FAST PARALLEL SOLUTION OF FULLY IMPLICT RUNGE-KUTTA AND DISCONTINOUS GALERKIN IN TIME FOR NUMERICAL PDES, PART I: THE LINEAR SETTING*

BEN S. SOUTHWORTH†, OLIVER KRZYSIK ‡, AND WILL PAZNER§

Abstract.

{sec:intro}

1. Introduction.

{sec:intro:irk}

1.1. Fully implicit Runge-Kutta. Consider the method-of-lines approach to the numerical solution of linear partial differential equations (PDEs), where we discretize in space and arrive at a system of ordinary differential equations (ODEs) in time,

{eq:problem}
$$M\mathbf{u}'(t) = \mathcal{L}(t)\mathbf{u} + \mathbf{f}(t) \quad \text{in } (0,T], \quad \mathbf{u}(0) = \mathbf{u}_0, \tag{1}$$

where M is a mass matrix, $\mathcal{L}(t) \in \mathbb{R}^{N\times N}$ a discrete linear operator, and $\mathbf{f}(t)$ a time-dependent forcing function.

[1] Then, consider time propagation using an s-stage Runge-Kutta scheme, characterized by the Butcher tableaux

$$\begin{array}{c|c} \mathbf{c}_0 & A_0 \\ \hline & \mathbf{b}_0^T \end{array},$$

with Runge-Kutta matrix $A_0 = (a_{ij})$, weight vector $\mathbf{b}_0^T = (b_1, \ldots, b_s)^T$, and quadrature nodes $\mathbf{c}_0 = (c_0, \ldots, c_s)$.

Runge-Kutta methods update the solution using a sum over stage vectors,

{eq:update}
$$\mathbf{u}_{n+1} = \mathbf{u}_n + \delta t \sum_{i=1}^{s} b_i \mathbf{k}_i, \tag{2}$$

{eq:stages}
$$\mathbf{k}_i = \mathcal{L}\left(\mathbf{u}_n + \delta t \sum_{j=1}^{s} a_{ij}\mathbf{k}_j, t_n + \delta t c_i\right) + \mathbf{f}(t_n + \delta t c_i). \tag{3}$$

Expanding, solving for the stages $\mathbf{k}$ can then be expressed as the solution of the block linear system,

{eq:kblock}
$$\left(\begin{bmatrix} M & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & M \end{bmatrix} - \delta t \begin{bmatrix} a_{11}\mathcal{L}_1 & \ldots & a_{1s}\mathcal{L}_1 \\ \vdots & \ddots & \vdots \\ a_{s1}\mathcal{L}_s & \ldots & a_{ss}\mathcal{L}_s \end{bmatrix}\right) \begin{bmatrix} \mathbf{k}_1 \\ \vdots \\ \mathbf{k}_s \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_s \end{bmatrix}, \tag{4}$$

where $\mathcal{L}_i := \mathcal{L}(t_n + \delta t c_i)$. Primarily this paper focuses on spatial operators that are independent of time; however, some of the results hold for commuting operators, such

*This research was conducted ...

†Department of Applied Mathematics, University of Colorado, U.S.A. (ben.s.southworth@gmail.com), http://orcid.org/0000-0002-0283-4928

‡School of Mathematical Sciences, Monash University, Australia (oliver.krzysik@monash.edu)

§Lawrence Livermore National Laboratory, U.S.A. (pazner1@llnl.gov)

[1] Note, PDEs with an algebraic constraint, for example, the divergence-free constraint in Navier Stokes, instead yield a differential algebraic equation (DAE), which requires separate careful treatment and will be the subject of a forthcoming paper.

as may arise in, for example, a time-dependent reaction term, so for now we maintain this generality.

The difficulty in fully implicit Runge-Kutta methods (which we will denote IRK) lies in solving the $Ns \times Ns$ block linear system in (4). This paper focuses on the parallel simulation of numerical PDEs, where N is typically very large and $\mathcal{L}$ is highly ill-conditioned. In such cases, direct solution techniques to solve (4) are not a viable option, and fast, parallel iterative methods must be used. However, IRK methods are rarely employed in practice due to the difficulties of solving (4). Even for relatively simple parabolic PDEs where $\mathcal{L}$ is symmetric positive definite (SPD), (4) instead yields a large nonsymmetric system with significant block coupling. For nonsymmetric systems $\mathcal{L}$ that already have variable coupling, fast iterative methods are even less likely to yield acceptable performance in solving (4).

sec:intro:dg}

1.2. Discontinuous Galerkin in time. Another field that has seen growing interest recently for numerical PDEs is discretizing in time using finite elements, rather than standard ODE techniques. Some of the more interesting features of using finite-elements in time is the natural ability to handle time-dependent domains, unstructured meshes in space-time, and adaptive mesh refinement in space and time. However, for so-called slab-based meshing in time with discontinuous Galerkin finite elements, where the time domain is discretized using spatial slabs (that is, the same time step is applied to the entire spatial domain), the resulting linear systems that must be solved for each time step take the same structure as (4) (for example, see [23]). A handful of works have looked at linear solvers for such discretizations, primarily for parabolic problems, including block preconditioning approaches [2, 21, 24], and direct space-time multigrid methods [9].

Although this paper focuses on fully implicit Runge-Kutta, the algorithms developed here can be directly applied to discontinuous Galerkin discretizations in time on fixed slab-based meshes. In fact, the principles used in this paper are similar to those used in [2] for space-time DG discretizations of linear parabolic problems, and some of the theory derived therein...

but here we consider much more general settings, including both non-parabolic and nonlinear problems.

ntro:outline}

1.3. Outline. This paper develops fast, parallel preconditioning techniques for the solution of fully implicit Runge-Kutta methods in numerical PDEs. Although we focus on implicit Runge-Kutta, the techniques developed extend naturally to DG finite elements in time as well, as discussed in Subsection 1.2. First, Subsection 1.4 provides background on why IRK methods are desirable over the simpler and more commonly used DIRK methods, and also provides a historical context for the preconditioners developed in this work. Subsection 1.5 then briefly discusses stable integration from a method-of-lines perspective and introduces two key elements that will be used throughout the paper.

Section 2 introduces an effective method to solve for the IRK update (2) directly for linear operators $\mathcal{L}$ that are independent of time, such as those that typically arise in linear PDEs. The new method effectively requires the preconditioning of s real-valued matrices along the lines of $\eta M - \delta t \mathcal{L}$, for some $\eta > 1$, and is easily implemented using existing preconditioners and parallel software libraries. In contrast to other works that have considered the preconditioning of (5), the proposed algorithm here (i) is amenable to short-term Krylov recursion (conjugate gradient (CG)/MINRES) if $\eta M - \mathcal{L}$ is, and (ii) only operates on the solution, thus not requiring the storage of each stage vector. Theory is developed that guarantees rapid convergence of GMRES and

CG under basic assumptions on stability from Subsection 1.5. Numerical results for the linear setting are provided in Section 3, demonstrating up to 10th-order accuracy using Gauss integration with a variety of problems and preconditioners. In addition, by using the new method with 2-stage Gauss integration, we are able to halve the number of AMG preconditioning iterations necessary to obtain 4th-order accuracy on an advection-diffusion example compared with standard 4th-order SDIRK schemes.

{sec:intro:hist}

1.4. Why fully implicit and previous work. [TODO: Add citations] Although difficult to solve, there are a number of desirable properties of IRK schemes, particularly in terms of accuracy and stabillity ([TODO: why stability?]). In practice, people typically use diagonally implicit Runge-Kutta (DIRK) methods, where A_0 is lower triangular, or singly implicit Runge Kutta (SIRK) methods, where A_0 has exactly one positive real eigenvalue. For such schemes, the solution of (4) only requires s linear solves along the lines of $M-\delta t a_{ii}\mathcal{L}_i$. Unfortunately, SIRK and DIRK schemes suffer from stage-order one (stage-order two with one explicit stage; see EDIRK methods), and for stiff nonlinear PDEs, the observed global order of accuracy in practice can be limited to $\approx \min\{p, q+1\}$, for integration order p and stage-order q. Thus, even a 6th-order DIRK method may only yield 2nd-order accuracy. Accuracy is even worse for index-2 DAEs, where the algebraic variable is limited to first-order accuracy with DIRK methods [TODO: cite Hairer] (which will be addressed as a follow-up to this paper). In contrast, IRK methods yield high stage order and, thus, formally high-order accuracy on stiff, nonlinear problems, and even index-2 DAEs. Furthermore, for less stiff problems, IRK methods can yield accuracy as high as order $2s$ for an s stage method, compared with a maximum of s or $s+1$ for SDIRK methods with reasonable stability properties [11, Section IV.6].

One simplification for using IRK methods in practice is to assume $\mathcal{L}_i = \mathcal{L}_j$ for all i, j, that is, $\mathcal{L}$ has no dependence on time. Such an assumption is natural for linear problems with no time-dependence in the spatial differential components, or when applying a simplified Newton method to nonlinear problems, where the Jacobian is only evaluated at one time-point per outer RK time step. Either case yields a simplified form of (4) that can be expressed in Kronecker product form,

$$(I \otimes M - \delta t A_0 \otimes \mathcal{L})\mathbf{k} = \mathbf{f}, \tag{5}$$

{eq:kron1}

where $\mathcal{L}$ is a fixed real-valued spatial operator or Jacobian.

