Technische Universität Braunschweig

Partitioned Methods for Multifield Problems

Rang, 3.5.2017

Contents

- **Linear iteration schemes**
- **A motivating example**

Contents

- **Linear iteration schemes**
- **A motivating example**

Linear iteration schemes

- Let $A \in \mathbb{R}^{n,n}$ be a regular matrix, $\mathbf{b} \in \mathbb{R}^n$, and let $\mathbf{x} \in \mathbb{R}^n$ be the exact solution of the linear system

$$A\mathbf{x} = \mathbf{b}.$$

- Let $\mathbf{x}^{(k)}$ be a given approximation of the solution of the linear system.

Linear iteration schemes

In the following we are looking for a corrected vector $\mathbf{v}^{(k)}$ such that the approximation

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{v}^{(k)}$$

is more precise. Therefore let

$$\mathbf{d}^{(k)} := \mathbf{b} - A\mathbf{x}^{(k)}$$

be the defect of $\mathbf{x}^{(k)}$.

Linear iteration schemes

The best correction $\mathbf{v}_*^{(k)}$ would be the correction which satisfies the equation

$$\mathbf{x}^{(k)} + \mathbf{v}_*^{(k)} = \mathbf{x}.$$

Inserting yields

$$A\left(\mathbf{x}^{(k)} + \mathbf{v}_*^{(k)}\right) = \mathbf{b}$$

and

$$A\mathbf{v}_*^{(k)} = \mathbf{b} - A\mathbf{x}^{(k)} = \mathbf{d}^{(k)}.$$

Thus $\mathbf{v}_*^{(k)}$ is the solution of $A\mathbf{v}_*^{(k)} = \mathbf{d}^{(k)}$, i.e.

$$\mathbf{v}_*^{(k)} = A^{-1}\mathbf{d}^{(k)}.$$

Linear iteration schemes

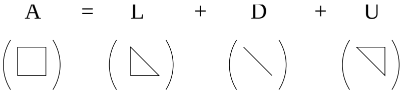
- The calculation of A^{-1} would be too expensive and with the knowledge of A^{-1} the solution of a linear system can be computed.
- Therefore we replace A^{-1} with C^{-1}, where C or C^{-1} should satisfy the following conditions:
  1. The calculation of $\mathbf{v}^{(k)} = C^{-1}\mathbf{d}^{(k)}$, i.e. solving $C\mathbf{v}^{(k)} = \mathbf{d}^{(k)}$, takes only small expense.
  2. $C \approx A$, such that $\mathbf{v}^{(k)} = C^{-1}\mathbf{d}^{(k)} \approx A^{-1}\mathbf{d}^{(k)} = \mathbf{v}_*^{(k)}$.

Then we can formulate the general linear iteration scheme

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \underbrace{C^{-1}\left(\mathbf{b} - A\mathbf{x}^{(k)}\right)}_{\mathbf{v}^{(k)}}.$$

Classical iteration schemes

First we decompose A into the form $A = L + D + U$, where L is a lower triangular matrix with main diagonal equals 0, D is a diagonal matrix and U is an upper triangular matrix with main diagonal equals zero. Decomposition of A:



$L_{ij} = 0, j \geqslant i$ $\quad$ $\operatorname{diag}(a_{ii})$ $\quad$ $U_{ij} = 0, j \leqslant i$

Jacobi method

Let $C := D = \operatorname{diag}(a_{ii})$. Then it follows

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \underbrace{D^{-1}\left(\mathbf{b} - A\mathbf{x}^{(k)}\right)}_{\mathbf{v}^{(k)}}.$$

With $D^{-1} = \operatorname{diag}\left(a_{ii}^{-1}\right)$ and

$$\left(\mathbf{b} - A\mathbf{x}^{(k)}\right)_i = b_i - \sum_{j=1}^{n} a_{ij} x_j^{(k)}$$

we obtain the representation in components

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{\substack{j=1 \\ i \neq j}}^{n} a_{ij} x_j^{(k)} \right), \qquad i = 1, \ldots, n.$$

The values a_{ii} should not be equal to zero.

