Electronic Transactions on Numerical Analysis.
Volume 55, pp. 365–390, 2022.
Copyright © 2022, Kent State University.
ISSN 1068–9613.
DOI: 10.1553/etna_vol55s365

# A MONOLITHIC ALGEBRAIC MULTIGRID FRAMEWORK FOR MULTIPHYSICS APPLICATIONS WITH EXAMPLES FROM RESISTIVE MHD[*]

PETER OHM[†], TOBIAS A. WIESNER[‡], ERIC C. CYR[†], JONATHAN J. HU[§], JOHN N. SHADID[†][¶], AND RAYMOND S. TUMINARO[§]

**Abstract.** We consider monolithic algebraic multigrid (AMG) algorithms for the solution of block linear systems arising from multiphysics simulations. While the multigrid idea is applied directly to the entire linear system, AMG operators are constructed by leveraging the matrix block structure. In particular, each block corresponds to a set of physical unknowns and physical equations. Multigrid components are constructed by first applying existing AMG procedures to matrix sub-blocks. The resulting AMG sub-components are then composed together to define a monolithic AMG preconditioner. Given the problem-dependent nature of multiphysics systems, different blocking choices may work best in different situations, and so software flexibility is essential. We apply different blocking strategies to systems arising from resistive magnetohydrodynamics in order to demonstrate the associated trade-offs.

**Key words.** multigrid, algebraic multigrid, multiphysics, magnetohydrodynamics

**AMS subject classifications.** 68Q25, 68R10, 68U05

## 1. Introduction and motivation.

Multigrid methods are among the fastest and most scalable techniques for solving linear systems that arise from many discretized partial differential equation (PDE) systems [55]. The multigrid idea is to accelerate convergence by performing relaxation (i.e., simple iterative schemes) on a hierarchy of different resolution systems. Algebraic multigrid methods (AMG) are popular as they build the hierarchy automatically, requiring little effort from the application developer. While algebraic multigrid methods have seen many successes, further developments are needed to more robustly adapt them to multiphysics PDE systems.

Multigrid's rapid convergence relies on constructing its components such that relaxation sweeps applied to the different fidelity discrete representations are complementary to each other. That is, errors not easily damped by relaxation on one discrete representation can be damped effectively by relaxation on another discrete representation. However, constructing AMG components with desirable complementary properties can be complicated for PDE systems. A discussion of AMG issues for PDE systems and some different approaches can be found in [23]. The idea of blocking is an important theme for multiphysics systems, where each block linear system corresponds to separate sets of physical unknowns and equations. Numerous block preconditioning strategies have been proposed such as those based on block

[*] Received April 27, 2021. Accepted December 29, 2021. Published online on March 1, 2022. Recommended by Ulrich Langer.
*Funding:* This work was supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Applied Mathematics program and by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research and Office of Fusion Energy Sciences, Scientific Discovery through Advanced Computing (SciDAC) program. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under grant DE-NA-0003525. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

[†] Sandia National Laboratories, P.O. Box 5800, MS 1320, Albuquerque, NM 87185
({pohm, eccyr, jnshadi}@sandia.gov).
[‡] Leica Geosystems AG, Heinrich-Wild-Strasse 201, 9435 Heerbrugg/SG, Switzerland
(tobias@tawiesn.de).
[§] Sandia National Laboratories, P.O. Box 969, MS 9159, Livermore, CA 94661
({jhu, rstumin}@sandia.gov).
[¶] Department of Mathematics and Statistics, University of New Mexico, Albuquerque NM, 87123.

factorizations, which manipulate the constituent Jacobian blocks and use inner solvers on simpler problems. We instead focus on monolithic AMG but still consider blocking ideas when developing the multigrid components. While a monolithic approach applies AMG directly to the entire linear system (as opposed to applying multigrid to sub-linear systems), block-oriented algorithms can be used to construct AMG components such as the relaxation methods and the grid transfer operators. In this way, one can leverage existing algorithms and software developed for simpler sub-systems that can then be combined or composed together to address more complex problems. In fact, it is possible to construct a monolithic multigrid method even when different grids are used for separate components. Given the problem-dependent nature of different multiphysics systems, one may need to consider different blocking schemes for different PDE systems, and so it is essential that solvers provide flexible mechanisms for defining and manipulating blocks in order to tailor the AMG strategy to specific situations.

We demonstrate the importance of a flexible blocking scheme in the context of solving difficult linear systems associated with resistive magnetohydrodynamics (MHD). The governing partial differential equations consist of conservation of mass, momentum, and energy augmented by the low-frequency Maxwell's equations and are often highly ill-conditioned [9, 25, 32, 47, 48]. Depending on the particular MHD scenario, different solver adaptations might be appropriate. Specifically, we highlight blocking AMG ideas using the MueLu package [3, 4] (found within Trilinos [29]), which facilitates different block strategies. In one case, a special block ILU relaxation method is devised that has significant computational advantages over a more black-box relaxation technique. Here, an ILU factorization is applied separately to the Navier-Stokes block and to the magnetics block as opposed to applying the ILU factorization to the entire system. This effectively ignores the coupling between the Navier-Stokes and the magnetics equations during the incomplete factorization, which leads to a significant reduction in the time required to actually perform the ILU factorization. The overall monolithic smoother couples the physics together with a Gauss-Seidel iteration. In another case, we show how to adapt the solver to address situations where different finite element basis functions are used to represent the different physical fields of the MHD system. Specifically, one AMG algorithm is applied to a Q2 Navier-Stokes block while another AMG algorithm is applied to a Q1 magnetics block, and the results of the two invocations are then composed together to define the interpolation for the entire MHD system. In this case, the blocking choice avoids a limitation in applying the existing AMG algorithm/software to a PDE system where the number of degrees of freedom per spatial location is not constant. That is, the blocking allows us to leverage algorithms and solve problems that could not be previously addressed with the existing algorithms. In the future, we plan to expand upon the current block strategies and consider different AMG schemes for pressures, velocities, magnetics, and the magnetics' Lagrange multipliers. This will allow us to employ more sophisticated interpolation algorithms for only the pressure and the Lagrange multipliers, where simpler schemes can adversely affect the convergence rate.

In Section 2 we introduce the MHD equations and the accompanying discrete systems of equations. In Section 3 we give an overview of the algebraic multigrid method. We discuss the implementation of these algebraic multigrid methods to multiphysics PDE systems in a truly monolithic manner through the use of blocked operators in Section 4. We demonstrate the numerical and computational performance benefits of this approach for various test problems and present the results in Section 5. Finally, we end with concluding remarks in Section 6.

## 2. The MHD equations.

The model of interest for this paper is the 3D resistive isothermal MHD equations including dissipative terms for the momentum and the magnetic induction equations [25, 48]. This model provides a base-level continuum description of conducting fluids in the presence of electromagnetic fields and is useful in the context of mod-

TABLE 2.1
*Residual form of the governing resistive 3D MHD equations.*

Momentum

$$\mathbf{r}_m = \frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot [\rho \mathbf{u} \otimes \mathbf{u} - \frac{1}{\mu_0} \mathbf{B} \otimes \mathbf{B} + (p + \frac{1}{2\mu_0} \|\mathbf{B}\|^2)\mathbf{I} - \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)] = \mathbf{0}$$

Continuity Constraint

$$r_P = \frac{\partial \rho}{\partial t} + \nabla \cdot \rho \mathbf{u} = 0$$

Magnetic Induction

$$\mathbf{r}_I = \frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot [\mathbf{u} \otimes \mathbf{B} - \mathbf{B} \otimes \mathbf{u} - \frac{\eta}{\mu_0} \left( \nabla \mathbf{B} - (\nabla \mathbf{B})^T \right) + \psi \mathbf{I}] = \mathbf{0}$$

Solenoidal Constraint

$$r_\psi = \nabla \cdot \mathbf{B} = 0$$

eling naturally occurring plasma physics systems (e.g., astrophysics and planetary dynamos) as well as for technology (e.g., magnetic confinement fusion). The system of equations is shown in Table 2.1 in residual form. The primitive variables are the velocity vector $\mathbf{u}$, the hydrodynamic pressure $p$, the magnetic induction $\mathbf{B}$ (hereafter also termed the magnetic field), and the Lagrange multiplier $\psi$. The associated plasma current $\mathbf{J}$ is obtained from Ampère's law as $\mathbf{J} = \frac{1}{\mu_0} \nabla \times \mathbf{B}$.

Satisfying the solenoidal involution $\nabla \cdot \mathbf{B} = 0$ in the discrete representation to machine precision is a topic of considerable interest in both structured and unstructured finite-volume and unstructured finite-element contexts (see, e.g., [8, 15, 53]). In the formulation discussed in this study, a scalar Lagrange multiplier ($\psi$ in Table 2.1) is introduced into the induction equation that enforces the solenoidal involution as a divergence-free constraint for the magnetic field [2, 8, 10, 11, 15, 48, 53]. This procedure is common in both the finite volume (see, e.g., [8, 15, 53]) and in finite element methods (see, e.g., [10, 11]). We focus on the incompressible limit of this system, i.e., $\nabla \cdot \mathbf{u} = 0$. This limit is useful to model applications such as low-Lundquist-number liquid-metal MHD flows [14, 39] and high-Lundquist-number large-guide-field fusion plasmas [18, 28, 52]. Together, the incompressibility constraint for the fluid velocity and the solenoid involution for the magnetic field (enforced as a constraint) produce a dual saddle point structure for the systems of equations [48].

The spatial discretization is based on the variational multiscale (VMS) finite element (FE) method [17, 30]. The semi-discretized system is integrated in time with a method-of-lines approach based on BDF schemes. The weak form of the VMS / stabilized FE formulation for the resistive MHD equation (see equation (2.1)) is given by

$$\text{(2.1a)} \quad \mathbf{F}_{\mathbf{u}}^h = \int_\Omega \mathbf{w}^h \cdot \mathbf{r}_m^h d\Omega + \sum_e \int_{\Omega_e} \rho \hat{\tau}_m \mathbf{r}_m^h \otimes \mathbf{u}^h : \nabla \mathbf{w}^h d\Omega + \sum_e \int_{\Omega_e} \hat{\tau}_P (\nabla \cdot \mathbf{w}^h) \mathrm{r}_P^h \, d\Omega,$$

$$\text{(2.1b)} \quad F_P^h = \int_\Omega q^h \mathrm{r}_P^h d\Omega + \sum_e \int_{\Omega_e} \rho \hat{\tau}_m \nabla q^h \cdot \mathbf{r}_m^h d\Omega,$$

$$\text{(2.1c)} \quad \mathbf{F}_I^h = \int_\Omega \mathbf{C}^h \cdot \mathbf{r}_I^h d\Omega - \sum_e \int_{\Omega_e} \hat{\tau}_I (\mathbf{r}_I^h \otimes \mathbf{u}^h - \mathbf{u}^h \otimes \mathbf{r}_I^h) : \nabla \mathbf{C}^h \, d\Omega + \sum_e \int_{\Omega_e} \hat{\tau}_\psi (\nabla \cdot \mathbf{C}^h) \mathrm{r}_\psi^h d\Omega,$$

$$F_{\psi}^{h} = \int_{\Omega} s^{h} \mathrm{r}_{\psi}^{h} d\Omega + \sum_{e} \int_{\Omega_e} \hat{\tau}_I \nabla s^{h} \cdot \mathbf{r}_I^{h} d\Omega, \tag{2.1d}$$

where $\hat{\tau}_i$ are the stabilization parameters. Here $[\mathbf{w}^h, q^h, \mathbf{C}^h, s^h]$ are the FE weighting functions for the velocity, pressure, magnetic field, and the Lagrange multiplier, respectively. The sum $\sum_e$ indicates that the integrals are taken only over element interiors $\Omega_e$ and that integration by parts is not performed. A full development and examination of this formulation is presented in [48].

**2.1. The Block discrete system.** A FE discretization of the stabilized equations gives rise to a system of coupled, nonlinear, non-symmetric algebraic equations, the numerical solution of which can be very challenging. At each stage of a Newton's method iteration, the discrete linearized block system has the following form:

$$\begin{bmatrix} \mathbf{J_u} & \mathbf{G} & \mathbf{Z} & \mathbf{0} \\ \mathbf{D} & \mathbf{L_P} & \mathbf{0} & \mathbf{0} \\ \mathbf{Y} & \mathbf{0} & \mathbf{J_I} & \mathbf{G} \\ \mathbf{0} & \mathbf{0} & \mathbf{G^T} & \mathbf{L}_{\psi} \end{bmatrix} \begin{bmatrix} \delta\hat{\mathbf{u}} \\ \delta\hat{p} \\ \delta\hat{\mathbf{B}} \\ \delta\hat{\psi} \end{bmatrix} = - \begin{bmatrix} \mathbf{r_u} \\ \mathrm{r}_p \\ \mathbf{r}_I \\ \mathrm{r}_{\psi} \end{bmatrix}. \tag{2.2}$$

Here the block matrix $\mathbf{J_u}$ corresponds to the discrete transient, convection, diffusion, and stress terms acting on the unknowns $\delta\hat{\mathbf{u}}$. The matrix $\mathbf{G}$ corresponds to the discrete gradient operator, $\mathbf{D}$ corresponds to the discrete representation of the continuity equation terms with velocity (note for a true incompressible flow this would be the divergence operator denoted as $\mathbf{G^T}$), the block matrix $\mathbf{J_I}$ corresponds to the discrete transient, convection, diffusion terms acting on the magnetic induction, and the matrices $\mathbf{L_P}, \mathbf{L}_{\psi}$ are stabilization Laplacians, which are described in the next paragraph.