Many papers have considered the solution of (5). In 1976, Butcher [7] used the Jordan normal form, $A_0 = U_0 L_0 U_0^{-1}$, where L_0 is lower triangular with eigenvalues of A_0 on the diagonal, to transform (5) to the problem $(I \otimes M - \delta t L_0 \otimes \mathcal{L})(U_0^{-1} \otimes I)\mathbf{k} = (U_0 \otimes I)^{-1}\mathbf{f}$, where the inner operator is now block lower triangular. Such a transformation reduces the solution of an $Ns \times Ns$ system to s linear systems of size $N \times N$ in a block forward solve. The downside is that IRK schemes with greater accuracy and stability than DIRK schemes have at most one real eigenvalue [8, 11]. Thus, for IRK methods such as Gauss or Radau integration, the original real-valued system is transformed into a set of smaller but primarily complex systems. There are various ways to handle the complex systems, but for numerial PDEs, the overhead in computational cost and implementation is typically too high to make the transformation a practical approach.

Published shortly after (and independently from) Butcher, Bickart proposed a similar way to invert (5) [3]. If we define $Q_s(x)$ as the characteristic polynomial of A_0, then the inverse of (5) can be computed via a specific set of matrix-vector multiplications in addition to the action of $Q_s(\mathcal{L})^{-1}$. In principle this is similar to Butcher's

result, as one can invert $Q_s(\mathcal{L})$ by inverting each term in the factored polynomial, $(\mu_1 I - \mathcal{L})^{-1}$, $(\mu_2 I - \mathcal{L})^{-1}$, ..., for eigenvalues $\{\mu_i\}_{i=1}^s$ of A_0. Although Bickart's paper received less attention than Butcher's over time (currently 2.5× less citations), the polynomial form provides a more natural way to handle complex eigenvalues, particularly for numerical PDEs in the modern high-performance computing landscape, where direct LU inverses are rare and most linear systems are solved via preconditioning and/or Krylov methods. We present a similar result in Lemma 3 and use this to develop an effective preconditioning for linear problems in Subsection 2.2.

Because significant research has been done on IRK methods, it is worth briefly reviewing some of the literature to put this work in context. In the field of time integration, SIRK and DIRK methods overcome the difficulty of solving IRK methods [1, 16]. In [18], it is shown that in considering RK methods with all real eigenvalues (wherein the Butcher transformation [7] remains real-valued), the best approximation to exponential is obtained by having all eigenvalues equal (SIRK methods [16]). Although SIRK methods offer some advantages over DIRK methods, they generally lack the favorable stability and accuracy of IRK methods [5, 18]. ESIRK methods use one explicit stage and can offer stage-order two [6] (one higher than standard SIRK methods), which provide an improved practical option, but still lack the robustness of fully implicit methods.

{sec:intro:stab}

1.5. A preconditioning framework and stability. Throughout the paper, we use the reformulation used in, for example, [19], where we can pull an $A_0 \otimes I$ out of the fully implicit system in (4), yielding the equivalent problem

$$\left(A_0^{-1} \otimes M - \delta t \begin{bmatrix} \mathcal{L}_1 & & \\ & \ddots & \\ & & \mathcal{L}_s \end{bmatrix}\right) (A_0 \otimes I) \begin{bmatrix} \mathbf{k}_1 \\ \vdots \\ \mathbf{k}_s \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_s \end{bmatrix}. \tag{6}$$

The off-diagonal block coupling in (6) now consists of mass matrices rather than differential operators, which makes the analysis and solution more tractable.[2] The algorithms developed here depend on the eigenvalues of A_0 and A_0^{-1}, leading to our first assumption.

{ass:eig}

Assumption 1. Assume that all eigenvalues of A_0 have positive real part.

Recall that if an IRK method is A-stable, irreducible, and A_0 is invertible (which includes Gauss, RadauIIA, and LobattoIIIC integration, among others), then Assumption 1 holds [11]; that is, this assumption is straightforward to satisfy in practice.

Stability must be taken into consideration when applying ODE solvers within a method-of-lines approach to numerical PDEs. The Dalhquist test problem extends naturally to this setting, where we are interested in the stability of the linearized operator $\mathcal{L}$, for the ODE(s) $\mathbf{u}'(t) = \mathcal{L}\mathbf{u}$, with solution $e^{t\mathcal{L}}\mathbf{u}$. A necessary condition for stability is that the eigenvalues of $\mathcal{L}$ lie within distance $\mathcal{O}(\delta t)$ of the region of stability for the Runge-Kutta scheme of choice (e.g., see [20]). Here we are interested in implicit schemes and, because the majority of implicit Runge-Kutta schemes used in practice are A- or L-stable, an effectively necessary condition for stability is that the eigenvalues of $\mathcal{L}$ be nonpositive. For normal operators, this requirement ends up being a necessary and sufficient condition for stability, as the eigenvectors form an orthogonal basis.

[2] For some of the methods introduced in this paper, more-or-less equivalent algorithms can be developed for (4) (that is, without extracting $A_0 \otimes I$). However, the derivation and analysis therein is more difficult, and the resulting algorithms offer no clear benefits over the presented methods.

For non-normal or non-diagonalizable operators, the analysis is more complicated. One of the best known works on the subject is by Reddy and Trefethen [20], where necessary and sufficient conditions for stability are derived as the ε pseudo-eigenvalues of $\mathcal{L}$ being within $\mathcal{O}(\varepsilon) + \mathcal{O}(\delta t)$ of the stability region as $\varepsilon, \delta t \to 0$. Here we relax this assumption to something that is more tractable to work with by noting that the ε pseudo-eigenvalues are contained within the field-of-values to $\mathcal{O}(\varepsilon)$ [25, Eq. (17.9)], where the field of values are defined as

$$W(\mathcal{L}) := \{\langle \mathcal{L}\mathbf{x}, \mathbf{x}\rangle : \|\mathbf{x}\| = 1\}. \tag{7}$$ {eq:fov}

This motivates the following assumption for the analysis done in this paper. {ass:fov}

Assumption 2. Let $\mathcal{L}$ be the linearized spatial operator representing a single time step. Assume $W(\mathcal{L}) \leq 0$.

It should be noted that the field-of-values have an additional connection to stability. From [25, Theorem 17.1], we have that $\|e^{t\mathcal{L}}\| \leq 1$ for all $t \geq 0$ i.f.f. $W(\mathcal{L}) \leq 0$. This is analogous to the "strong stability" discussed by Leveque [12, Chapter 9.5], as opposed to the weaker (but still sufficient) condition $\|e^{t\mathcal{L}}\| \leq C$ for all $t \geq 0$ and some constant C. In practice, Assumption 2 is likely to hold when simulating numerical PDEs, and in Subsection 2.2, it is proven that Assumption 1 and 2 provide sufficient conditions to guarantee fast Krylov convergence of the proposed methods.

2. Fast parallel preconditioning. {sec:solve}

For ease of notation, let us scale both sides of the (6) by a block-diagonal mass matrix inverse and let

$$\widehat{\mathcal{L}}_i := \delta t M^{-1} \mathcal{L}_i,$$

for $i = 1, .., s$. Note the time step δt for the given Runge-Kutta step is now included in $\widehat{\mathcal{L}}_i$. Now let α_{ij} denote the ij-element of A_0^{-1} (assuming A_0 is invertible). Then, solving (4) can be effectively reduced to inverting the operator[3]

$$\mathcal{M}_s := A_0^{-1} \otimes I - \begin{bmatrix} \widehat{\mathcal{L}}_1 & & \\ & \ddots & \\ & & \widehat{\mathcal{L}}_s \end{bmatrix}$$

$$= \begin{bmatrix} \alpha_{11} I - \widehat{\mathcal{L}}_1 & \alpha_{12} I & \dots & \alpha_{1s} I \\ \alpha_{21} I & \alpha_{22} I - \widehat{\mathcal{L}}_2 & & \alpha_{2s} I \\ \ddots & & \ddots & \vdots \\ \alpha_{s1} I & \dots & \alpha_{s(s-1)} I & \alpha_{ss} I - \widehat{\mathcal{L}}_s \end{bmatrix}. \tag{8}$$ {eq:k1}

We proceed by deriving a closed form inverse (8), demonstrating how the Runge-Kutta update in (2) can then be performed directly (without forming and saving each stage vector), and developing a preconditioning strategy to apply this update using existing fast, parallel preconditioners. In Subsection 2.3, analysis proves that with the right choice of constant in the preconditioner, the preconditioned operator has condition number < 9, independent of problem size, order, and number of stages.

[3]Note, there are a number of methods with one explicit stage preceded or followed by several fully implicit and coupled stages. In such cases, A_0 is not invertible, but the explicit stage can be eliminated from the system (by doing an explicit time step). The remaining operator can then be reformulated as in (6), and the inverse that must be applied takes the form of (8) but based on a principle submatrix of A_0.

{sec:solve:inv}

2.1. An inverse and update for commuting operators. This section introduces a result similar to Bickart's [3], but using a different framework and generalized to hold for commuting operators. If $\widehat{\mathcal{L}}_i = \widehat{\mathcal{L}}_j$ for all i, j, we show that the inverse of (8) can be expressed in terms of $P_s(\widehat{\mathcal{L}})^{-1}$, where $P_s(\widehat{\mathcal{L}})$ is the characteristic polynomial of A_0^{-1}. Note, we consider $\mathcal{M}_s$ as a matrix over the commutative ring of linear combinations of $\{I, \mathcal{L}\}$, and the determinant and adjugate referred to in Lemma 3 are defined over matrix-valued elements rather than scalars.

