CISM COURSE COMPUTATIONAL ACOUSTICS

Solvers

Part 2: Iterative Solvers

Ulrich Langer and Martin Neumüller
Institute of Computational Mathematics
Johannes Kepler University Linz

Udine, May 23-27, 2016

Outline

Outline

1. Classical Iteration Methods

2. Gradient and Conjugate Gradient Methods for SPD Systems

Summary

General Idea and Questions

- Idea:
  - Given initial guess $\underline{u}^0 \in \mathbb{R}^n$
  - Generate (how ?) successively a sequence of vectors

$$\underline{u}^1, \underline{u}^2, \ldots, \underline{u}^k \longrightarrow \underline{u} \in \mathbb{R}^n : \mathbf{K}\underline{u} = \underline{f} \quad \text{for } k \to \infty\,!$$

- Questions
  1. Construction principles
  2. Convergence analysis
  3. Convergence rate and iteration error estimates
     q-linear: $\exists q \in [0,1)$: $\|\underline{u} - \underline{u}^k\| \le q\|\underline{u} - \underline{u}^{k-1}\| \le q^k\|\underline{u} - \underline{u}^0\|$
     r-linear: $\exists q \in [0,1)$ and $c = const > 0$: $\|\underline{u} - \underline{u}^k\| \le c\, q^k$
  4. In practice: Convergence tests, e.g., defect tests

$$\|\underline{d}^k\| = \|\underline{d}^k\|_{R^n} = \|\underline{e}^k\|_{\mathbf{K}^T\mathbf{K}} = (\mathbf{K}^T\mathbf{K}\underline{e}^k, \underline{e}^k)_{R^n}^{0.5} \le \varepsilon\|\underline{d}^0\|$$

     with the defect $\underline{d}^k = \underline{f} - \mathbf{K}\underline{u}^k = \mathbf{K}(\underline{u} - \underline{u}^k) = \mathbf{K}\underline{e}^k$
  5. Choice of the norm $\|\underline{e}^k\|$ in which we control the iteration ?

Jacobi = Mother of Additive Schwarz

Idea: Solve the i-th eqn $K_{i1}u_1 + \ldots + K_{ii}u_i + \ldots + K_{in}u_n = f_i$ for u_i yielding the fixed point eqn: $u_i = K_{ii}^{-1}(f_i - \sum_{j\neq i} K_{ij}u_j)$

Algorithm (Jacobi iteration method)

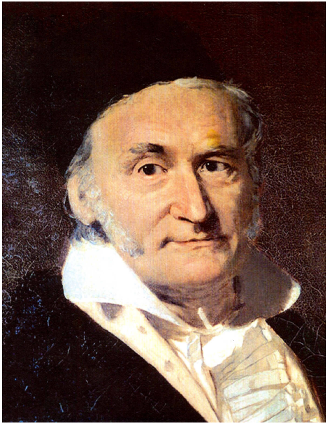
Given initia guess $\underline{u}^0 = (u_1^0, \ldots, u_n^0)^T \in \mathbb{R}^n$,
iterate $k = 0, 1, \ldots, k_{stop}$ *until convergence (defect test):*
$\underline{u}^{k+1} = (u_1^{k+1}, \ldots, u_n^{k+1})^T \in \mathbb{R}^n$:

$$u_i^{k+1} = \frac{1}{K_{ii}}\left(f_i - \sum_{j=1, j\neq i}^{n} K_{ij}u_j^k\right)$$

for $i = 1, 2, \ldots, n$ *(in parallel).*

Slow convergence (see our analysis below), but the damped version has an excellent smoothing property (see LN4) !

Gauss and his new iterative method



„Fast jeden Abend mache ich eine neue Auflage des Tableau, wo immer leicht nachzuhelfen ist. Bei der Einförmigkeit des Messungsgeschäfts gibt dies immer eine angenehme Unterhaltung; man sieht daran auch immer gleich, ob etwas Zweifelhaftes eingeschlichen ist, was noch wünschenswert bleibt usw. Ich empfehle Ihnen diesen Modus zur Nachahmung. Schwerlich werden Sie je wieder direct eliminiren, wenigstens nicht, wenn Sie mehr als zwei Unbekannte haben. Das indirecte Verfahren läßt sich halb im Schlafe ausführen oder man kann während desselben an andere Dinge denken."

