Richardson vs. Jacobi Iteration for $A\underline{u} = \underline{f}$

- We've already seen Jacobi iteration:

$$\underline{u}_{k+1} \;=\; \underline{u}_k + D^{-1}(\underline{f} - A\underline{u}_k). \tag{1}$$

- For the case where $-A$ is the centered-difference approximation to the Laplacian with uniform grid spacing, h, we know that

$$D \;=\; a_{11} I. \tag{2}$$

- That is, multiplying by D^{-1} is equivalent to multiplying by a constant, $\alpha = 1/a_{11}$.

- This simple iteration scheme is known as *Richardson iteration*,

$$\underline{u}_{k+1} \;=\; \underline{u}_k + \alpha(\underline{f} - A\underline{u}_k). \tag{3}$$

$$\tag{4}$$

- Richardson has a direct relationship to time stepping as we illustrate shortly.

- Assuming that we start with $\underline{u}_0 = 0$, Richardson iteration also yields a solution that is a polynomial in the matrix A times the data, $\underline{f}$.

Polynomial Approximation Spaces

- Consider the first few Richardson iterates, starting with $\underline{u}_0 = 0$,

$$\begin{aligned}
\underline{u}_1 &= 0 + \alpha\underline{f} && \in \mathbb{P}_0(A)\underline{f} \\
\underline{u}_2 &= u_1 + \alpha\left(\underline{f} - A\underline{u}_1\right) \\
&= \alpha\underline{f} + \alpha\left(\underline{f} - \alpha A\underline{f}\right) \\
&= 2\alpha\underline{f} - \alpha^2 A\underline{f} && \in \mathbb{P}_1(A)\underline{f} \\
\underline{u}_3 &= 2\alpha\underline{f} - \alpha^2 A\underline{f} + \alpha\left(\underline{f} - \alpha A\underline{u}_2\right) \\
&= 3\alpha\underline{f} - 3\alpha^2 A\underline{f} + \alpha^3 A^2\underline{f} && \in \mathbb{P}_2(A)\underline{f}
\end{aligned} \tag{5}$$

- It appears that $\underline{u}_k \in \mathbb{P}_{k-1}(A)\underline{f}$, where $\mathbb{P}_j(x)$ is the space of polynomials of degree $\leq k$ in the argument x.

- A more direct way of seeing this polynomial form is to consider the *error*,

$$\underline{e}_k := \underline{u} - \underline{u}_k = (I - \alpha A)^k \underline{e}_0 \tag{6}$$

$$= (I - \alpha A)^k \underline{u} \tag{7}$$

$$= \left[I + c_1 A + c_2 A^2 + \cdots + c_k A^k\right] \underline{u}, \tag{8}$$

  where the coefficients c_j come from the binomial expansion (and involve powers of α).

- Solving for $\underline{u}_k$ and exploiting the fact that $A\underline{u} = \underline{f}$, we have:

$$\underline{u}_k = -\left[c_1 A + c_2 A^2 + \cdots + c_k A^k\right] \underline{u} \tag{9}$$

$$= -\left[c_1 + c_2 A + \cdots + c_k A^{k-1}\right] \underline{f} \tag{10}$$

$$\in \mathbb{P}_{k-1}(A)\underline{f}. \tag{11}$$

- The space $\mathbb{P}_{k-1}(A)\underline{f}$ is referred to as the k-dimensional *Krylov subspace*,

$$K_k(A)\underline{f} = span\left\{\underline{f}, A\underline{f}, A^2\underline{f}, \ldots, A^{k-1}\underline{f}\right\}. \tag{12}$$

- Note that a *preconditioned* iteration scheme of the form

$$\underline{u}_{k+1} = \underline{u}_k + M^{-1}(\underline{f} - A\underline{u}_k) \tag{13}$$

  would lead to

$$\underline{u}_k \in K_k(M^{-1}A)M^{-1}\underline{f}, \tag{14}$$

  where M is the preconditioning matrix or *preconditioner*.

- For Jacobi iteration, D is the preconditioner.

- All of these methods are Krylov-subspace methods, because they produce a solution in K_k.

- They are *not*, however, *Krylov-subspace projection* (KSP) methods, because they do not produce the best approximation in the space.

- KSPs are the next topic of interest.
- These methods generate the *projection* of $\underline{u}$ onto K_k in a *computable norm*, and thus find the closest element in K_k to the unknown $\underline{u}$.
- For A SPD, we will consider CG and also preconditioned CG (PCG), for the case where M is also SPD.
- For nonsymmetric systems, we will consider GMRES and other alternative methods.

- Before moving on to projection, we make a few more comments about Richardson iteration and its relationship to time stepping.