{lem:inv}

LEMMA 3. *Let α_{ij} denote the (i,j)th entry of A_0^{-1} and assume $\{\widehat{\mathcal{L}}_i\}_{i=1}^s$ are commuting operators. Define $\mathcal{M}_s$ as in (8). Let det($\mathcal{M}_s$) be the determinant of $\mathcal{M}_s$ (in this case a block-diagonal matrix) and let adj($\mathcal{M}_s$) be the adjugate of $\mathcal{M}_s$. Then, $\mathcal{M}_s$ is invertible if and only if det($\mathcal{M}_s$) is invertible, and*

$$\mathcal{M}_s^{-1} = \det(\mathcal{M}_s)^{-1}\mathrm{adj}(\mathcal{M}_s).$$

Now, suppose $\widehat{\mathcal{L}}_i = \widehat{\mathcal{L}}_j$ for all i, j, and let $P_s(x)$ be the characteristic polynomial of A_0^{-1}. Then,

$$\mathcal{M}_s^{-1} = \mathrm{diag}(P_s(\widehat{\mathcal{L}})^{-1})\mathrm{adj}(\mathcal{M}_s),$$

where "diag" indicates a block diagonal matrix, with diagonal blocks given by $P_s(\widehat{\mathcal{L}})^{-1}$.

Proof. Notice in (8) that if $\widehat{\mathcal{L}}_i$ and $\widehat{\mathcal{L}}_j$ commute for all i, j, then $\mathcal{M}_s$ is a matrix over the commutative ring of linear combinations of I and $\{\widehat{\mathcal{L}}_i\}$. Let adj($\mathcal{M}_s$) denote the matrix adjugate. A classical result in matrix analysis then tells us that

$$\mathrm{adj}(\mathcal{M}_s)\mathcal{M}_s = \mathcal{M}_s\mathrm{adj}(\mathcal{M}_s) = \det(\mathcal{M}_s)I.$$

Moreover, $\mathcal{M}_s$ is invertible if and only if if the determinant of $\mathcal{M}_s$ is invertible, in which case $\mathcal{M}_s^{-1} := \det(\mathcal{M}_s)^{-1}\mathrm{adj}(\mathcal{M}_s)$ [4, Theorem 2.19 & Corollary 2.21] For the case of time-independent operators ($\widehat{\mathcal{L}}_i = \widehat{\mathcal{L}}_j$), notice that $\mathcal{M}_s$ takes the form $A_0^{-1} - \widehat{\mathcal{L}}I$ over the commutative ring defined above. Analogous to a scalar matrix, the determinant of $A_0^{-1} - \widehat{\mathcal{L}}I$ is the characteristic polynomial of A_0^{-1} evaluated at $\widehat{\mathcal{L}}$. ∎

Returning to (6), we can express the direct solution for the set of all stage vectors $\mathbf{k} = [\mathbf{k}_1; ...; \mathbf{k}_s]$ as

$$\mathbf{k} := \det(\mathcal{M}_s)^{-1}(A_0^{-1} \otimes I)\mathrm{adj}(\mathcal{M}_s)\mathbf{f},$$

where $\mathbf{f} = [\mathbf{f}_1; ...; \mathbf{f}_s]$ (note that $A_0 \otimes I$ commutes with $\det(\mathcal{M}_s)^{-1}$). Excusing the slight abuse in notation, let $\det(\mathcal{M}_s)^{-1}$ now denote just the diagonal block (rather than a block-diagonal matrix). The Runge-Kutta update is then given by

$$\begin{aligned}\mathbf{u}_{n+1} &= \mathbf{u}_n + \delta t \sum_{i=1}^{s} b_i \mathbf{k}_i \\ &= \mathbf{u}_n + \delta t \det(\mathcal{M}_s)^{-1}(\mathbf{b}_0^T A_0^{-1} \otimes I)\mathrm{adj}(\mathcal{M}_s)\mathbf{f}. \end{aligned} \tag{9}$$

Remark 4 (Implementation & complexity). The adjugate consists of linear combinations of I and $\widehat{\mathcal{L}}$, and an analytical form can be derived for an arbitrary $s \times s$ matrix, where $s \sim \mathcal{O}(1)$. Applying its action requires a set of vector summations and matrix-vector multiplications. In particular, the diagonal elements of adj($\mathcal{M}_s$) are

monic polynomials in $\widehat{\mathcal{L}}$ of degree $s-1$ (or linear combinations of comparable degree if $\widehat{\mathcal{L}}_i \neq \widehat{\mathcal{L}}_j$) and off-diagonal terms are polynomials in $\widehat{\mathcal{L}}$ of degree $s-2$.

Returning to (9), we consider two cases. First, if a given Runge-Kutta scheme is stiffly accurate (for example, RadauIIA methods), then $\mathbf{b}_0^T A_0^{-1} = [0, ..., 0, 1]$. This yields the nice simplification that computing the update in (9) only requires applying the last row of adj$(\mathcal{M}_s)$ to $\mathbf{f}$ (in a dot-product sense) and applying $\det(\mathcal{M}_s)^{-1}$ to the result. From the discussion above regarding the adjugate structure, applying the last row of adj$(\mathcal{M}_s)$ requires $(s-2)(s-1)+(s-1) = (s-1)^2$ matrix-vector multiplications. Because this only happens once, followed by the linear solve(s), these multiplications are typically of relatively marginal cost.

In the more general case of non stiffly accurate (for example, Gauss methods), one can obtain an analytical form for $(\mathbf{b}_0^T A_0^{-1} \otimes I)adj(\mathcal{M}_s)$. Each element in this block $1 \times s$ matrix consists of polynomials in $\widehat{\mathcal{L}}$ of degree $s-1$ (although typically not monic). Compared with stiffly accurate schemes, this now requires $(s-1)s$ matrix-vector multiplications, which is $s-1$ more than for stiffly accurate schemes, but still typically of marginal overall computational cost.

{sec:solve:prec}

2.2. Preconditioning by conjugate pairs. Following the discussion and algorithm developed in Subsection 2.1, the key outstanding point is inverting $\det(\mathcal{M}_s)^{-1}$. Moving forward, we restrict our attention to the case $\widehat{\mathcal{L}}_i = \widehat{\mathcal{L}}_j$ for all i, j, in which case $\det(\mathcal{M}_s)^{-1} = P_s(\widehat{\mathcal{L}})^{-1}$, where $P_s(x)$ is the characteristic polynomial of A_0^{-1} (see Lemma 3).

In contrast to much of the original work on solving IRK systems, where LU factorizations were the dominant cost and system sizes relatively small, explicitly forming and inverting $P_s(\widehat{\mathcal{L}})$ for numerical PDEs is typically not a viable option in high-performance simulation on modern computing architectures. Instead, by computing the eigenvalues $\{\lambda_i\}$ of A_0^{-1}, we can express $P_s(\widehat{\mathcal{L}})$ in a factored form,

$$P_s(\widehat{\mathcal{L}}) = \prod_{i=1}^{s} (\lambda_i I - \widehat{\mathcal{L}}), \tag{10}$$

{eq:fac}

and its inverse can then be computed by successive applications of $(\lambda_i I - \widehat{\mathcal{L}})^{-1}$, for $i = 1, ..., s$. As discussed previously, eigenvalues of A_0 and A_0^{-1} will often be complex, making the inverse of individual factors $(\lambda_i I - \widehat{\mathcal{L}})^{-1}$ more difficult and often impractical.

Here, we propose combining pairs of conjugate eigenvalues into quadratic polynomials that we must precondition, which take the form

$$\begin{aligned} \mathcal{Q}_\eta :&= ((\eta + i\beta)I - \widehat{\mathcal{L}})((\eta - i\beta)I - \widehat{\mathcal{L}}) \\ &= (\eta^2 + \beta^2)I - 2\eta\widehat{\mathcal{L}} + \widehat{\mathcal{L}}^2 = (\eta I - \widehat{\mathcal{L}})^2 + \beta^2 I. \end{aligned} \tag{11}$$

{eq:imag1}

In practice, we typically do not want to directly form or precondition a quadratic operator like (11), due to (i) the overhead cost of large parallel matrix multiplication, and (ii) the fact that many fast parallel methods such as multigrid are not well-suited for solving a polynomial in $\widehat{\mathcal{L}}$. The point of (11) is that by considering conjugate pairs of eigenvalues, the resulting operator is real-valued. Thus, consider preconditioning (11) with the inverse of the real-valued quadratic polynomial, $(\eta I - \widehat{\mathcal{L}})^2$, dropping the $+\beta^2 I$ term. Expanding, the preconditioned operator then takes the form

$$\mathcal{P}_\eta := (\eta I - \widehat{\mathcal{L}})^{-2} \left[(\eta^2 + \beta^2)I - 2\eta\widehat{\mathcal{L}} + \widehat{\mathcal{L}}^2 \right]$$

{eq:prec1}
$$= I + \beta^2(\eta I - \widehat{\mathcal{L}})^{-2} = I + \frac{\beta^2}{\eta^2}\left(I - \tfrac{1}{\eta}\widehat{\mathcal{L}}\right)^{-2}. \tag{12}$$

It turns out, under assumptions introduced in Subsection 1.5, the preconditioned operator in (12) is very well-conditioned. Theorem 6 analyzes the field-of-values of $\mathcal{P}_\eta$ (7) as a measure of the preconditioning, and Corollaries 7 and 8 extend this preconditioning to prove fast convergence of GMRES and conjugate gradient (CG).

Remark 5 (Convergence independent number of stages/DG polynomial order). It turns out the conditioning of $\mathcal{P}_\eta$ has some dependence on the ratio β^2/η^2, particularly for $\beta > \eta$, which typically increases with number of stages/DG polynomial order. Subsection 2.3 introduces a modification that yields conditioning independent of β and η (and thus, number of stages/polynomial order) at the expense of less rigorous analysis of GMRES convergence than provided for $\mathcal{P}_\eta$ in this section.