C. F. Gauss in [2]

Gauss-Seidel = Mother of Multipl. Schwarz

Idea: Use the already computed new components

$u_1^{k+1}, \dots, u_{i-1}^{k+1}$ in the iteration:

Algorithm (Gauss-Seidel iteration method)

Given initial guess $\underline{u}^0 = (u_1^0, \dots, u_n^0)^T \in \mathbb{R}^n$*,*
iterate $k = 0, 1, \dots, k_{stop}$ *until convergence (defect test):*
$\underline{u}^{k+1} = (u_1^{k+1}, \dots, u_n^{k+1})^T \in \mathbb{R}^n$*:*

$$u_i^{k+1} = \frac{1}{K_{ii}} \left(f_i - \sum_{j=1}^{i-1} K_{ij} u_j^{k+1} - \sum_{j=i+1}^{n} K_{ij} u_j^k \right)$$

for $i = 1, 2, \dots, n$ *(sequentially), with* $\sum_{j=1}^{0} = \sum_{j=n+1}^{n} = 0$*.*

Slow convergence, but an excellent smoothing property (LN4) !

Richardson and Preconditioned Richardson

- **Motivation:** Solving ODE system

$$\frac{\partial \underline{u}(t)}{\partial t} + \mathbf{K}\underline{u}(t) = \underline{f}$$

by explicite Euler gives Richardson method:

$$\frac{\underline{u}^{k+1} - \underline{u}^{k}}{\tau} + \mathbf{K}\underline{u}^{k} = \underline{f}, \quad k = 1, 2, \ldots \tag{1}$$

- Application of Richardson (1) to the preconditioned system

$$\mathbf{C}^{-1}\mathbf{K}\underline{u} = \mathbf{C}^{-1}\underline{f} \quad \Longleftrightarrow \quad \mathbf{K}\underline{u} = \underline{f}$$

gives the **preconditioned Richardson method**:

$$\mathbf{C}\frac{\underline{u}^{k+1} - \underline{u}^{k}}{\tau} + \mathbf{K}\underline{u}^{k} = \underline{f}, \quad k = 1, 2, \ldots \tag{2}$$

where the preconditioner $\mathbf{C}$ should reduce the stiffness and should be easily invertable (see LN3) !

Preconditioned Richardson: Algorithm

Algorithm (Preconditioned Richardson method)

Given initia guess $\underline{u}^0 = (u_1^0, \ldots, u_n^0)^T \in \mathbb{R}^n$*,*
iterate $k = 0, 1, \ldots, k_{stop}$ *until convergence (defect test):*

$$\begin{aligned} \underline{d}^k &= \underline{f} - \mathbf{K}\underline{u}^k \\ \mathbf{C}\underline{w}^k &= \underline{d}^k \\ \underline{u}^{k+1} &= \underline{u}^k + \tau \underline{w}^k \end{aligned}$$

Convergence rate heavily depends on the quality of the preconditioner, see our analysis below !

Preconditioned Richardson: Preconditioners

Special choices of the preconditioner $\mathbf{C}$ in the **preconditioned Richardson method**:

$$\mathbf{C}\frac{\underline{u}^{k+1}-\underline{u}^{k}}{\tau}+\mathbf{K}\underline{u}^{k}=\underline{f}, \quad k=1,2,\ldots$$

yields well-known classical iteration methods:

1. $\mathbf{C}=\mathbf{I}$: Classical Richardson method
2. $\mathbf{C}=\mathbf{D}:=\text{diag}\,\mathbf{K}$: τ-Jacobi method ($\tau=1$: Jacobi method)
3. $\mathbf{C}=\mathbf{L}+(1/\omega)\mathbf{D}$: SOR preconditioner ($\mathbf{K}=\mathbf{L}+\mathbf{D}+\mathbf{U}$):
   $\tau=1$: SOR = Successive OverRelaxation (D. Young, 1950)
   $\tau=1$ and $\omega=1$: Gauss-Seidel
4. $\mathbf{C}=\widetilde{\mathbf{L}}\widetilde{\mathbf{U}}$: ILU decomposition of $\mathbf{K}$, see LN1
5. Modern preconditioners: see LN3

Convergence Analysis

From the preconditioned Richardson iteration (1) and $\mathbf{K}\underline{u} = \underline{f}$, we can immediately derive the error iteration scheme:

$$\underline{e}^{k+1} = \underline{u} - \underline{u}^{k+1} = \underline{u} - (\underline{u}^k - \tau \mathbf{C}^{-1}\mathbf{K}(\underline{u} - \underline{u}^k)) = \mathbf{E}\,\underline{e}^k \qquad (3)$$

with the error propagation (iteration) matrix $\mathbf{E} = \mathbf{I} - \tau \mathbf{C}^{-1}\mathbf{K}$.