The Gauß-Seidel method

- Let $C := L + D$. Then it follows

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + (L+D)^{-1}\left(\mathbf{b} - A\mathbf{x}^{(k)}\right).$$

- Multiply with $(L + D)$ from the left. Then

$$(L+D)\mathbf{x}^{(k+1)} = (L+D)\mathbf{x}^{(k)} + \mathbf{b} - A\mathbf{x}^{(k)}.$$

- With $A = L + D + U$ we obtain

$$(L+D)\mathbf{x}^{(k+1)} = (L+D)\mathbf{x}^{(k)} + \mathbf{b} - (L+D+U)\mathbf{x}^{(k)} = \mathbf{b} - U\mathbf{x}^{(k)}$$

The Gauß-Seidel method

Then

$$D\mathbf{x}^{(k+1)} = \mathbf{b} - L\mathbf{x}^{(k+1)} - U\mathbf{x}^{(k)}.$$

The componentwise representation is now given by

$$a_{ii}x_i^{(k+1)} = b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}, \qquad i = 1, \ldots, n.$$

and

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left[b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\right], \qquad i = 1, \ldots, n.$$

Example

- Consider the problem

$$\begin{cases} -u'' = f, & x \in (0,1) \\ u = 0, & x \in \{0,1\} \end{cases}$$

- Exact solution:

$$u(x) = \frac{x(1-x)}{(x-1/4)+1/10}$$

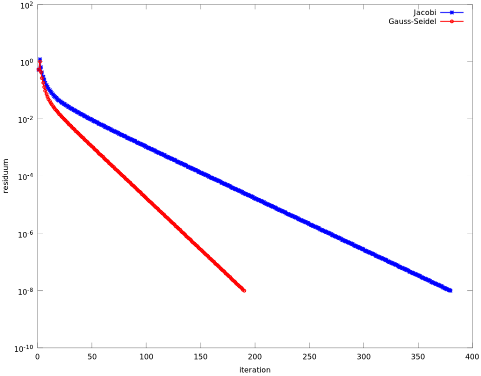
- Then:

$$f(x) = -160\frac{3200x^3+3100x^2-3120x+351}{(80x^2-40x+13)^2}$$

- Space discretisation: central differences

Example

Dimension: 10



The SOR-method (Successive Over-Relaxation)

The SOR-scheme tries to improve the convergence properties of the Gauss-Seidel method. The idea is to introduce a relaxation parameter $\omega > 0$ which is used in the following way $C := \frac{1}{\omega} D + L$. Inserting yields

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \left(\frac{1}{\omega} D + L \right)^{-1} \left(\mathbf{b} - A\mathbf{x}^{(k)} \right) .$$

We multiply from the left with $\frac{1}{\omega} D + L$, use $A = L + D + U$ and get

$$\begin{aligned} \left(\frac{1}{\omega} D + L \right) \mathbf{x}^{(k+1)} &= \left(\frac{1}{\omega} D + L \right) \mathbf{x}^{(k)} + \mathbf{b} - (L + D + U)\mathbf{x}^{(k)} \\ &= \left(\frac{1}{\omega} - 1 \right) D\mathbf{x}^{(k)} + \mathbf{b} - U\mathbf{x}^{(k)} \end{aligned}$$

The SOR-method (Successive Over-Relaxation)

Then

$$\frac{1}{\omega}D\mathbf{x}^{(k+1)} = \frac{1-\omega}{\omega}D\mathbf{x}^{(k)} - L\mathbf{x}^{(k+1)} + \mathbf{b} - U\mathbf{x}^{(k)}$$

and

$$D\mathbf{x}^{(k+1)} = (1-\omega)D\mathbf{x}^{(k)} + \omega\left(\mathbf{b} - L\mathbf{x}^{(k+1)} - U\mathbf{x}^{(k)}\right).$$

The representation with components is now given by

$$a_{ii}x_i^{(k+1)} = (1-\omega)a_{ii}x_i^{(k)} + \omega\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\right),$$

and

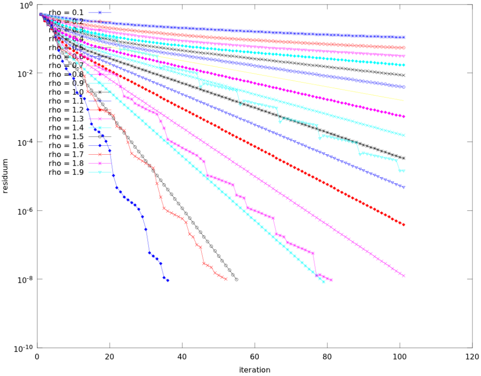
$$x_i^{(k+1)} = (1-\omega)x_i^{(k)} + \frac{\omega}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\right).$$

