

# Von Neumann Analysis of Jacobi and Gauss-Seidel Iterations

We consider the FDA to the 1D Poisson equation on a grid  $x_j$  covering  $[0,1]$  with uniform spacing  $h$ ,

$$h^{-2}(u_{j+1} - 2u_j + u_{j-1}) = f_j$$

whose 'exact' solution (to the FDA) is  $u_j^*$ . We let  $e_j^{[k]} = u_j^{[k]} - u_j^*$  be the 'error' in an iterative solution at step  $k = 0, 1, 2, \dots$

## Jacobi

For Jacobi,

$$\begin{aligned} u_j^{[k+1]} &= \frac{1}{2}(u_{j-1}^{[k]} + u_{j+1}^{[k]} - h^2 f_j) \\ u_j^* &= \frac{1}{2}(u_{j-1}^* + u_{j+1}^* - h^2 f_j) \\ e_j^{[k+1]} &= \frac{1}{2}(e_{j-1}^{[k]} + e_{j+1}^{[k]}) \end{aligned} \tag{1}$$

Forgetting for the moment about the boundary conditions, this iteration has eigenfunctions  $\exp(iqx_j)$  so we can seek solutions in the form

$$e_j^{[k]} = a^k \exp(iqx_j) \tag{2}$$

Here  $a(q)$  is the *amplification factor* of wavenumber  $q$ , which we'd like to determine. An iteration will be convergent if  $|a(q)| < 1$  for errors of all wavenumber  $q_n$  that can be supported by the grid for the given FDA. In particular, with Dirichlet BCs at  $x = 0, 1$ , the error can be expressed (using odd extension plus a DFT) purely in terms of sines with wavenumbers  $q_n = n\pi, n = 1, \dots, m$ . Harmonic  $n = 1$  ( $q_1 = \pi$ ) corresponds to a single half-wavelength across the domain, and harmonic  $m + 1$  ( $q_{m+1} = \pi/h$ ) would correspond to a half-wavelength across a single grid spacing; this is called the Nyquist wavenumber.

Substituting (2) into (1), we obtain an expression for  $a$ :

$$\begin{aligned} a^{k+1} \exp(iqx_j) &= \frac{1}{2}(a^k \exp(iq[x_j - h]) + a^k \exp(iq[x_j + h])) \\ a &= \cos(qh) \end{aligned} \tag{3}$$

Thus the amplification factors of the  $m$  wavenumbers supported by the grid are:

$$a_n = \cos(q_n h) = \cos(n\pi h), \quad n = 1, \dots, m, \quad h = \frac{1}{m+1}$$

All of these amplification factors have magnitude less than 1. The lowest wavenumber has  $a_1 = \cos(\pi h)$  and the highest wavenumber has  $a_m = -\cos(\pi h)$ .

To visualize how the Jacobi iteration affects an error with very long wavelength, consider the action of the Jacobi iteration of a sequence of errors all of which are 1, i. e. not varying in  $x$  at all.

$$\begin{array}{lcl} e_j^{[k]} : & 1 & 1 & 1 & 1 & 1... \\ e_j^{[k+1]} : & 1 & 1 & 1 & 1 & 1... \end{array}$$

In this limit, the Jacobi iteration is unable to reduce the error.

Similarly , consider the action of the Jacobi iteration of a sequence of errors oscillating with wavelength  $2h$ :

$$\begin{array}{lcl} e_j^{[k]} : & +1 & -1 & +1 & -1 & +1... \\ e_j^{[k+1]} : & -1 & +1 & -1 & +1 & -1... \end{array}$$

Again, the Jacobi iteration is unable to reduce the error, though now it flips its sign.

On the other hand, there is a sweet spot at intermediate wavelengths, e. g.  $4h$ :

$$\begin{array}{lcl} e_j^{[k]} : & +1 & 0 & -1 & 0 & +1... \\ e_j^{[k+1]} : & 0 & 0 & 0 & 0 & 0... \end{array}$$

For this wavelength a single Jacobi iteration removes all of the error.

The overall convergence rate of the Jacobi iteration is limited by the  $|a|$  of the slowest-converging wavenumber allowed by the grid. This may sound familiar, and in fact it is just the analysis of the previous lecture with ‘maximum amplification factor’ replacing ‘spectral radius of the iteration matrix  $G$ ’. We are just repeating that analysis in a somewhat easier perspective to visualize.