{th:fov}
THEOREM 6 (Preconditioned field of values). *Suppose Assumptions 1 and 2 hold, that is, $\eta > 0$ and $W(\mathcal{L}) \leq 0$ (7). Let $\mathcal{P}_\eta$ denote the preconditioned operator, where $((\eta + i\beta)I - \widehat{\mathcal{L}})((\eta - i\beta)I - \widehat{\mathcal{L}})$ is preconditioned with $(\eta I - \widehat{\mathcal{L}})^{-2}$. Then $W(\mathcal{P}_\eta)$ is bounded by Ω as shown in Figure 1.*

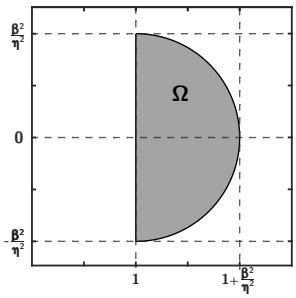


Fig. 1

{fig:bound}

Proof. Let $\sigma(B)$ denote the spectrum of operator B, $\sigma_{\min}(B)$ and $\sigma_{\max}(B)$ the minimum and maximum eigenvalues, and $\rho(B)$ the spectral radius. Also, define the symmetric/skew-symmetric splitting $B = B_s + B_k$, where $B_s := (B + B^T)/2$ and $B_k := (B - B^T)/2$, and the numerical radius as $r(B) = \sup\{|\lambda| : \lambda \in W(B)\}$. Recall the following properties of $W(B)$ [10, 15]:

1. $Bs \leq 0$ in the symmetric negative semi-definite sense if and only if $W(B) \leq 0$.
2. $W(B) \subset [\sigma_{\min}(B_s), \sigma_{\max}(B_s)] \times [-\rho(B_k)\mathrm{i}, \rho(B_k)\mathrm{i}]$.
3. $\sigma(B) \subset W(B)$.
4. If B is invertible and $B_s \leq 0$ in the symmetric negative semi-definite sense, then the symmetric part of B^{-1} is also negative semi-definite.
5. $r(B) \leq \|B\|_2$.
6. $W(I + B) = 1 + W(B)$.

Note that an exact inverse yields $\mathcal{P}_\eta = I$, with spectrum and field-of-values given by $\sigma(\mathcal{P}_S) = W(\mathcal{P}_S) = \{1\}$. Appealing to (12) and the final property stated above, $W(\mathcal{P}_\eta) = 1 + \frac{\beta^2}{\eta^2}W(E)$, for error term $E := (I - \frac{1}{\eta}\widehat{\mathcal{L}})^{-2}$ and real-valued constant $\beta^2/\eta^2 > 0$. Next we will bound $W(E)$ in the complex plane.

Assume that $\eta > 0$ and the symmetric part of $\widehat{\mathcal{L}}$ satisfies $(\widehat{\mathcal{L}} + \widehat{\mathcal{L}}^T)/2 \leq 0$. It follows that the real part of eigenvalues of $\widehat{\mathcal{L}}$ are non-positive and, thus, $(I - \frac{1}{\eta}\widehat{\mathcal{L}})$

cannot have a zero eigenvalue and must be invertible. Furthermore, it also follows that the symmetric part of $(I - \frac{1}{\eta}\widehat{\mathcal{L}})$ is symmetric positive definite and thus the symmetric part of $(I - \frac{1}{\eta}\widehat{\mathcal{L}})^{-2}$ is as well. This yields a lower bound of zero on the real-axis for $W(E)$, that is, $\mathrm{Re}(W(E)) > 0$.

Now, note that by the assumption $(\widehat{\mathcal{L}} + \widehat{\mathcal{L}}^T)/2 \leq 0$, we have

$$\frac{\left\langle (I - \frac{1}{\eta}\widehat{\mathcal{L}})\mathbf{x}, (I - \frac{1}{\eta}\widehat{\mathcal{L}})\mathbf{x} \right\rangle}{\langle \mathbf{x}, \mathbf{x} \rangle} = 1 - \frac{\langle (\widehat{\mathcal{L}} + \widehat{\mathcal{L}}^T)\mathbf{x}, \mathbf{x} \rangle}{\eta \langle \mathbf{x}, \mathbf{x} \rangle} + \frac{\langle \frac{1}{\eta^2}\widehat{\mathcal{L}}^T\widehat{\mathcal{L}}\mathbf{x}, \mathbf{x} \rangle}{\langle \mathbf{x}, \mathbf{x} \rangle} \geq 1 \tag{13}$$ {eq:norm1}

for all $\mathbf{x} \neq \mathbf{0}$. Then,

$$\begin{aligned} \|(I - \tfrac{1}{\eta}\widehat{\mathcal{L}})^{-2}\| \leq \|(I - \tfrac{1}{\eta}\widehat{\mathcal{L}})^{-1}\|^2 &= \sup_{\mathbf{x}\neq\mathbf{0}} \frac{\left\langle (I - \frac{1}{\eta}\widehat{\mathcal{L}})^{-1}\mathbf{x}, (I - \frac{1}{\eta}\widehat{\mathcal{L}})^{-1}\mathbf{x} \right\rangle}{\langle \mathbf{x}, \mathbf{x} \rangle} \\ &= \sup_{\mathbf{y}\neq\mathbf{0}} \frac{\langle \mathbf{y}, \mathbf{y} \rangle}{\left\langle (I - \frac{1}{\eta}\widehat{\mathcal{L}})\mathbf{y}, (I - \frac{1}{\eta}\widehat{\mathcal{L}})\mathbf{y} \right\rangle} \leq 1. \end{aligned}$$

This yields a bound on the numerical radius $r(E) = r((I - \frac{1}{\eta}\widehat{\mathcal{L}})^{-2}) \leq \|(I - \frac{1}{\eta}\widehat{\mathcal{L}})^{-2}\| \leq 1$. Combining with $\mathrm{Re}(W(E)) > 0$, the field of values of the error term, $W(E)$, is contained in the positive half of the unit circle in the complex plane, which completes the proof. ∎

{cor:gmres}

COROLLARY 7 (GMRES convergence bounds). *Let π_k denote the set of consistent polynomials of degree k. Then the ideal GMRES bound (an upper bound in operator norm of worst-case convergence) on convergence after k iterations applied to the preconditioned operator $\mathcal{P}_\eta$ (12) is bounded by*

$$\min_{p\in\pi_k} \|p(\mathcal{P}_\eta)\| \leq 2\left(\frac{\beta^2/\eta^2}{2 + \beta^2/\eta^2}\right)^k.$$

Proof. For operator B, let $\nu(B)$ denote the distance of $W(B)$ from the origin and define $\cos(\zeta) := \nu(B)/r(B)$. In [13, Lemma 3.2], it is proven that worst-case convergence of GMRES applied to operator B is bounded by

$$\min_{p\in\pi_k} \|p(B)\| \leq 2\left(\frac{1 - \cos\zeta}{1 + \cos\zeta}\right)^k. \tag{14}$$ {eq:gmres}

For $B = \mathcal{P}_\eta$, we have $\nu(\mathcal{P}_\eta) = 1$ and $r(\mathcal{P}_\eta) \leq 1 + \beta^2/\eta^2$. Plugging into (14) completes the proof.[4] ∎

{cor:cg}

COROLLARY 8 (CG convergence bounds). *Define $\mathcal{Q}_\eta$ as in (11) and $\mathcal{P}_\eta$ as in (12), and assume $(\eta I - \widehat{\mathcal{L}})$ is SPD. Then, the error $\mathbf{e}_k$ in the $\mathcal{Q}_\eta$-norm after k iterations of preconditioned conjugate gradient is bounded by*

$$\frac{\|\mathbf{e}_k\|_{\mathcal{Q}_\eta}}{\|\mathbf{e}_0\|_{\mathcal{Q}_\eta}} \leq 2\left(\frac{\sqrt{1 + \beta^2/\eta^2} - 1}{\sqrt{1 + \beta^2/\eta^2} + 1}\right)^k.$$

[4] Note, classical GMRES convergence results based on $\lambda_{\min}((\mathcal{P}_\eta + \mathcal{P}_\eta^T)/2)$ and $\lambda_{\max}(\mathcal{P}_\eta^T\mathcal{P}_\eta)$ can also be applied, yielding the bound $\left(\frac{\beta^2/\eta^2}{1+\beta^2/\eta^2}\right)^{k/2}$, $2 - 4\times$ slower convergence than Theorem 6.

Proof. Note that if $(\eta I - \widehat{\mathcal{L}})$ is SPD, then $\mathcal{Q}_\eta$ is also SPD. Then, recall that for conjugate gradient applied to SPD matrix A, error is bounded via

$$\frac{\|\mathbf{e}_k\|_A}{\|\mathbf{e}_0\|_A} \leq 2\left(\frac{\sqrt{\kappa(A)}-1}{\sqrt{\kappa(A)}+1}\right)^k, \tag{15}$$

{eq:cg_th}

where $\kappa(A)$ denotes the condition number of A. For preconditioned CG with SPD preconditioner B^{-1} and Cholesky decomposition $B = R^T R$, PCG applied to $A\mathbf{x} = \mathbf{b}$ is equivalent to applying CG to the modified SPD system $(R^{-T}AR^{-1})R\mathbf{x} = R^{-T}\mathbf{b}$. The condition number is then given by

$$\begin{aligned}\kappa(R^{-T}AR^{-1}) &= \lambda_{max}(R^{-T}AR^{-1})/\lambda_{min}(R^{-T}AR^{-1}) \\ &= \lambda_{max}(R^{-1}R^{-T}A)/\lambda_{min}(R^{-1}R^{-T}A) \\ &= \lambda_{max}(B^{-1}A)/\lambda_{min}(B^{-1}A).\end{aligned}$$

for largest and smallest eigenvalues λ_{max} and λ_{min}, respectively. Then, recall that eigenvalues of an operator are contained in the field-of-values, and from Theorem 6. This yields $\lambda_{max}(\mathcal{P}_\eta)/\lambda_{min}(\mathcal{P}_\eta) \leq 1 + \beta^2/\eta^2$, and by monotonicity of (15) in κ this completes the proof. ∎

Table 1 provides the values of η, β^2/η^2, and the CG and GMRES bounds from Corollary 7 and Corollary 8 for Gauss, RadauIIA, and LobattoIIIC integration.

