier parameter setup, Table 2, we can see that both preconditioners yield scalable iterations. We note that for the latest mesh level ($\ell = 6$) for both Tables 2 and 3 the approximate inverse preconditioner yields a faster solution method. In fact, for the more difficult problem ($\mathrm{Ha} = \sqrt{1000}$) the approximate inverse preconditioner is almost exactly two times quicker than the block triangular preconditioner for $\ell = 6$.

Computational cost of preconditioners. From the definition of the approximate inverse preconditioner it is obvious that each iteration is more expensive than the block triangular preconditioner. However, as seen in Table 2 and particularly in Table 3, the scalability of the results for the approximate inverse preconditioner produces a significant speed up in solution time for the larger problems.

ℓ	DoFs	time$^{\mathrm{A}}$	$it^{\mathrm{A}}_{\mathrm{NL}}$	$it^{\mathrm{A}}_{\mathrm{O}}$	$it^{\mathrm{A}}_{\mathrm{MX}}$	$it^{\mathrm{A}}_{\mathrm{F}}$	time$^{\mathrm{B}}$	$it^{\mathrm{B}}_{\mathrm{NL}}$	$it^{\mathrm{B}}_{\mathrm{O}}$	$it^{\mathrm{B}}_{\mathrm{MX}}$	$it^{\mathrm{B}}_{\mathrm{F}}$
1	14,012	7.6	4	57.0	2.0	2.0	5.6	4	146.2	2.0	2.0
2	28,436	22.2	4	56.2	2.0	2.0	14.9	4	147.2	2.0	2.0
3	64,697	66.0	4	56.0	2.0	2.0	47.8	4	154.2	2.0	2.0
4	245,276	271.5	4	56.0	2.1	2.0	205.6	4	160.5	2.2	2.0
5	937,715	1255.2	4	55.5	2.8	2.0	1003.9	4	168.8	2.9	2.0
6	5,057,636	17656.4	4	58.5	3.0	2.1	35563.2	4	217.5	3.0	2.0

Table 3: 3D Cavity Driven using both the approximate inverse and block triangular preconditioner with parameters $\kappa =$ 1e1, $\nu =$ 1e-1, $\nu_m =$ 1e-1 and Ha $= \sqrt{1000}$

Linear operation	$\mathcal{P}_1 : it^{\mathrm{A}}_{\mathrm{O}} = 58.5$		$\mathcal{P}_2 : it^{\mathrm{B}}_{\mathrm{O}} = 217.5$	
	it^{A}	total$^{\mathrm{A}}$	it^{B}	total$^{\mathrm{B}}$
$\hat{F}^{-1}$	4	234.0	1	217.5
A_p^{-1}	3	175.5	1	217.5
Q_p^{-1}	3	175.5	1	217.5
M_X^{-1}	2	117.0	1	217.5
W^{-1}	2	117.0	1	217.5
$\hat{C}^T$ or C	2	117.0	1	217.5
B^T or B	4	234.0	1	217.5
G^T or G	2	117.0	2	435.0
Total execution time	17656.4		35563.2	

Table 4: Computational cost using $\mathcal{P}_1$ and $\mathcal{P}_2$ for mesh level $\ell = 6$ in Table 3. Here it^* denotes the number of block operations (application of the inverse of a matrix or matrix vector multiplication) and the total number of block operations is denoted by total* for the applications of the approximate inverse preconditioner (A) or block triangular (B).

Table 4 shows the number of solves and matrix-vector products associated the individual block matrices that make up the approximate inverse and the block triangular preconditioners for mesh level $\ell = 6$ in Table 3. We can see that the approximate inverse preconditioner has a smaller number of solves for both systems associated with vector ($\hat{F}$ and M_X) and scalar (A_p, Q_p and W) valued matrices. Also, the total number of matrix-vector products is smaller for the approximate inverse preconditioner. This is reflected in the time it takes to solve this system, where the approximate inverse preconditioner is faster than the block triangular one. We also note that the iterations in Table 3 remain constant for the approximate inverse preconditioner compared to the block triangular one. Therefore, for harder problems on larger meshes it seems that the approximate inverse preconditioner seems more efficient with respect to time and iterations.

