Technische Universität München
Lehrstuhl für Informatik V
Univ.-Prof. Dr. M. Bader
Carsten Uphoff, Nikola Tchipev

SS 2018
Sheet 1
13.04.2018

Scientific Computing II

Iterative Solvers

Exercise 1: Repetition "Finite Differences"

Consider the one-dimensional Poisson equation with homogeneous Dirichlet conditions

$$\begin{aligned} -\frac{d^2u}{dx^2} &= f(x), \quad x \in (0,1), \\ u(0) = u(1) &= 0. \end{aligned} \tag{1}$$

(a) Discretise the Poisson equation by finite differences using an equidistant mesh size $h = 1/N$ and $N+1$ grid points.

(b) Write the finite difference approximation from (a) in matrix-vector form $Au = b$. Therefore, define the entries of the matrix $A \in \mathbb{R}^{N+1 \times N+1}$.

(c) Write the finite difference approximation as $Au = b$, where $A \in \mathbb{R}^{N-1 \times N-1}$ and $b \in \mathbb{R}^{N-1}$, by substituting the values for $u(0)$ and $u(1)$.

Solution:

(a) The discrete approximation reads

$$\begin{aligned} \frac{-u_{i+1} + 2u_i - u_{i-1}}{h^2} &= f_i, \quad i = 1, \dots, N-1, \\ u_0 = u_1 &= 0, \end{aligned} \tag{2}$$

where $u_i := u(ih)$, $f_i := f(ih)$.

(b) Eq. (2) corresponds to one line in the equation system; each line for inner points $i = 1, \dots, N-1$ is thus of the form

$$\begin{pmatrix} \dots & -\frac{1}{h^2} & \frac{2}{h^2} & -\frac{1}{h^2} & \dots \end{pmatrix} \cdot \begin{pmatrix} \vdots \\ u_{i-1} \\ u_i \\ u_{i+1} \\ \vdots \end{pmatrix} = f_i.$$

The right hand side vector b thus corresponds to the function f, i.e. $b_i := f_i$ for $i = 1, \ldots, N-1$. the boundary points yield an identity entry in the matrix and a right hand side $b_0 := 0$, $b_N := 0$. The matrix A thus arises as

$$A_{ij} := \begin{cases} 1 & \text{if} \quad (i,j) = (0,0),\ (i,j) = (N,N), \\ -\dfrac{1}{h^2} & \text{if} \quad |i-j| = 1 \text{ and } i \neq 0, N, \\ \dfrac{2}{h^2} & \text{if} \quad i = j \text{ and } i \neq 0, N, \\ 0 & \text{otherwise,} \end{cases}$$

with $0 \leq i, j \leq N$.

(c) As $u(0) = u(1) = 0$ we have that $u_0 = u_N = 0$. Hence,

$$A_{ij} := \begin{cases} -\frac{1}{h^2} & \text{if} \quad |i-j| = 1, \\ \frac{2}{h^2} & \text{if} \quad i = j, \\ 0 & \text{otherwise,} \end{cases}$$

with $1 \leq i, j \leq N-1$.

Row-Wise Derivation of Smoothers

Besides the matrix-based derivation (see lecture slides), most smoother methods can also be easily derived row-wise. Each row of the linear system reads:

$$\sum_j A_{ij} u_j = b_i, \quad i = 1, ..., N \tag{3}$$

The i-row can be separated into $A_{ii} u_i + \sum_{i \neq j} A_{ij} u_j = b_i$. Rearranging yields

$$u_i = \frac{1}{A_{ii}} \left(b_i - \sum_{j \neq i} A_{ij} u_j \right). \tag{4}$$

- *Jacobi method*
  For the right hand side of Eq. (4) we use the iterative solution at iteration step (n) and obtain the new iterative solution $(n+1)$:

$$u_i^{(n+1)} = \frac{1}{A_{ii}} \left(b_i - \sum_{j \neq i} A_{ij} u_j^{(n)} \right).$$

- *Weighted Jacobi method*
  We introduce a weighting factor ω and split the left hand side $u_i = \frac{1}{\omega} u_i + \left(1 - \frac{1}{\omega}\right) u_i$.

We can now evaluate parts of u_i at (n) or $(n+1)$. We obtain:

$$\begin{aligned} \tfrac{1}{\omega} u_i^{(n+1)} + \left(1 - \tfrac{1}{\omega}\right) u_i^{(n)} &= \frac{1}{A_{ii}} \left(b_i - \sum_{j \neq i} A_{ij} u_j^{(n)} \right) \\ \Leftrightarrow \qquad u_i^{(n+1)} &= \frac{\omega}{A_{ii}} \left(b_i - \sum_{j \neq i} A_{ij} u_j^{(n)} \right) + (1-\omega)\, u_i^{(n)}. \end{aligned}$$

The right hand side of the last equation corresponds to a weighted average of the last solution $u_i^{(n)}$ and the solution predicted by the (non-weighted) Jacobi method.

