# MA3227 Numerical Analysis II

## Lecture 10: Jacobi Method

Simon Etter

2019/2020

# Jacobi Method

**Introduction**

Recap: our overarching goal is to find an $\mathcal{O}(N)$ solver for $-\Delta_n^{(d)} u_n = f$.

So far, we have the following:

| | LU | Krylov |
|---|---|---|
| $d = 1$ | $\mathcal{O}\big(N\big)$ | $\mathcal{O}(N^2)$ |
| $d = 2$ | $\mathcal{O}\big(N^{3/2}\big)$ | $\mathcal{O}(N^{3/2})$ |
| $d = 3$ | $\mathcal{O}\big(N^2\big)$ | $\mathcal{O}(N^{4/3})$ |

Jacobi's method is yet another method for the same problem which scales even worse than LU or Krylov methods.

The reason why we discuss Jacobi is because it provides the foundation for an algorithm which finally achieves our $\mathcal{O}(N)$ goal.

# Jacobi Method

**Jacobi iteration for Poisson equation**

Assume we have an initial guess $x_0$ for the linear system

$$(3+1)^2 \begin{pmatrix} 2 & -1 & \\ -1 & 2 & -1 \\ & -1 & 2 \end{pmatrix} \begin{pmatrix} x[1] \\ x[2] \\ x[3] \end{pmatrix} = \begin{pmatrix} b[1] \\ b[2] \\ b[3] \end{pmatrix}.$$

Solving each equation for the "diagonal" unknown yields

$$\begin{aligned} x_1[1] &= \tfrac{1}{2}\big(\tfrac{b[1]}{(3+1)^2} + x_0[2]\big), \\ x_1[2] &= \tfrac{1}{2}\big(\tfrac{b[2]}{(3+1)^2} + x_0[1] + x_0[3]\big), \\ x_1[3] &= \tfrac{1}{2}\big(\tfrac{b[3]}{(3+1)^2} + x_0[2]\big). \end{aligned}$$

The resulting $x^{(1)}$ is not the exact solution in general, but we may hope that it is a better approximation to $x$ than $x^{(0)}$.

Idea: iterate the map $x^{(0)} \mapsto x^{(1)}$ until convergence.
See `jacobi_step()`.

# Jacobi Method

**Gauss-Seidel iteration**

Minor modification of Jacobi:

on each line, we use the most recent version of $x[i]$ currently available.

$$\begin{aligned} x_1[1] &= \tfrac{1}{2}\left(\tfrac{b[1]}{(3+1)^2} + x_0[1]\right), \\ x_1[2] &= \tfrac{1}{2}\left(\tfrac{b[2]}{(3+1)^2} + x_1[1] + x_0[3]\right), \\ x_1[3] &= \tfrac{1}{2}\left(\tfrac{b[3]}{(3+1)^2} + x_1[2]\right). \end{aligned}$$

See `gauss_seidel_step()`.

Gauss-Seidel is a well-known algorithm that you should have heard of.

It has some minor advantages and disadvantages compared to Jacobi, but overall the two methods are very similar. If you understand Jacobi, you will have no trouble understanding Gauss-Seidel.

I will only discuss Jacobi in the following for simplicity.

# Jacobi Method

**Discussion of Jacobi-type methods**

- Good: single iterations run very fast.
- Bad: many iterations are needed to reach a reasonable accuracy.

See `plot_convergence()`.

**Next steps**

As usual, we want to quantify the rate of convergence through an estimate of the form $\|x_k - x\| = \mathcal{O}\big(f(k)\big)$.

A useful tool to derive this estimate is a matrix formula for the Jacobi iteration as provided on the next slide.

# Jacobi Method

**Jacobi iteration (abstract definition)**

Let $A$ be an invertible matrix with nonzero diagonal $D$.
The Jacobi iteration can then be written as follows.

**Algorithm 1** Jacobi iteration

1: **for** $k = 1, 2, \ldots$ **do**
2: $\quad x_k = D^{-1}\Big(b - (A - D)\, x_{k-1}\Big)$
3: **end for**

The next slide demonstrates that the general definition reduces to the concrete Jacobi iteration when applied to the Poisson matrix.

# Jacobi Method

**Example: Jacobi iteration for Poisson matrix**

Consider the matrix

$$A = (3+1)^2 \begin{pmatrix} 2 & -1 & \\ -1 & 2 & -1 \\ & -1 & 2 \end{pmatrix}$$

The Jacobi iteration takes the form

$$\begin{aligned} x^{(1)} &= D^{-1}\left(b - (A-D)\,x^{(0)}\right) \\ &= \begin{pmatrix} 2 & & \\ & 2 & \\ & & 2 \end{pmatrix}^{-1} \left( \frac{1}{(3+1)^2} \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} - \begin{pmatrix} & -1 & \\ -1 & & -1 \\ & -1 & \end{pmatrix} \begin{pmatrix} x_1^{(0)} \\ x_2^{(0)} \\ x_3^{(0)} \end{pmatrix} \right) \\ &= \begin{pmatrix} \frac{1}{2}\left(\frac{b[1]}{(3+1)^2} + x_0[2]\right), \\ \frac{1}{2}\left(\frac{b[2]}{(3+1)^2} + x_0[1] + x_0[3]\right), \\ \frac{1}{2}\left(\frac{b[3]}{(3+1)^2} + x_0[2]\right). \end{pmatrix}. \end{aligned}$$

This is precisely the iteration that we had before.