5.2. Fichera corner. The second solution we consider is a smooth solution on a nonconvex domain. Specically, the domain is a cube missing a corner, and it is known as the Fichera corner. It is defined as

$$\Omega = (0,1)^3/[0.5,1) \times [0.5,1) \times [0.5,1).$$

An example of such a domain is given in Figure 2. The exact solution is

$$\begin{aligned} &\mathbf{u} = \nabla \times (u_1, u_1, u_1) \quad \text{on} \quad \Omega, \\ &p = xyz(x-1)(y-1)(z-1)\exp(x) \quad \text{on} \quad \Omega \\ &\mathbf{b} = \nabla \times (b_1, b_1, b_1) \quad \text{on} \quad \Omega, \\ &r = xyz(x-1)(y-1)(z-1)\exp(x+y+z) \quad \text{on} \quad \Omega, \end{aligned} \tag{5.2}$$

where

$$\begin{aligned} u_1 &= x^2y^2z^2(x-1)^2(y-1)^2(z-1)^2\cos(x), \\ b_1 &= x^2y^2z^2(x-1)^2(y-1)^2(z-1)^2\sin(y), \end{aligned}$$

which defines the inhomogeneous Dirichlet boundary conditions and forcing terms $\boldsymbol{f}$ and $\boldsymbol{g}$.

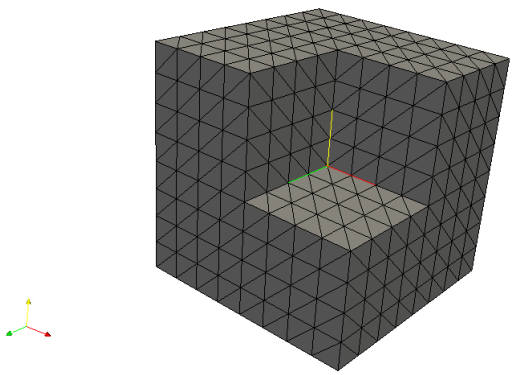


Figure 2: Example Fichera corner domain for mesh level $\ell = 3$

Table 5 shows the timing and iteration results for the following two setups:

- Setup 1: κ = 1e1, ν = 1e-2, ν_m = 1e-2 and Ha = $\sqrt{1e5}$,
- Setup 2: κ = 1e1, ν = 1e-2, ν_m = 1e-3 and Ha = 1000.

We can see that for Setup 1 (Ha = $\sqrt{1e5}$) that the outer FGMRES iterations remain constant. However, when considering Setup 2 (Ha = 1000) we start to see a large degradation in terms of the iteration counts. As mentioned in Section 5.1, as the Hartmann number increases the numerical solution procedure becomes more challenging. Hence for Ha = 1000, it is not unexpected to see loss of scalability for the solver.

5.3. MHD generator. The final test example considered is the more physically relevant MHD generator problem. It describes unidirectional flow in a duct which induces an electromagnetic field. We consider the channel $[0,5] \times [0,1] \times [0,1]$. On the left and right boundaries

ℓ	DoFs	Setup 1					Setup 2				
		time$^{\mathrm{A}}$	$it^{\mathrm{A}}_{\mathrm{NL}}$	$it^{\mathrm{A}}_{\mathrm{O}}$	$it^{\mathrm{A}}_{\mathrm{MX}}$	$it^{\mathrm{A}}_{\mathrm{F}}$	time$^{\mathrm{A}}$	$it^{\mathrm{A}}_{\mathrm{NL}}$	$it^{\mathrm{A}}_{\mathrm{O}}$	$it^{\mathrm{A}}_{\mathrm{MX}}$	$it^{\mathrm{A}}_{\mathrm{F}}$
1	34,250	15.6	4	29.2	2.6	2.0	102.3	7	211.7	2.0	2.0
2	57,569	30.4	4	29.2	2.7	2.0	242.3	7	237.0	2.1	2.0
3	89,612	52.9	4	28.8	2.9	2.0	440.7	7	252.1	2.1	2.0
4	332,744	232.2	4	27.8	3.0	2.0	2361.4	7	294.3	2.4	2.0
5	999,269	1026.3	4	27.8	3.0	3.0	8657.9	7	303.9	2.8	2.1
6	5,232,365	11593.5	5	28.6	3.0	3.0	111675.3	7	321.4	2.9	2.5