A closer examination of the VMS terms generated by the induction equation and in the enforcement of the solenoidal constraint through the Lagrange multiplier $\psi$ exhibit the presence of a weak Laplacian operator acting on the Lagrange multiplier

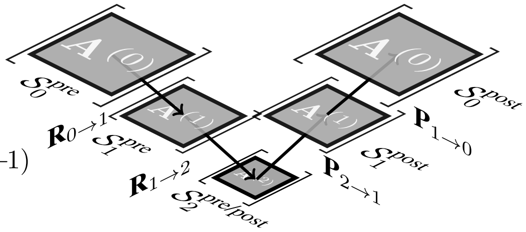
$$L_{\psi} = \sum_{e} \int_{\Omega_e} \hat{\tau}_I \nabla s \cdot \nabla \psi d\Omega.$$

This term is the analogue of the weak pressure Laplacian $L_P = \sum_e \int_{\Omega_e} \rho \hat{\tau}_m \nabla \Phi \cdot \nabla p d\Omega$ appearing in the total mass continuity equation (see the general discussion for stabilized FE CFD in [17] and [47, 48] for MHD). These VMS operators are critical in the elimination of oscillatory modes from the null space of the resistive MHD saddle point system for both $(\mathbf{u}, p)$ and $(\mathbf{B}, \psi)$ and allow equal-order interpolation of all the unknowns (see [17] for incompressible CFD and [2, 10, 11] for resistive MHD). The $\mathbf{L_P}$ and $\mathbf{L}_{\psi}$ operators also help facilitate the solution of the linear systems with a number of algebraic and domain decomposition-type preconditioners that rely on non-pivoting ILU factorization, Jacobi relaxation, or Gauss-Seidel iterations as sub-domain solvers [46, 48, 49].

The difficulty of producing robust and efficient preconditioners for (2.2) has motivated many different types of decoupled solution methods. Often, transient schemes combine semi-implicit methods with fractional-step (operator splitting) approaches or use fully-decoupled solution strategies [1, 26, 27, 31, 33, 35, 41, 43, 44, 51, 54]. In these cases, the motivation is to reduce memory usage and to produce a simplified equation set for which efficient solution strategies already exist. Unfortunately, these simplifications place significant limitations on the broad applicability of these methods. A detailed presentation of the characteristics of different linear and nonlinear solution strategies is beyond our current scope. Here, we wish to highlight that our approach of fully coupling the resistive MHD PDEs in the nonlinear solver

FIG. 3.1. *Multigrid V-cycle algorithm and a graphical representation for a 3-level method.*

```
MGV$(u_\ell, b_\ell, \ell)$ :
  if $\ell \neq \ell_{max}$
    $u_\ell \leftarrow \mathcal{S}_\ell^{\text{pre}}(\mathbf{A}_\ell, u_\ell, b_\ell)$
    $r_\ell \leftarrow b_\ell - \mathbf{A}_\ell u_\ell$
    $u_{\ell+1} \leftarrow 0$
    $u_{\ell+1} \leftarrow$ MGV$(u_{\ell+1}, \mathbf{R}_{\ell\to\ell+1} r_\ell, \ell{+}1)$
    $u_\ell \leftarrow u_\ell + \mathbf{P}_{\ell+1\to\ell} u_{\ell+1}$
    $u_\ell \leftarrow \mathcal{S}_\ell^{\text{post}}(\mathbf{A}_\ell, u_\ell, b_\ell)$
  else
    $u_\ell \leftarrow \mathbf{A}_\ell^{-1} b_\ell$
```



preserves the inherently strong coupling of the physics with the goal to produce a more robust solution methodology [46, 47, 48]. Preservation of this strong coupling, however, places a significant burden on the linear solution procedure.

## 3. Multigrid methods.

Multigrid methods leverage the fact that many simple iterative methods can effectively eliminate high-frequency error components relative to the "mesh" resolution used for discretization. That is, different error components are efficiently reduced by essentially applying a simple iterative method to the appropriate resolution approximation.

Multigrid methods generally come in two varieties: geometric multigrid (GMG) and algebraic multigrid (AMG). GMG grid transfers are based on geometric relationships such as a linear interpolation to transfer coarse solutions to finer meshes. With AMG, coarse-level information and grid transfers are developed automatically by analyzing the supplied fine-level discretization matrix. This normally involves a combination of graph heuristics for coarsening followed by some approximation algorithm to develop operators that accurately transfer information between the meshes. This paper focuses on AMG as it can be more easily adapted to complex application domains by non-multigrid scientists.

Figure 3.1 depicts what is referred to as a multigrid V-cycle to solve a linear system $\mathbf{A}_\ell u_\ell = b_\ell$. Subscripts distinguish between different resolutions. $\mathbf{P}_{\ell+1\to\ell}$ interpolates from level $\ell{+}1$ to level $\ell$. $\mathbf{R}_{\ell\to\ell+1}$ restricts from level $\ell$ to level $\ell{+}1$. $\mathbf{A}_\ell$ is the discrete problem on level $\ell$, and for coarse levels it is defined by a Petrov-Galerkin projection

$$\mathbf{A}_{\ell+1} = \mathbf{R}_{\ell\to\ell+1}\mathbf{A}_\ell\mathbf{P}_{\ell+1\to\ell}.$$

$\mathcal{S}_\ell^{\text{pre}}$ and $\mathcal{S}_\ell^{\text{post}}$ denote a basic iterative scheme (e.g., a Gauss-Seidel iteration) that is applied to damp or relax some error components. The overall efficiency is governed by the interplay of the two main multigrid ingredients: grid transfer operators and the smoothing methods. For a general overview on multigrid methods, the reader is referred to [7, 55] and the references therein.

For PDE and multiphysics systems, applying AMG to the entire PDE system (i.e., monolithic multigrid) can be problematic or even impossible for a number of reasons, especially when the coupling between different solution types (e.g., pressures and velocities) is strong. Classical simple iterative methods may not necessarily reduce all oscillatory error components and might even amplify some oscillatory error components. In fact, methods requiring the inversion of the matrix diagonal (e.g., the Jacobi iteration) are not even well defined when applied to incompressible fluid discretizations that give rise to zeros on the matrix diagonal. Further, many standard AMG algorithms for defining grid transfers might lead to transfers that do not accurately preserve smooth functions. For example, methods such as smoothed aggregation rely on a Jacobi-like step to generate smooth grid transfer basis functions. This Jacobi step, however, will obviously not generate smoother basis functions when the iteration is not well defined (or when the matrix diagonal is *small* in a relative sense). The AMG software itself may not even be applicable to the PDE system when different FE basis functions are used to represent different fields within the system. For PDE systems, most AMG codes assume that the different equations are discretized in a similar way. In particular, AMG coarsening algorithms (or for our approach *aggregation* algorithms) are often applied by first grouping all degrees of freedom (DoFs) at each mesh node together and then applying the coarsening algorithms to the graph induced from the block matrix. This has the advantage that all unknowns at a mesh point are coarsened in the same fashion. However, the approach breaks down when the number of DoFs associated with different fields varies or if all DoFs are not co-located, e.g., when using quadratic basis functions to represent velocities while pressures employ only linear basis functions. In previous versions of our AMG software, the only alternative to this simple PDE system technique would be to completely ignore the multiphysics coupling and effectively treat the entire system as if it were a scalar PDE, which almost always leads to poor convergence.

## 4. Truly monolithic block multigrid for the MHD equations.

In this section we propose a multigrid method for multiphysics systems that can be represented by block matrices allowing one to adapt both the relaxation algorithms and the grid transfer construction algorithms to the structure of the system.

### 4.1. Block matrices and multigrid for PDE systems.

As illustrated in the previous MHD discussion, PDE systems are often represented by block matrices as in, for example, equation (2.2). More generally, block systems can be written as

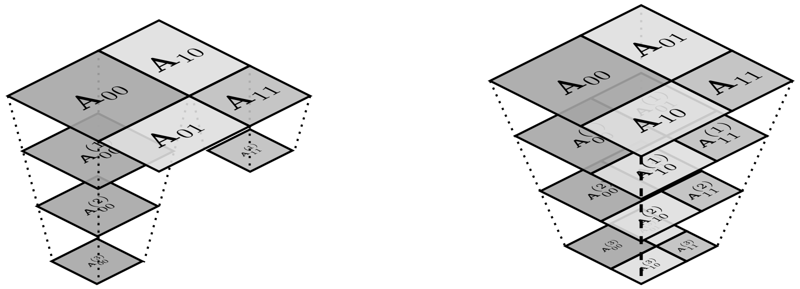
$$\begin{bmatrix} A_{00} & A_{01} & \cdots & A_{0N} \\ A_{10} & A_{11} & \cdots & A_{1N} \\ \vdots & \vdots & \ddots & \vdots \\ A_{N0} & A_{N1} & \cdots & A_{NN} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_N \end{bmatrix} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_N \end{bmatrix}, \tag{4.1}$$

where each component of the vector $x_i$ is a field in the multiphysics PDE and the sub-matrices $A_{ij}$ are approximations of operators in the governing equations.

One preconditioning approach to PDE systems follows a so-called *physics-based* strategy (see Figure 4.1a). These techniques can be viewed as approximate block factorizations (involving Schur complement approximations) to the block matrix equations (4.1). The Schur complement arises naturally during a block LU factorization of a linear system, in which the block upper triangular factor has a block diagonal entry referred to as the Schur complement. The exact Schur complement involves a matrix inverse, which is not feasible for large sparse systems. In such cases, the inverse is typically approximated, e.g., with the application of an iterative solve. More detailed information can be found, for example, in [60]. The factorizations are usually constructed based on the underlying physics. Here, different AMG V-cycle sweeps are used to approximate the different sub-matrix inverses that appear within the approximate block factors. As the sub-matrices correspond to single physics or scalar PDE operators, application-specific modifications to the multigrid algorithm are often not necessary. This makes the physics-based strategy particularly easy to implement as one can leverage ready-to-use multigrid packages. Several methods that follow this popular strategy are described in [21] and the references therein. While the physics-based approach has some practical advantages, the efficiency of the preconditioner relies heavily on how well the coupling between different equations within the PDE is approximated by the block factorization.

In this paper, we instead consider a monolithic multigrid alternative to a physics-based approach. A monolithic scheme applies a multigrid algorithm to the entire block PDE system, and so the AMG scheme effectively develops a hierarchy of block PDE matrices associated with different resolutions. Figure 4.1 contrasts the two approaches and highlights the monolithic approach's key potential advantage, namely an explicit representation of the cross-coupling defined by $\mathbf{A}_{01}$ and $\mathbf{A}_{10}$ on all multigrid hierarchy levels. In contrast, a physics-based scheme must represent the coupling by developing an effective Schur complement (whose inverse might then be approximated via multigrid sweeps), which can be non-trivial for complex applications. This is because AMG is only applied to sub-components of the entire block system within a physics-based scheme, and hence cross-coupling terms are not explicitly projected to coarser levels. On the other hand, a monolithic approach introduces its own set of application-specific mathematical challenges such as the construction of relaxation procedures for monolithic systems and the development of coarsening schemes for different fields within a multiphysics system. The design of efficient multiphysics preconditioners often requires one to make use of the specific knowledge about the block structure, the mathematical models of the underlying physics, and the problem-specific coupling of the equations. The mathematical challenges of multiphysics systems are often further compounded by non-trivial software challenges. Unfortunately, most AMG packages cannot be customized to particular multi-physics scenarios without having an in-depth knowledge of the AMG software.

FIG. 4.1. *Two multigrid approaches to address multiphysics applications.*



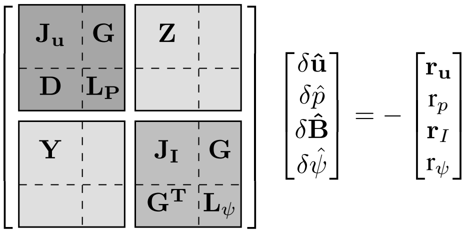
(a) Physics-based approach with multiple multigrid approximations to the sub-matrix inverses within an approximate block factorization. Example with 4 and 2 multigrid levels for $\mathbf{A}_{00}$ and $\mathbf{A}_{11}$.

(b) Monolithic multigrid with PDE coupling represented on all multigrid levels.

In the next sections, we propose a monolithic algorithm for the MHD equations that can be easily adapted and customized using the MueLu package within the Trilinos framework [3, 4]. Though we focus on a concrete MHD case, we emphasize the importance of the software's generality in enabling a monolithic approach for those with limited knowledge of the multigrid package internals.

**4.2. Algebraic representation of the MHD problem.** In adapting a monolithic multigrid strategy to the MHD equations, a natural approach would be to interpret the system (2.2) as a $2 \times 2$ block system where the Navier-Stokes equations are separated from the Maxwell equations. That is, we treat the Navier-Stokes part and the Maxwell part as separate entities that are coupled by the off-diagonal blocks as shown in the $2 \times 2$ block representation of Figure 4.2. In this way, we can leverage existing ideas/solvers for the Navier-Stokes equations

FIG. 4.2. *Representation of the MHD system* (2.2) *as a* $2 \times 2$ *block system.*



and for the Maxwell equations. In making this $2 \times 2$ decomposition, we are effectively emphasizing the significance of the coupling between fields within the Navier-Stokes block and within the Maxwell block as compared to the coupling between Navier-Stokes unknowns and Maxwell unknowns. This is due to the importance of the coupling constraint equations (e.g., incompressibility conditions involving velocities or contact constraints [58]) to the associated evolution equations. These constraint equations often give rise to saddle-point-like block systems. Of course, there are physical situations where the coupling between the Navier-Stokes equations and the Maxwell equations is quite significant, and so a block $2 \times 2$ decomposition might be less appropriate. There might also be situations where coupling relationships are more complex. If, for example, the MHD equations are embedded in another larger more complex multiphysics problem, then one might need to consider a hierarchy of coupling configurations, which might require different arrangements/blockings of multigrid ingredients. Given the problem-specific nature of multiphysics preconditioning, our emphasis here is on the importance of a flexible software framework to facilitate different types of blocking within the preconditioner.