	Stages	2	3		4		5		
Gauss	η	3.0	4.64	3.68	4.21	5.79	7.29	4.65	6.70
	β^2/η^2	0.33	0	0.91	1.59	0.09	0	2.36	0.27
	CG	0.072	0	0.160	0.234	0.022	0	0.294	0.060
	GMRES	0.143	0	0.313	0.444	0.043	0	0.541	0.119
RadauIIA	η	2.0	3.64	2.68	3.21	4.79	6.29	3.66	5.70
	β^2/η^2	0.50	0	1.29	2.21	0.11	0	3.20	0.32
	CG	0.101	0	0.205	0.283	0.025	0	0.344	0.069
	GMRES	0.2	0	0.393	0.525	0.051	0	0.616	0.137
LobattoIIIC	η	1.0	2.63	1.69	2.22	3.78	5.28	2.66	4.70
	β^2/η^2	1	0	2.21	3.51	0.13	0	4.88	0.38
	CG	0.172	0	0.284	0.360	0.031	0	0.416	0.081
	GMRES	0.333	0	0.525	0.637	0.063	0	0.709	0.161

Table 1: η, β^2/η^2, and the convergence factors in Corollary 7 and Corollary 8 for GMRES and CG, respectively (without the leading constants of 2) for Gauss, RadauIIA, and LobattoIIIC integration, with 2–5 stages. Each column within a given set of stage columns corresponds to a single eigenvalue of A_0^{-1}. For odd numbers of stages, one eigenvalue is real, corresponding to the column where $\beta^2/\eta^2 = 0/\eta^2 = 0$.

{tab:beta}

:solve:gamma}

2.3. Conditioning independent of η and β. In the previous section, we considered two applications of $(\eta I - \widehat{\mathcal{L}})^{-1}$ as a preconditioner for $\mathcal{Q}_\eta$ (11). Theorem 6 and the corresponding Corollaries proved that preconditioned GMRES is guaranteed

to converge using such an approach, with bounds on convergence provided in Table 1. However, one can note from Figure 1 and Table 1 that convergence depends on values of β and η and, in particular, that convergence degrades as the number of stages increase. Ideally, we would like to observe convergence independent of β and η, analogous to spatial solvers that achieve convergence independent of spatial mesh and discretization order. This section is motivated by [2], where a similar algorithm is developed for linear parabolic problems using the real Schur decomposition, and, for SPD operators, a modified constant $\eta \mapsto \sqrt{\eta^2+\beta^2}$ is proven to be optimal in terms of minimizing the condition number of the preconditioned operator. In this section, we derive bounds on the condition number of the preconditioned operator for preconditioner $(\gamma I - \widehat{\mathcal{L}})^{-2}$, $\gamma \neq \eta$, and general operators satisfying Assumption 2 (Theorem 9). A corollary then proves that the constant

$$\gamma_* := \sqrt{\eta^2+\beta^2} \tag{16}$$ {{eq:gam_opt}}

yields a preconditioned operator with condition number < 9, independent of η and β (Corollary 10).

Consider a similar preconditioner as in Subsection 2.1, but with some modified constant $\gamma \mapsto (\gamma I - \widehat{\mathcal{L}})^{-2}$. The resulting preconditioned operator takes the form

$$\begin{aligned}
\mathcal{P}_\gamma &= (\gamma I - \mathcal{L})^{-2}\Big[(nI-\mathcal{L})^2 + \beta^2 I\Big] \\
&= (\gamma I - \mathcal{L})^{-2}\Big[((\eta-\gamma)I + (\gamma I - \mathcal{L}))^2 + \beta^2 I\Big] \\
&= (\gamma I - \mathcal{L})^{-2}\Big[(\gamma-\eta)^2 I - 2(\gamma-\eta)(\gamma I - \mathcal{L}) + (\gamma I - \mathcal{L})^2 + \beta^2 I\Big] \\
&= I - 2(\gamma-\eta)(\gamma I - \mathcal{L})^{-1} + (\beta^2 + (\gamma-\eta)^2)(\gamma I - \mathcal{L})^{-2} \\
&= I - 2\frac{\gamma-\eta}{\gamma}\left(I - \tfrac{1}{\gamma}\mathcal{L}\right)^{-1} + \frac{\beta^2+(\gamma-\eta)^2}{\gamma^2}\left(I - \tfrac{1}{\gamma}\mathcal{L}\right)^{-2}.
\end{aligned} \tag{17}$$ {{eq:preL_k2}}

Note that in (17) we have a quadratic polynomial in $(I - \frac{1}{\gamma}\mathcal{L})^{-1}$. Although this provides nice structure, the field-of-values analysis applied in Subsection 2.1 becomes much more complicated due to no necessary relation between $\langle A\mathbf{x},\mathbf{x}\rangle$ and $\langle A^2\mathbf{x},\mathbf{x}\rangle$ for general operators A. Thus, here we take a different approach, analyzing the condition number of the preconditioned operator, $\mathcal{P}_\gamma$, similar to as done for SPD matrices in [2]. Although the conditioning does not yield immediate GMRES bounds as the field-of-values analysis does (or as conditioning does for bounds on CG convergence), it still provides a robust measure of the effectiveness and scalability of the preconditioner.

As a preliminary result, note that working out the roots of the polynomial in (17), it can be expressed in factored form as

$$\mathcal{P}_\gamma = \left[I - \overline{\alpha}\left(I - \tfrac{1}{\gamma}\mathcal{L}\right)^{-1}\right]\left[I - \alpha\left(I - \tfrac{1}{\gamma}\mathcal{L}\right)^{-1}\right],$$

where

$$\alpha, \overline{\alpha} := 1 - \frac{\eta}{\gamma} \pm \mathrm{i}\beta, \tag{18}$$ {{eq:alpha}}

and

$$\alpha + \overline{\alpha} = 2(1-\tfrac{\eta}{\gamma}), \qquad \alpha\overline{\alpha} = \tfrac{\beta^2+(\gamma-\eta)^2}{\gamma^2}. \tag{19}$$ {{eq:alpha_eq}}

Moving forward we will limit ourselves to considering $\eta \leq \gamma \leq \frac{\eta^2+\beta^2}{\eta}$, which limits to the natural case of $\gamma = \eta$ as $\beta \to 0$. Similar analysis can be performed for γ outside of this range, but the derivations are different and there does not appear to be any good reason for γ outside of this range (see discussion following Theorem 9).

{th:cond}

THEOREM 9 (Conditioning of preconditioned operator). *Suppose Assumptions 1 and 2 hold, that is, $\eta > 0$ and $W(\mathcal{L}) \leq 0$ (7). Let $\eta \leq \gamma \leq \frac{\eta^2+\beta^2}{\eta}$ and let $\mathcal{P}_\gamma$ denote the preconditioned operator, where $((\eta + i\beta)I - \widehat{\mathcal{L}})((\eta - i\beta)I - \widehat{\mathcal{L}})$ is preconditioned with $(\gamma I - \widehat{\mathcal{L}})^{-2}$. Then,*

$$\operatorname{cond}(\mathcal{P}_\gamma) \leq (1 + \alpha\overline{\alpha}) \left(1 + \frac{\alpha\overline{\alpha}}{(1-\alpha)(1-\overline{\alpha})}\right) \tag{20}$$

with α and $\overline{\alpha}$ defined as in (18).

Proof. The proof proceeds as follows: first, we factor $\mathcal{P}_\gamma$ (17), and proceed to use the factored form to derive an upper bound on $\|\mathcal{P}_\gamma\|$ and $\|\mathcal{P}_\gamma^{-1}\|$, which immediately yields a bound on

$$\operatorname{cond}(\mathcal{P}_\gamma) = \|\mathcal{P}_\gamma\|\|\mathcal{P}_\gamma^{-1}\|. \tag{21}$$

Note that following from the discussion in Subsection 2.1 and Theorem 6, for real $k > 0$, $W\left[\left(I - \frac{1}{k}\mathcal{L}\right)^{-1}\right]$ and $W\left[\left(I - \frac{1}{k}\mathcal{L}\right)^{-2}\right]$ are contained in the positive half unit circle, and $\|(I - \frac{1}{\gamma}\mathcal{L})^{-1}\| \leq 1$ (see proof of Theorem 6).