Table 5: Fichera corner using the approximate inverse preconditioner. Setup 1: κ = 1e1, ν = 1e-2, ν_m = 1e-2 and Ha = $\sqrt{1e5}$ and Setup 2: κ = 1e1, ν = 1e-2, ν_m = 1e-3 and Ha = 1000.

ℓ	DoFs	time$^{\mathrm{A}}$	$it^{\mathrm{A}}_{\mathrm{NL}}$	$it^{\mathrm{A}}_{\mathrm{O}}$	$it^{\mathrm{A}}_{\mathrm{MX}}$	$it^{\mathrm{A}}_{\mathrm{F}}$
1	2,199	1.5	3	172.7	1.5	1.9
2	13,809	13.3	3	108.0	1.5	2.0
3	96,957	260.8	4	105.2	1.9	2.0
4	724,725	1693.3	3	70.7	2.0	2.8
5	5,600,229	8515.7	3	68.0	2.1	2.6

Table 6: MHD generator using the approximate inverse preconditioner with parameters κ = 1, ν = 1e-1, ν_m = 1e-1 and Ha = 10

we enforce the boundary condition $\mathbf{u} = (1, 0, 0)$ and on the other walls a no slip boundary condition is applied. Defining $\delta = 0.1$, $b_0 = 1$, $x_{\mathrm{on}} = 2$ and $x_{\mathrm{off}} = 2.5$ then the boundary condition associated with the magnetic unknowns is $\mathbf{n} \times \mathbf{b} = \mathbf{n} \times (0, \mathbf{b}_y, 0)$ where

$$\mathbf{b}_y = \frac{b_0}{2}\left[\tanh\left(\frac{x - x_{\mathrm{on}}}{\delta}\right) - \tanh\left(\frac{x - x_{\mathrm{off}}}{\delta}\right)\right].$$

The timing and iteration results for the approximate inverse preconditioner are presented in Table 6. From the table we can see that the iteration counts decrease as the problem gets larger. We suspect this is due to the fact the the mesh size h is becoming small enough for mesh level $\ell \geq 3$ that the fluid and magnetic viscosities are correctly captured on these meshes.

6. Conclusions. We have introduced new block preconditioning techniques for the MHD model (1.1)–(1.2). Our goal was to develop block preconditioning approaches that utilize null space properties for the Maxwell equations and adapt them to the MHD model. Using the maximal nullity result in [9], we are able to find exact expressions for both a block Schur complement of the MHD coefficient matrix and its inverse. Using this Schur complement we have derived a block triangular preconditioner as well as an approximate inverse preconditioner.

The numerical experiments demonstrate the viability and effectiveness of both the approximate inverse and block triangular preconditioners. We present several numerical results which show strong scalability with respect to mesh, convex domains and/or large Hartmann numbers. Developing robust solvers for this problem is a highly challenging task, and we believe that our approach shows promise in terms of its ability to strongly scale and tackle large-scale 3D problems with high Hartmann numbers.

Further developments could include parallelization of the code. From the structure of the approximate inverse preconditioner in (3.6), we observe that each block column has some repetition for the sequence of systems which are solved. Therefore, one could apply each block column of (3.6) in parallel. This may greatly reduce the the overall computational time for the application of the inverse preconditioner to one or two vector solves per block column, which is comparable to the block triangular preconditioner.

Finally, one may choose to employ a different block matrix dropping strategy, which does not depend entirely on the approximate order with respect to the mesh size, h, but takes into account the nondimensional parameter setup (ν, ν_m or κ) of the problem. This, in turn, may provide a more specialized preconditioning approach based on the specific problem considered.

References.