The error iteration scheme (3) has the following consequences wrt convergence:

1. Richardson (1) converges **iff** the spectral radius $\rho(\mathbf{E}) := \max_{i=1,\dots,n} |\lambda_i(\mathbf{E})|$ of $\mathbf{E}$ is less than 1.
2. Error estimate wrt some norm and q-linear convergence:

$$\|\underline{e}^{k+1}\| \le \|\mathbf{I} - \tau \mathbf{C}^{-1}\mathbf{K}\|\,\|\underline{e}^k\| = q\|\underline{e}^k\| \le q^{k+1}\|\underline{e}^0\| \to 0 \quad (4)$$

   provided that $q = \|\mathbf{E}\| < 1$ in some norm $\|\cdot\|$ (?)

Convergence Analysis: SPD case (I)

- Let **K** and **C** be SPD. Then

$$\mathbf{K}\underline{u} = \underline{f} \quad \Longleftrightarrow \quad \tilde{\mathbf{K}}\underline{\tilde{u}} = \underline{\tilde{f}} \tag{5}$$

with $\underline{\tilde{f}} = \mathbf{C}^{-1/2}\underline{f}$, $\underline{\tilde{u}} = \mathbf{C}^{1/2}\underline{u}$, and the preconditioned stiffness matrix $\tilde{\mathbf{K}} = \mathbf{C}^{-1/2}\mathbf{K}\mathbf{C}^{-1/2}$ that is obviously SPD !

- Thus it is sufficient to derive iteration error estimates for the classical Richardson method !
- Let us consider expansion of the k-th error $\underline{e}^k$ into a Fourier series wrt the eigenvectors of $\mathbf{K}$ (resp. $\tilde{\mathbf{K}}$):

$$\underline{e}^k = \sum_{j=1}^{n} \alpha_j \underline{\varphi}_j \tag{6}$$

with the Fourier coefficients $\alpha_j = (\underline{e}^k, \underline{\varphi}_j)_{\mathbb{R}^n}$, $j = 1, 2, \ldots, n$.

Convergence Analysis: SPD case (II)

- Inserting the Fourier expansion (6) into the error scheme (3) with $\mathbf{C} = \mathbf{I}$, we get

$$\underline{e}^{k+1} = \mathbf{E}\underline{e}^{k} = (\mathbf{I} - \tau\mathbf{K})\underline{e}^{k} = \sum_{j=1}^{n} \alpha_j (1 - \tau\lambda_j)\underline{\varphi}_j \qquad (7)$$

- We choose the following class of norms

$$\|\underline{v}\|_s := (\mathbf{K}^s \underline{v}, \underline{v})_{\mathbb{R}^n}^{1/2}, \quad s \in \mathbb{R} \quad \text{(special interest: } s = 0, 1, 2\text{)}$$

in which we want to derive sharp iteration error estimates !

- Show that $\|\cdot\|_s$ is indeed a norm ?

Convergence Analysis: SPD case (III)

Richardson method

- Using (7), we get the sharp estimate

$$
\begin{aligned}
\|\underline{e}^{k+1}\|_s^2 &= (\mathbf{K}^s \underline{e}^{k+1}, \underline{e}^{k+1})_{\mathbb{R}^n} = (\mathbf{K}^s \underline{e}^{k+1}, \underline{e}^{k+1}) \\
&= (\mathbf{K}^s \sum_{j=1}^{n} \alpha_j (1-\tau\lambda_j)\underline{\varphi}_j, \sum_{i=1}^{n} \alpha_i (1-\tau\lambda_i)\underline{\varphi}_i) \\
&= (\sum_{j=1}^{n} \alpha_j (1-\tau\lambda_j)\lambda_j^s \underline{\varphi}_j, \sum_{i=1}^{n} \alpha_i (1-\tau\lambda_i)\underline{\varphi}_i) \\
&= \sum_{j=1}^{n} \alpha_j^2 \lambda_j^s (1-\tau\lambda_j)^2 \\
&\leq \max_{i=1,\ldots,n} (1-\tau\lambda_i)^2 \sum_{j=1}^{n} \alpha_j^2 \lambda_j^s = \max_{i=1,\ldots,n} (1-\tau\lambda_i)^2 \|\underline{e}^k\|_s^2 \\
&= (\max\{|1-\tau\lambda_1|, |1-\tau\lambda_n|\})^2 \|\underline{e}^k\|_s^2 = q(\tau)^2 \|\underline{e}^k\|_s^2
\end{aligned}
$$