The SOR-method (Successive Over-Relaxation)

- But how to choose the relaxation parameter ω?
- $\omega = 1$ ... Gauß-Seidel method.
- With an optimal choice of ω we can improve the convergence of the SOR-method in comparison with the Gauß-Seidel method.
- But the optimal value for ω depends on the problem, i.e. in general one has to try several values. One should start with larger values, i.e. $\omega = 1.7$, then $\omega = 1.3$ ... and take a look on the convergence behaviour.

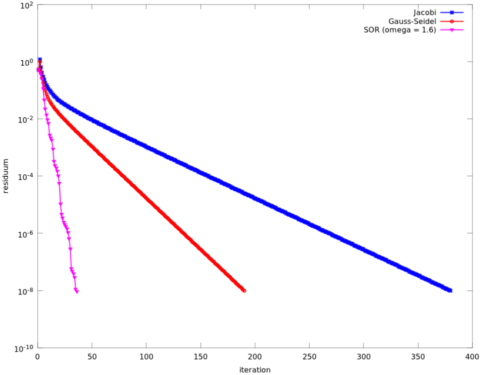
Example

Dimension: 10



Example

Dimension: 10



The symmetric Gauß-Seidel method

The symmetric Gauß-Seidel method is a combination of the forward Gauß-Seidel method

$$\mathbf{x}^{(k+1/2)} = (D+L)^{-1}(\mathbf{b} - U\mathbf{x}^{(k)})$$

and the backward Gauß-Seidel method

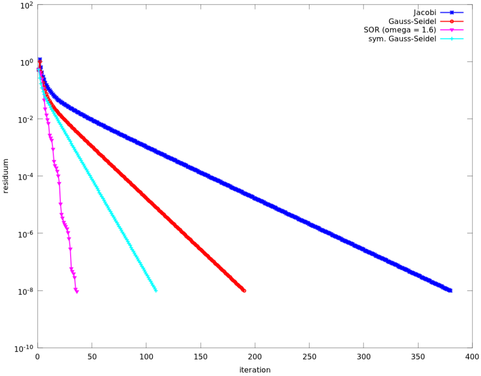
$$\mathbf{x}^{(k+1)} = (D+U)^{-1}(\mathbf{b} - L\mathbf{x}^{(k+1/2)}).$$

Then it follows

$$\mathbf{x}^{(k+1)} = (U+L)^{-1}D(D+L)^{-1}\mathbf{b} + (U+L)^{-1}L(D+L)^{-1}U\mathbf{x}^{(k)},$$

Example

Dimension: 10



Convergence results for linear iteration methods

Definition: Let $\mathbf{x} \in \mathbb{R}^n$ be the exact solution of the linear system $A\mathbf{x} = \mathbf{b}$. An iteration method which produces the sequence $\mathbf{x}^{(k)}$, $k = 0, 1 \ldots,$ of approximations, is called convergent, if

$$\lim_{k\to\infty} \left\|\mathbf{x}^{(k)} - \mathbf{x}\right\| = 0$$

holds, where $\|\cdot\|$ is some vector norm on $\mathbb{R}^n$. The condition is equivalent to the componentent-wise convergence, i.e.

$$\lim_{k\to\infty} x_i^{(k)} = x_i, \qquad \forall i = 1, \ldots, n.$$

Convergence results for linear iteration methods

- The equation

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + C^{-1}\left(\mathbf{b} - A\mathbf{x}^{(k)}\right).$$

for the general iteration method can be transformed into

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + C^{-1}\left(\mathbf{b} - A\mathbf{x}^{(k)}\right) = C^{-1}\mathbf{b} + (I - C^{-1}A)\mathbf{x}^{(k)}$$

and

$$\mathbf{x}^{(k+1)} = M\mathbf{x}^{(k)} + \mathbf{g}$$

with $M = I - C^{-1}A$ and $\mathbf{g} = C^{-1}\mathbf{b}$.

- This equation is often called normalform.
- Every linear iteration method can be written in normal form.
- The matrix M is called iteration matrix of the method.

