1 Multigrid for fully implicit Runge-Kutta

1.1 The problem

Here we explore a semi-algebraic (block) multigrid solver for the block linear systems that arise in fully implicit Runge-Kutta and discontinuous Galerkin in time. Consider the method-of-lines approach to the numerical solution of partial differential equations (PDEs), where we discretize in space and arrive at a system of ordinary differential equations (ODEs) in time,

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

where M is a mass matrix and $\mathcal{N} \in \mathbb{R}^{N\times N}$ is a discrete, time-dependent, nonlinear operator depending on t and $\mathbf{u}$ (including potential forcing terms). Note, PDEs with an algebraic constraint, for example, the divergence-free constraint in Navier Stokes, instead yield a system of differential algebraic equations (DAEs), which are not yet addressed. Now, consider time propagation of (1) 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,

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

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

For nonlinear PDEs, $\mathcal{N}$ is linearized using, for example, a Newton or a Picard linearization, and each nonlinear iteration then consists of solving the linearized system of equations. In most cases, such a linearization is designed to approximate (or equal) the Jacobian of (3). Applying a chain rule to (3) for the partial $\partial(M\mathbf{k}_i - \mathcal{N}_i)/\partial\mathbf{k}_j$, we see that the linearized system should take the form

$$\left(\begin{bmatrix} M & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & M \end{bmatrix} - \delta t \begin{bmatrix} a_{11}\mathcal{L}_1 & \ldots & a_{1s}\mathcal{L}_1 \\ \vdots & \ddots & \vdots \\ a_{s1}\mathcal{L}_s & \ldots & a_{ss}\mathcal{L}_s \end{bmatrix}\right) \begin{bmatrix} \mathbf{k}_1 \\ \vdots \\ \mathbf{k}_s \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_s \end{bmatrix}, \tag{4}$$ {{eq:k0}}

where $\mathcal{L}_i \in \mathbb{R}^{N\times N}$ denotes a linearization of the nonlinear function corresponding to the ith stage vector, $\mathcal{N}_i \coloneqq \mathcal{N}\left(\mathbf{u}_n + \delta t \sum_{j=1}^{s} a_{ij}\mathbf{k}_j, t_n + \delta t c_i\right)$ (the main point being that the spatially linearized operators, $\mathcal{L}_i$, should be fixed for a given block row of the full linearized system, as in (4)).

Define the field of values of operator $\mathcal{L}$ as the set

$$W(\mathcal{L}) \coloneqq \{\langle \mathcal{L}\mathbf{x}, \mathbf{x}\rangle : \|\mathbf{x}\| = 1\}. \tag{5}$$ {{eq:fov}}

Here, we make two reasonable assumptions on A_0 and $\mathcal{L}_i$:

Assumption 1. *Assume that all eigenvalues of A_0 (and equivalently A_0^{-1}) have positive real part.*

Assumption 2. *Let $\mathcal{L}$ be a linearized spatial operator, and assume that $W(\mathcal{L}) \leq 0$.*

In addition, it is worth pointing out that A_0 typically has a dominant lower triangular structure.

For ease of notation, let us scale on the left by a block-diagonal matrix with diagonal blocks M^{-1} and redefine $\mathcal{L}_i \mapsto \delta t M^{-1}\mathcal{L}_i$. The scaled system then takes the form

$$\left(\begin{bmatrix} I & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & I \end{bmatrix} - \begin{bmatrix} a_{11}\mathcal{L}_1 & \ldots & a_{1s}\mathcal{L}_1 \\ \vdots & \ddots & \vdots \\ a_{s1}\mathcal{L}_s & \ldots & a_{ss}\mathcal{L}_s \end{bmatrix}\right) \begin{bmatrix} \mathbf{k}_1 \\ \vdots \\ \mathbf{k}_s \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_s \end{bmatrix}. \tag{6}$$ {{eq:k1}}

Unfortunately, we cannot say anything about the field-of-values of the latter operator. This is easily verified by noting that the eigenvalues of $A_0 \otimes \mathcal{L}$ are given by the product of eigenvalues of A_0 and $\mathcal{L}$, and one can easily construct A_0 and $\mathcal{L}$ that satisfy Assumptions 1 and 2 with eigenvalues in both half planes. This also confuses what the "small"/"smooth" modes of (6) are.