[1] J. H. Adler, T. R. Benson, E. C. Cyr, S. P. MacLachlan, and R. S. Tuminaro. Monolithic multigrid methods for two-dimensional resistive magnetohydrodynamics. *SIAM Journal on Scientific Computing*, 38(1):B1–B24, 2016. ISSN 1064-8275. doi: 10.1137/151006135. URL https://doi.org/10.1137/151006135.

[2] F. Armero and J. C. Simo. Long-term dissipativity of time-stepping algorithms for an abstract evolution equation with applications to the incompressible MHD and Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 131(1):41–90, 1996. doi: 10.1016/0045-7825(95)00931-0. URL https://doi.org/10.1016/0045-7825(95)00931-0.

[3] S. Balay, M. F. Adams, J. Brown, P. Brune, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, K. Rupp, B. F. Smith, and H. Zhang. PETSc User's Manual. Technical Report ANL-95/11 - Revision 3.4, Argonne National Laboratory, 2013. URL http://www.mcs.anl.gov/petsc.

[4] S. M. Balay, S. Adams, J. Brown, P. Brune, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, K. Rupp, B. F. Smith, and H. Zhang. PETSc Web page. http://www.mcs.anl.gov/petsc, 2014. URL http://www.mcs.anl.gov/petsc.

[5] M. Benzi, G. Golub, and J. Liesen. Numerical solution of saddle point problems. *Acta Numerica*, 14:1–137, 2005. ISSN 1474-0508. doi: 10.1017/S0962492904000212. URL http://journals.cambridge.org/article_S0962492904000212.

[6] E. C. Cyr, J. N. Shadid, R. S. Tuminaro, R. P. Pawlowski, and L. Chacón. A new approximate block factorization preconditioner for two-dimensional incompressible (reduced) resistive MHD. *SIAM Journal on Scientific Computing*, 35(3):B701–B730, 2013. ISSN 1064-8275. doi: 10.1137/12088879X. URL https://doi.org/10.1137/12088879X.

[7] P. A. Davidson. *An Introduction to Magnetohydrodynamics*. Cambridge Texts in Applied Mathematics. Cambridge University Press, Cambridge, 2001. ISBN 0-521-79487-0. doi: 10.1017/CBO9780511626333. URL http://dx.doi.org/10.1017/CBO9780511626333.

[8] H. C. Elman, D. J. Silvester, and A. J. Wathen. *Finite Elements and Fast Iterative Solvers: with Applications in Incompressible Fluid Dynamics*. Oxford University Press, second edition, 2014. URL http://www.oxfordscholarship.com/view/10.1093/acprof:oso/9780199678792.001.0001/acprof-9780199678792.

[9] R. Estrin and C. Greif. On Nonsingular Saddle-point Systems with a Maximally Rank Deficient Leading Block. *SIAM Journal on Matrix Analysis and Applications*, 36(2): 367–384, 2015. doi: 10.1137/140989996. URL https://doi.org/10.1137/140989996.

[10] R. D. Falgout and U. Yang. *hypre*: A library of high performance preconditioners. In *Computational Science ICCS 2002*, volume 2331 of *Lecture Notes in Computer Science*, pages 632–641. Springer Berlin Heidelberg, 2002. ISBN 978-3-540-43594-5. doi: 10.1007/3-540-47789-6_66. URL http://dx.doi.org/10.1007/3-540-47789-6_66.

[11] J.-F. Gerbeau, C. Le. Bris, and T. Lelièvre. *Mathematical Methods for the Magnetohydrodynamics of Liquid Metals*. Oxford University Press, 2006. URL http://www.oxfordscholarship.com/view/10.1093/acprof:oso/9780198566656.001.0001/acprof-9780198566656.

[12] C. Greif and D. Schötzau. Preconditioners for the discretized time-harmonic Maxwell equations in mixed form. *Numerical Linear Algebra with Applications*, 14(4):281–297, 2007. doi: 10.1002/nla.515. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/nla.515.

[13] R. Hiptmair and J. Xu. Nodal auxiliary space preconditioning in $H(\mathrm{curl})$ and $H(\mathrm{div})$ spaces. *SIAM Journal on Numerical Analysis*, 45(6):2483–2509, 2007. doi: 10.1137/060660588. URL https://doi.org/10.1137/060660588.