Relationship to Time Stepping

- Consider the system of ODEs (with $L = -A$)

$$\frac{d\underline{u}}{dt} \;=\; L\underline{u} \;+\; \underline{f}, \quad \underline{u}_0 = 0, \tag{15}$$

- Three common time steppers for this problem are

Euler Forward (EF): $\underline{u}_{k+1} \;=\; (I + \Delta t L)\,\underline{u}_k \;+\; \Delta t \underline{f}$

Euler Backward (EB): $(I - \Delta t L)\,\underline{u}_{k+1} \;=\; \underline{u}_k \;+\; \Delta t \underline{f}$

Crank-Nicolson (CN): $(I - \frac{\Delta t}{2} L)\,\underline{u}_{k+1} \;=\; (I + \frac{\Delta t}{2} L)\,\underline{u}_k \;+\; \Delta t \underline{f}$

- EF and EB are $O(\Delta t)$ (first-order) accurate.

- CN (also known as the trapezoidal rule) is $O(\Delta t^2)$ (second-order) accurate.

- pth-order accurate means that $\|\tilde{\underline{u}}(T) - \underline{u}(T)\| = O(\Delta t^p)$ as $\lambda_L \Delta t \longrightarrow 0$.

- Under the assumption that $Re(\lambda_L) < 0$, the differential equation (15) evolves to a steady-state solution, $\underline{u}_\infty$, which satisfies

$$-L\underline{u}_\infty \;=\; \underline{f}. \tag{16}$$

- The error, $\underline{e}_k := \underline{u}_\infty - \underline{u}_k$ satisfies the homogeneous equation,

$$\frac{d\underline{e}}{dt} \;=\; L\underline{e}, \quad \underline{e}_0 = \underline{u}_\infty. \tag{17}$$

- Let's look at Euler-Forward,

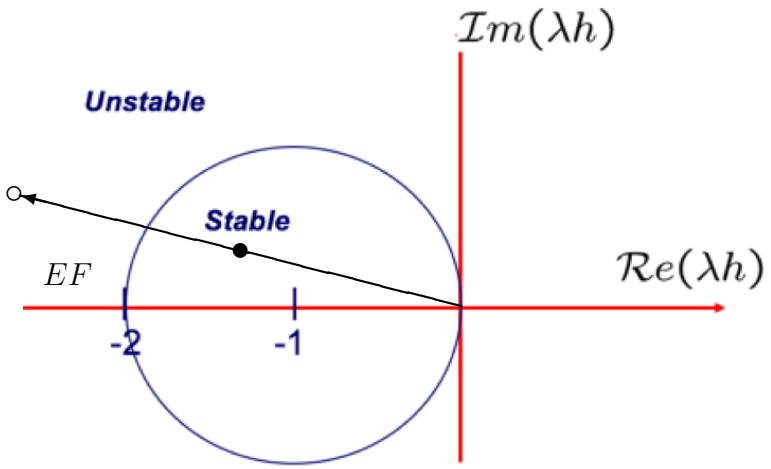
$$\underline{u}_{k+1} = (I + \Delta t L)\,\underline{u}_k + \Delta t \underline{f} \tag{18}$$

$$= (I - \Delta t A)\,\underline{u}_k + \Delta t \underline{f} \tag{19}$$

$$= \underline{u}_k + \Delta t(\underline{f} - A\underline{u}_k). \tag{20}$$

- We see that it is exactly the same as Richardson iteration with $\alpha \equiv \Delta t$.

- Notice that, if $\mathcal{R}e(\lambda_L) < 0$, it is always possible to find a Δt for which $\lambda_L \Delta t$ is inside the EF stability region.



- For example, in the figure above, the circle ($\circ$) represents a value of $\lambda_L \Delta t$ that is outside the stability region, i.e.,

$$|\,1 + \lambda_L \Delta t\,| \;>\; 1. \tag{21}$$

- We can rectify this situation by reducing Δt.

- In this example, reducing Δt by a factor of 2 leads to the value of $\lambda_L \Delta t$ represented by the bullet ($\bullet$), which is inside the stability region.

- For an n-dimensional system of equations, we must choose Δt such that *all* values of $\lambda_{L,k} \Delta t$ are in the stability region:

$$|\,1 + \lambda_{L,k} \Delta t\,| \;<\; 1, \quad k = 1, \ldots, n, \tag{22}$$

which is always possible.

- As a consequence, we can assert that it always possible to find an α such that Richardson iteration will converge, provided that $\mathcal{R}e(\lambda_{A,k}) > 0$, $k = 1, \ldots, n$.

- Assuming that A has real positive eigenvalues with $0 < \lambda_{A,1} \leq \cdots \leq \lambda_{A,n}$, the optimal value of α is given by

$$\alpha_{opt} = \frac{2}{\lambda_{A,1} + \lambda_{A,n}}. \tag{23}$$

  (See Saad, Eq. (4.33).)

- Let $\lambda_L = -\lambda_A$ denote the eigenvalues of L.