For this reason, we also consider a modified set of equations obtained by pulling out an $A_0 \otimes I$ from (6). Let $\{\alpha_{ij}\}$ denote the entries of A_0^{-1}, and consider an equivalent formulation of (6),

$$\left(\begin{bmatrix} \alpha_{11} I & \dots & \alpha_{1s} I \\ \vdots & \ddots & \vdots \\ \alpha_{s1} I & \dots & \alpha_{ss} I \end{bmatrix} - \begin{bmatrix} \mathcal{L}_1 & & \\ & \ddots & \\ & & \mathcal{L}_s \end{bmatrix} \right) (A_0 \otimes I) \begin{bmatrix} \mathbf{k}_1 \\ \vdots \\ \mathbf{k}_s \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_s \end{bmatrix}. \tag{7}$$ {{eq:k2}}

We consider the system matrix we have to solve in (7) as the leading block matrix. Inverting this solves for the scaled stage vectors $(A_0 \times I)\mathbf{K}$, and we can then easily invert $A_0 \otimes I$ to get the real stage vectors. Here we have nicer field-of-value measures. In particular, note that the first operator $A_0^{-1} \otimes I$ satisfies $W(A_0^{-1} \otimes I) \geq 0$ and the second operator satisfies $W(\text{diag}(\mathcal{L}_i)) \leq 0$. To that end, if we consider "smooth" modes to be the smallest modes with respect to eigenvalues or FOVs, the "smooth" modes of (7) are largely defined by the "smooth" modes of $\text{diag}(\mathcal{L}_i)$. If $\mathcal{L}_i = \mathcal{L}_j$ for all i, j, it stands to reason that the block constant vector is a good representation of the null space. For $\mathcal{L}_i \neq \mathcal{L}_j$, we still expect that the operators are similar in some sense, so the block constant vector is still likely to be a reasonable approximation to the near null space.

1.2 Current solvers

1.3 Multigrid

The main question here is how do we develop a block multigrid method to precondition the block operator

$$\begin{bmatrix} \alpha_{11} I & \dots & \alpha_{1s} I \\ \vdots & \ddots & \vdots \\ \alpha_{s1} I & \dots & \alpha_{ss} I \end{bmatrix} - \begin{bmatrix} \mathcal{L}_1 & & \\ & \ddots & \\ & & \mathcal{L}_s \end{bmatrix} = \begin{bmatrix} \alpha_{11} I - \mathcal{L}_1 & \dots & \alpha_{1s} I \\ \vdots & \ddots & \vdots \\ \alpha_{s1} I & \dots & \alpha_{ss} I - \mathcal{L}_s \end{bmatrix}. \tag{8}$$ {{eq:k3}}

In particular, we want to project down such that we only solve one system involving $\mathcal{L}$ as a coarse-grid operator, and couple this with some kind of relaxation scheme on the "fine" grid to address the coupling between stages.

Even in the block setting, the matrix in (8) isn't symmetric and also isn't triangular like AIR likes. Thus, a first thought is to use a smoothed-aggregation type approach, where we define a single coarse-grid operator. Continuing with the idea from Section 1.1, the block constant vector is likely a good approximation of the near null space of (8), which suggests a plausible $P = \begin{bmatrix} I & \dots & I \end{bmatrix}^T$. For restriction, let us define a scaled version, $R = \begin{bmatrix} r_1 I & \dots & r_s I \end{bmatrix}^T$. Then, the coarse-grid operator is defined by

$$\begin{aligned} RAP &\coloneqq \begin{bmatrix} r_1 I & \dots & r_s I \end{bmatrix} \begin{bmatrix} \alpha_{11} I - \mathcal{L}_1 & \dots & \alpha_{1s} I \\ \vdots & \ddots & \vdots \\ \alpha_{s1} I & \dots & \alpha_{ss} I - \mathcal{L}_s \end{bmatrix} \begin{bmatrix} I \\ \vdots \\ I \end{bmatrix} \\ &= \sum_{i=1}^{s} \left(r_i \left(\sum_{j=1}^{s} \alpha_{ij} \right) I - r_i \mathcal{L}_i \right) \end{aligned} \tag{9}$$ {{eq:cg}}

Here, we propose to choose R under three constraints:

1. $\sum_{i=1}^{s} r_i = 1$ – this means that we are defining our coarse grid with some form of weighted average over $\{\mathcal{L}_i\}$.
2. $\sum_{i=1}^{s} r_i \left(\sum_{j=1}^{s} \alpha_{ij} \right) = 1$ – This ensures that the leading constant in (9) is positive (not always true if $R = P^T$) and unity. Moreover, if $\mathcal{L}_i = \mathcal{L}_j$ for all i, j, constraints (1) and (2) yield backward Euler as a coarse-grid operator.
3. Obviously for more than two stages, the above constraints are underdetermined. To constrain the full system, we propose to constrain the constants $\{r_i\}$ to all be as close as possible. This makes R as close to a scaled P^T as possible, while satisfying (1) and (2). One way to formalize this is to find

the minimum variance solution to the underdetermined set of equations. This is nicely formalized in https://arxiv.org/pdf/1906.09121.pdf. In particular, the minimum variance solution to $M\mathbf{x} = \mathbf{b}$ is given by

$$\mathbf{x}_V \coloneqq M^T(MM^T)^{-1}(\mathbf{b} - \alpha M\mathbf{1})) + \alpha\mathbf{1}, \qquad \text{where}$$
$$\alpha = \frac{\mathbf{1}^T M^T (MM^T)^{-1}\mathbf{b}}{\mathbf{1}^T M^T (MM^T)^{-1} M\mathbf{1}}.$$

Here, the first row of M is all ones, corresponding to constraint (1), the second row of M corresponds to the row sums of the Butcher tableaux, and $\mathbf{b} = [1, 1]^T$.

Note that by satisfying constraints (1) and (2), the coarse-grid operator takes the simplified form

$$RAP = I - \sum_{i=1}^{s} \mathcal{L}_i, \tag{10}$$ {{eq:cg2}}

and

$$RA = \begin{bmatrix} \left(\sum_{i=1}^s r_i\alpha_{i1}\right) I - r_1\mathcal{L}_1 & \dots & \left(\sum_{i=1}^s r_i\alpha_{is}\right) I - r_s\mathcal{L}_s \end{bmatrix}.$$

For ease of notation, define constants

$$c_j \coloneqq \sum_{i=1}^{s} r_i\alpha_{ij} - r_j.$$

Then, RA can be expressed in the simplified representation

$$RA = \begin{bmatrix} r_1(I - \mathcal{L}_1) + c_1 I & \dots & r_s(I - \mathcal{L}_s) + c_s I \end{bmatrix}. \tag{11}$$ {{eq:simp}}

1.3.1 $\mathcal{L}_i = \mathcal{L}_j$

Suppose $\mathcal{L}_i = \mathcal{L}_j$. Then, (10) reduces to $RAP = I - \mathcal{L}$, and we have

$$\begin{aligned} I - P(RAP)^{-1}RA &= I - \begin{bmatrix} I \\ \vdots \\ I \end{bmatrix} (I - \mathcal{L})^{-1} \begin{bmatrix} r_1(I - \mathcal{L}_1) + c_1 I & \dots & r_s(I - \mathcal{L}_s) + c_s I \end{bmatrix} \\ &= I - \begin{bmatrix} I \\ \vdots \\ I \end{bmatrix} \left(\begin{bmatrix} r_1 I & \dots & r_s I \end{bmatrix} + \begin{bmatrix} c_1(I - \mathcal{L})^{-1} & \dots & c_s(I - \mathcal{L})^{-1} \end{bmatrix} \right) \\ &= I - \begin{bmatrix} r_1 I & \dots & r_s I \\ \vdots & \ddots & \vdots \\ r_1 I & \dots & r_s I \end{bmatrix} - \begin{bmatrix} (I - \mathcal{L})^{-1} & & \\ & \ddots & \\ & & (I - \mathcal{L})^{-1} \end{bmatrix} \begin{bmatrix} c_1 I & \dots & c_s I \\ \vdots & \ddots & \vdots \\ c_1 I & \dots & c_s I \end{bmatrix}. \end{aligned} \tag{12}$$ {{eq:cgc}}