Convergence results for linear iteration methods

Usually M and $\mathbf{g}$ are of the form such that

$$\mathbf{x} = M\mathbf{x} + \mathbf{g}$$

holds. The linear iteration method satisfies the above equation, since

$$\mathbf{x} = M\mathbf{x} + \mathbf{g} = (I - C^{-1}A)\mathbf{x} + C^{-1}\mathbf{b}) = \mathbf{x} + C^{-1}(A\mathbf{x} - \mathbf{b}) = \mathbf{x}.$$

The linear iteration method converges, if $\lim_{k\to\infty} \mathbf{x}^{(k)} = \mathbf{x}$.

Convergence results for linear iteration methods

For the derivation of convergence criteria we need the spectral radius of a matrix which can be defined in the following way

Definition For a given matrix $M \in \mathbb{R}^{n,n}$ let λ_i, $i = 1, \ldots, n$ be the eigenvalues of M. Then

$$\rho(M) := \max_{i=1,\ldots,n} |\lambda_i|$$

is called the spectral radius of M.

Convergence results for linear iteration methods

Theorem: (Convergence criteria) The linear iteration method with the iteration matrix M converges for an arbitrary startvector $\mathbf{x}^{(0)}$ if and only if $\rho(M) < 1$ and if in a dedicated matrixnorm $\|\cdot\|$, the condition $\|M\| \leqslant 1$ is satisfied.

In this case the following error estimates are true:

$$\left\|\mathbf{x}^{(k)} - \mathbf{x}\right\| \leqslant \frac{\|M\|^k}{1-\|M\|} \left\|\mathbf{x}^{(1)} - \mathbf{x}^{(0)}\right\|, \tag{1}$$

$$\left\|\mathbf{x}^{(k)} - \mathbf{x}\right\| \leqslant \frac{\|M\|}{1-\|M\|} \left\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\right\|. \tag{2}$$

Inequality (1) is called a priori estimate and (2) is called a posteriori estimate.

Banach's Fixedpoint Theorem

Let J be a closed set, such that $\varphi(J) \subset J$, i.e. from $\mathbf{x} \in J$ it follows $\varphi(\mathbf{x}) \in J$. Moreover let φ be contractive, i.e. the inequality

$$\|\varphi(\mathbf{x}) - \varphi(\mathbf{y})\| \leqslant \|\mathbf{x} - \mathbf{y}\| \quad \forall \mathbf{x}, \mathbf{y} \in J$$

is satisfied. Then the following is true:

- There exists a unique fixedpoint $\mathbf{x}^* = \varphi(\mathbf{x}^*)$ in J.
- For all starting values $\mathbf{x}_0 \in J$ the fixedpoint iteration $\mathbf{x}_{n+1} = \varphi(\mathbf{x}_n)$ converges to the fixedpoint $\mathbf{x}^*$.
- The following estimates are valid

$$\begin{aligned} \|\mathbf{x}_n - \mathbf{x}^*\| &\leqslant \frac{\rho^n}{1-\rho} \|\mathbf{x}_1 - \mathbf{x}_0\|, \\ \|\mathbf{x}_n - \mathbf{x}^*\| &\leqslant \frac{\rho}{1-\rho} \|\mathbf{x}_n - \mathbf{x}_{n-1}\|. \end{aligned}$$

Proof

We consider the fixed point iteration

$$\mathbf{x}_{n+1} = \varphi(\mathbf{x}_n) := M\mathbf{x}_n + \mathbf{g}.$$

Since $\|M\| < 1$ and $\rho(M) < 1$ the fixed point iteration converges.

Contents

- **Linear iteration schemes**

- **A motivating example**

A motivating example

- Consider the ODE

$$\dot{u} = \lambda u, \qquad u(0) = 1, \quad \lambda \ll 0$$

Dahlquist's problem

- Exact solution: $u(t) = \exp(\lambda t)$
- time discretisation: fully implicit Runge–Kutta method with s internal stages
- Aim: Compute one timestep

Runge–Kutta methods

- A Runge-Kutta method with s internal stages is given by

$$\mathbf{k}_i = \mathbf{f}\left(t_m + c_i\tau_m, \mathbf{u}_m + \tau_m \sum_{j=1}^{s} a_{ij}\mathbf{k}_j\right), \quad i = 1, \dots, s,$$

$$\mathbf{u}_{m+1} = \mathbf{u}_m + \tau_m \sum_{i=1}^{s} b_i\mathbf{k}_i.$$

- s ... number of stages
- $\mathbf{c}$ ... includes the grid points of the time discretisation
- $\mathbf{b}$ ... vector with weights
- a_{ij}, b_i and c_i should be chosen in such a way that some order conditions are satisfied to obtain a sufficient consistency order.