# Jacobi Method

**Towards a convergence estimate for Jacobi iteration**

Using the Jacobi iteration formula and $b = Ax$, we obtain

$$\begin{aligned} x_k - x &= D^{-1}\left(b - (A-D)\,x_{k-1}\right) - x \\ &= D^{-1}\left(Ax - Dx - (A-D)\,x_{k-1}\right) \\ &= -D^{-1}(A-D)\left(x_{k-1} - x\right). \end{aligned}$$

Applying this formula repeatedly yields

$$x_k - x = R^k\left(x_0 - x\right) \qquad \text{where} \qquad R = -D^{-1}(A-D).$$

Let us expand initial error in terms of eigenvectors $u_\ell$ of $R$,

$$x_0 - x = \sum_{\ell=1}^{N} c_\ell\, u_\ell.$$

Denoting the associated eigenvalues by $\lambda_\ell$, we obtain

$$x_k - x = R^k\left(x_0 - x\right) = \sum_{\ell=1}^{N} c_\ell\, R^k\, u_\ell = \sum_{\ell=1}^{N} c_\ell\, \lambda_\ell^k\, u_\ell.$$

# Jacobi Method

**Towards a convergence estimate for Jacobi iteration**

From previous slide:

$$x_k - x = \sum_{\ell=1}^{N} c_\ell \, \lambda_\ell^k \, u_\ell .$$

Assume $\|u_\ell\| = 1$ and eigenvalues are sorted such that $|\lambda_1| \geq \ldots \geq |\lambda_N|$. Then,

$$\|x_k - x\| \leq \sum_{\ell=1}^{N} |c_\ell| \, |\lambda_\ell|^k \leq \left( \sum_{\ell=1}^{N} |c_\ell| \right) |\lambda_1|^k .$$

**Conclusion**

Jacobi iterates $x_k$ satisfy

$$\|x_k - x\| \leq C \, |\lambda_{\max}|^k$$

where $\lambda_{\max}$ is the eigenvalue of largest absolute value of

$$R = -D^{-1}(A - D).$$

# Jacobi Method

**Jacobi's method applied to Poisson's equation**

$$A = -\Delta_n^{(1)} = (n+1)^2 \begin{pmatrix} 2 & -1 & & \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{pmatrix} \quad \Longrightarrow \quad D = (n+1)^2 \begin{pmatrix} 2 & & \\ & \ddots & \\ & & 2 \end{pmatrix}.$$

Let us introduce the following notation:

- $\lambda_\ell = (n+1)^2 \left(2 - 2\cos\left(\pi \frac{\ell}{n+1}\right)\right)$: eigenvalues of $-\Delta_n^{(1)}$.
- $\hat{\lambda}_\ell$: eigenvalues of $R = -D^{-1}(A - D)$.

Since $D \propto I$, we obtain

$$\hat{\lambda}_\ell = -\frac{1}{2\,(n+1)^2}\left(-\lambda_\ell - 2\,(n+1)^2\right) = \cos\left(\pi \frac{\ell}{n+1}\right).$$

Largest absolute value is achieved for $\ell = 1$ and $\ell = n$ for which we have

$$\hat{\lambda}_1 = -\hat{\lambda}_n = \cos\left(\frac{\pi}{n+1}\right) = 1 - \mathcal{O}(n^{-2}).$$

The same result holds for $\Delta_n^{(d)}$ because we have $\lambda_{\max}^{(d)} = d\,\lambda_{\max}^{(1)}$ but also $D^{(d)} = 2d\,(n+1)^2\, I$ and thus the $d$ cancel.

# Jacobi Method

**Jacobi's method applied to Poisson's equation (continued)**

It follows from the above that the Jacobi iterates satisfy

$$\|x_k - x\| \leq C \cos\left(\tfrac{\pi}{n+1}\right)^k = C\left(1 - \mathcal{O}(n^{-2})\right)^k$$

In particular, to guarantee $\|x_k - x\| \leq \varepsilon$ we must choose

$$k = \frac{\log(\varepsilon/C)}{\log\left(\cos\left(\tfrac{\pi}{n+1}\right)\right)} = \mathcal{O}(n^2).$$

This yields the following runtimes.

| | LU | Krylov | Jacobi |
|---|---|---|---|
| $d = 1$ | $\mathcal{O}(N)$ | $\mathcal{O}(N^2)$ | $\mathcal{O}(N^3)$ |
| $d = 2$ | $\mathcal{O}(N^{3/2})$ | $\mathcal{O}(N^{3/2})$ | $\mathcal{O}(N^2)$ |
| $d = 3$ | $\mathcal{O}(N^2)$ | $\mathcal{O}(N^{4/3})$ | $\mathcal{O}(N^{5/3})$ |

# Jacobi Method

**Summary**

- Jacobi's method is

$$x_k = D^{-1}\left(b - (A - D)\, x_{k-1}\right)$$

  with $D = \operatorname{diag}(A)$.
- Jacobi iterates satisfy

$$\|x_k - x\| \le C\, |\lambda_{\max}|^k$$

  where $\lambda_{\max}$ is the eigenvalue of largest absolute value of $R = -D^{-1}\,(A - D)$.
- For $A = -\Delta_n^{(d)}$, we have $|\lambda_{\max}| = 1 - \mathcal{O}(n^2)$.