Recall $\sum_{i=1}^s r_i = 1$ and, thus, the leading $I - \mathcal{R}$ term has row-sum zero. The matrix with $\{c_i\}$ constants also has row sum zero. Thus, as expected, if the error modes are fixed across stages, coarse-grid correction (12) is exact. Moreover, for all schemes I have tested $r_i > 0$ (this should always be the case, but I do not have a proof) and the leading $I - \mathcal{R}$ has eigenvalues $\{0, 1, ..., 1\}$, so if error modes are moderately similar across stages, this term should remain small. Similarly, by assumption $W(\mathcal{L}) \leq 0$ and it follows that $\|(I - \mathcal{L})^{-1}\| \leq 1$. Even better, the error we expect to vary more noticeably across stages is high-frequency error, which corresponds to large eigenvalues of $\mathcal{L}$. However, applying $(I - \mathcal{L})^{-1}$ to such error is $\ll 1$, this such error in the second term should be made small by the diagonal scaling by $(I - \mathcal{L})^{-1}$. The key is to thus eliminate high-frequency error in stages and leave low frequency error moderately similar across stages.

Complementary relaxation: The next question is how to develop a complementary relaxation scheme. In particular, we want to make the error over all stages roughly similar. One heuristic to do so is by noting that $\{\mathcal{L}_i\}$ should be similar operators for all i and, thus, we particularly expect the "smooth" modes of these operators to be similar for all $\mathcal{L}_i$. Thus consider a relaxation scheme that eliminates

high frequency error on each stage, which would hypothetically leave similar low-frequency error on all stages. Due to the coupling in the larger system, I think we want something better than just a straight Gauss-Seidel type relaxation on the full system. In particular, we don't want to smooth error on the full system, we want to smooth error on each stage. To that end, let $A_0 = Q_0 R_0 Q_0^T$ be the real Schur composition of A_0, where $Q_0 Q_0^T = I$ and R_0 is block triangular, with 1×1 blocks corresponding to real eigenvalues of A_0 and 2×2 blocks corresponding to complex eigenvalues of A_0. Returning to (7) with $\mathcal{L}_i = \mathcal{L}_j$, we have

$$\begin{aligned} A_0^{-1} \otimes I - I \otimes \mathcal{L} &= (Q_0 \otimes I)\,(R_0 \otimes I - I \otimes \mathcal{L})\,(Q_0^T \otimes I), \\ \left(A_0^{-1} \otimes I - I \otimes \mathcal{L}\right)^{-1} &= (Q_0 \otimes I)\,(R_0 \otimes I - I \otimes \mathcal{L})^{-1}\,(Q_0^T \otimes I). \end{aligned} \tag{13}$$ {{eq:inv_kron}}

Applying $Q_0 \otimes I$ and $Q_0^T \otimes I$ are fairly trivial computations and only require linear combinations of different stage vectors. Thus a simple approximation to (13) is to replace $\mathcal{L} \mapsto M$, where M is some easy to invert approximation of $\mathcal{L}$, such as the lower-triangular part. This yields a relaxation scheme

$$\mathcal{M}^{-1} = (Q_0 \otimes I)\,(R_0 \otimes I - I \otimes M)^{-1}\,(Q_0^T \otimes I). \tag{14}$$ {{eq:relax1}}

A similar relaxation scheme can be developed for the more general setting, and we simple replace $\mathcal{L}_i \mapsto M_i$

$$\mathcal{M}^{-1} = (Q_0 \otimes I)\left(R_0 \otimes I - \begin{bmatrix} M_1 & & \\ & \ddots & \\ & & M_s \end{bmatrix}\right)^{-1} (Q_0^T \otimes I). \tag{15}$$ {{eq:relax2}}

Note this version is less exact because $Q_0 \otimes I$ does not commute with $\mathrm{diag}\{\mathcal{L}_i\}$, but it should accomplish a similar objective.

The one outstanding question of relaxation schemes in (14) and (15) is inverting the block 2×2 systems that arise corresponding to complex eigenvalues of A_0, $\lambda = \eta \pm \mathrm{i}\beta$, e.g.,

$$\begin{bmatrix} \eta I - M_1 & -\phi I \\ -\frac{\beta^2}{\phi} I & \eta I - M_2 \end{bmatrix} \tag{16}$$ {{eq:rel_sys}}

If we choose M_i to be the lower triangular portion of $\mathcal{L}_i$, we can reorder (16) to be block lower triangular with 2×2 blocks. This is not currently implemented, but probably would not be too hard. Alternatively, we could just apply the theory in the nonlinear paper and use a modified γ_* relaxation constant in the $(2,2)$-block.