We start with $\|\mathcal{P}_\gamma\|$, where

$$\|\mathcal{P}_\gamma\| \leq \left\|I - \overline{\alpha}\left(I - \tfrac{1}{\gamma}\mathcal{L}\right)^{-1}\right\| \left\|I - \alpha\left(I - \tfrac{1}{\gamma}\mathcal{L}\right)^{-1}\right\|. \tag{22}$$

Recalling that $W\left[(I - \frac{1}{k}\mathcal{L})^{-1}\right] \geq 0$ and $\|(I - \frac{1}{\gamma}\mathcal{L})^{-1}\| \leq 1$, and using the assumption $\gamma \geq \eta$, we can expand the norm squared in inner-product form, yielding

$$\begin{aligned}
&\left\|I - \overline{\alpha}\left(I - \tfrac{1}{\gamma}\mathcal{L}\right)^{-1}\right\|^2 \\
&\quad = 1 + \max_{\mathbf{x}\neq\mathbf{0}} \left(\alpha\overline{\alpha} \frac{\left\|\left(I - \frac{1}{\gamma}\mathcal{L}\right)^{-1}\mathbf{x}\right\|^2}{\|\mathbf{x}\|^2} - (\alpha + \overline{\alpha}) \frac{\operatorname{Re}\left(\left\langle \left(I - \frac{1}{\gamma}\mathcal{L}\right)^{-1}\mathbf{x}, \mathbf{x}\right\rangle\right)}{\|\mathbf{x}\|^2} \right) \\
&\quad \leq 1 + \alpha\overline{\alpha}\left\|\left(I - \tfrac{1}{\gamma}\mathcal{L}\right)^{-1}\right\|^2 \\
&\quad \leq 1 + \alpha\overline{\alpha}.
\end{aligned}$$

Note that the above derivation is identical for α and $\overline{\alpha}$, which yields

$$\|\mathcal{P}_\gamma\| \leq 1 + \alpha\overline{\alpha}. \tag{23}$$

To bound $\|\mathcal{P}_\gamma^{-1}\|$, we use an analogous approach as above (22), developing bounds on the two polynomial factors separately. First, note that

$$\left[I - \alpha\left(I - \tfrac{1}{\gamma}\mathcal{L}\right)^{-1}\right]^{-1} = \left(I - \tfrac{1}{\gamma}\mathcal{L}\right)\left[(1-\alpha)I - \tfrac{1}{\gamma}\mathcal{L}\right]^{-1}$$

$$= \left[\alpha I + ((1-\alpha)I - \tfrac{1}{\gamma}\mathcal{L})\right]\left[(1-\alpha)I - \tfrac{1}{\gamma}\mathcal{L}\right]^{-1}$$

$$= I + \alpha\left[(1-\alpha)I - \tfrac{1}{\gamma}\mathcal{L}\right]^{-1}$$

$$= I + \frac{\alpha}{1-\alpha}\left(I - \tfrac{1}{\gamma(1-\alpha)}\mathcal{L}\right)^{-1}. \tag{24}$$

For condensed notation, let $c_\alpha := \frac{\alpha}{1-\alpha}$, and note that

$$c_\alpha + \overline{c_\alpha} = \frac{\alpha + \overline{\alpha} - 2\alpha\overline{\alpha}}{1 + \alpha\overline{\alpha} - (\alpha + \overline{\alpha})} = \frac{2\eta\gamma - 2(\eta^2 + \beta^2)}{\eta^2 + \beta^2},$$

which is < 0 when $\eta\gamma \le (\eta^2 + \beta^2)$ and < 0 otherwise.

Expanding the squared norm of (24) in inner-product form, we have

$$\min_{\mathbf{x}\neq\mathbf{0}} \left[\frac{\left\|\left(I - \frac{1}{\gamma(1-\alpha)}\mathcal{L}\right)\mathbf{x}\right\|^2}{\|\mathbf{x}\|^2} + (c_\alpha + \overline{c_\alpha})\frac{\operatorname{Re}\left(\left\langle (I - \frac{1}{\gamma(1-\alpha)}\mathcal{L})^{-1}\mathbf{x}, \mathbf{x}\right\rangle\right)}{\|\mathbf{x}\|^2} + c_\alpha\overline{c_\alpha}\frac{\left\|(I - \frac{1}{\gamma(1-\alpha)}\mathcal{L})^{-1}\mathbf{x}\right\|^2}{\|\mathbf{x}\|^2} \right]. \tag{25}$$

Bounding the norm term in (25) requires care because α is complex. Note that the norm is the maximum singular value, which is equivalent to one over the minimum nonzero singular value of the inverse. The squared minimum nonzero singular value of the inverse of the inverse is given by

$$\min_{\mathbf{x}\neq\mathbf{0}} \frac{\left\|\left(I - \frac{1}{\gamma(1-\alpha)}\mathcal{L}\right)\mathbf{x}\right\|^2}{\|\mathbf{x}\|^2}$$

$$= 1 + \min_{\mathbf{x}\neq\mathbf{0}} \left[\frac{1}{|\gamma(1-\alpha)|^2}\frac{\|\mathcal{L}\mathbf{x}\|^2}{\|\mathbf{x}\|^2} - \left(\frac{1}{\gamma(1-\alpha)} + \frac{1}{\overline{\gamma}(1-\overline{\alpha})}\right)\frac{\operatorname{Re}\left(\langle\mathcal{L}\mathbf{x}, \mathbf{x}\rangle\right)}{\|\mathbf{x}\|^2} \right].$$

Noting that $\operatorname{Re}(\langle\mathcal{L}\mathbf{x}, \mathbf{x}\rangle) \le 0$, $\gamma \ge \eta > 0$, and $\operatorname{Re}(1-\alpha) \ge 0$, it follows that the squared minimum singular value above is ≥ 1, which implies $\|(I - \frac{1}{\gamma(1-\alpha)}\mathcal{L})^{-1}\| \le 1$.

Returning to (25), consider the inner-product term with leading constant $(c_\alpha + \overline{c_\alpha}) \le 0$. Note that letting $\mathbf{x} := (I - \frac{1}{\gamma(1-\alpha)}\mathcal{L})\mathbf{y}$, we have

$$\frac{\left\langle (I - \frac{1}{\gamma(1-\alpha)}\mathcal{L})^{-1}\mathbf{x}, \mathbf{x}\right\rangle}{\|\mathbf{x}\|^2} = \frac{\left\langle \mathbf{y}, (I - \frac{1}{\gamma(1-\alpha)}\mathcal{L})\mathbf{y}\right\rangle}{\|(I - \frac{1}{\gamma(1-\alpha)}\mathcal{L})\mathbf{y}\|^2} = \frac{\|\mathbf{y}\|^2 - \frac{1}{\gamma(1-\overline{\alpha})}\langle\mathbf{y}, \mathcal{L}\mathbf{y}\rangle}{\|(I - \frac{1}{\gamma(1-\alpha)}\mathcal{L})\mathbf{y}\|^2}.$$

By assumption, $\operatorname{Re}(\langle\mathbf{y}, \mathcal{L}\mathbf{y}\rangle) \le 0$, while $\gamma \ge 0$ and $\operatorname{Re}(1-\alpha) \ge 0$. It follows that

$$\operatorname{Re}\left(\frac{\left\langle (I - \frac{1}{\gamma(1-\alpha)}\mathcal{L})^{-1}\mathbf{x}, \mathbf{x}\right\rangle}{\|\mathbf{x}\|^2}\right) \ge 0.$$

Thus in (25) because $(c_\alpha + \overline{c_\alpha}) \le 0$, we can drop the corresponding term for an upper bound of

$$\left\|\left[I - \alpha\left(I - \tfrac{1}{\gamma}\mathcal{L}\right)^{-1}\right]^{-1}\right\|^2 \le 1 + c_\alpha\overline{c_\alpha}\left\|(I - \tfrac{1}{\gamma(1-\alpha)}\mathcal{L})^{-1}\right\|^2$$

$$\le 1 + c_\alpha\overline{c_\alpha}$$

$$= 1 + \frac{\alpha\overline{\alpha}}{(1-\alpha)(1-\overline{\alpha})}.$$

Again noting that analogous derivations holds for $\overline{\alpha}$ with identical bounds, we have for $(c_\alpha + \overline{c_\alpha}) < 0$,

$$\|\mathcal{P}_\gamma^{-1}\| \leq 1 + \frac{\alpha\overline{\alpha}}{(1-\alpha)(1-\overline{\alpha})}. \tag{26}$$

{{eq:lower_alpha2}}

Combining (21), (23), and (26) completes the proof. ∎

In [2], a similar analysis as Theorem 9 is done under the assumption that $-\mathcal{L}$ is SPD, in which case the conditioning can be derived based on eigenvalues. There, they derive the (exact) conditioning of the preconditioned operator for $\gamma = \eta$ to be $\text{cond}(\mathcal{P}_\eta) = 1 + \frac{\beta^2}{\eta^2}$. If we let $\gamma = \eta$, from (19) we have $\alpha + \overline{\alpha} = 0$, $\alpha\overline{\alpha} = \frac{\beta^2}{\eta^2}$, and Theorem 9 yields the bound

$$\text{cond}(\mathcal{P}_{\gamma=\eta}) \leq 1 + 2\frac{\beta^2}{\eta^2}. \tag{27}$$

In particular, this indicates that Theorem 9 is fairly tight (it is unclear if the additional factor of 2 in (27) is necessary for non-SPD operators, or a flaw in the line of proof). Considering the upper limit on γ in Theorem 9, $\gamma \mapsto \frac{\eta^2+\beta^2}{\eta}$, we have a bound

$$\text{cond}\left(\mathcal{P}_{\gamma=\frac{\eta^2+\beta^2}{\eta}}\right) \leq \left(1 + \frac{\beta^2}{\eta^2}\right)\left(1 + \frac{\beta^2}{\beta^2+\eta^2}\right). \tag{28}$$

Note, in both (27) and (28) we see a dependence on η and β.

Using the tight derivation of conditioning of the preconditioned operator in [2], they also derive an optimal constant to minimize the condition number for SPD matrices, $\gamma_* := \sqrt{\eta^2+\beta^2}$ (16). Considering γ_* here, (19) yields $\alpha + \overline{\alpha} = \alpha\overline{\alpha} = 2 - 2\frac{\eta}{\sqrt{\eta^2+\beta^2}}$, and Theorem 9 yields the bound

$$\text{cond}(\mathcal{P}_{\gamma_*}) \leq \left(3 - 2\frac{\eta}{\sqrt{\eta^2+\beta^2}}\right)^2. \tag{29}$$

Noting that $\text{cond}(\mathcal{P}_{\gamma_*}) < 9$ for all $\eta > 0, \beta \geq 0$, we have the following corollary.