- *Gauss-Seidel method*
  We solve the right hand side of Eq. (4) with both new and old values $u_j^{(n)}$, $u_j^{(n+1)}$. By this method, we can only use one array to store the solution u since we can immediately write the entries at $(n+1)$ into the original positions of the solutions $u_j^{(n)}$. The method reads:
  $$u_i^{(n+1)} = \frac{1}{A_{ii}} \left(b_i - \sum_{j<i} A_{ij} u_j^{(n+1)} - \sum_{j>i} A_{ij} u_j^{(n)} \right).$$

- *Successive-Over-Relaxation method*
  Similar to weighted Jacobi, we split the left hand side $u_i = \frac{1}{\omega} u_i + \left(1 - \frac{1}{\omega}\right) u_i$ and evaluate parts of u_i at (n) or $(n+1)$, but use both old and new values on the right hand side.

$$\begin{aligned} \tfrac{1}{\omega} u_i^{(n+1)} + \left(1 - \tfrac{1}{\omega}\right) u_i^{(n)} &= \frac{1}{A_{ii}} \left(b_i - \sum_{j<i} A_{ij} u_j^{(n+1)} - \sum_{j>i} A_{ij} u_j^{(n)} \right) \\ \Leftrightarrow \qquad u_i^{(n+1)} &= \frac{\omega}{A_{ii}} \left(b_i - \sum_{j<i} A_{ij} u_j^{(n+1)} - \sum_{j>i} A_{ij} u_j^{(n)} \right) + (1-\omega)\, u_i^{(n)}. \end{aligned}$$

  So the right hand side is an average of the last solution $u_i^{(n)}$ and the solution predicted by the Gauß-Seidel method.

Exercise 2: Eigenvalues and eigenvectors

Show that the discretised sine, i.e. $u_i = sin(k\pi ih)$, is an eigenvector with eigenvalue $\lambda = (4/h^2)\sin^2(k\pi h/2)$ of the finite difference matrix A in Exercise 1(c). You may use the following trigonometric identities:

$$\sin(a+b) + \sin(a-b) = 2\sin(a)\cos(b) \tag{5}$$

$$\cos(2x) = 1 - 2\sin^2(x) \tag{6}$$

Solution:
An eigenvector of A has to satisfy the equation $Au = \lambda u$. So for each row of A we have

$$\frac{-u_{i+1} + 2u_i - u_{i-1}}{h^2} = \lambda u_i. \tag{7}$$

By rearranging terms in (7), we obtain

$$(2 - h^2\lambda)u_i = u_{i-1} + u_{i+1}. \tag{8}$$

Using the ansatz $u_i = \sin(cih)$, i.e. we assume that the eigenvectors have sinusoidal shape, we can simplify (8) in the following way:

$$(2 - h^2\lambda)\sin(cih) = \sin\left(c(i-1)h\right) + \sin\left(c(i+1)h\right) \overset{(5)}{=} 2\sin(cih)\cos(ch)$$

The equation is satisfied if $\sin(cih) = 0$. Otherwise we may divide by $\sin(cih)$ and obtain

$$\lambda = \frac{2}{h^2}\left(1 - \cos(ch)\right) \overset{(6)}{=} \frac{4}{h^2}\sin^2\left(\frac{ch}{2}\right).$$

The question remains if there are restrictions on c. Here, we have keep in mind that (7) is slightly different on the boundary as u_0 and u_N are not part of the equation system as derived in Exercise 1(c). However, if u_0 and u_N are added as zeros to (7), then the equation holds also on the boundary. But then we have to require that

$$\begin{aligned} u_0 &= \sin(c \cdot 0 \cdot h) = 0 \\ u_N &= \sin(c \cdot N \cdot h) \overset{h=1/N}{=} \sin(c) = 0, \end{aligned}$$

which leads to the requirement that $c = k\pi, k \in \mathbb{Z} \setminus \{0\}$.

Exercise 3: Fourier Analysis for Jacobi Methods

In this exercise we are interested in the smoothing properties of the weighted Jacobi method applied on the Problem of Exercise 1 with zero right-hand side ($f(x) = 0$).

(a) Formulate the weighted Jacobi method and write the iteration scheme in the form

$$u_i^{(n+1)} = \sum_j M_{ij} u_i^{(n)}.$$

(b) Determine the eigenvalues and eigenvectors of M.

Hints: From the lecture you know that $M := I - \omega \operatorname{diag}(A)^{-1} A$ for weighted Jacobi. Combine this fact with the results from Exercise 2 in order to compute the eigendecomposition easily.

(c) How can the eigendecomposition be used to calculate the reduction of error in each smoothing iteration?