[14] I. Ipsen. A note on preconditioning nonsymmetric matrices. *SIAM Journal on Scientific Computing*, 23(3):1050–1051, 2001. doi: 10.1137/S1064827500377435. URL https://doi.org/10.1137/S1064827500377435.

[15] D. Li. *Numerical Solution of the Time-Harmonic Maxwell equations and Incompressible Magnetohydrodynamics Problems*. PhD thesis, University of British Columbia, 2010. URL https://open.library.ubc.ca/collections/ubctheses/24/items/1.0051989.

[16] A. Logg, K. A. Mardal, and G. N. Wells, editors. *Automated solution of differential equations by the finite element method*, volume 84 of *Lecture Notes in Computational Science and Engineering*. Springer, Heidelberg, 2012. ISBN 978-3-642-23098-1; 978-3-642-23099-8. doi: 10.1007/978-3-642-23099-8. URL http://dx.doi.org/10.1007/978-3-642-23099-8.

[17] U. Müller and L. Bühler. *Magnetofluiddynamics in Channels and Containers*. Springer, 2001. doi: 10.1007/978-3-662-04405-6. URL https://doi.org/10.1007/978-3-662-04405-6.

[18] M. F. Murphy, G. H. Golub, and A. J. Wathen. A note on preconditioning for indefinite linear systems. *SIAM Journal on Scientific Computing*, 21(6):1969–1972, 2000. doi: 10.1137/S1064827599355153. URL https://doi.org/10.1137/S1064827599355153.

[19] J. C. Nédélec. Mixed finite elements in $\mathbb{R}^3$. *Numerische Mathematik*, 35(3):315–341, 1980. doi: 10.1007/BF01396415. URL https://doi.org/10.1007/BF01396415.

[20] E. G. Phillips, H. C. Elman, E. C. Cyr, J. N. Shadid, and R. P. Pawlowski. A block preconditioner for an exact penalty formulation for stationary MHD. *SIAM Journal on Scientific Computing*, 36(6):B930–B951, 2014. doi: 10.1137/140955082. URL http://dx.doi.org/10.1137/140955082.

[21] E. G. Phillips, H. C. Elman, E. C. Cyr, J. N. Shadid, and R. P. Pawlowski. Block Pre-

conditioners for Stable Mixed Nodal and Edge finite element Representations of Incompressible Resistive MHD. *SIAM Journal on Scientific Computing*, 38(6):B1009–B1031, 2016. doi: 10.1137/16M1074084. URL https://doi.org/10.1137/16M1074084.

[22] Y. Saad. A flexible inner-outer preconditioned GMRES algorithm. *SIAM Journal on Scientific Computing*, 14(2):461–469, 1993. doi: 10.1137/0914028. URL https://doi.org/10.1137/0914028.

[23] C. Taylor and P. Hood. A numerical solution of the Navier-Stokes equations using the finite element technique. *Computers and Fluids*, 1(1):73–100, 1973. doi: https://doi.org/10.1016/0045-7930(73)90027-3. URL http://www.sciencedirect.com/science/article/pii/0045793073900273.

[24] M. Wathen. Iterative Solution of a Mixed Finite Element Discretisation of an Incompressible Magnetohydrodynamics Problem. Master's thesis, University of British Columbia, 2014. URL https://open.library.ubc.ca/cIRcle/collections/ubctheses/24/items/1.0135538.

[25] M. Wathen. *Preconditioners for incompressible magnetohydrodynamics*. PhD thesis, University of British Columbia, 2018. URL https://open.library.ubc.ca/cIRcle/collections/ubctheses/24/items/1.0375762.

[26] M. Wathen, C. Greif, and D. Schötzau. Preconditioners for Mixed Finite Element Discretizations of Incompressible MHD Equations. *SIAM Journal on Scientific Computing*, 39(6):A2993–A3013, 2017. ISSN 1064-8275. doi: 10.1137/16M1098991. URL https://doi.org/10.1137/16M1098991.