For the remainder of the paper the block notation

$$\begin{bmatrix} \mathbf{A}_{00} & \mathbf{A}_{01} \\ \mathbf{A}_{10} & \mathbf{A}_{11} \end{bmatrix} \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{x}_1 \end{bmatrix} = - \begin{bmatrix} \mathbf{b}_0 \\ \mathbf{b}_1 \end{bmatrix} \tag{4.2}$$

is used to represent the corresponding blocks from Figure 4.2. The velocity and pressure increments $\delta\hat{\mathbf{u}}$ and $\delta\hat{\mathbf{p}}$ are grouped in $\mathbf{x}_0$, and the Maxwell information is represented by $\mathbf{x}_1$. In a similar way, the block notation for the residual vector is adopted.

**4.3. Monolithic multigrid ingredients for volume-coupled problems.** The multiphysics solver that we propose is generally applicable to *volume-coupled* problems. Volume coupled means that the different physics blocks are defined on the same domain. Volume-coupled examples include thermo-structure-interaction (TSI) problems (see [13]) or in our case the MHD equations. Specifically, within our MHD formulation, all physics equations (i.e., Navier-Stokes and Maxwell parts) are defined throughout the entire domain. This is in contrast to interface-coupled problems such as fluid-structure interaction (FSI) applications (see [16, 24, 34, 37]) or structural contact problems (see [58, 59]), where different equation sets are valid over distinct domains that are only coupled through a common interface. From a multigrid perspective, special interface coarsening methods are necessary for interface-coupled applications, which is not the focus of this paper.

Two multigrid ingredients must be specified to fully define the monolithic solver: the inter-grid transfer operators and the relaxation or smoother procedures.

**4.3.1. Inter-grid transfers for the MHD system.** Following the $2 \times 2$ decomposition of the MHD system from equation (4.2), we consider rectangular block diagonal inter-grid transfer operators

$$\mathbf{R}_{i\rightarrow i+1} = \begin{bmatrix} \mathbf{R}_{00} & \\ & \mathbf{R}_{11} \end{bmatrix} \qquad \text{and} \qquad \mathbf{P}_{i+1\rightarrow i} = \begin{bmatrix} \mathbf{P}_{00} & \\ & \mathbf{P}_{11} \end{bmatrix}$$

for the restriction and prolongation between multigrid levels, respectively. The basic idea uses the MueLu multigrid package to produce grid transfers for the Navier-Stokes equations and the Maxwell equations and then leverages MueLu's flexibility to combine these grid transfer operators into a composite block diagonal operator.

The block perspective allows us first to build individual components (e.g., $\mathbf{P}_{00}$ and $\mathbf{P}_{11}$) with completely separate invocations of MueLu and then combine or compose them together. As discussed earlier, we rely on underlying core AMG kernels that are applicable to either a scalar PDE or a PDE system with co-located unknowns. Mixed finite element schemes may not normally satisfy this co-located requirement, so the ability to separately invoke these core components to produce $\mathbf{P}_{00}$ and $\mathbf{P}_{11}$ alleviates this restriction, allowing us to apply monolithic AMG to a wider class of PDE systems. That is, a mixed basis function discretization can be approached without having to erroneously treat the entire system as a scalar PDE. We demonstrate this capability at the end of Section 5 through a mixed formulation utilizing Q2/Q2 VMS for the Navier-Stokes degrees of freedom and Q1/Q1 VMS for the Maxwell degrees of freedom.

In the case where the unknowns between blocks are co-located (e.g., Q1/Q1 VMS for both the Navier-Stokes and Maxwell system), we have the option to correlate the grid transfer construction by having the multigrid invocations share the same aggregates (or coarsening definition) as depicted in Figure 4.3. In this way, we guarantee that there is a one-to-one relationship of the coarse Maxwell degrees of freedoms and the associated Navier-Stokes degrees of freedom. That is, we obtain the same coarsening rate for both, and so the ratio between Navier-Stokes and Maxwell degrees of freedom is constant on all multigrid levels. It should be noted that this is relatively straightforward when all equations are defined on the same mesh using a first-order nodal finite element discretization method. Thus, there are 8 degrees of freedom at each mesh node (four associated with the fluid flow and four associated with electromagnetics).

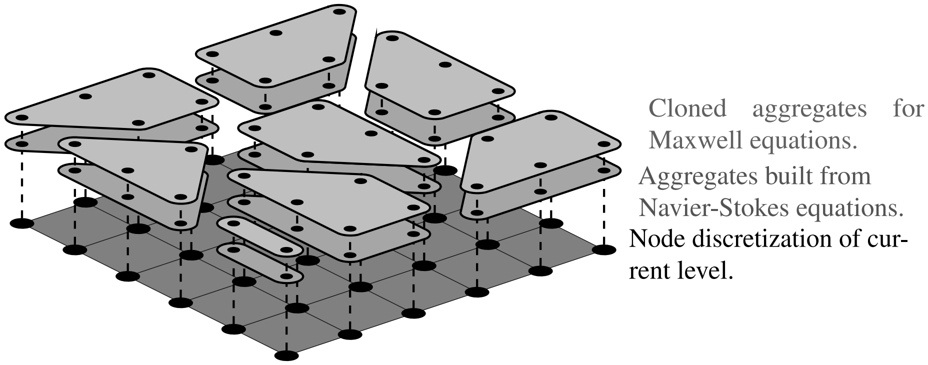
Our approach uses aggregation-based multigrid. In contrast to classic multigrid, in which a subset of fine-level unknowns are selected as coarse-grid unknowns, aggregation-based approaches group fine-level unknowns into *aggregates* that form the coarse-grid unknowns. Specifically, mesh vertices on a given level are assigned to *aggregates* $\mathcal{A}_\ell^i$ such that

$$\bigcup_{i=1}^{N_{\ell+1}} \mathcal{A}_\ell^i = \{1, ..., N_\ell\}, \quad \mathcal{A}_\ell^i \cap \mathcal{A}_\ell^j = \emptyset, \qquad 1 \le i < j \le N_{\ell+1},$$

where $N_\ell$ denotes the number of mesh vertices on level $\ell$. Each aggregate $\mathcal{A}_\ell^i$ on level $\ell$ gives rise to one node on level $\ell+1$. The $\mathcal{A}_\ell^i$ are formed by applying greedy algorithms to the graph associated with a matrix discretization. Specifically, an unaggregated unknown that has no direct connections via the matrix graph to unknowns already in an existing aggregate is chosen as an aggregate *root*. Unknowns that are directly connected to the root via the matrix graph are grouped with the root to form an aggregate. There are a number of heuristics that may be used to ignore certain graph connections, based on an analysis involving matrix entries. Such heuristics are necessary for handling anisotropies in the PDE and/or the underlying mesh. After the initial greedy sweep phase to assign unknowns to aggregates, there are later phases that place all remaining unassigned unknowns into existing and/or new aggregates. Typically, one wants aggregates to be approximately the same size and roughly spherical in shape (for isotropic problems). Piecewise-constant interpolation can then be defined over each aggregate for each solution component.

Within the current approach, we first build aggregates using the matrix graph of the sub-block $\mathbf{A}_{00}$ on level $i$ as input. Next, we build separate aggregates based on the graph of the sub-block $\mathbf{A}_{11}$ or instead clone the Navier-Stokes aggregates when degrees of freedom are co-located as depicted in Figure 4.3.

FIG. 4.3. *Cloned aggregation strategy for volume-coupled multiphysics problems.*


Cloned aggregates for Maxwell equations.
Aggregates built from Navier-Stokes equations.
Node discretization of current level.

As discussed later, this type of multigrid adaptation is relatively straightforward with MueLu using an XML input file. While cloning is often not essential for many problems, our plan is to enhance the cloning capability for mixed finite element systems where unknowns between blocks are not co-located. This enhancement would allow for some correlation between the aggregates of the different blocks, though they will no longer be identical. We have found that correlated-aggregation can help avoid stability issues that might arise with the AMG-generated coarse discretizations when the finest-level discrete operator corresponds to a saddle point system that does not employ stabilized finite elements. In addition to consistently coarsening both fluid and electromagnetic variables, the reuse or cloning of aggregates saves a modest amount of time within the multigrid setup phase. Additional reductions in the setup cost can be achieved if one defines $\mathbf{R}_{11} = \mathbf{R}_{00}$ and $\mathbf{P}_{11} = \mathbf{P}_{00}$. Again, this is relatively easy to do within the MueLu framework, though not necessary. In our experiments, piecewise constant interpolation is the basis for all grid transfers. This simple grid transfer choice is more robust for highly convective flows.

**4.3.2. Block smoother for the MHD system.** A block Gauss-Seidel (BGS) iteration forms the basis of the multigrid smoother using the notion of blocks already introduced via the $2 \times 2$ decomposition. A standard block Gauss-Seidel iteration would *solve* for an entire block of unknowns simultaneously, recompute residuals, and then *solve* for the other block of equations simultaneously. This corresponds to alternating between a Navier-Stokes sub-block solve and a Maxwell sub-block solve while performing residual updates after each sub-solve. Effectively, the $2 \times 2$ block perspective emphasizes the coupling within the Navier-Stokes block and within the Maxwell part as each of these blocks correspond to a saddle point system due to the presence of constraint equations. That is, the difficulties associated with saddle point systems will be addressed by an approximate sub-block solve. Further, the block Gauss-Seidel iteration considers the Navier-Stokes block and the Maxwell block equally important as the iteration simply alternates equally between the two sub-solves. As the exact solution of each sub-system is computationally expensive, an additive Schwarz domain decomposition (DD) method with an ILU(0) solve on each domain is used to generate approximate sub-solutions, associating one domain with each computing core.

While not necessarily an optimal smoother, we have found that ILU(0) is often effective for the saddle point systems associated with incompressible flow. While additive Schwarz DD with ILU(0) could be applied to the entire $2 \times 2$ matrix, this involves significantly more computation and memory during the setup and apply phases of the smoother. This more expensive process is generally not needed as the coupling between the block sub-systems is often much less important than the coupling within the sub-blocks, though there could be MHD situations with significant cross-coupling between the Navier-Stokes and Maxwell part that would warrant the smoother applied to the whole system.

**Algorithm 1** Damped blocked Gauss-Seidel smoother.

```
Require: $\mathbf{A}, \mathbf{b}, \omega$, #sweeps
 1: Set initial guess: $\mathbf{x} := \mathbf{0}$
 2: for $s = 0$, $s <$ #sweeps do
 3:     % Calculate update for Navier-Stokes part
 4:     Calculate residual: $\mathbf{r}_0 := \mathbf{b}_0 - \mathbf{A}_{00}\mathbf{x}_0 - \mathbf{A}_{01}\mathbf{x}_1$
 5:     Solve approximately $\mathbf{A}_{00}\widetilde{\mathbf{x}_0} = \mathbf{r}_0$ for $\widetilde{\mathbf{x}_0}$
 6:     Update intermediate solution: $\mathbf{x}_0 := \mathbf{x}_0 + \omega\,\widetilde{\mathbf{x}_0}$
 7:     % Calculate update for Maxwell part
 8:     Calculate residual: $\mathbf{r}_1 := \mathbf{b}_1 - \mathbf{A}_{10}\mathbf{x}_0 - \mathbf{A}_{11}\mathbf{x}_1$
 9:     Solve approximately $\mathbf{A}_{11}\widetilde{\mathbf{x}_1} = \mathbf{r}_1$ for $\widetilde{\mathbf{x}_1}$
10:     Update intermediate solution: $\mathbf{x}_1 := \mathbf{x}_1 + \omega\,\widetilde{\mathbf{x}_1}$
11: end for
12: return $\mathbf{x} := \begin{bmatrix}\mathbf{x}_0 \\ \mathbf{x}_1\end{bmatrix}$
```

Algorithm 1 shows the outline of the damped Gauss-Seidel block smoothing algorithm. First, an approximate solution update $\widetilde{\mathbf{x}_0}$ of the Navier-Stokes part is built in line 5 of Algorithm 1 and then scaled by a damping parameter $\omega$ in line 6. Similarly, an approximate solution update $\widetilde{\mathbf{x}_1}$ is built for the Maxwell part and scaled by the same damping parameter $\omega$ in lines 9 and 10. Please note, that the residual calculation in line 8 employs the intermediate solution update from line 6. Similarly, an intermediate solution is used for the residual calculation in line 4 if we apply more than one sweep with the block Gauss-Seidel smoothing algorithm.

As one can see from Algorithm 1, we only need approximate inverses of the diagonal blocks $\mathbf{A}_{00}$ and $\mathbf{A}_{11}$. A flexible implementation allows one to choose appropriate local smoothing methods to approximately invert the blocks $\mathbf{A}_{00}$ and $\mathbf{A}_{11}$. The coupling is guaranteed by the off-diagonal blocks $\mathbf{A}_{01}$ and $\mathbf{A}_{10}$ in the residual calculations in lines 4 and 8.