{:independent}

COROLLARY 10 (Conditioning independent of η and β). *Preconditioning $\mathcal{Q}_\eta$ (11) with $(\gamma_* I - \widehat{\mathcal{L}})^{-2}$, for $\gamma_* = \sqrt{\eta^2+\beta^2}$ yields a preconditioned operator $\mathcal{P}_{\gamma_*}$ such that* $\text{cond}(\mathcal{P}_{\gamma_*}) < 9$*, independent of η and β.*

It is also worth pointing out that as $\beta \to 0$, $\gamma \to \eta$ in all three of the above cases, and (as expected) (27), (28), and (29) all $\to 1$.

Remark 11 ("Optimal" γ). A natural question is what is the optimal γ in general? Let $f(\gamma)$ denote the upper bound from Theorem 9 (20). Some algebra shows that

$$f'(\gamma) = \frac{4\gamma^4 - 6\eta\gamma^3 + 4\eta(\eta^2+\beta^2)\gamma - 2(\eta^2+\beta^2)^2}{(\eta^2+\beta^2)\gamma^3}.$$

For $\beta > 0$,

$$f'(\eta) = -\frac{2\beta^4}{\eta^3(\eta^2+\beta^2)} < 0, \qquad f'(\gamma_*) = \frac{2(\sqrt{\eta^2+\beta^2}-\eta)}{\eta^2+\beta^2} > 0.$$

Thus there exists a critical point $\gamma \in (\eta, \gamma_*)$, and it is likely not advantageous to choose $\gamma > \gamma_*$. The equation for the root is a quartic polynomial and is not examined analytically here, however, depending on η and β, the sign change (and, thus, root) is likely between $0.8\gamma_* - 0.9\gamma_*$, suggesting γ_* is indeed a good approximation to the optimal γ with respect to (20).

Moreover, the analysis from [2] proves that γ_* (16) *is* optimal for SPD operators, and we prove γ_* also provides very good preconditioning independent of the number of stages/order of integration in the more general setting (Corollary 10).

2.4. Preconditioning in practice.

2.4.1. Choosing γ.

2.4.2. Inexact preconditioning. In practice, fully converging $(\eta I - \widehat{\mathcal{L}})^{-1}$ as a preconditioner is not desirable – even if a Krylov method converges rapidly, if each iteration requires a full linear solve, the resulting method remains too expensive to be competitive in practice. Here, we propose applying a Krylov method to $\mathcal{Q}_\eta := (\eta^2+\beta^2)I - 2\eta\widehat{\mathcal{L}} + \widehat{\mathcal{L}}^2$ by computing the operator's action (that is, not fully constructing it), and preconditioning each Krylov iteration with *two* applications of a sparse parallel preconditioner for $(\eta I - \widehat{\mathcal{L}})$, approximating the action of $(\eta I - \widehat{\mathcal{L}})^{-2}$.

To motive this heuristically, suppose $(\eta I - \widehat{\mathcal{L}}) = B + N$, where B^{-1} corresponds to the action of a preconditioner and N the error term that is not captured. For a good preconditioner, we expect $B^{-1}(\eta I - \widehat{\mathcal{L}}) = I + B^{-1}N$ to be well-conditioned and, thus, $B^{-1}N$ to be small in some sense. Then,

$$B^{-2}\mathcal{Q}_\eta = I + \beta^2 B^{-2} + \left(B^{-2}NB + B^{-1}N + B^{-2}N^{-2}\right).$$

In addition to wanting N to be small, it is also important that the error terms $B^{-2}NB + B^{-1}N + B^{-2}N^{-2}$ are generally positive in a field-of-values sense. If these terms are large and positive in a field-of-values sense, the outer Krylov iteration applied to $B^{-2}\mathcal{Q}_\eta$ can correct for these modes. However, if error terms $B^{-2}NB + B^{-1}N + B^{-2}N^{-2}$ in (2.4.2) have negative components, the field-of-values of the preconditioned operator is shifted towards the origin, wherein Krylov acceleration is often of progressively less use.

Thus, theory developed in Subsection 2.2 guarantees that the proposed algorithm will have robust and fairly rapid convergence under reasonable assumptions if each linear system is inverted exactly. Approximate inverses prove to be much faster in practice, but *it is important that the underlying preconditioner provides a good inverse approximation.* Fortunately, for difficult problems with only okay preconditioners, it is straightforward to apply either multiple inner fixed-point iterations or an inner Krylov iteration (wrapped with a flexible outer Krylov method [17, 22]) to ensure robust (outer) iterations, analogous to techniques used in block preconditioning. In Subsection 3.2.2, a numerical example is shown where the proposed method diverges using a single inner fixed-point iteration as a preconditioner for $(\eta I - \widehat{\mathcal{L}})$, but three (or more) inner fixed-point iterations yields fast, scalable convergence.

2.4.3. Mass matrices. Recall in the finite element context where mass matrices are involved, we defined $\widehat{\mathcal{L}} := \delta t M^{-1}\mathcal{L}$. For a given conjugate pair of eigenvalues, the quadratic polynomial (11) can be expressed as

$$\mathcal{Q}_\eta = M^{-1}((\eta + i\beta)M - \delta t\mathcal{L})M^{-1}((\eta - i\beta)M - \delta t\widehat{\mathcal{L}}). \tag{30}$$

In this context, it is best to first scale both sides of the linear system by M. This halves the number of times M^{-1} must applied for each Krylov iteration, and if M and $\mathcal{L}$ are Hermitian, the resulting quadratic system is SPD and can be solved using preconditioned conjugate gradient or MINRES, preconditioned with one application of a preconditioner, the action of M, and a second application of the preconditioner.

sec:numerics}

3. Numerical results.

:numerics:fd}

3.1. Finite-difference advection-diffusion.

:numerics:dg}

3.2. Discontinuous Galerkin advection(-diffusion). This section considers a more difficult advection-diffusion problem, discretized using high-order discontinuous Galerkin finite elements. In particular, we demonstrate the effectiveness and scalability of the new preconditioning on more complex flows and DG discretizations (Figure 2 and Subsection 3.2.1), the reduction in computational work that can be achieved using large time steps and very high-order integration (Subsection 3.2.1), and how using multiple "inner" preconditioning iterations to approximate $\widehat{L}^{-1}$ (or even inner Krylov acceleration) can be important for the fast, scalable application of IRK integration (Subsection 3.2.2).

[TODO: Will, can you please add a problem/FEM discretization description here?]

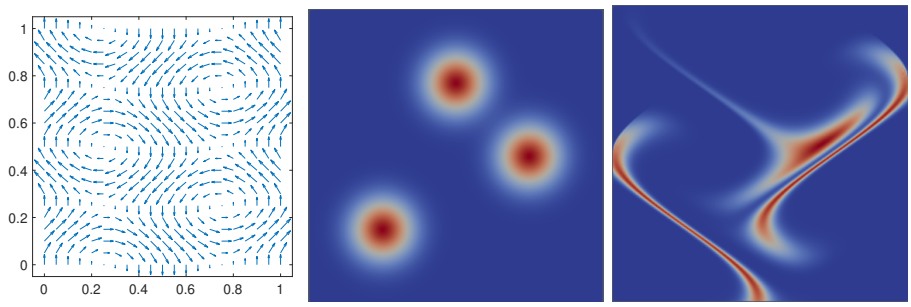
Figure 2 plots the velocity field $\Omega(x, y)$ followed by the spatial solution at five points in time (including the initial condition at $t = 0$) using diffusion coefficient $\kappa = 10^{-6}$.

erics:dg:adv}

3.2.1. High-order & advection-dominated. First, we demonstrate high-order IRK integration applied to the advection-dominated problem in Figure 2, using AIR as a preconditioner for the linear systems. AIR was originally designed for upwind DG discretizations of advection and is well-suited for this problem. We use the *hypre* implementation, with distance 1.5 restriction with strength tolerance $\theta_R = 0.01$, one-point interpolation (type 100), Falgout coarsening (type 6) with strength tolerance $\theta_C = 0.1$, no pre-relaxation, and forward Gauss Seidel post-relaxation (type 3), first on F-points followed by a second sweep on all points. The domain is discretized using 4th-order finite elements on a structured mesh, and the time step for each integration scheme is chosen such that the spatial and temporal orders of accuracy match; for example, for 8th-order integration we choose $\delta t = \sqrt{h}$, for mesh spacing h, so that $\delta t^8 = h^4$. All linear systems are solved to a relative tolerance of 10^{-12}.

Figure 3a shows the total numer of AIR iterations required to compute one time step using six different Runge-Kutta schemes, from 4th to 10th order, as a funciton of mesh spacing h. Note that as $h \to 0$, there is only modest growth in the number of AIR iterations per time step, indicating good (but not perfect) scalability. Moreover, note that there is small growth in the number of iterations for SDIRK4 as well (increasing from 20 to 25), which suggests the growth in iteration count is due to imperfect scalability of AIR iterations rather than the new IRK preconditioning.

Figure 3b then shows the relative number of AIR iterations to compute a given accuracy compared to SDIRK4. For example, if $h = 0.01$, then SDIRK4 uses $\delta t_4 = 0.01$ and Gauss8 uses $\delta t_8 = \sqrt{0.01} = 0.1 = 10\delta t_4$, that is, $10\times$ less time steps to achieve comparable accuracy. Note that as $h \to 0$, this factor becomes progressively larger. For quite moderate h, we see how very high-order integration can quickly lead to a reduction in total AIR iterations compared to a standard SDIRK4 iteration.