Convergence Analysis: SPD case (IV)

Lemma (Convergence rate estimate)

The $\|\cdot\|_s$ *norm of the iteration matrix* $\mathbf{E} = \mathbf{I} - \tau\mathbf{K}$ *is given by*

$$\|\mathbf{E}\|_s := \max_{\underline{v}\in\mathbb{R}^n} \frac{\|\mathbf{E}\underline{v}\|_s}{\|\underline{v}\|_s} = q(\tau) := \max\{|1-\tau\lambda_1|, |1-\tau\lambda_n|\} < 1$$

for fixed $\tau \in (0, 2/\lambda_n)$ *and* $s \in \mathbb{R}$.

Remark: $\|\underline{e}^{k+1}\|_s \le q(\tau)\|\underline{e}^k\|_s \le \ldots \le (q(\tau))^{k+1}\|\underline{e}^0\|_s$

$s = 0 : \|\underline{u} - \underline{u}^{k+1}\|_{\mathbb{R}^n} \le (q(\tau))^{k+1}\|\underline{u} - \underline{u}^0\|_{\mathbb{R}^n}$ (not computable)

$s = 1 : \|\underline{u} - \underline{u}^{k+1}\|_{\mathbf{K}} \le (q(\tau))^{k+1}\|\underline{u} - \underline{u}^0\|_{\mathbf{K}}$ (not computable)

$s = 2 : \|\underline{u} - \underline{u}^{k+1}\|_{\mathbf{K^2}} = \|\underline{d}^{k+1}\|_{\mathbb{R}^n} \le (q(\tau))^{k+1}\|\underline{d}^0\|_{\mathbb{R}^n}$ (comp.)

Convergence Analysis: SPD case (V)

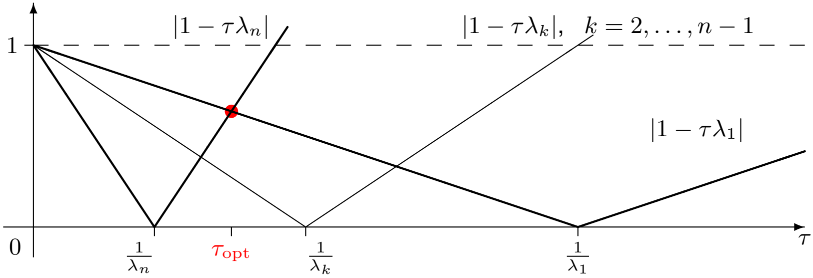


Figure: Functions $|1-\tau\lambda_k|$

- Eqn $1-\tau\lambda_1 = \tau\lambda_n - 1$ yields $\tau_{\text{opt}} = 2/(\lambda_1+\lambda_n)$
- $q_{\text{opt}} = q(\tau_{\text{opt}}) = (\lambda_n-\lambda_1)/(\lambda_n+\lambda_1) = (\kappa_2-1)/(\kappa_2+1)$ with the spectral condition number $\kappa_2 = \kappa_2(\mathbf{K}) = \lambda_n/\lambda_1$.

Convergence Analysis: SPD case (VI)

Theorem (Optimal convergence rates)

In the SPD case, the classical Richardson method (1) converges for all $\tau \in (0, 2/\lambda_{max}(bfK)) = (0, 2/\lambda_n)$*, and, for every fixed* $s \in (0,1)$*, the iteration error estimate*

$$\|\underline{u} - \underline{u}^{k+1}\|_s \leq q(\tau)\|\underline{u} - \underline{u}^k\|_s \tag{8}$$

holds with $q(\tau) := \max\{|1 - \tau\lambda_1|, |1 - \tau\lambda_n|\} < 1$*. The optimal (minimal) rate*

$$q_{\text{opt}} = q(\tau_{\text{opt}}) = \frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1} = \frac{\kappa_2(\mathbf{K}) - 1}{\kappa_2(\mathbf{K}) + 1} \tag{9}$$

is attained at $\tau_{\text{opt}} = 2/(\lambda_1 + \lambda_n)$*, where* λ_1 *and* λ_2 *are the minimal and maximal eigenvalues of the matrix* $\mathbf{K}$*.*

Convergence Analysis: SPD case (VII)