**4.4. Software.** Though this paper demonstrates the multigrid approach for certain MHD formulations, it is clear that the concept is more general. Due to the problem-dependent nature of preconditioning multiphysics problems, it is logical to provide a general software framework that helps design application-specific preconditioning methods. The proposed methods in this paper are implemented using the next-generation multigrid framework MueLu from the Trilinos software libraries. In contrast to other publications like [56], which share the same core idea of a general software framework, MueLu is publicly available through the Trilinos library. Furthermore, it is fully embedded in the Trilinos software stack and has full native access to all features provided by the other Trilinos packages. It aims at next-generation HPC platforms and automatically benefits from all performance improvements in the underlying core linear algebra packages. The core design concept of MueLu is based on building blocks that can be combined to construct complex preconditioner layouts. Each building block processes certain input data and produces output data which serves as input for the downstream building blocks. The relations between the building blocks are easy to modify via an XML input file. For application-specific adaptions it is usually sufficient to replace specific algorithmic components while the majority of building blocks can be reused.

## 5. Experimental results.

All simulations were run on a multi-node cluster composed of nodes with dual 16 core sockets with Intel Broadwell E5-2695 (2.1 GHz) processors. Each node contains 128 GB RAM, and network communication is performed on an Intel Omni-Path high-speed interconnect.

### 5.1. The MHD generator.

This problem is a steady-state MHD duct flow configuration representing an idealized MHD generator, where an electrical current is induced by pumping a conducting fluid (mechanical work) through an externally applied vertical magnetic field [48]. The bending of the magnetic field lines produces a horizontal electrical current. The geometric domain for this problem is a square cross-sectional duct of dimensions $[0, 15] \times [0, 1] \times [0, 1]$. The velocity boundary conditions are set with Dirichlet inlet velocity of $\mathbf{u} = (u, 0, 0)$, no-slip conditions on the top, bottom, and sides of the channel and natural boundary conditions on the outflow. The magnetic field on the top and bottom boundaries is specified as $\mathbf{B} = (0, B_y^{\text{gen}}, 0)$, where

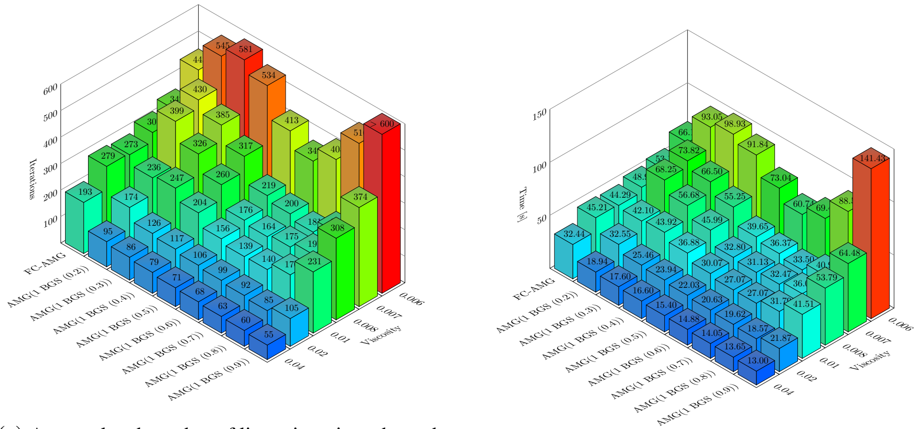
$$B_y^{\text{gen}} = \frac{1}{2} B_0 \left[ \tanh\left(\frac{x - x_{\text{on}}}{\delta}\right) - \tanh\left(\frac{x - x_{\text{off}}}{\delta}\right) \right].$$

Here, $B_0$ is the strength of the field, and $\delta$ is a measure of the transition length-scale for the application of the field. The inlet, outlet, and sides are perfect conductors with $\mathbf{B} \cdot \hat{\mathbf{n}} = \mathbf{0}$ and $\mathbf{E} \times \hat{\mathbf{n}} = \mathbf{0}$, where $\hat{\mathbf{n}}$ is the outward facing unit normal vector. Zero Dirichlet boundary conditions are applied on all surfaces for the Lagrange multiplier. The problem is defined by three non-dimensional parameters: the Reynolds number $Re = \rho u L/\mu$, the magnetic Reynolds number $Re_m = \mu_0 u L/\eta$, and the Hartmann number $Ha = B_0 L/\sqrt{\rho \nu \eta}$. Here, $u$ is the maximum $x$-direction velocity. The parameters in this problem are taken to be $u = 1.0$, $\rho = 1$, $B_0 = 3.354$, $\mu_0 = 1$, $\eta = 1$, $x_{\text{on}} = 4.0$, $x_{\text{off}} = 6.0$, and $\delta = 0.5$. At each stage of the Newton method, a non-restarted GMRES iterative Krylov solver is used. The convergence criteria for each of the linear solves is taken as a relative residual reduction of $10^{-3}$, which is consistent with an inexact Newton-type procedure [19, 20, 50]. The reference preconditioner is a fully-coupled AMG method (FC-AMG), where the relaxation method, additive Schwarz DD with overlap one and ILU(0) on each domain, is applied to the entire system that includes the off-diagonal coupling between the fluids and magnetics (see [38]). For the blocked variant presented in this manuscript (Section 4), we use BGS as a relaxation method (AMG(BGS)). The approximate solves for the sub-blocks are handled with additive Schwarz DD with overlap one and ILU(0) on each domain. The coarse grid for both the FC-AMG and AMG(BGS) is solved directly.

In the first study, we investigate the number of iterations and solution time as a function of the block Gauss-Seidel (BGS) damping parameter $\omega$ for various values of the viscosity $\mu \in \{0.006, 0.007, 0.008, 0.01, 0.02, 0.04\}$, which effectively sets the range for the non-dimensional parameters: $25 \leq Re \leq 167$, $17 \leq Ha \leq 43$, and $Re_m = 1$.

Figure 5.1 displays the accumulated number of linear iterations and solver timings for the MHD generator problem on a $240 \times 16 \times 16$ mesh using 32 processors. The numbers on the side of the columns in Figure 5.1a denote the number of nonlinear iterations. The

FIG. 5.1. *Solver performance for the AMG(BGS) preconditioner with 1 BGS coupling iteration versus the fully-coupled FC-AMG preconditioner for the MHD generator example on a* $240 \times 16 \times 16$ *mesh using* 32 *processors.*



(a) Accumulated number of linear iterations depending on the BGS damping parameter and viscosity. The numbers on the sides of the column denote the number of nonlinear iterations to solve the system.

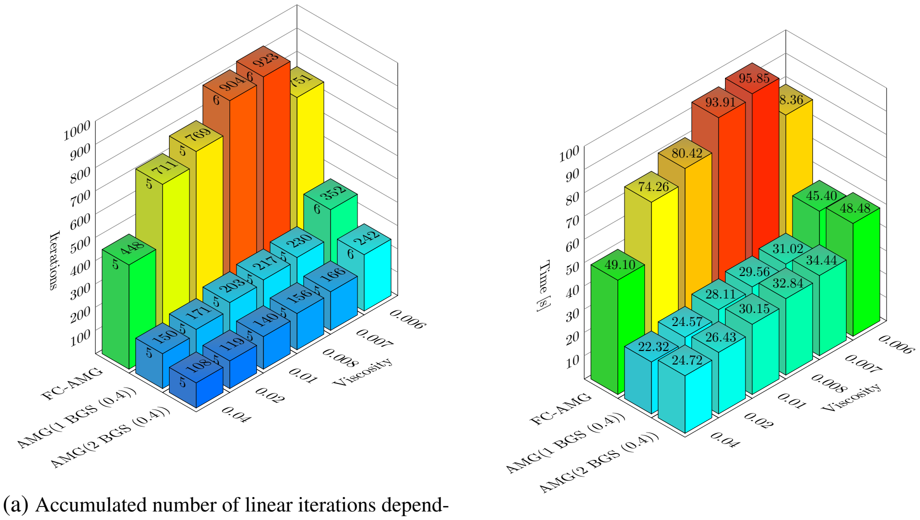
(b) Solver timings (setup + iteration phase) depending on the BGS damping parameter and viscosity.

number on the $z$-axis represents the accumulated number of linear iterations. Figure 5.1b displays the accumulated solver timings (setup and iteration phase) of the corresponding preconditioning variants. The timings are averaged over 5 simulations. As one can see from Figure 5.1, the right choice of the BGS damping parameter is crucial for smaller viscosities. Even though the optimal damping parameter depends on the problem, in practice a damping parameter $\omega$ near 0.5 seems to work well. Figure 5.2 shows the accumulated linear iterations and the solver times for the MHD generator example on a finer $480 \times 32 \times 32$ mesh using 256 processors. One can see in Figure 5.2a that a higher number of BGS iterations reduces the overall number of linear iterations, but not enough to compensate for the higher costs per iteration (see Figure 5.2b). For reference, the fully-coupled approach is also shown in the back row (labeled as *FC-AMG*). This option generally takes more time and iterations than the multiphysics solver, varying somewhere between 2 and 3 times slower. In experiments not shown here, tightening the linear solver tolerance lead to comparable scalability, albeit with higher runtime for all preconditioners. As a result we have focused on the looser tolerances to minimize runtime for these engineering applications.

### 5.2. The hydromagnetic Kevin-Helmholtz (HMKH) problem.

A hydromagnetic Kelvin-Helmholtz unstable shear layer problem is a configuration used to study magnetic reconnection [48] and is posed in a domain of $[0, 4] \times [-2, 2] \times [0, 2]$. It is described by an initial condition defined by two counter flowing conducting fluid streams with constant velocities $\mathbf{u}(x, y > 0, z, 0) = (5, 0, 0)$ and $\mathbf{u}(x, y < 0, z, 0) = (-5, 0, 0)$ and a Harris-sheet-sheared magnetic field configuration given by $\mathbf{B}(x, y, z, 0) = (0, B_0 \tanh(y/\delta), 0)$. The boundary conditions are periodic on the right and left as well as on the front and back. The top and bottom are impenetrable for the fluid velocity, and the magnetic field is defined by the Harris sheet. The magnetic Lagrange multiplier is taken as zero on all boundaries. The parameters in this problem are $\rho = 1$, $\mu_0 = 1$, $\mu = 10^{-4}$, $\eta = 10^{-4}$, $B_0 = 0.3333$, $\delta = 0.1$ to produce $Re = 5 \times 10^4$, $Re_m = 5 \times 10^4$, and an Alfven velocity, $u_A = B_0/\sqrt{\rho\mu_0} = 0.333$, resulting in an Alfvenic Mach number $M_A = u/u_A = 15$. For these non-dimensional parameters, the

FIG. 5.2. *Solver performance for the AMG(BGS) preconditioner with a fixed damping parameter and 1 or 2 BGS coupling iterations versus the fully coupled FC-AMG preconditioner for the MHD generator example on a* $480 \times 32 \times 32$ *mesh using* $256$ *processors.*

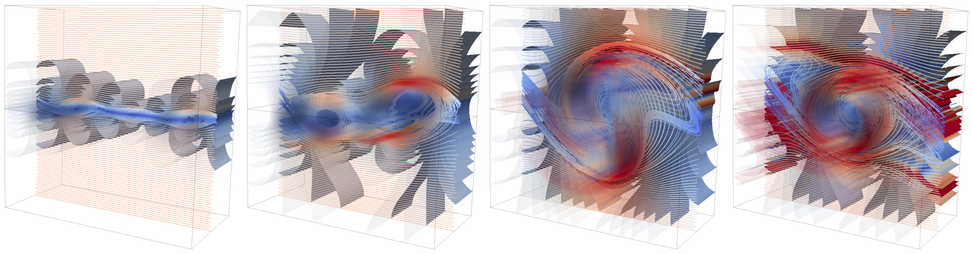


(a) Accumulated number of linear iterations depending on the BGS iterations and viscosity. The numbers on the sides of the column denote the number of nonlinear iterations to solve the system.

(b) Solver timings (setup + iteration phase) depending on the BGS iterations parameter and viscosity.

shear layer is Kelvin-Helmholtz unstable and forms a vortex sheet that evolves with time and undergoes thin current sheet formation, vortex rollup, and merging. Figure 5.3 displays the unstable shear layer evolving from smaller vortices to a larger vortex.

FIG. 5.3. *Hydromagnetic KH problem with* $Re = 5 \times 10^4$, $Re_m = 5 \times 10^4$, $M_A = 15$. *Pressure contour lines and velocity streamlines.*



(a) $t = 3.0s$.

(b) $t = 3.5s$.