## Gauss-Seidel

For GS,

$$e_j^{[k+1]} = \frac{1}{2}(e_{j-1}^{[k+1]} + e_{j+1}^{[k]}) \quad (4)$$

For a Von Neumann analysis, we substitute (2) into (4):

$$\begin{aligned} a &= \frac{1}{2}(a \exp(-iqh) + \exp(iqh)) \\ a &= \frac{\exp(iqh)}{2 - \exp(-iqh)} \end{aligned} \quad (5)$$

Now  $a$  is a complex function of  $q$ , but we can still calculate its magnitude:

$$|a| = \frac{|\exp(iqh)|}{|2 - \exp(-iqh)|} = [(2 - \cos qh)^2 + \sin^2 qh]^{-1/2} = [5 - 4 \cos qh]^{-1/2} \quad (6)$$

The convergence of GS is analyzed by calculating  $a_n$  for all the grid-supported wavenumbers  $q_n$ . One technical wrinkle is that when  $a$  is complex, applying an iteration of GS to a sinusoidal solution which satisfies the Dirichlet BCs will not exactly preserve its sinusoidal character at the boundaries. Thus, unlike with Jacobi there is no longer a precise correspondence between the eigenvalues of the iteration matrix  $G$  and the amplification factors  $a_n$  of the  $m$  grid-supported wavenumbers. Nevertheless, the analysis still provides (and motivates) and correct answers in the limit of large  $m$ .

The lowest grid-supported wavenumber  $q_1 = \pi$  gives the largest  $a$ . Assuming  $\pi h \ll 1$ ,

$$\rho^{GS} = |a_1| = [5 - 4 \cos \pi h]^{-1/2} \approx [5 - 4(1 - \pi^2 h^2/2)]^{-1/2} = [1 + \pi^2 h^2/2]^{-1/2} \approx 1 - \pi^2 h^2 \quad (7)$$

The number of iterations needed for GS to converge to a given tolerance is inversely proportional to  $-\log \rho^{GS} \approx \pi^2 h^2$ , which is twice as large as  $-\log \rho^J$ . Thus GS converges twice as fast (with half as many iterations) as Jacobi, as claimed without proof in the last lecture.

The action of GS on very long wavelengths is similar to Jacobi, but for a sequence of errors oscillating with the wavelength  $2h(q = \pi/h)$ , with  $\exp(iqh) = -1$ ,  $a = -1/3$  for GS compared to -1 for Jacobi, i. e. GS is able to strongly damp the shortest wavelengths supported by the grid:

$$\begin{array}{lcccccc} e_j^{[k]} : & +1 & -1 & +1 & -1 & +1 \dots \\ e_j^{[k+1]} : & -\frac{1}{3} & +\frac{1}{3} & -\frac{1}{3} & +\frac{1}{3} & -\frac{1}{3} \dots \end{array}$$

For an error component with an intermediate wavelength  $4h$  for which  $\exp(iqh) = i$ ,  $a = i/(2 + i)$  has magnitude  $5^{-1/2}$  and hence this error component is also strongly damped at each iteration.

## Take-home points

Two important insights from the above analysis are:

1. Both Jacobi and GS are least efficient at damping the longest-wavelength components of the error, and this efficiency worsens as the grid spacing  $h$  is reduced
2. Both Jacobi and GS overdamp the shortest-wavelength ( $2h$ ) component of the error, but GS still decreases its amplitude substantially in each iteration.

## Multigrid iteration

The first of these insights inspired the *multigrid* method to be discussed in a later lecture, in which the FDA is first formulated on a very coarse grid  $h = 1/2$  (decreasing the error at the longest wavelengths), on which a Jacobi or GS iteration is taken, then the grid is refined to spacing  $h/4$ , and an iteration is taken on this grid, etc., until we reach the final desired grid spacing. In each such cycle, the error at all wavelengths is reduced, greatly speeding up the convergence of the method, and since the coarse-grid iterations involve much less computation, there is little added overall computational expense. It will turn out that multigrid converges to the FDA solution of Poisson's equation in  $O(\log m)$  flops per grid point, similarly efficient to the DFT approach, but with much broader applicability (at the expense of much more algorithmic complexity).

## SOR iteration

The second of these insights inspired the *successive over-relaxation* (SOR) method, in which the GS correction at step  $k + 1$  is multiplied by a factor  $\omega > 1$  that is designed to improve the convergence at long wavelengths while still keeping  $|a| < 1$  at short wavelengths:

$$\begin{aligned} u_j^{GS} &= \frac{1}{2}(u_{j-1}^{[k+1]} + u_{j+1}^{[k]} - h^2 f_j) \\ u_j^{[k+1]} &= u_j^{[k]} + \omega(u_j^{GS} - u_j^{[k]}) \end{aligned} \tag{8}$$

See RJL4.2.2 for a brief discussion of SOR for the 1D Poisson equation, including the optimal choice

$$\omega_{\text{opt}} = \frac{2}{1 + \sin \pi h} \approx 2 - 2\pi h,$$

for which the spectral radius of the SOR iteration matrix  $G$  is

$$\rho_{\text{opt}} = \omega_{\text{opt}} \approx 1 - 2\pi h,$$

and the required number of iterations to converge to  $O(h^2)$  error is (remembering  $h = O(m^{-1})$ )

$$k_{\text{opt}} = \log h / \log \rho_{\text{opt}} = O(m \log m).$$

This is much more efficient than the  $O(m^2 \log m)$  iterations required for convergence of Jacobi and Gauss-Seidel, with no increase in complexity. SOR still is much less efficient in 1D than using a tridiagonal solver, but it is competitive in 2D and superior in 3D than a direct sparse LU solver.