The conjugate gradient method from different perspectives

Petr Tichý

Faculty of Mathematics and Physics
Charles University

December 7, 2016
Seminář ústavu matematiky VŠCHT

Problem formulation

Consider a system

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

where $\mathbf{A} \in \mathbb{R}^{n \times n}$ is **symmetric, positive definite**.

- $\mathbf{A}$ is large and sparse
- look for an approximation
- in each iteration perform $\mathbf{A}v$

Without loss of generality, $\|b\| = 1$, $x_0 = 0$.

The conjugate gradient method

input $\mathbf{A}$, b

$r_0 = b$, $p_0 = r_0$

for $k = 1, 2, \ldots$ **do**

$$\begin{aligned}
\gamma_{k-1} &= \frac{r_{k-1}^T r_{k-1}}{p_{k-1}^T \mathbf{A}\, p_{k-1}} \\
x_k &= x_{k-1} + \gamma_{k-1} p_{k-1} \\
r_k &= r_{k-1} - \gamma_{k-1} \mathbf{A}\, p_{k-1} \\
\delta_k &= \frac{r_k^T r_k}{r_{k-1}^T r_{k-1}} \\
p_k &= r_k + \delta_k p_{k-1}
\end{aligned}$$

test quality of x_k

end for

Coefficients matrix $\mathbf{R}_k$

$$\begin{bmatrix}
\frac{1}{\sqrt{\gamma_0}} & \sqrt{\frac{\delta_1}{\gamma_0}} & & \\
& \ddots & \ddots & \\
& & \ddots & \sqrt{\frac{\delta_{k-1}}{\gamma_{k-2}}} \\
& & & \frac{1}{\sqrt{\gamma_{k-1}}}
\end{bmatrix}$$

Vectors $\in \mathcal{K}_k(\mathbf{A}, b)$

$\mathrm{span}\{b, \mathbf{A}b, \ldots, \mathbf{A}^{k-1}b\}$

Orthogonality

$r_i \perp r_j \qquad p_i \perp_{\mathbf{A}} p_j$

CG as the Lanczos method

The Lanczos algorithm

Let $\mathbf{A}$ be symmetric, compute orthonormal basis of $\mathcal{K}_k(\mathbf{A}, b)$

input $\mathbf{A}$, b
$v_1 = b/\|b\|$
$\beta_0 = 0$, $v_0 = 0$
for $k = 1, 2, \ldots$ **do**
$\alpha_k = v_k^T \mathbf{A} v_k$
$w = \mathbf{A} v_k - \alpha_k v_k - \beta_{k-1} v_{k-1}$
$\beta_k = \|w\|$
$v_{k+1} = w/\beta_k$
end for

$$\begin{array}{c} \mathbf{T}_k \\ \begin{bmatrix} \alpha_1 & \beta_1 & & \\ \beta_1 & \ddots & & \\ & & \ddots & \beta_{k-1} \\ & & \beta_{k-1} & \alpha_k \end{bmatrix} \end{array}$$

$$\mathbf{A} v_k = \beta_k v_{k+1} + \alpha_k v_k + \beta_{k-1} v_{k-1} \,.$$

The Lanczos algorithm can be represented by

$$\mathbf{A}\mathbf{V}_k = \mathbf{V}_k \mathbf{T}_k + \beta_k v_{k+1} e_k^T \,, \qquad \mathbf{V}_k^* \mathbf{V}_k = \mathbf{I} \,.$$