Butcher table

Butcher table (Butcher 1964):

$$
\begin{array}{c|ccc}
c_1 & a_{11} & \dots & a_{1s} \\
c_2 & a_{21} & \dots & a_{2s} \\
\vdots & \vdots & & \vdots \\
c_s & a_{s1} & \dots & a_{ss} \\
\hline
 & b_1 & \dots & b_s
\end{array}
=
\begin{array}{c|c}
\mathbf{c} & A \\
\hline
 & \mathbf{b}^\top
\end{array}.
$$

In dependency of *A* Runge-Kutta methods can be classified:

- Explicit Runge-Kutta methods (ERK)
- Diagonal implicit methods (DIRK)
- Fully implicit methods (FIRK)
- ...

Example

- Explicit Euler method (order 1)

$$\begin{array}{c|c} 0 & 0 \\ \hline & 1 \end{array}$$

- implicit Euler method (order 1)

$$\begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array}$$

- Gauß-Legendre method (s=1, order 2)

$$\begin{array}{c|c} 1/2 & 1/2 \\ \hline & 1 \end{array}$$

Example

- Radau-IIA (s=2, order 3)

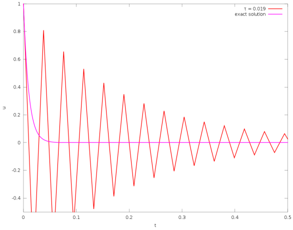
$$\begin{array}{c|cc} 1/3 & 5/12 & -1/12 \\ 1 & 3/4 & 1/4 \\ \hline & 3/4 & 1/4 \end{array}$$

- classical Runge-Kutta method (order 4)

$$\begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 & 0 \\ 1/2 & 0 & 1/2 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ \hline & 1/6 & 1/3 & 1/3 & 1/6 \end{array}$$

Numerical error

- Consider $\dot{u} = \lambda u$ with $u(0) = 1$ and $\lambda = -100$
- use explicit Euler method with $\tau = 0.019$



- Question: Which numerical error should be used?
- maximal error
- error at final time
- discrete L_2-error

$$\epsilon := \sqrt{\frac{1}{N}\sum_{k=1}^{N} |u(t_k) - u_k|^2}$$

Analysis

- Consider $\dot{u} = \lambda u$ with $u(0) = 1$
- use θ-methods, i.e.

$$u_{m+1} = u_m + \tau\lambda\theta u_{m+1} + \tau\lambda(1-\theta)u_m$$

- Stability function:

$$R(z) = \frac{1 + z(1-\theta)}{1 - z\theta}, \qquad z := \tau\lambda$$

- Examples:

$$\begin{aligned}
\theta = 0: \quad & R(z) &= 1 + z, & \qquad \text{explicit Euler}\\
\theta = 1/2: \quad & R(z) &= \frac{1+z/2}{1-z/2}, & \qquad \text{trapezoidal rule}\\
\theta = 1: \quad & R(z) &= \frac{1}{1-z}, & \qquad \text{implicit Euler}
\end{aligned}$$

Stability

- Stability function:

$$R(z) = \frac{1 + z(1-\theta)}{1 - z\theta}, \qquad z := \tau\lambda$$

- Stable, if

$$-1 \leqslant R(z) \leqslant +1$$

- Upper bound: always satisfied
- Lower bound: $\Longrightarrow$

$$0 < 2 + z(1 - 2\theta)$$

- $\theta \in [1/2, 1]$: satisfied since $z < 0$
- $\theta \in [0, 1/2)$: $\Longrightarrow$

$$\tau < \frac{2}{(1-2\theta)|\lambda|}$$

Stability

- A Runge-Kutta method is called A-stable, if the stability function $R_0(z)$ satisfies

$$|R_0(z)| \leqslant 1, \quad \forall z \in \mathbb{C}^- := \{z \in \mathbb{C} \mid \text{Re } z \leqslant 0\}.$$