(d) For a multigrid algorithm we are interested in removing the "high frequencies", say $N/2 \leq k \leq N$. Show that $\omega = 2/3$ is the best choice in the sense that it solves

$$\omega = \arg\min_{\omega'} \max_{N/2 \leq k \leq N} |\lambda_k(\omega')|,$$

where λ_k is the k-th eigenvalue. What is the reasoning of this criterion?

Solution:

(a) Using the row-wise derivation method, the weighted Jacobi scheme reads

$$u_i^{(n+1)} = \frac{\omega}{2}\left(u_{i-1}^{(n)} + u_{i+1}^{(n)}\right) + (1-\omega)u_i^{(n)}.$$

Hence, the entries of M are

$$M_{ij} = \begin{cases} \frac{\omega}{2} & \text{if} \quad |i-j| = 1, \\ 1-\omega & \text{if} \quad i = j, \\ 0 & \text{otherwise,} \end{cases}$$

with $1 \leq i, j \leq N-1$.

(b) We already know the eigendecomposition of A from Exercise 2, i.e. $Av_k = \lambda_k v_k$ where v_k is the k-th eigenvector and λ_k is the k-th eigenvalue, respectively, and $k = 1, \ldots, N-1$. Futhermore, we know that

$$\operatorname{diag}(A)^{-1} = \frac{h^2}{2} I.$$

It can be easily shown that the following is true for every $k = 1, \ldots, N-1$:

$$Mv_k = \left(I - \omega\frac{h^2}{2}A\right)v_k = \left(1 - \omega\frac{h^2}{2}\lambda_k\right)v_k =: \mu_k v_k.$$

Hence, the eigenvectors of M are v_k and the eigenvalues are

$$\mu_k = 1 - \omega\frac{h^2}{2}\lambda_k = 1 - \omega\frac{h^2}{2}\frac{4}{h^2}\sin^2\left(\frac{k\pi h}{2}\right) = 1 - 2\omega\sin^2\left(\frac{k\pi h}{2}\right).$$

(c) As derived in the lecture, the (point-wise) error in the n-th iteration is

$$e^{(n)} = M^n e^{(0)}.$$

We may decompose any error into a linear combination of eigenvectors v_k, as they are a basis of $\mathbb{R}^{N-1}$. That is,

$$e^{(0)} = \sum_{k=1}^{N-1} \alpha_k v_k.$$

Inserting this into the error equation yields

$$e^{(n)} = M^n\left(\sum_{k=1}^{N-1}\alpha_k v_k\right) = \sum_{k=1}^{N-1}\alpha_k M^n v_k = \sum_{k=1}^{N-1}\alpha_k \lambda_k^n v_k.$$

If we take the norm on both sides and use the triangle inequality we obtain

$$\|e^{(n)}\| = \left\|\sum_{k=1}^{N-1}\alpha_k\lambda_k^n v_k\right\| \leq \sum_{k=1}^{N-1}|\alpha_k||\lambda_k|^n\|v_k\|. \tag{9}$$

Equation (9) leads to the following interpretation: As the eigenvectors follow a sine function, we may think of them as waves with a single frequency. The error can be decomposed into frequency components, similar to a Fourier analysis. The eigenvalues then give an upper bound on the error reduction of each frequency component. So even if we are unaware of the error (because the true solution is unknown in general) and of the α_k values, we still know that the error in the k-th frequency component is reduced by the factor $|\lambda_k|$ in one iteration.

(d) When seeing a minimax problem, one can think of it in terms of a protagonist and an antagonist. Here, the protagonist (you) tries to minimise the eigenvalues of M such that the error is reduced fastest in each iteration. Then, the antagonist (Murphy) picks the largest eigenvalue, i.e. the one leading to the worst error reduction. (In a sense, Murphy picks the worst of all the things that can go wrong.) The good thing about a minimax problem is that you get a guaranteed bound that holds even in the worst case.

Coming to the problem at hand: As the eigenvalues are monotonically decreasing, it is sufficient to take the maximum over the first and the last eigenvalue in the interval, i.e.

$$\omega = \arg\min_{\omega'} \max \left\{ |\lambda_{N/2}(\omega')|, |\lambda_N(\omega')| \right\} = \arg\min_{\omega'} \max \left\{ |1-\omega'|, |1-2\omega'| \right\}.$$

The extreme points of the maximum function must be at at one of the kinks or intersection points of $|1-\omega'|$ and $|1-2\omega'|$, because otherwise one could follow the slope to a smaller or larger point (hint: create a plot of $\max\{|\lambda_{N/2}(\omega')|, |\lambda_N(\omega')|\}$). Here, the only relevant case (for finding the minimum) is

$$\lambda_{N/2} = 1-\omega' = -\left(1-2\omega'\right) = -\lambda_N,$$

which leads to $\omega = 2/3$. With $\omega = 2/3$, the maximum absolute eigenvalue for the high frequencies is $1/3$. Hence, all errors in high frequencies ($k \geq N/2$) are guaranteed to be three times lower after one iteration.