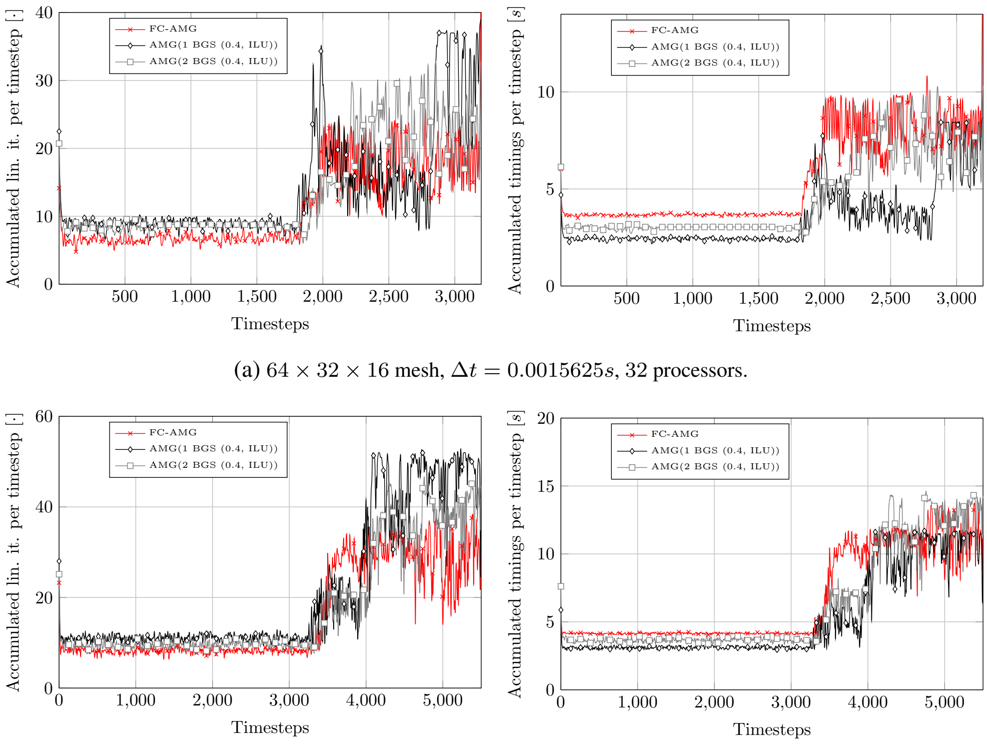
(c) $t = 4.0s$.

(d) $t = 4.5s$.

To study the behavior of the nonlinear- and linear solver, we perform transient simulations of the problem with $\text{CFL}_{max} = 0.25$ and $\text{CFL}_{max} = 0.5$. The step sizes were chosen to resolve the advected time scale. Below, we compare the timings and number of iterations of non-restarted GMRES with a solver tolerance of $10^{-3}$ for the reference fully-coupled AMG preconditioner (FC-AMG) and the blocked AMG(BGS) preconditioner.

Figures 5.4 and 5.5 illustrate the solver performance for the different preconditioning strategies over a time sequence for the HMKH problem with a maximum CFL number of $0.25$ and $0.5$, respectively. We ran the problem both on a $64 \times 32 \times 16$ mesh on 32 processors and a $128 \times 64 \times 32$ mesh on 256 processors. Generally, the FC-AMG method needs the least number of iterations, whereas the blocked AMG(BGS) variant with only 1 coupling iteration needs the highest number of iterations. Increasing the number of BGS coupling iterations reduces the number of linear iterations getting closer to the reference method. However, looking at the linear solver time (setup and iteration phase), we see the opposite picture. The FC-AMG method is the slowest, and the blocked AMG(1 BGS) is the most time-efficient method. Comparing the solver behavior for the different meshes, there is a slight increase in the linear iteration count for the finer meshes. So, while the weak scaling is not optimal, the iteration growth with problem size is mild.

FIG. 5.4. *HMKH example ($CFL_{max} = 0.25$). The left plots show the accumulated linear iterations over time steps. The right plots show the accumulated solution time (setup + iteration phase) per time step.*

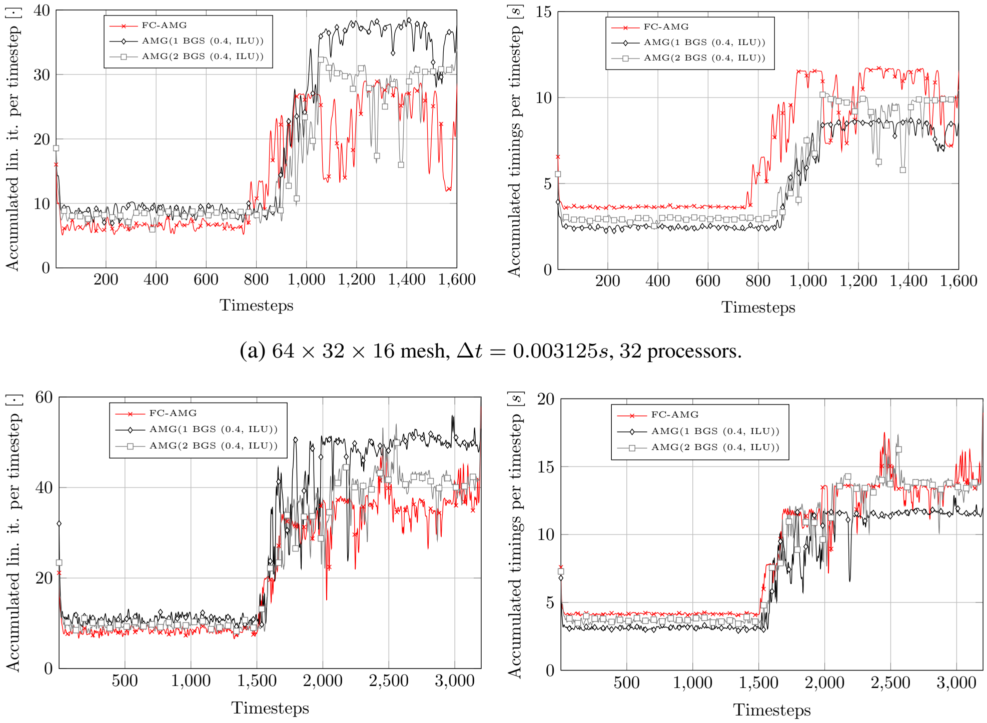


(a) $64 \times 32 \times 16$ mesh, $\Delta t = 0.0015625s$, 32 processors.

(b) $128 \times 64 \times 32$ mesh, $\Delta t = 0.00078125s$, 256 processors.

Numerical results are further summarized in Tables 5.1 and 5.2. The first column denotes the average number of nonlinear iterations per time step for the full simulation. One can see that the number of BGS coupling iterations has some influence on the nonlinear solver, even though there is no clear trend. The second column represents the average number of linear iterations per nonlinear iteration. As one would expect, a higher number of BGS coupling iterations reduces the number of linear iterations necessary to solve the problem. The next

FIG. 5.5. *HMKH example ($CFL_{max} = 0.5$). The left plots show the accumulated linear iterations over time steps. The right plots show the accumulated solution time (setup + iteration phase) per time step.*



(a) $64 \times 32 \times 16$ mesh, $\Delta t = 0.003125s$, 32 processors.

(b) $128 \times 64 \times 32$ mesh, $\Delta t = 0.0015625s$, 256 processors.

three columns give the average setup time, the average solve time, and the average overall time for the linear solver per nonlinear iteration. The AMG(BGS) variants have a clear advantage in the setup costs, but the iteration costs are higher. That is, fully-coupled AMG using an additive Schwarz method with ILU(0) that includes Navier-Stokes and Maxwell coupling provides some convergence benefits, but at the cost of a very large setup time. The last three columns show the absolute setup, solve, and overall solver time for finishing the simulation. It should be noted that the FC-AMG method is generally more competitive with the block-oriented solvers for this problem, and it can even modestly outperform some of the block-oriented variants. That is, a block approach to monolithic AMG often yields a faster method than a traditional (non-block) AMG strategy, but certainly not all of the time. More generally, a block-oriented approach provides additional algorithm possibilities to address complex multiphysics problems by combining existing algorithms for the components. These block algorithms can be critical for some discrete systems (such as the mixed finite element system shown in Section 5.4) or provide significant gains for other systems such as those shown in Figure 5.2. In still other cases, such as those in Figure 5.5, the gains may be more modest.

**5.3. Island coalescence.** The island coalescence problem is a prototype problem used to study magnetic reconnection. While seemingly dominated by transient dynamics, a scalable simulation of this problem requires correctly handling the bidirectional coupling between the fluid and the magnetics equations. This is evident from the larger Alfven wave CFLs used in

TABLE 5.1
*HMKH problem for CFL*$_{max} = 0.25$.

Legend:

| | |
|---|---|
| $n_T$ | Number of time steps |
| $n_N$ | Accumulated number of all nonlinear iterations |
| $n_L$ | Accumulated number of all linear iterations |
| $t_{Se}$ | Multigrid setup time |
| $t_{So}$ | Multigrid solution time (iteration phase) |
| $t_\Sigma$ | Solver time (setup + iteration phase) |

| | Preconditioner | $\frac{n_N}{n_T}$ | $\frac{n_L}{n_N}$ | $\frac{t_{Se}}{n_N}$ | $\frac{t_{So}}{n_N}$ | $\frac{t_\Sigma}{n_N}$ | $t_{Se}$ | $t_{So}$ | $t_\Sigma$ |
|---|---|---|---|---|---|---|---|---|---|
| $64 \times 32 \times 16$ | FC-AMG | 1.47 | 7.72 | 3.08 | 0.68 | 3.76 | 14443.59 | 3186.92 | 17630.51 |
| | AMG(1 BGS (0.4, ILU)) | 1.37 | 10.18 | 1.46 | 1.11 | 2.57 | 6393.33 | 4854.98 | 11248.30 |
| | AMG(2 BGS (0.4, ILU)) | 1.47 | 9.25 | 1.47 | 1.65 | 3.12 | 6909.57 | 7780.94 | 14690.51 |
| | AMG(3 BGS (0.4, ILU)) | 1.49 | 8.53 | 1.47 | 2.14 | 3.61 | 7034.32 | 10238.93 | 17273.25 |
| $128 \times 64 \times 32$ | FC-AMG | 1.53 | 10.51 | 3.37 | 0.99 | 4.36 | 28307.94 | 8318.34 | 36626.28 |
| | AMG(1 BGS (0.4, ILU)) | 1.55 | 13.96 | 1.75 | 1.73 | 3.48 | 14884.65 | 14781.09 | 29665.74 |
| | AMG(2 BGS (0.4, ILU)) | 1.51 | 11.87 | 1.74 | 2.39 | 4.13 | 14484.49 | 19942.69 | 34427.18 |
| | AMG(3 BGS (0.4, ILU)) | 1.42 | 10.69 | 1.74 | 3.07 | 4.81 | 13619.80 | 23986.51 | 37606.32 |

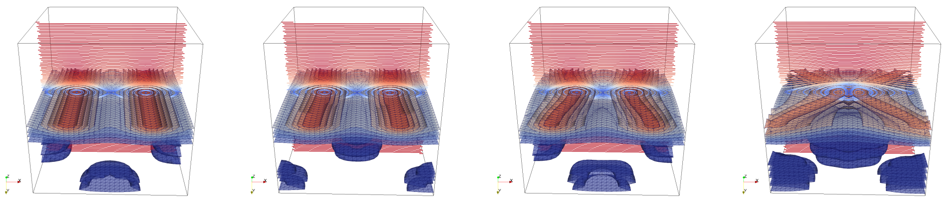
TABLE 5.2
*HMKH problem for CFL*$_{max} = 0.5$.

Legend:

| | |
|---|---|
| $n_T$ | Number of time steps |
| $n_N$ | Accumulated number of all nonlinear iterations |
| $n_L$ | Accumulated number of all linear iterations |
| $t_{Se}$ | Multigrid setup time |
| $t_{So}$ | Multigrid solution time (iteration phase) |
| $t_\Sigma$ | Solver time (setup + iteration phase) |

| | Preconditioner | $\frac{n_N}{n_T}$ | $\frac{n_L}{n_N}$ | $\frac{t_{Se}}{n_N}$ | $\frac{t_{So}}{n_N}$ | $\frac{t_\Sigma}{n_N}$ | $t_{Se}$ | $t_{So}$ | $t_\Sigma$ |
|---|---|---|---|---|---|---|---|---|---|
| $64 \times 32 \times 16$ | FC-AMG | 1.81 | 8.06 | 3.07 | 0.70 | 3.77 | 8878.08 | 2013.02 | 10891.10 |
| | AMG(1 BGS (0.4, ILU)) | 1.77 | 11.19 | 1.47 | 1.24 | 2.71 | 4166.08 | 3513.20 | 7679.28 |
| | AMG(2 BGS (0.4, ILU)) | 1.74 | 9.39 | 1.46 | 1.68 | 3.14 | 4081.49 | 4667.58 | 8749.07 |
| | AMG(3 BGS (0.4, ILU)) | 1.75 | 8.68 | 1.48 | 2.20 | 3.68 | 4143.28 | 6192.65 | 10335.92 |
| $128 \times 64 \times 32$ | FC-AMG | 1.94 | 11.19 | 3.36 | 1.06 | 4.42 | 20883.40 | 6616.06 | 27499.46 |
| | AMG(1 BGS (0.4, ILU)) | 1.91 | 15.05 | 1.74 | 1.91 | 3.65 | 10691.61 | 11677.32 | 22368.94 |
| | AMG(2 BGS (0.4, ILU)) | 1.89 | 12.74 | 1.74 | 2.58 | 4.32 | 10572.95 | 15626.96 | 26199.91 |
| | AMG(3 BGS (0.4, ILU)) | 1.80 | 11.33 | 1.74 | 3.19 | 4.93 | 10038.69 | 18408.73 | 28447.42 |

FIG. 5.6. *Structure of the current tubes in the 3D island coalescence problem with* $S = 2 \times 10^4$ *for the initial condition and for the times in the evolution of the problem of* $t = 2, 3, 4$. *The 3D current tubes have bent in the z-direction and form current sheets.*



(a) $t = 0.0s$. (b) $t = 2.0s$. (c) $t = 3.0s$. (d) $t = 4.0s$.

this section. Incorrectly accounting for the coupling leads to poor scalability (see [12, 48] for a further discussion).