- A Runge-Kutta method is called strongly A-stable, if the RK-method is A-stable and the stability function $R_0(z)$ satisfies

$$\lim_{z \to -\infty} |R_0(z)| < 1.$$

- A Runge-Kutta method is called L-stable (left-stability, see Ehle (1973), if the RK-method is stongly A-stable and the stability function $R_0(z)$ satisfies

$$\lim_{z \to -\infty} |R_0(z)| = 0.$$

Dahlquist's equation

- Consider the ODE (Dahlquist's problem):

$$\dot{u} = \lambda u, \qquad u(0) = 1, \quad \lambda \ll 0$$

- Linear system:

$$k_i = \lambda \left(\mathbf{u}_m + \tau_m \sum_{j=1}^{s} a_{ij} k_j \right), \quad i = 1, \ldots, s$$

or

$$\left(k_i - \tau_m \lambda \sum_{j=1}^{s} a_{ij} k_j \right) = \lambda \mathbf{u}_m.$$

Two situations

- Solve the problem with a Radau-IIA method of order 19, i.e. 10 internal stages!!!
- first situation: We have access to Matlab/Octave. Then we are solving the system direct with the backslash operator and that is all. This idea is called in the context of coupled problems the monolythical approach.
- second situation: We do not access to any linear solver, we have only a simple programming language with for-loops. We do not want to implement a complicated linear solver. Therefore we use a partitioned approach

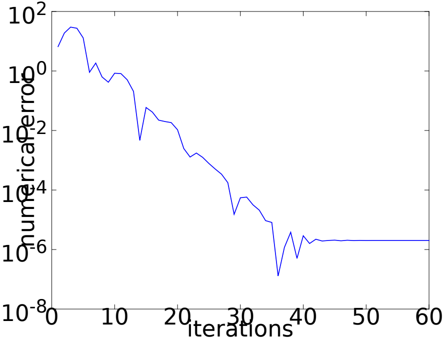
Numerical results

- Numerical errors:

	$\lambda = -100, \tau = 0.1$	$\lambda = -1.0E + 6, \tau = 0.01$
monolythical	4.2287e-08	9.8030e-04
Jacobi (10 it.)	0.056014	2.0262e+05
Jacobi (20 it.)	6.8069e-05	9.6596e+08
Jacobi (40 it.)	4.2314e-08	-

- For the partitioned method we have to use a smaller stepsize τ, for example $\tau = 1/(7.0E + 4)$
- Result: A partitioned approach leads to a stepsize restriction.

Convergence for $\tau = 1/(7.0E + 4)$



Convergence result for the motivating example

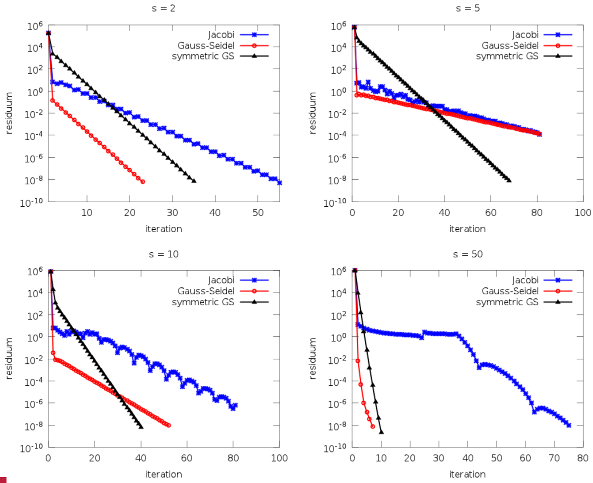
- Dahlquist's problem: $\dot{u} = \lambda u$
- Iteration matrix:

$$M = (m_{ij})_{i,j=1}^{s}, \qquad m_{ij} = \begin{cases} 0, & i = j \\ za_{ij}/(1 - za_{ii}), & i \neq j \end{cases}$$

- spectral radius: $\rho(M) \approx 2.34$
- maximal stepsize: $\tau \approx 2.0E - 5$