CG versus Lanczos

Let $\mathbf{A}$ be symmetric, positive definite

Both compute an orthogonal basis of $\mathcal{K}_k(\mathbf{A}, b)$. It holds that

$$v_{k+1} = (-1)^k \frac{r_k}{\|r_k\|} \,.$$

It can be shown that

$$\mathbf{T}_k = \mathbf{R}_k^T \mathbf{R}_k$$

where

$$\overbrace{\begin{bmatrix} \alpha_1 & \beta_1 & & \\ \beta_1 & \ddots & \ddots & \\ & \ddots & \ddots & \beta_{k-1} \\ & & \beta_{k-1} & \alpha_k \end{bmatrix}}^{\mathbf{T}_k}, \quad \overbrace{\begin{bmatrix} \frac{1}{\sqrt{\gamma_0}} & \sqrt{\frac{\delta_1}{\gamma_0}} & & \\ & \ddots & \ddots & \\ & & \ddots & \sqrt{\frac{\delta_{k-1}}{\gamma_{k-2}}} \\ & & & \frac{1}{\sqrt{\gamma_{k-1}}} \end{bmatrix}}^{\mathbf{R}_k}.$$

Lanczos, CG and the eigenvalues approximations

The Lanczos algorithm can be represented by

$$\mathbf{A}\mathbf{V}_k \;=\; \mathbf{V}_k\mathbf{T}_k + \beta_k v_{k+1} e_k^T, \qquad \mathbf{V}_k^*\mathbf{V}_k = \mathbf{I}\,.$$

Let

$$\mathbf{T}_k\, y \;=\; \mu\, y, \qquad \|y\| = 1\,.$$

Then

$$\mathbf{A}\,\overbrace{\mathbf{V}_k y}^{z} = \mu\,\overbrace{\mathbf{V}_k y}^{z} + \beta_k\, v_{k+1} e_k^T\, y$$

and

$$\|\mathbf{A}\,z - \mu\,z\| = \beta_k\,|e_k^T\, y|.$$

Connection to CG $\;\longrightarrow\;$ $\mathbf{T}_k = \mathbf{R}_k^T\mathbf{R}_k$

Eigenvalues of $\mathbf{T}_k$ are squared singular values of $\mathbf{R}_k$.

CG as a projection method

Projection process

and approximation to the solution of $\mathbf{A}x = b$

Given $x_0 = 0$ (for simplicity), look for x_k,

$$x_k \in \mathcal{S}_k \quad \text{s.t.} \quad r_k \perp \mathcal{C}_k$$

- $r_k = b - \mathbf{A}x_k$
- $\mathcal{S}_k$ ... k-dimensional **search** space
- $\mathcal{C}_k$ ... k-dimensional **constraints** space

Conjugate gradients: $\mathcal{S}_k = \mathcal{C}_k = \mathcal{K}_k(\mathbf{A}, b)$.

$$r_k \perp \mathcal{K}_k(\mathbf{A}, b) \quad \Leftrightarrow \quad (x - x_k) \perp_{\mathbf{A}} \mathcal{K}_k(\mathbf{A}, b)$$

i.e., $\|x - x_k\|_{\mathbf{A}}$ is minimal ($\mathbf{A}$ is SPD).

CG as an optimization procedure

$\mathbf{A}x = b$

... and minimization of a quadratic functional

Let $\mathbf{A} \in \mathbb{R}^{n \times n}$ be **symmetric** and **positive definite**.

- $\mathbf{A}$ and b define the quadratic functional

$$\mathcal{F}(y) \;\equiv\; \frac{1}{2}\, y^T \mathbf{A} y - y^T b\,, \qquad \mathcal{F} : \mathbb{R}^n \longmapsto \mathbb{R}\,.$$

- Gradient, Hessian

$$\nabla \mathcal{F}(y) = \mathbf{A} y - b\,, \qquad \nabla^2 \mathcal{F}(y) = \mathbf{A}\,.$$

- $\mathcal{F}(y)$ is **strictly convex**.
- $\mathcal{F}(y)$ attains its minimum at $\nabla \mathcal{F}(x) = 0$,

$$\nabla \mathcal{F}(y) = 0 \quad \Leftrightarrow \quad \mathbf{A}x = b\,.$$

Quadratic functional

... and derivation of the conjugate gradient method

- $\mathcal{F}(y)$ and the $\mathbf{A}$-norm of the error,

$$\mathcal{F}(y) \quad = \quad \frac{1}{2}\|x-y\|_{\mathbf{A}}^2 - \frac{1}{2}\|x\|_{\mathbf{A}}^2 \,.$$

- An efficient **strategy** to find the minimum of $\mathcal{F}(y)$?