Since the preconditioned Richardson (2) can be interpreted as the application of the classical Richardson method (1) to

$$\tilde{\mathbf{K}}\underline{\tilde{u}} = \underline{\tilde{f}} \quad \Longleftrightarrow \quad \mathbf{K}\underline{u} = \underline{f}$$

with $\tilde{\mathbf{K}} = \mathbf{C}^{-1/2}\mathbf{K}\mathbf{C}^{-1/2}$ $\underline{\tilde{f}} = \mathbf{C}^{-1/2}\underline{f}$, $\underline{\tilde{u}} = \mathbf{C}^{1/2}\underline{u}$, we get the convergence results as presented in the Theorem, but now with

$$\lambda_1 = \lambda_{min}(\mathbf{C}^{-1/2}\mathbf{K}\mathbf{C}^{-1/2}) = \lambda_{min}(\mathbf{C}^{-1}\mathbf{K}),$$

$$\lambda_n = \lambda_{max}(\mathbf{C}^{-1/2}\mathbf{K}\mathbf{C}^{-1/2}) = \lambda_{max}(\mathbf{C}^{-1}\mathbf{K}).$$

Remark: $s = 2$ gives computable norm. Indeed,

$$\|\underline{\tilde{u}} - \underline{\tilde{u}}^j\|_2^2 = (\tilde{\mathbf{K}}^2\underline{\tilde{e}}^j, \underline{\tilde{e}}^j) = (\mathbf{K}\mathbf{C}^{-1}\mathbf{K}\underline{e}^j, \underline{e}^j) = (\underline{w}^j, \underline{d}^j)$$

Jacobi = special preconditiond Richardson

- τ-Jacobi = preRichardson with $\mathbf{C} = \mathbf{D} :=$ diag $\mathbf{K}$:

  $\Rightarrow \kappa_2(\mathbf{C}^{-1}\mathbf{K}) = O(h^{-2})$

  $\Rightarrow q_{\text{opt}} = q(\tau_{\text{opt}}) = \frac{\kappa_2(\mathbf{C}^{-1}\mathbf{K})-1}{\kappa_2(\mathbf{C}^{-1}\mathbf{K})+1} = \frac{1-O(h^2)}{1+O(h^2)} = 1 - O(h^2)$

- Example from LN1: $\mathbf{K} = h^{-1}$ tridiag $(-1, 2, -1)$

  $\Rightarrow \mathbf{C} =$ diag $\mathbf{K} =$ diag $(2h^{-1})$

  $\Rightarrow \lambda_{min}(\mathbf{C}^{-1}\mathbf{K}) = (h/2)\, 4h^{-1} \sin^2 \frac{\pi h}{2} = 2 \sin^2 \frac{\pi h}{2}$

  $\Rightarrow \lambda_{max}(\mathbf{C}^{-1}\mathbf{K}) = (h/2)\, 4h^{-1} \cos^2 \frac{\pi h}{2} = 2 \cos^2 \frac{\pi h}{2}$

  $\Rightarrow \tau_{\text{opt}} = \frac{2}{\lambda_{min}+\lambda_{max}} = \frac{2}{2(\sin^2(\cdot)+\cos^2(\cdot))} = 1$

  $\Rightarrow$ Classical Jacobi is optimal !

  $\Rightarrow q_{\text{opt}} = \frac{\kappa_2(\mathbf{C}^{-1}\mathbf{K})-1}{\kappa_2(\mathbf{C}^{-1}\mathbf{K})+1} = \frac{1-\tan^2 \frac{\pi h}{2}}{1+\tan^2 \frac{\pi h}{2}} = 1 - 2\sin^2 \frac{\pi h}{2} \approx 1 - \frac{\pi^2 h^2}{2}$

  $\Rightarrow$ Slow convergence: $I(\varepsilon) = O(h^{-2} \ln \varepsilon^{-1})$

  $\Rightarrow$ **BUT**: Fast reduction of the high frequency modes for damped Jacobi, e.g., $\tau = 1/\lambda_{max}(\mathbf{C}^{-1}\mathbf{K}) = \frac{1}{2\cos^2 \frac{\pi h}{2}} \approx \frac{1}{2}$, see Figure Functions $|1 - \tau\lambda_k|$ and LN4 (MGM) !

Krylov instead of Richardson

In practice, we use

- **preconditioned Krylov subspace iteration methods**

instead of preconditioned Richardson iteration methods since

1. they don't need spectral information to determine iteration parameters like τ in Richardson,
2. they converge faster !