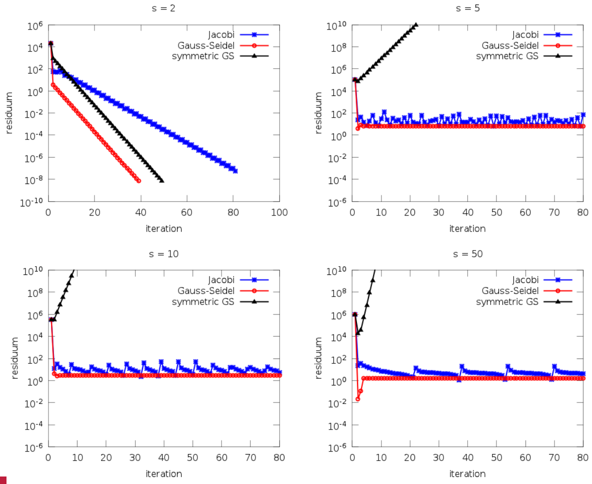
Dahlquist's problem: Convergence for fixed τ and s

$\tau = 2 \cdot 10^{-5}$

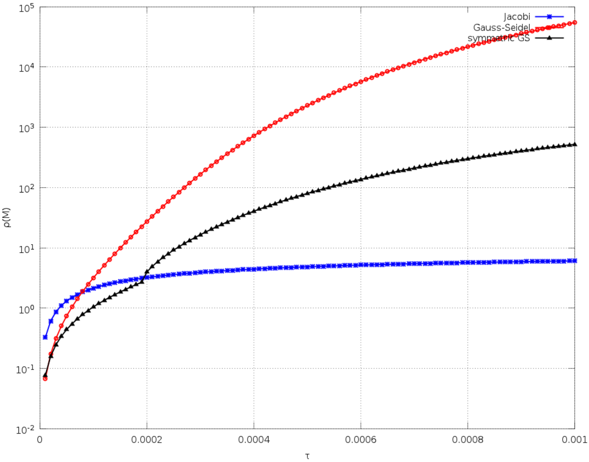


Dahlquist's problem: Convergence for fixed τ and s

$\tau = 2 \cdot 10^{-4}$

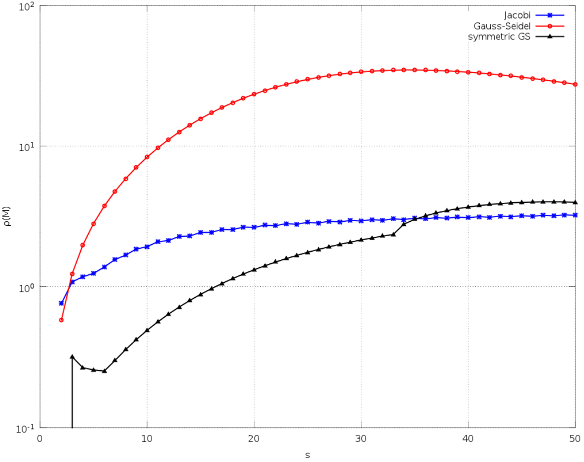


Dahlquist’s problem: Spectral radius for fixed s



Dahlquist's problem: Spectral radius for fixed τ

$\tau = 2 \cdot 10^{-4}$



Theorem

Let A be a diagonally dominant matrix, i.e. the condition

$$a_{ii} > \sum_{\substack{j=1 \\ i \neq j}} |a_{ij}|.$$

Then the Jacobi method converges.
Proof: This theorem can be proven by an easy calculation, see for example the book of Axelsson.

Theorem

Let A be a symmetric positive definite matrix. The the SOR-scheme converges for all ω with $0 < \omega < 2$.

Remarks

There are cases known for that only the Jacobi- but not the Gauss-Seidel method converges and vice versa.

- Let
$$A = \begin{pmatrix} 2 & 1 & 1 \\ -2 & 2 & -2 \\ -1 & 1 & 2 \end{pmatrix}$$
Then $\rho(M_J) = \frac{\sqrt{5}}{2}$ and $\rho(M_{GS}) = \frac{1}{2}$. It converges only the Gauss-Seidel method.
- Let
$$A = \begin{pmatrix} 1 & -2 & 2 \\ -1 & 1 & -1 \\ -2 & -2 & 1 \end{pmatrix}$$
Then $\rho(M_J) \approx 10^{-5}$ and $\rho(M_{GS}) = 2 + \sqrt{8}$. It converges only the Jacobi method.