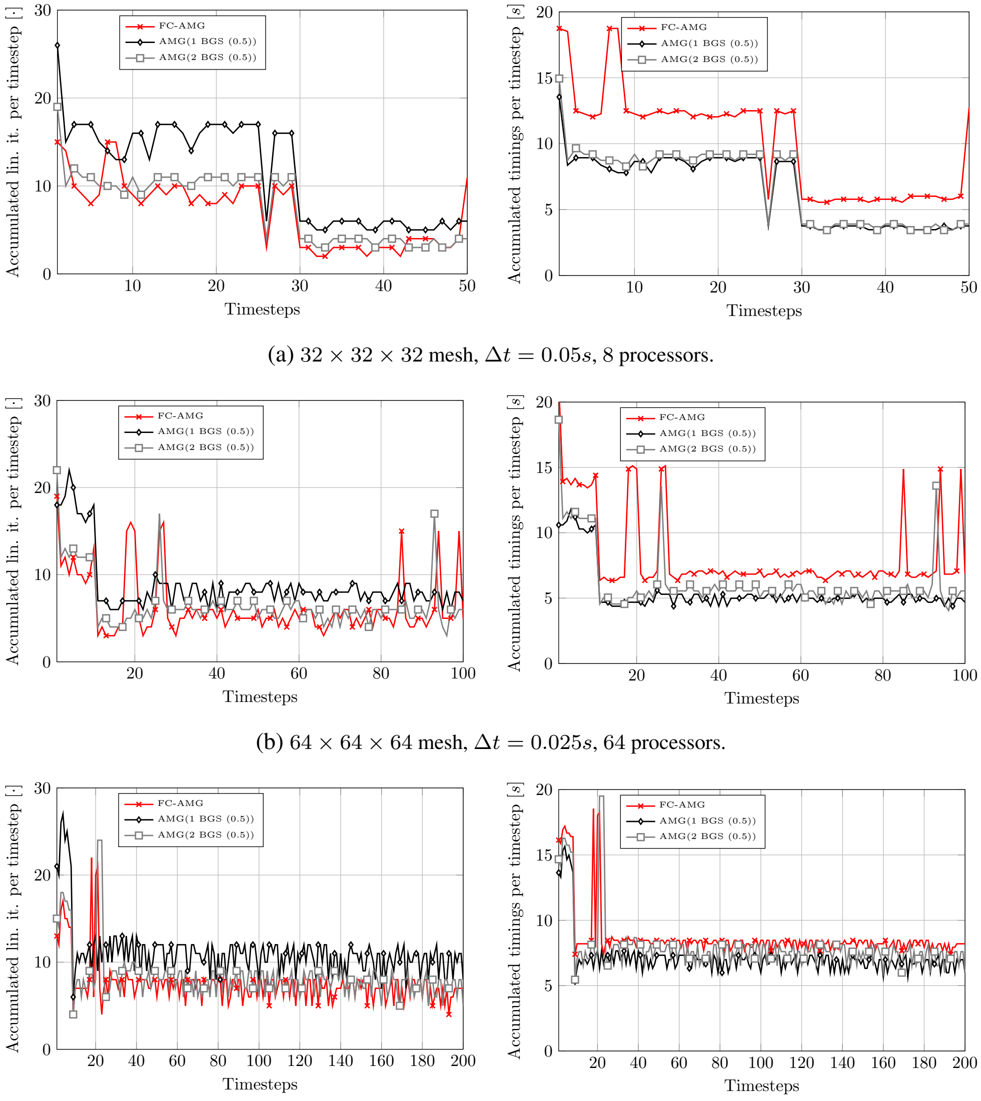
Figure 5.6 displays different stages of the island coalescence reconnection event using an iso-surface of $p$ and iso-lines of $\mathbf{B}$ at $z = 0$. At $t = 0s$, the initial equilibrium is described by two 3D current tubes (islands in the 2D plane) embedded in a Harris current sheet (as in the HMKH problem of Section 5.2) in a $[-1, 1] \times [-1, 1] \times [-1, 1]$ domain [9, 22]. The initial condition for the island coalescence problem consists of zero fluid velocities ($\mathbf{u}^0 = \mathbf{0}$), zero Lagrange multipliers ($\psi = 0$), and a Fadeev magnetic equilibrium [9, 22] that defines the magnetic field $\mathbf{B}$ and the fluid pressure $p$. More details of this setup can be found in [48]. The dynamics of island coalescence changes as a function of resistivity. For larger resistivities, the islands monotonically approach each other and the X-point in the center. For low resistivities, fluid-plasma pressure builds up, and the islands "slosh", no longer monotonically approaching the X-point, leading to a lower reconnection rate (for more details on the physics, see, e.g., [5]). Figure 5.6 displays different stages of the reconnection event. Clearly evident is the formation of the X-point in the intersecting planes between the islands (see the images at $t = 4$), the development of thin current sheets at that same X-point location (and the corresponding 3D surface), and the movement of the center of the tubes (island O-points) towards the X-point [5, 36]. In this study, we have taken $\rho = 1$, $\mu = \eta = 10^{-3}$, $\mu_0 = 1$, and, using the spacing of the O-points, we have $L = 1$, resulting in $Re = Re_m = 10^3$. As in [36], these choices imply the resistivity $\eta = 1/S$, where $S$ is the Lundquist number defined as $S = \mu_0 L u_A/\eta$, where $u_A$ is the Alfven velocity. We perform transient simulations of the problem with time step sizes of $\Delta t \in \{0.05, 0.025, 0.0125\}$. The mesh sizes used were $32 \times 32 \times 32$, $64 \times 64 \times 64$ and $128 \times 128 \times 128$ and were run on 8, 64, and 512 processors, respectively on 2, 16, and 128 cluster nodes. This provides simulations with Alfven wave CFLs ranging from 1.6 to 12.8. We compare the number of iterations and timings of non-restarted GMRES, using a relative linear solve tolerance of $\varepsilon = 10^{-3}$ when combined with different preconditioning strategies.

Figure 5.7 displays the solver performance for different preconditioning strategies over a time sequence for the island coalesce problem with a CFL number of 3.2. As with the HMKH example, we see that while the AMG(BGS) variants require more iterations than the FC-AMG reference, the cost per iteration is low enough that the time savings ends up in favor of the AMG(BGS).

Results for the island coalescence problem with various CFL numbers are summarized in Tables 5.3 through Tables 5.6. The first column denotes the average number of nonlinear iterations per time step for the full simulation. The second column represents the average number of linear iterations per nonlinear iteration. Again, as one would expect, increasing the BGS coupling iterations results in faster convergence or fewer required iterations. The next three columns give the average setup time, the average solve time, and the average overall time for the linear solver per nonlinear iteration. The last three columns show the absolute setup, solve, and overall solver time for finishing the simulation. While the FC-AMG boasts faster solve times, it has significant setup costs. The AMG(BGS) demonstrates a significant reduction in setup time, though the approach is only slightly faster than the FC-AMG approach due to the higher solve times.

**5.4. Mixed finite elements.** Next we illustrate a formulation for which the hydrodynamics and electromagnetics systems are discretized by differing FE spaces. In this example we consider Q2/Q2 VMS for the hydrodynamics (saddle point and convective stabilization) and Q1/Q1 VMS for the induction (electromagnetics) systems (saddle point and convective stabilization). In this problem the difference in the order of accuracy is motivated by the desire to minimize the overall computational time while still maintaining higher accuracy for

FIG. 5.7. *Island coalescence example (CFL = 3.2). The left plots show the accumulated linear iterations over time steps. The right plots show the accumulated solution time (setup + iteration phase) per time step.*



(a) $32 \times 32 \times 32$ mesh, $\Delta t = 0.05s$, 8 processors.

(b) $64 \times 64 \times 64$ mesh, $\Delta t = 0.025s$, 64 processors.

(c) $128 \times 128 \times 128$ mesh, $\Delta t = 0.0125s$, 512 processors.

the MHD simulation in appropriate applications. For example, when a liquid metal is the conducting fluid in an MHD generator, the flow Reynolds number can be significantly higher than the corresponding magnetic Reynolds number due to the very high magnetic diffusivity of liquid metals. In general, the low magnetic Reynolds number is indicative of diffusive-dominated transport for the magnetics in the liquid metal. Thus, a mixed discretization with a lower-order approximation for the induction subsystem may be appropriate. Other cases that employ disparate discretizations for hydrodynamics and magnetics would be various forms of structure-preserving methods where, for example, nodal FE are employed for flow variables and face or even edge FE are used for the magnetic field (see, e.g., [6, 40, 42, 45]).

TABLE 5.3
*Island coalescence problem for CFL 1.6.*

Legend:

| | |
|---|---|
| $n_T$ | Number of time steps |
| $n_N$ | Accumulated number of all nonlinear iterations |
| $n_L$ | Accumulated number of all linear iterations |
| $t_{Se}$ | Multigrid setup time |
| $t_{So}$ | Multigrid solution time (iteration phase) |
| $t_\Sigma$ | Solver time (setup + iteration phase) |

| | Preconditioner | $\frac{n_N}{n_T}$ | $\frac{n_L}{n_N}$ | $\frac{t_{Se}}{n_N}$ | $\frac{t_{So}}{n_N}$ | $\frac{t_\Sigma}{n_N}$ | $t_{Se}$ | $t_{So}$ | $t_\Sigma$ |
|---|---|---|---|---|---|---|---|---|---|
| $32 \times 32 \times 32$ | FC-AMG | 1.16 | 3.96 | 5.07 | 0.91 | 5.98 | 588.12 | 105.67 | 693.79 |
| | AMG(1 BGS (0.4)) | 1.12 | 7.12 | 2.06 | 2.00 | 4.07 | 230.83 | 224.47 | 455.30 |
| | AMG(2 BGS (0.4)) | 1.14 | 5.11 | 2.06 | 2.37 | 4.43 | 234.84 | 269.82 | 504.66 |
| | AMG(3 BGS (0.4)) | 1.15 | 4.29 | 2.06 | 2.74 | 4.80 | 237.36 | 314.98 | 552.34 |
| $64 \times 64 \times 64$ | FC-AMG | 1.04 | 5.23 | 5.62 | 1.28 | 6.89 | 1174.16 | 266.58 | 1440.74 |
| | AMG(1 BGS (0.4)) | 1.04 | 8.77 | 2.55 | 2.66 | 5.21 | 533.16 | 555.24 | 1088.40 |
| | AMG(2 BGS (0.4)) | 1.04 | 6.64 | 2.55 | 3.31 | 5.86 | 533.16 | 692.22 | 1225.38 |
| | AMG(3 BGS (0.4)) | 1.04 | 5.56 | 2.55 | 3.87 | 6.42 | 532.95 | 808.61 | 1341.56 |

Here, we again consider the MHD Generator problem from Section 5.1 with the modest intention of demonstrating the ability of our proposed methods to handle disparate discretizations, which in this case are Q2/Q2 VMS for the hydrodynamics and Q1/Q1 VMS for the induction (electromagnetics) systems. The system is difficult to approach through standard fully-coupled AMG methods due to the mixed FE spaces with DoFs that are no longer co-located. The blocked approach outlined in this manuscript allows for the separate construction of aggregates for the hydrodynamics block and the electromagnetics block. The monolithic multigrid hierarchy then naturally provides us with a coupling between the blocks on all levels of the hierarchy.

The study is carried out for the same set of physical, geometrical, and solver parameters as in Section 5.1, with varying viscosities $\mu \in \{0.006, 0.01, 0.04\}$. The relaxation method is a blocked Gauss-Seidel scheme with a damping parameter of 0.6. For the sub-block solves, a single iteration of an additive Schwarz method with an overlap of one and ILU(0) was used to generate approximate sub-block solutions. The current implementation that we are using lacks parallel load rebalancing, which is problematic on higher core counts. To circumvent this issue, the maximum number of AMG levels was capped at 4 levels, as a further coarsening of the 2048 processor case requires rebalancing. The coarsest level problem is still relatively large in the 2048 processor case (16,384 rows after 3 levels of refinement or 8 DoFs per processor), so the coarse level solve is handled with an iteration of the smoother instead of a direct solve.

We explore the weak scaling of the method in Table 5.7, showing iterations and timings for various preconditioner configurations. For comparison, we also consider an additive Schwarz domain decomposition method (Schwarz-DD) with overlap one and ILU(0) domain solve for the entire $2 \times 2$ block system as a preconditioner. The use of Blocked AMG provides a significant reduction in setup time over the monolithic additive Schwarz method, as the block off-diagonal terms are no longer considered in the factorization. While the number of linear iterations does degrade as the problem size increases, the ability to apply Blocked AMG to this mixed FE space problem provides a significant linear solve time speed-up compared to the use of an additive Schwarz method on the entire $2 \times 2$ block system.