- The growth factors for the three time steppers in terms of λ_L are

$$G_{EF} = 1 + \Delta t \lambda_L \tag{24}$$

$$G_{EB} = \frac{1}{1 - \Delta t \lambda_L} \tag{25}$$

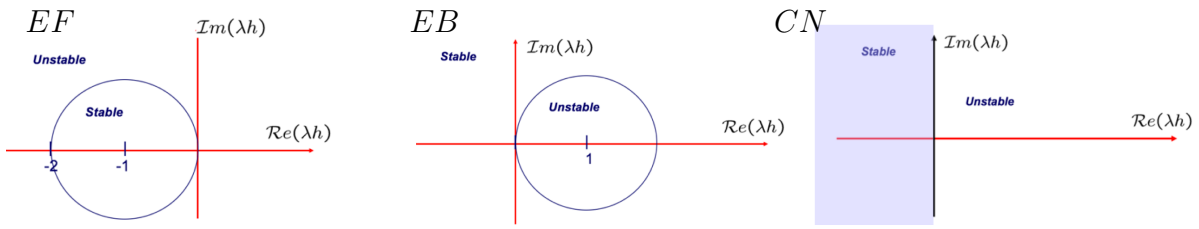
$$G_{CN} = \frac{(1 + \frac{\Delta t}{2}\lambda_L)}{(1 - \frac{\Delta t}{2}\lambda_L)} \tag{26}$$

- Note that the Taylor series for each of these expressions agrees with the first few terms of the *analytical growth factor*,

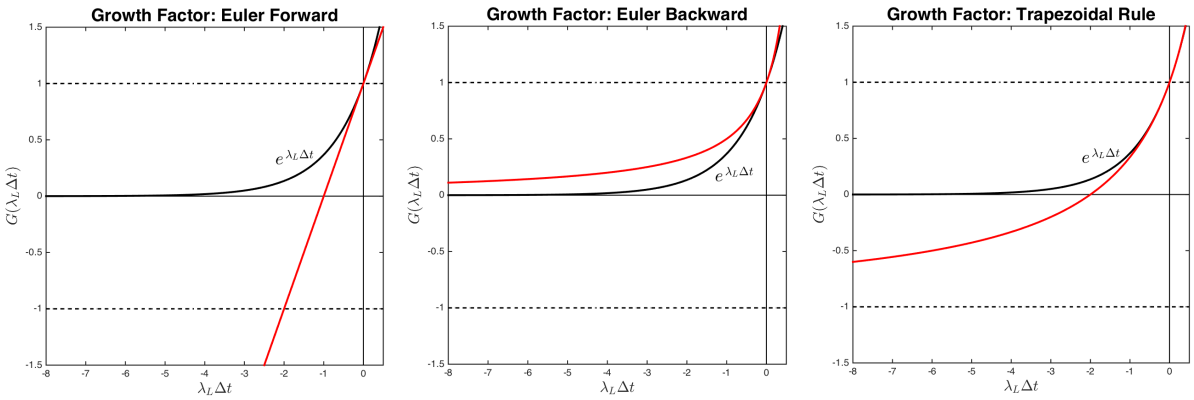
$$\widetilde{G} = e^{\lambda_L \Delta t} = 1 + \lambda_L \Delta t + \frac{(\lambda_L \Delta t)^2}{2!} + \frac{(\lambda_L \Delta t)^3}{3!} + \cdots \tag{27}$$

- EB and CN are *stable*, which means that $|G| < 1$ for all $Re(\lambda_L \Delta t) < 0$.

- The corresponding stability regions in the complex $\lambda_L \Delta t$ plane are shown below (with $h = \Delta t$ the time step size).



- For *real* $\lambda_L < 0$, we can also plot the growth factors as a function of $\lambda_L \Delta t$:



- Notice that the growth factor for EB does not change signs nor cross the x-axis.

- The growth factor for EF changes sign whenever $\lambda_L \Delta t = -1$ and exceeds 1 in modulus if $\lambda_L \Delta t < -2$.

- The growth factor for CN changes sign whenever $\lambda_L \Delta t = -2$.

- For a given λ_L, we can therefore zero out the error associated with that value by taking a single step with $\Delta t = \frac{2}{\lambda_L}$.

- Notice that, from an iterative solver perspective, EF is interesting because it doesn't require solving a system on each "time step" (which is analogous to a single Richardson iteration with $\alpha = \Delta t$).

- On the other hand, the trapezoidal rule (also known as *Crank-Nicolson*), is interesting because it has a zero crossing, it is stable for all $\lambda_L \Delta t$, and, for tensor-product grids, and there exist approximate solvers for the "implicit" part of the CN update, which are known as *alternating-direction implicit* (ADI) methods.

- ADI can be very fast, as we will see in future exercises.