In the SPD case,

- **Preconditioned Conjugate Gradient (PCG) method**,

proposed by Magnus Hestenes and Eduard Stiefel in 1952,
is the method of choice (#3 under the top 10 num. alg.)

Outline

SPD Systems and Minimization Problems

- Let us consider the linear system: Find $\underline{u} \in \mathbb{R}^n$:

$$\mathbf{K}\underline{u} = \underline{f}$$

  with given rhs $\underline{f} \in \mathbb{R}^n$ and SPD system matrix **K**, i.e. $\mathbf{K} = \mathbf{K}^T$ and $(\mathbf{K}\underline{v}, \underline{v}) > 0 \; \forall \underline{v} \in \mathbb{R}^n : \underline{v} \neq \underline{0}$.

- Then the SPD system $\mathbf{K}\underline{u} = \underline{f}$ is equivalent to the energy minimization problem

$$J(\underline{u}) = \min_{\underline{v}\in\mathbb{R}^n} \frac{1}{2}(\mathbf{K}\underline{v}, \underline{v}) - (\underline{f}, \underline{v}) = \min_{\underline{v}\in\mathbb{R}^n} \frac{1}{2} \sum_{i,j=1}^{n} K_{ij} v_j v_i - \sum_{i=1}^{n} f_i v_i$$

Proof: $\nabla J(\underline{v}) = \left(\frac{\partial J(\underline{v})}{\partial v_i}\right)_{i=1,\ldots,n} = \mathbf{K}\underline{v} - \underline{f} = \underline{0}\,!$,

$$\nabla^2 J(\underline{v}) = \left(\frac{\partial^2 J(\underline{v})}{\partial v_i \partial v_j}\right)_{i,j=1,\ldots,n} = \mathbf{K} \quad \text{SPD !}$$

Gradient (Steepest Descent) Method

- Idea for a Steepest Descent Method (Gradient Method):
  - Given initial guess $\underline{u}^0 = (u_1^0, \ldots, u_n^0)^T \in \mathbb{R}^n$,
  - Compute steepest descent $\underline{d}^0$ at $\underline{u}^0$:
    $\underline{d}^0 = -\nabla J(\underline{u}^0) = \underline{f} - \mathbf{K}\underline{u}^0$
  - $\underline{s}^0 = \underline{d}^0$ (search direction)
  - $\underline{u}^1 = \underline{u}^0 + \alpha_1 \underline{s}^0$ (next iterate)
  - Compute step size α_1: $J(\underline{u}^0 + \alpha_1 \underline{s}^0) = \min_\alpha J(\underline{u}^0 + \alpha \underline{s}^0)$
    $\frac{dJ(\underline{u}^0 + \alpha \underline{s}^0)}{d\alpha} = (\mathbf{K}\underline{u}^0, \underline{s}^0) - (\underline{f}, \underline{s}^0) + \alpha\,(\mathbf{K}\underline{s}^0, \underline{s}^0) = 0$ gives

$$\alpha_1 = \frac{(\underline{d}^0, \underline{s}^0)}{(\mathbf{K}\underline{s}^0, \underline{s}^0)}$$

- The new steepest descent $\underline{d}^1$ at $\underline{u}^1$ can be computed by recursion as follows

$$\begin{aligned} \underline{d}^1 &= \underline{f} - \mathbf{K}\underline{u}^1 = \underline{f} - \mathbf{K}(\underline{u}^0 + \alpha_1 \underline{s}^0) \\ &= \underline{f} - \mathbf{K}\underline{u}^0 - \alpha_1 \mathbf{K}\underline{s}^0 = \underline{d}^0 - \alpha_1 \mathbf{K}\underline{s}^0 \end{aligned}$$

Gradient Method: Algorithm

Algorithm (Gradient method = steepest descent method)

Initialization:

$\underline{u}^0 \in \mathbb{R}^n$ *- given initial guess,*

$\underline{d}^0 = \underline{f} - \mathbf{K}\underline{u}^0$ *- initial defect = steepest descent,*

$\underline{s}^0 = \underline{d}^0$ *- search direction,*

Iteration: $k = 0, 1, \ldots, k_{stop}$;

If $\|\underline{d}^k\| \leq \varepsilon \|\underline{d}^0\|$, *then STOP (defect test),*

$\alpha_k = (\underline{d}^k, \underline{s}^k)/(\mathbf{K}\underline{s}^k, \underline{s}^k)$ *- new step size,*

$\underline{u}^{k+1} = \underline{u}^k + \alpha_k \underline{s}^k$ *- new iterate,*