Additional work is needed to better understand the smoother and aggregation choices in the mixed FE case as evidenced by the increase in iterations as the problem size increases

TABLE 5.4
*Island coalescence problem for CFL 3.2.*

| | | |
|---|---|---|
| Legend: | $n_T$ | Number of time steps |
| | $n_N$ | Accumulated number of all nonlinear iterations |
| | $n_L$ | Accumulated number of all linear iterations |
| | $t_{Se}$ | Multigrid setup time |
| | $t_{So}$ | Multigrid solution time (iteration phase) |
| | $t_\Sigma$ | Solver time (setup + iteration phase) |

| | Preconditioner | $\frac{n_N}{n_T}$ | $\frac{n_L}{n_N}$ | $\frac{t_{Se}}{n_N}$ | $\frac{t_{So}}{n_N}$ | $\frac{t_\Sigma}{n_N}$ | $t_{Se}$ | $t_{So}$ | $t_\Sigma$ |
|---|---|---|---|---|---|---|---|---|---|
| $32 \times 32 \times 32$ | FC-AMG | 1.66 | 4.27 | 5.10 | 0.98 | 6.08 | 423.22 | 81.27 | 504.49 |
| | AMG(1 BGS (0.4)) | 1.60 | 8.54 | 2.07 | 2.42 | 4.48 | 165.44 | 193.33 | 358.77 |
| | AMG(2 BGS (0.4)) | 1.60 | 5.85 | 2.06 | 2.69 | 4.76 | 165.12 | 215.45 | 380.57 |
| | AMG(3 BGS (0.4)) | 1.68 | 4.70 | 2.06 | 2.99 | 5.05 | 173.12 | 251.34 | 424.46 |
| $64 \times 64 \times 64$ | FC-AMG | 1.19 | 5.35 | 5.63 | 1.29 | 6.92 | 670.33 | 153.54 | 823.87 |
| | AMG(1 BGS (0.4)) | 1.10 | 8.95 | 2.55 | 2.72 | 5.27 | 280.50 | 299.32 | 579.82 |
| | AMG(2 BGS (0.4)) | 1.10 | 6.43 | 2.55 | 3.24 | 5.79 | 280.61 | 356.08 | 636.69 |
| | AMG(3 BGS (0.4)) | 1.13 | 5.36 | 2.56 | 3.71 | 6.27 | 288.72 | 419.54 | 708.26 |
| $128 \times 128 \times 128$ | FC-AMG | 1.05 | 7.16 | 6.34 | 1.91 | 8.24 | 1336.90 | 402.68 | 1739.57 |
| | AMG(1 BGS (0.4)) | 1.04 | 11.75 | 3.31 | 3.91 | 7.21 | 687.65 | 812.75 | 1500.40 |
| | AMG(2 BGS (0.4)) | 1.05 | 8.73 | 3.32 | 4.69 | 8.01 | 697.83 | 984.75 | 1682.58 |
| | AMG(3 BGS (0.4)) | 1.05 | 7.38 | 3.34 | 5.47 | 8.80 | 700.35 | 1148.35 | 1848.70 |

for high viscosity, indicating potential inefficiencies. One area of concern is that the AMG coarsening rate for the higher-order variables (fluid variables in this case) is faster than the coarsening rate for the lower-order variables. This is a consequence of having denser matrix rows associated with the Q2 hydrodynamics problem when compare with the Q1 electromagnetics problem. More generally, the coarsening of the hydrodynamics equations and the electromagnetics equations are completely independent of each other. In future work, we will consider two enhancements to improve the AMG method. The first centers on applying AMG to a low-order discretization in order to precondition the higher-order discretization. This type of approach, often referred to as defect correction, requires some relaxation sweeps for the high-order matrix as well as a mechanism for transferring residuals and solutions between the low-order and high-order spaces. A second enhancement will consider correlated coarsening algorithms, where the Q1/Q1 aggregates influence the Q2/Q2 aggregation scheme. In this example there is some partial overlap in the location of DoFs on the mesh. Some mesh nodes have 8 DoFs, corresponding to four DoFs for hydrodynamics and four DoFs for electromagnetics while others only have 4 DoFs, corresponding only to the hydrodynamics. A natural extension is to force the aggregation scheme to preserve this partial co-location aspect of the hydrodynamics and the electromagnetics Dofs. In future work, we plan to incorporate some ability to partially share aggregation information between the two AMG invocations. In this case, one might share the aggregate root (or central) vertices generated during the AMG invocation for electomagnetics. These root vertices could then be used to construct an initial set of aggregates for the hydrodynamics. As there are more hydrodynamic unknowns, many hydrodynamic unknowns might remain unaggregated, and so further aggregation would be needed to complete the set of aggregates for the hydrodynamics.

It should be noted that the mixed finite example that we show here is a somewhat simpler case of more general mixed finite element problems. In particular, our formulation still includes stabilization terms $\mathbf{L_P}$ and $\mathbf{L}_\psi$ in equation (2.2). Many mixed finite element formulations are

TABLE 5.5
*Island coalescence problem for CFL 6.4.*

Legend:
$n_T$ Number of time steps
$n_N$ Accumulated number of all nonlinear iterations
$n_L$ Accumulated number of all linear iterations
$t_{Se}$ Multigrid setup time
$t_{So}$ Multigrid solution time (iteration phase)
$t_\Sigma$ Solver time (setup + iteration phase)

| | Preconditioner | $\frac{n_N}{n_T}$ | $\frac{n_L}{n_N}$ | $\frac{t_{Se}}{n_N}$ | $\frac{t_{So}}{n_N}$ | $\frac{t_\Sigma}{n_N}$ | $t_{Se}$ | $t_{So}$ | $t_\Sigma$ |
|---|---|---|---|---|---|---|---|---|---|
| $64 \times 64 \times 64$ | FC-AMG | 1.50 | 5.61 | 5.60 | 1.35 | 6.96 | 420.15 | 101.61 | 521.76 |
| | AMG(1 BGS (0.4)) | 1.30 | 10.80 | 2.55 | 3.32 | 5.87 | 165.69 | 215.60 | 381.29 |
| | AMG(2 BGS (0.4)) | 1.34 | 6.99 | 2.56 | 3.52 | 6.07 | 171.39 | 235.52 | 406.90 |
| | AMG(3 BGS (0.4)) | 1.54 | 5.86 | 2.56 | 4.07 | 6.63 | 196.81 | 313.53 | 510.34 |
| $128 \times 128 \times 128$ | FC-AMG | 1.25 | 7.38 | 6.35 | 2.02 | 8.37 | 793.62 | 252.00 | 1045.63 |
| | AMG(1 BGS (0.4)) | 1.09 | 12.27 | 3.31 | 4.09 | 7.40 | 361.01 | 445.69 | 806.70 |
| | AMG(2 BGS (0.4)) | 1.21 | 9.04 | 3.31 | 4.85 | 8.16 | 400.27 | 587.34 | 987.61 |
| | AMG(3 BGS (0.4)) | 1.24 | 7.30 | 3.32 | 5.40 | 8.72 | 411.93 | 669.75 | 1081.67 |

TABLE 5.6
*Island coalescence problem for CFL 12.8.*

Legend:
$n_T$ Number of time steps
$n_N$ Accumulated number of all nonlinear iterations
$n_L$ Accumulated number of all linear iterations
$t_{Se}$ Multigrid setup time
$t_{So}$ Multigrid solution time (iteration phase)
$t_\Sigma$ Solver time (setup + iteration phase)

| | Preconditioner | $\frac{n_N}{n_T}$ | $\frac{n_L}{n_N}$ | $\frac{t_{Se}}{n_N}$ | $\frac{t_{So}}{n_N}$ | $\frac{t_\Sigma}{n_N}$ | $t_{Se}$ | $t_{So}$ | $t_\Sigma$ |
|---|---|---|---|---|---|---|---|---|---|
| $128 \times 128 \times 128$ | FC-AMG | 1.58 | 7.80 | 6.35 | 2.08 | 8.44 | 502.04 | 164.66 | 666.71 |
| | AMG(1 BGS (0.4)) | 1.34 | 15.48 | 3.31 | 5.16 | 8.47 | 221.84 | 345.86 | 567.69 |
| | AMG(2 BGS (0.4)) | 1.42 | 9.61 | 3.31 | 5.15 | 8.46 | 235.15 | 365.40 | 600.55 |
| | AMG(3 BGS (0.4)) | 1.48 | 7.62 | 3.31 | 5.65 | 8.95 | 244.72 | 417.88 | 662.60 |

chosen precisely to avoid these types of stabilization terms. These mixed formulations pose additional AMG challenges that must be addressed in order to produce an optimally performing AMG solver. This includes AMG issues associated with the zero diagonal sub-matrix and with possible stability concerns of the automatically generated coarse discretization operators. Some of these issues are discussed in [57], and a second paper is under preparation to discuss the remaining issues. In the current manuscript, we emphasize that the block matrix framework is even necessary to apply AMG to these more complex mixed finite element systems, as most AMG software cannot be applied (without generating some error condition) to systems where DoFs are not co-located. Further, the block framework described here is the most natural way to address the additional AMG issues when stabilization terms are not present. For example, the zero diagonal entries are typically restricted to one or two sub-blocks of the entire system such as the pressure unknowns within the continuity equation. Thus, the block-oriented AMG approach can employ a special procedure to generate a pressure interpolation operator while using a more standard AMG approach for generating interpolation for the other physical quantities. The non-trivial aspects associated with doing this are deferred to a future paper that relies heavily on the block-oriented AMG framework.

TABLE 5.7
*MHD Generator using mixed finite elements.*

Legend:

| | |
|---|---|
| $n_N$ | Accumulated number of all nonlinear iterations |
| $n_L$ | Accumulated number of all linear iterations |
| $t_{Se}$ | Multigrid setup time |
| $t_{So}$ | Multigrid solution time (iteration phase) |

| | Preconditioner | Processors | visc | $n_N$ | $n_L$ | $n_L/n_N$ | $t_{S_e}$ | $t_{So}$ |
|---|---|---|---|---|---|---|---|---|
| $64 \times 32 \times 32$ | AMG(1 BGS (0.6)) | 32 | 0.04 | 5 | 458 | 91.6 | 52.40 | 327.15 |
| | AMG(1 BGS (0.6)) | 32 | 0.01 | 5 | 421 | 84.2 | 52.27 | 295.11 |
| | AMG(1 BGS (0.6)) | 32 | 0.006 | 6 | 970 | 161.7 | 62.73 | 709.46 |
| | Schwarz-DD | 32 | 0.04 | 7 | 3016 | 430.9 | 323.84 | 913.46 |
| | Schwarz-DD | 32 | 0.01 | 11 | 5016 | 456.0 | 510.68 | 1514.91 |
| | Schwarz-DD | 32 | 0.006 | 21 | 10016 | 477.0 | 978.11 | 3029.31 |
| $128 \times 64 \times 64$ | AMG(1 BGS (0.6)) | 256 | 0.04 | 5 | 663 | 132.6 | 56.97 | 519.58 |
| | AMG(1 BGS (0.6)) | 256 | 0.01 | 5 | 475 | 95.0 | 56.80 | 357.89 |
| | AMG(1 BGS (0.6)) | 256 | 0.006 | 6 | 723 | 120.5 | 68.18 | 556.24 |
| | Schwarz-DD | 256 | 0.04 | 15 | 7024 | 468.3 | 758.51 | 2200.64 |
| | Schwarz-DD | 256 | 0.01 | 19 | 9024 | 474.9 | 963.58 | 2831.34 |
| | Schwarz-DD | 256 | 0.006 | 8 | 3524 | 440.5 | 404.60 | 1098.83 |
| $256 \times 128 \times 128$ | AMG(1 BGS (0.6)) | 2048 | 0.04 | 5 | 1080 | 216.0 | 59.93 | 1003.43 |
| | AMG(1 BGS (0.6)) | 2048 | 0.01 | 6 | 993 | 165.5 | 71.45 | 856.84 |
| | AMG(1 BGS (0.6)) | 2048 | 0.006 | 6 | 944 | 157.3 | 71.17 | 805.97 |
| | Schwarz-DD | 2048 | 0.04 | 21 | 10032 | 477.7 | 1103.45 | 3554.11 |
| | Schwarz-DD | 2048 | 0.01 | 22 | 10532 | 478.7 | 1153.60 | 3729.02 |
| | Schwarz-DD | 2048 | 0.006 | 11 | 5032 | 457.4 | 573.59 | 1760.38 |

**6. Conclusion.** A new framework for developing multiphysics multigrid preconditioners is developed and demonstrated on a number of MHD problems. The key idea is to develop the multigrid components in a block fashion that mirrors the blocks in a multiphysics system. Our approach has been to develop block smoothers and apply them to a multigrid hierarchy constructed using block restriction/prolongation operators. In many cases, the blocked multiphysics multigrid hierarchy allows for faster solution times than a non-blocked approach. For mixed spatial discretizations, the multiphysics framework provides the only genuine avenue to leverage pre-existing multigrid software to produce a monolithic multigrid preconditioner. Here, the AMG engine is invoked multiple times for different sub-blocks, and the resulting individual grid transfers are combined into one composite operator that can be employed in a monolithic AMG fashion. The run time benefits of the blocked multilevel approach are more evident for the mixed spatial discretization, as a non-blocked multilevel approach is not suitable, leaving non-multilevel preconditioners for comparison. While this paper has focused on specific examples and MHD, the goal of the framework is to be able to easily construct, adapt, and tailor different monolithic multigrid preconditioners to various PDE systems.

## REFERENCES

[1] A. Y. AYDEMIR AND D. C. BARNES, *An implicit algorithm for compressible three-dimensional magnetohydrodynamic calculations*, J. Comput. Phys., 59 (1985), pp. 108–119.

[2] S. BADIA, R. CODINA, AND R. PLANAS, *On an unconditionally convergent stabilized finite element approxi-*

*mation of resistive magnetohydrodynamics*, J. Comput. Phys., 234 (2013), pp. 399–416.