{sub{fig:v}

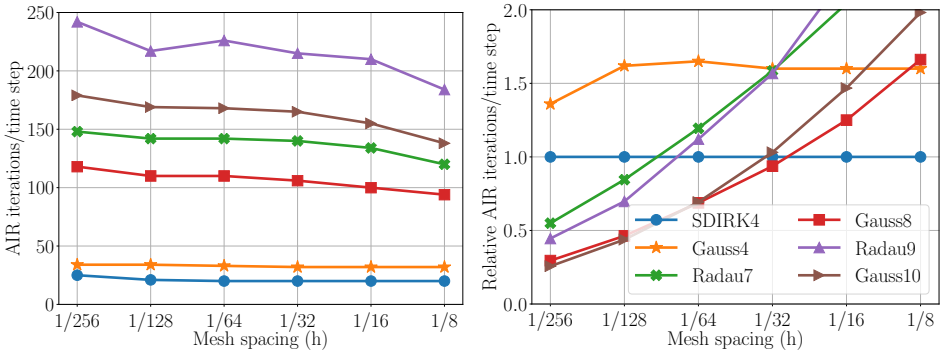
(a) Velocity field (b) $t = 0$ (c) $t = 0.3$

(d) $t = 0.8$ (e) $t = 2.0$ (f) $t = 7.0$

Fig. 2: Advection-diffusion problem with velocity field shown in subplot (a) and the solution plotted for various time points from $t = 0$ to $t = 7.0$ in subplots (b-f). Heatmap indicates solution in the 2d domain, with blue $\mapsto 0$ and red $\mapsto 1$.

g:ad_advdiff}

Gauss8, for example, requires $\approx 29\%$ of the total number of AIR iterations as SDIRK4 at $h = \delta t_4 = 1/256$, and this ratio continues to decrease for smaller δt. Although this does not account for additional setup time, where Gauss8 requires solving two different linear systems and SDIRK4 only one, very high-order integration permitted through the new preconditioning still offers the opportunity for significant speedups.



(a) Total AIR iterations per time step {sub{fig:dg_o4_abs} (b) AIR iterations relative to SDIRK4 {sub{fig:dg_o4_rel}

Fig. 3

{fig:dg_o4}

rics:dg:diff}

3.2.2. Diffusive problems and inner Krylov. In [14], AIR was shown to be effective on some DG advection-diffusion problems, and classical AMG is known to be effective on diffusion-dominated problems. However, the region of comparable levels of advection and diffusion remains the most difficult from a multigrid perspective. We use this to demonstrate how methods developed here require a "good" preconditioner for a backward Euler time step in order to converge on more general IRK methods. However, ensuring a preconditioner is sufficiently good can be resolved analogous to block preconditioning techniques, where an inner iteration is used that applies multiple AIR iterations as a single preconditioner.

Here we consider an analogous problem to above, but set the diffusion coefficient to $\kappa = 0.01$. We use a mesh with spacing $h \approx 0.001$, 2nd-order DG finite elements, a time step of $\delta t = 0.1$, and three-stage 6th-order Gauss integration. Altogether, this yields equal orders of accuracy, with time and space error $\sim 10^{-6}$. FGMRES [22] is used for the outer iteration, which allows for GMRES to be applied in an inner iteration as a preconditioner for $(\eta M - \mathcal{L})$. Figure 4 plots the total number of AIR iterations per time step as a function of the number of AIR iterations applied for each application of the preconditioner, using an inner GMRES or an inner fixed-point iteration.

Recall we have three stages, one of which is a single linear system coresponding to a real eigenvalue, and the other corresponding to a pair of complex conjugate eigenvalues, which we precondition as in Section 2. The latter ends up being the more difficult problem to solve – using one AIR iteration as a preconditioner, the outer FGMRES iteration for the complex conjugate quadratic does not converge in 1000 iterations. If two AIR iterations with GMRES are used as a preconditioner, the FGMRES iteration converges in approximately 130 iterations, each of which requires two applications of GMRES preconditioned with two AIR iterations, yielding just over 500 total AIR iterations to converge. Further increasing the number of AIR iterations per preconditioning yields nice convergence using inner fixed-point or GMRES, with 150 and 112 total AIR iterations per time step, respectively.

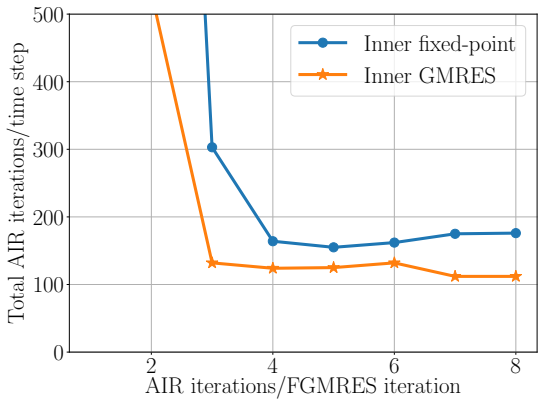


Fig. 4: AIR iterations per time step as a function of the number of AIR iterations applied for each application of the preconditioner.

{fig:dg_o2}

3.3. High-order matrix-free diffusion.

{sec:conc}

4. Conclusions. This paper introduced a theoretical and algorithmic framework for the fast, parallel solution of fully implicit Runge-Kutta methods in numerical PDEs (without algebraic constraints). A field-of-values analysis is derived to guarantee rapid Krylov convergence...[TODO: finish]

References.

[1] R. Alexander, *Diagonally implicit runge–kutta methods for stiff ode's*, SIAM Journal on Numerical Analysis, 14 (1977), pp. 1006–1021.

[2] S. Basting and E. Bänsch, *Preconditioners for the Discontinuous Galerkin time-stepping method of arbitrary order*, ESAIM: Mathematical Modelling and Numerical Analysis, 51 (2017), pp. 1173–1195, doi:10.1051/m2an/2016055.

[3] T. A. Bickart, *An Efficient Solution Process for Implicit Runge–Kutta Methods*, SIAM Journal on Numerical Analysis, 14 (1977), pp. 1022–1027, doi:10.1137/0714069.

[4] W. C. Brown, *Matrices over commutative rings*, Marcel Dekker, Inc., 1993.

[5] K. Burrage, *Efficiently Implementable Algebraically Stable Runge–Kutta Methods*, SIAM Journal on Numerical Analysis, 19 (1982), pp. 245–258, doi:10.1137/0719015.

[6] J. Butcher and D. Chen, *A new type of singly-implicit Runge–Kutta method*, Applied Numerical Mathematics, 34 (2000), pp. 179–188, doi:10.1016/s0168-9274(99)00126-9.

[7] J. C. Butcher, *On the implementation of implicit Runge-Kutta methods*, BIT Numerical Mathematics, 16 (1976), pp. 237–240, doi:10.1007/bf01932265.

[8] J. C. Butcher, *Numerical methods for ordinary differential equations*, John Wiley & Sons, 2016.

[9] M. J. Gander and M. Neumuller, *Analysis of a new space-time parallel multigrid algorithm for parabolic problems*, SIAM Journal on Scientific Computing, 38 (2016), pp. A2173–A2208.

[10] K. E. Gustafson and D. K. Rao, *Numerical range*, in Numerical Range, Springer, 1997, pp. 1–26.

[11] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II, Stiff and Differential-Algebraic Problems*, (1996), pp. 118–130, doi:10.1007/978-3-642-05221-7_8.

[12] R. J. LeVeque, *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems*, vol. 98, Siam, 2007.

[13] J. Liesen and P. Tichý, *The field of values bound on ideal GMRES*, arXiv preprint arXiv:1211.5969, (2012).

[14] T. A. Manteuffel, J. Ruge, and B. S. Southworth, *Nonsymmetric algebraic multigrid based on local approximate ideal restriction (ℓAIR)*, SIAM J. Sci. Comput., 40 (2018), pp. A4105–A4130.

[15] A. Mees and D. Atherton, *Domains containing the field of values of a matrix*, Linear Algebra and its Applications, 26 (1979), pp. 289–296.

[16] S. P. Nørsett, *Runge-kutta methods with a multiple real eigenvalue only*, BIT Numerical Mathematics, 16 (1976), pp. 388–393.

[17] Y. Notay, *Flexible conjugate gradients*, SIAM Journal on Scientific Computing, 22 (2000), pp. 1444–1460.

[18] B. Orel, *Real pole approximations to the exponential function*, BIT, 31 (1991), pp. 144–159, doi:10.1007/bf01952790.

[19] W. Pazner and P.-O. Persson, *Stage-parallel fully implicit Runge–Kutta*

solvers for discontinuous Galerkin fluid simulations, Journal of Computational Physics, 335 (2017), pp. 700–717, doi:10.1016/j.jcp.2017.01.050.

[20] S. C. Reddy and L. N. Trefethen, *Stability of the method of lines*, Numerische Mathematik, 62 (1992), pp. 235–267, doi:10.1007/bf01396228.

[21] T. Richter, A. Springer, and B. Vexler, *Efficient numerical realization of discontinuous Galerkin methods for temporal discretization of parabolic problems*, Numerische Mathematik, 124 (2013), pp. 151–182, doi:10.1007/s00211-012-0511-7.

[22] Y. Saad, *A flexible inner-outer preconditioned GMRES algorithm*, SIAM Journal on Scientific Computing, 14 (1993), pp. 461–469.

[23] D. Schötzau and C. Schwab, *Time Discretization of Parabolic Problems by the HP-Version of the Discontinuous Galerkin Finite Element Method*, SIAM Journal on Numerical Analysis, 38 (2000), pp. 837–875, doi:10.1137/s0036142999352394.

[24] I. Smears, *Robust and efficient preconditioners for the discontinuous Galerkin time-stepping method*, IMA Journal of Numerical Analysis, (2016), p. drw050, doi:10.1093/imanum/drw050.

[25] L. N. Trefethen and M. Embree, *Spectra and pseudospectra: the behavior of nonnormal matrices and operators*, Princeton University Press, 2005.