$\underline{d}^{k+1} = \underline{d}^k - \alpha_k \mathbf{K}\underline{s}^k$ *- new defect,*

$\underline{s}^{k+1} = \underline{d}^{k+1}$ *- new search direction.*

Gradient Method (GM): Convergence

- Since

$$J(\underline{v}) = 0.5(\mathbf{K}\underline{v},\underline{v}) - (\underline{f},\underline{v})$$
$$= 0.5(\mathbf{K}\underline{v},\underline{v}) - (\mathbf{K}\underline{u},\underline{v}) + 0.5(\mathbf{K}\underline{u},\underline{u}) - 0.5(\mathbf{K}\underline{u},\underline{u})$$
$$= 0.5\|\underline{u}-\underline{v}\|_{\mathbf{K}}^2 - 0.5\|\underline{u}\|_{\mathbf{K}}^2,$$

we conclude that

$$\min_{\underline{v}\in\mathbb{R}^n} J(\underline{v}) \quad \Leftrightarrow \quad \min_{\underline{v}\in\mathbb{R}^n} \|\underline{u}-\underline{v}\|_{\mathbf{K}}$$

- **Consequence:** Using $\underline{s}^k = \underline{d}^k$, we get

$$\|\underline{u}-\underline{u}^{k+1}\|_{\mathbf{K}} = \min_{\alpha\in\mathbb{R}} \|\underline{u} - (\underline{u}^k + \alpha_k \underline{s}^k\|_{\mathbf{K}}$$
$$\le \|\underline{u} - (\underline{u}^k + \tau_{\text{opt}}\underline{d}^k)\|_{\mathbf{K}} \le q(\tau_{\text{opt}})\|\underline{u}-\underline{u}^k\|_{\mathbf{K}},$$

i.e. GM converges at least as fast as Richardson !

Gradient Method (GM): Improvements

- Possible improvements:
  1. Preconditioning: Apply GM to the preconditioned system

$$\mathbf{C}^{-1}\mathbf{K}\underline{u} = \mathbf{C}^{-1}\underline{f} \quad \Longleftrightarrow \quad \mathbf{C}^{-0.5}\mathbf{K}\mathbf{C}^{-0.5}\underline{v} = \mathbf{C}^{-0.5}\underline{f}$$

     This means that the search direction in the Preconditioned Gradient Method is the preconditioned defect

$$\underline{s}^{k+1} = \underline{w}^{k+1} := \mathbf{C}^{-1}\underline{d}^{k+1}$$

  2. Use conjugate search directions defined by

$$\underline{s}^{k+1} = \underline{d}^{k+1} + \beta_k \underline{s}^k \;\perp\; \underline{s}^k \text{ wrt } (\cdot,\cdot)_{\mathbf{K}} := (\mathbf{K}\cdot,\cdot)$$

     i.e., $\beta_k \in \mathbb{R} : (\mathbf{K}\underline{s}^{k+1}, \underline{s}^k) = 0 \Rightarrow \beta_k = -\frac{(\mathbf{K}\underline{d}^{k+1}, \underline{s}^k)}{(\mathbf{K}\underline{s}^k, \underline{s}^k)}$

- Both improvements lead to the
  Preconditioned Conjugate Gradient (PCG) Method

PCG: Algorithm

Algorithm (Preconditioned conjugate gradient method)

Initialization:

$\underline{u}^0 \in \mathbb{R}^n$ - *given initial guess,*

$\underline{d}^0 = \underline{f} - \mathbf{K}\underline{u}^0$ - *initial defect = steepest descent,*

$\underline{s}^0 = \underline{w}^0 := \mathbf{C}^{-1}\underline{d}^0$ - *search direction = preconditioned defect,*

Iteration: $k = 0, 1, \ldots, k_{stop}$;

If $\|\underline{e}^k\|_{\mathbf{K}\mathbf{C}^{-1}\mathbf{K}} \leq \varepsilon \|\underline{e}^0\|_{\mathbf{K}\mathbf{C}^{-1}\mathbf{K}}$*, then STOP (*$\mathbf{K}\mathbf{C}^{-1}\mathbf{K}$ *norm test),*

$\alpha_k = (\underline{d}^k, \underline{s}^k)/(\mathbf{K}\underline{s}^k, \underline{s}^k) = (\underline{d}^k, \underline{w}^k)/(\mathbf{K}\underline{s}^k, \underline{s}^k)$ - *new step size,*