[3] L. Berger-Vergiat, C. A. Glusa, J. J. Hu, M. Mayr, A. Prokopenko, C. M. Siefert, R. S. Tuminaro, and T. A. Wiesner, *MueLu multigrid framework*, Software, 2019. http://trilinos.org/packages/muelu

[4] ———, *MueLu user's guide*, Tech. Rep. SAND2019-0537, Sandia National Laboratories, Albuquerque, 2019.

[5] D. Biskamp, *Magnetic Reconnection in Plasmas*, Cambridge University Press, Cambridge, 2000.

[6] P. B. Bochev, J. J. Hu, A. C. Robinson, and R. S. Tuminaro, *Towards robust 3D Z-pinch simulations: discretization and fast solvers for magnetic diffusion in heterogeneous conductors*, Electron. Trans. Numer. Anal., 15 (2003), pp. 186–210. http://etna.ricam.oeaw.ac.at/vol.15.2003/pp186-210.dir/pp186-210.pdf

[7] W. L. Briggs, V. E. Henson, and S. McCormick, *A Multigrid Tutorial*, 2nd ed., SIAM, Philadelphia, 2000.

[8] L. Chacón, *A non-staggered, conservative, $\nabla \cdot \vec{B} = 0$, finite-volume scheme for 3D implicit extended magnetohydrodynamics in curvilinear geometries*, Comput. Phys. Comm., 163 (2004), pp. 143–171.

[9] ———, *An optimal, parallel, fully implicit Newton-Krylov solver for three-dimensional visco-resistive magnetohydrodynamics*, Phys. Plasmas, 15 (2008), Art. 056103, 12 pages.

[10] R. Codina and N. Hernández, *Stabilized finite element approximation of the stationary magento-hydrodyanamics equations*, Comput. Mech., 38 (2006), pp. 344–355.

[11] ———, *Approximation of the thermally coupled MHD problem using a stabilized finite element method*, J. Comput. Phys., 230 (2011), pp. 1281–1303.

[12] E. C. Cyr, J. N. Shadid, R. S. Tuminaro, R. P. Pawlowski, and L. Chacón, *A new approximate block factorization preconditioner for 2D incompressible (reduced) resistive MHD*, SIAM J. Sci. Comput., 35 (2013), pp. B701–B730.

[13] C. Danowski, V. Gravemeier, L. Yoshihara, and W. A. Wall, *A monolithic computational approach to thermo-structure interaction*, Internat. J. Numer. Methods Engrg., 95 (2013), pp. 1053–1078.

[14] P. A. Davidson, *An Introduction to Magnetohydrodynamics*, Cambridge University Press, Cambridge, 2001.

[15] A. Dedner, F. Kemm, D. Kröner, C.-D. Munz, T. Schnitzer, and M. Wesenberg, *Hyperbolic divergence cleaning for the MHD equations*, J. Comput. Phys., 175 (2002), pp. 645–673.

[16] S. Deparis, D. Forti, G. Grandperrin, and A. Quarteroni, *FaCSI: a block parallel preconditioner for fluid-structure interaction in hemodynamics*, J. Comput. Phys., 327 (2016), pp. 700–718.

[17] J. Donea and A. Huerta, *Finite Element Methods for Flow Problems*, Wiley, Chichester, 2002.

[18] J. F. Drake and T. M. Antonsen, *Nonlinear reduced fluid equations for toroidal plasmas*, Phys. Fluids, 27 (1984), pp. 898–908.

[19] S. C. Eisenstat and H. F. Walker, *Globally convergent inexact Newton methods*, SIAM J. Optim., 4 (1994), pp. 393–422.

[20] ———, *Choosing the forcing terms in an inexact Newton method*, SIAM J. Sci. Comput., 17 (1996), pp. 16–32.

[21] 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, New York, 2005.

[22] V. M. Fadeev, I. F. Kvartskhava, and N. N. Komarov, *Self-focusing of local plasma currents*, Nuclear Fusion, 5 (1965), pp. 202–209.

[23] T. Füllenbach and K. Stüben, *Algebraic multigrid for selected PDE systems*, in Elliptic and Parabolic problems (Rolduc/Gaeta, 2001), J. Bemelmans, B. Brighi, A. Brillard, M. Chipot, F. Conrad, I. Shafrir, V. Valente, and G. Vergara-Caffarelli, eds., World Scientific, River Edge, 2002, pp. 399–410.

[24] M. W. Gee, U. Küttler, and W. A. Wall, *Truly monolithic algebraic multigrid for fluid-structure interaction*, Internat. J. Numer. Methods Engrg., 85 (2011), pp. 987–1016.

[25] H. Goedbloed and S. Poedts, *Principles of Magnetohydrodynamics*, Cambridge Univ. Press, Cambrigdge, 2004.

[26] D. S. Harned and W. Kerner, *Semi-implicit method for three-dimensional compressible magnetohydrodynamic simulation*, J. Comput. Phys., 60 (1985), pp. 62–75.

[27] D. S. Harned and Z. Mikic, *Accurate semi-implicit treatment of the Hall effect in magnetohydrodynamic computations*, J. Comput. Phys., 83 (1989), pp. 1–15.

[28] R. D. Hazeltine, M. Kotschenreuther, and P. J. Morrison, *A four-field model for tokamak plasma dynamics*, Phys. Fluids, 28 (1985), pp. 2466–2477.

[29] M. A. Heroux, R. A. Bartlett, V. E. Howle, R. J. Hoekstra, J. J. Hu, T. G. Kolda, R. B. Lehoucq, K. R. Long, R. P. Pawlowski, E. T. Phipps, A. G. Salinger, H. K. Thornquist, R. S. Tuminaro, J. M. Willenbring, A. Williams, and K. S. Stanley, *An overview of the Trilinos Project*, ACM Trans. Math. Software, 31 (2005), pp. 397–423.

[30] T. Hughes, *Multiscale phenomena: Green's function, the Dirichlet-to-Neumann map, subgrid scale models, bubbles and the origins of stabilized methods*, Comput. Methods Appl. Mech. Engrg., 127 (1995), pp. 387–401.

[31] A. Hujeirat, *IRMHD: an implicit radiative and magnetohydrodynamical solver for self-gravitating systems*, Monthly Not. Roy. Astr. Soc., 298 (1998), pp. 310–320.

[32] S. C. JARDIN, *Review of implicit methods for the magnetohydrodynamic description of magnetically confined plasmas*, J. Comput. Phys., 231 (2012), pp. 822–838.

[33] S. C. JARDIN AND J. A. BRESLAU, *Implicit solution of the four-field extended-magnetohydrodynamic equations using high-order high-continuity finite elements.*, Phys. Plasmas, 12 (2005), Art. 056101, 10 pages.

[34] D. JODLBAUER, U. LANGER, AND T. WICK, *Parallel block-preconditioned monolithic solvers for fluid-structure interaction problems*, Internat. J. Numer. Methods Engrg., 117 (2019), pp. 623–643.

[35] R. KEPPENS, G. TÓTH, M. A. BOTCHEV, AND A. V. D. PLOEG, *Implicit and semi-implicit schemes: algorithms*, Internat. J. Numer. Methods Fluids, 30 (1999), pp. 335–352.

[36] D. A. KNOLL AND L. CHACÓN, *Coalescence of magnetic islands, sloshing, and the pressure problem*, Phys. Plasmas, 13 (2006), Art. 032307, 7 pages.

[37] U. LANGER AND H. YANG, *Robust and efficient monolithic fluid-structure-interaction solvers*, Internat. J. Numer. Methods Engrg., 108 (2016), pp. 303–325.

[38] P. T. LIN, J. N. SHADID, R. S. TUMINARO, M. SALA, G. L. HENNIGAN, AND R. P. PAWLOWSKI, *A parallel fully coupled algebraic multilevel preconditioner applied to multiphysics PDE applications: drift-diffusion, flow/transport/reaction, resistive MHD*, Internat. J. Numer. Methods Fluids, 64 (2010), pp. 1148–1179.

[39] R. MOREAU, *Magneto-Hydrodynamics*, Kluwer, Dordrecht, 1990.

[40] J.-C. NÉDÉLEC, *Mixed finite elements in* $\mathbf{R}^3$, Numer. Math., 35 (1980), pp. 315–341.

[41] W. PARK, J. BRESLAU, J. CHEN, G. Y. FU, S. C. JARDIN, S. KLASKY, J. MENARD, A. PLETZER, B. C. STRATTON, D. STUTMAN, H. R. STRAUSS, AND L. E. SUGIYAMA, *Nonlinear simulation studies of tokamaks and STs*, Nuclear Fusion, 43 (2003), pp. 483–489.

[42] E. G. PHILLIPS, H. C. ELMAN, E. C. CYR, J. N. SHADID, AND R. P. PAWLOWSKI, *Block preconditioners for stable mixed nodal and edge finite element representations on incompressible resistive MHD*, SIAM J. Sci. Comput., 38 (2016), pp. B1009–B1031.

[43] A. C. ROBINSON ET. AL., *ALEGRA: An arbitrary Lagrangian-Eulerian multimaterial, multiphysics code*, in AIAA 2008-1235 46th AIAA Aerospace Sciences Meeting and Exhibit, AIAA, Reston, 2008.

[44] D. D. SCHNACK, D. C. BARNES, D. S. HARNED, AND E. J. CARAMANA, *Semi-implicit magnetohydrodynamic calculations*, J. Comput. Phys., 70 (1987), pp. 330–354.

[45] D. SCHÖTZAU, *Mixed finite element methods for stationary incompressible magneto-hydrodynamics*, Numer. Math., 96 (2004), pp. 771–800.

[46] J. N. SHADID, *A fully-coupled Newton-Krylov solution method for parallel unstructured finite element fluid flow, heat and mass transfer simulations*, Int. J. Comput. Fluid Dyn., 12 (1999), pp. 199–211.

[47] J. N. SHADID, R. P. PAWLOWSKI, J. W. BANKS, L. CHACÓN, P. T. LIN, AND R. S. TUMINARO, *Towards a scalable fully-implicit fully-coupled resistive MHD formulation with stabilized FE methods*, J. Comput. Phys., 229 (2010), pp. 7649–7671.

[48] J. N. SHADID, R. P. PAWLOWSKI, E. C. CYR, R. S. TUMINARO, L. CHACÓN, AND P. D. WEBER, *Scalable implicit incompressible resistive MHD with stabilized FE and fully-coupled Newton-Krylov-AMG*, Comput. Methods Appl. Mech. Engrg., 304 (2016), pp. 1–25.

[49] J. N. SHADID, R. S. TUMINARO, K. D. DEVINE, G. L. HENNIGAN, AND P. T. LIN, *Performance of fully coupled domain decomposition preconditioners for finite element transport/reaction simulations*, J. Comput. Phys., 205 (2005), pp. 24–47.

[50] J. N. SHADID, R. S. TUMINARO, AND H. F. WALKER, *On backtracking failure in Newton–*GMRES *methods with a demonstration for the Navier–Stokes equations*, J. Comput. Phys., 180 (2002), pp. 549–558.

[51] C. R. SOVINEC, A. H. GLASSER, T. A. GIANAKON, D. C. BARNES, R. A. NEBEL, S. E. KRUGER, D. D. SCHNACK, S. J. PLIMPTON, A. TARDITI, M. S. CHU, AND THE NIMROD TEAM, *Nonlinear magnetohydrodynamics simulation using high-order finite elements*, J. Comput. Phys., 195 (2004), pp. 355–386.

[52] H. R. STRAUSS, *Nonlinear, 3-dimensional magnetohydrodynamics of noncircular tokamaks*, Phys. Fluids, 19 (1976), pp. 134–140.

[53] G. TÓTH, *The* $\nabla \cdot B = 0$ *constraint in shock-capturing magnetohydrodynamics codes*, J. Comput. Phys., 161 (2000), pp. 605–652.

[54] G. TÓTH, R. KEPPENS, AND M. A. BOTCHEV, *Implicit and semi-implicit schemes in the Versatile Advection Code: numerical tests*, Astron. Astrophys., 332 (1998), pp. 1159–1170.

[55] U. TROTTENBERG, C. W. OOSTERLEE, AND A. SCHÜLLER, *Multigrid*, Academic Press, San Diego, 2001.

[56] F. VERDUGO AND W. A. WALL, *Unified computational framework for the efficient solution of* $n$*-field coupled problems with monolithic schemes*, Comput. Methods Appl. Mech. Engrg., 310 (2016), pp. 335–366.

[57] A. VORONIN, Y. HE, S. MACLACHLAN, L. OLSON, AND R. TUMINARO, *Low order preconditioners for the Stokes equations*, Preprint on arXiv, 2021. https://arxiv.org/abs/2103.11967

[58] T. WIESNER, *Flexible Aggregation-based Algebraic Multigrid Methods for Contact and Flow Problems*, PhD. Thesis, Institue for Computational Mechanics, Technische Universität München, Munich, 2015.

[59] T. A. WIESNER, M. MAYR, A. POPP, M. W. GEE, AND W. A. WALL, *Algebraic multigrid methods for*

*saddle point systems arising from mortar contact formulations*, Internat. J. Numer. Methods Engrg., 122 (2021), pp. 3749–3779.

[60] T. A. WIESNER, R. S. TUMINARO, W. A. WALL, AND M. W. GEE, *Multigrid transfers for nonsymmetric systems based on Schur complements and Galerkin projections*, Numer. Linear Algebra Appl., 21 (2014), pp. 415–438.