$\underline{u}^{k+1} = \underline{u}^k + \alpha_k \underline{s}^k$ - *new iterate,*

$\underline{d}^{k+1} = \underline{d}^k - \alpha_k \mathbf{K}\underline{s}^k$ - *new defect,*

$\underline{w}^{k+1} := \mathbf{C}^{-1}\underline{d}^{k+1}$ - *preconditioning,*

$\beta_k = -(\mathbf{K}\underline{w}^{k+1}, \underline{s}^k)/(\mathbf{K}\underline{s}^k, \underline{s}^k) = (\underline{w}^{k+1}, \underline{d}^{k+1})/(\underline{w}^k, \underline{d}^k)$,

$\underline{s}^{k+1} = \underline{w}^{k+1} + \beta_k \underline{s}^k$ - *new search direction.*

PCG: Algorithm

Algorithm (Preconditioned conjugate gradient method)

Initialization:

$\underline{u}^0 \in \mathbb{R}^n$ - *given initial guess,*

$\underline{d}^0 = \underline{f} - \mathbf{K}\underline{u}^0$ - *initial defect = steepest descent,*

$\underline{s}^0 = \underline{w}^0 := \mathbf{C}^{-1}\underline{d}^0$ - *search direction = preconditioned defect,*

Iteration: $k = 0, 1, \ldots, k_{stop}$;

If $(\underline{w}^k, \underline{d}^k) \leq \varepsilon(\underline{w}^k, \underline{d}^k)$, *then STOP (*$\mathbf{K}\mathbf{C}^{-1}\mathbf{K}$ *norm test),*

$\alpha_k = (\underline{d}^k, \underline{s}^k)/(\mathbf{K}\underline{s}^k, \underline{s}^k) = (\underline{d}^k, \underline{w}^k)/(\mathbf{K}\underline{s}^k, \underline{s}^k)$ - *new step size,*

$\underline{u}^{k+1} = \underline{u}^k + \alpha_k \underline{s}^k$ - *new iterate,*

$\underline{d}^{k+1} = \underline{d}^k - \alpha_k \mathbf{K}\underline{s}^k$ - *new defect,*

$\underline{w}^{k+1} := \mathbf{C}^{-1}\underline{d}^{k+1}$ - *preconditioning,*

$\beta_k = -(\mathbf{K}\underline{w}^{k+1}, \underline{s}^k)/(\mathbf{K}\underline{s}^k, \underline{s}^k) = (\underline{w}^{k+1}, \underline{d}^{k+1})/(\underline{w}^k, \underline{d}^k)$,

$\underline{s}^{k+1} = \underline{w}^{k+1} + \beta_k \underline{s}^k$ - *new search direction.*

PCG: Convergence

Theorem (PCG: convergence rate estimate)

Let $\mathbf{K}$ *and* $\mathbf{C}$ *be SPD matrices. Then not more than*

$$I(\varepsilon) = [|\ln(\varepsilon^{-1} + (\varepsilon^{-2} + 1)^{0.5}) / \ln(\tilde{q}^{-1})|]$$

iteration are necessary to reduce the inital error $\|\underline{u} - \underline{u}^0\|_{\mathbf{K}}$ *by the factor* $\varepsilon \in (0,1)$. *Moreover, the iteration error estimate*

$$\|\underline{u} - \underline{u}^{k+1}\|_{\mathbf{K}} \leq \eta^{(k+1)} \|\underline{u} - \underline{u}^0\|_{\mathbf{K}}$$

holds, where

$$\eta^{(k)} = \frac{2q^k}{1 + q^{2k}}, \quad \text{with } q = \frac{\sqrt{\kappa_2(\mathbf{C}^{-0.5}\mathbf{K}\mathbf{C}^{-0.5})} - 1}{\sqrt{\kappa_2(\mathbf{C}^{-0.5}\mathbf{K}\mathbf{C}^{-0.5})} + 1} < 1.$$

Outline

Summary

- Classical iteration methods: Jacobi & Gauss-Seidel
- Preconditioned Richardson method
- SPD systems and energy minimization
- Gradient or steepest descent method
- CG and PCG
- Non SPD systems: Krylov subspace like GMRES

References

[1] M. Jung and U. Langer.
Methode der finiten Elemente für Ingenieuer.
Springer Vieweg, Wiesbaden, 2013.

[2] G. Meurant.
Computer Solution of Large Linear Systems, volume 28 of *Studies in Mathematics and its Applications*.
North Holland, 2013.