- Use **line search**, $x_k = x_{k-1} - \gamma_{k-1} p_{k-1}$.

- Choose p_{k-1} to be **conjugate** ($\mathbf{A}$-orthogonal), then

$$\|x-x_k\|_{\mathbf{A}}^2 = \min_{y \in \mathrm{span}\{p_0,\ldots,p_{k-1}\}} \|x-y\|_{\mathbf{A}}^2 \,.$$

- CG $\rightarrow$ use r_k to construct p_k.

Nonlinear conjugate gradient method

For a general problem

$$\min_{x \in \mathbb{R}^n} f(x)$$

consider CG as a **quadratic approximation**.

Association

$$r_k \;\leftrightarrow\; -\nabla f(x_k)\,, \qquad \mathbf{A} \;\leftrightarrow\; \nabla^2 f(x_k)\,,$$

and use the same relations.

Ingredients

- Line search to determine γ_{k-1}.
- Compute gradients numerically.
- Avoid the use of the Hessian.
- Restart every nth iteration.

The (nonlinear) conjugate gradient method

input $\mathbf{A}$, b, x_0
$r_0 = b - \mathbf{A}x_0$
$p_0 = r_0$
for $k = 1, 2, \ldots$ **do**

$$
\begin{aligned}
\gamma_{k-1} &= \frac{r_{k-1}^T r_{k-1}}{p_{k-1}^T \mathbf{A}\, p_{k-1}} \\
x_k &= x_{k-1} + \gamma_{k-1} p_{k-1} \\
r_k &= r_{k-1} - \gamma_{k-1} \mathbf{A}\, p_{k-1} \\
\delta_k &= \frac{r_k^T r_k}{r_{k-1}^T r_{k-1}} \\
p_k &= r_k + \delta_k p_{k-1}
\end{aligned}
$$

test quality of x_k

end for

input f, x_0
$r_0 = -\nabla f(x_0)$
$p_0 = r_0$
for $k = 1, 2, \ldots$ **do**

$$
\begin{aligned}
\gamma_{k-1} &\leftarrow \text{line search} \\
x_k &= x_{k-1} + \gamma_{k-1} p_{k-1} \\
r_k &= -\nabla f(x_k) \\
\delta_k &= \frac{r_k^T r_k}{r_{k-1}^T r_{k-1}} \\
p_k &= r_k + \delta_k p_{k-1}
\end{aligned}
$$

test quality of $\nabla f(x_k)$

end for

CG as Gauss quadrature

(Normalized) orthogonal polynomials

A sequence of polynomials ψ_i of degree i such that

$$\langle \psi_i, \psi_j \rangle \; = \; \delta_{i,j} \, .$$

Usually, the inner product $\langle \cdot, \cdot \rangle$ defined by

$$\int_\zeta^\xi \psi_i \psi_j \, \mathrm{d}x, \qquad \int_\zeta^\xi \psi_i \psi_j w(x) \, \mathrm{d}x, \quad \text{or} \quad \int_\zeta^\xi \psi_i \psi_j \, \mathrm{d}\omega(x) \, .$$

- ψ_i **unique** up to a normalization
- roots $\in (a, b)$, **distinct**
- can be computed by the **three-term** recurrence

$$\beta_{k+1} \, \psi_{k+1}(x) = (x - \alpha_{k+1})\psi_k - \beta_k \, \psi_{k-1}(x)$$

Ortogonal polynomials and Jacobi matrices

Three-term recurrences can be written in the form

$$x \begin{bmatrix} \psi_0 \\ \psi_1 \\ \vdots \\ \vdots \\ \psi_{m-1} \end{bmatrix} = \begin{bmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \alpha_2 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \beta_{m-1} \\ & & & \beta_{m-1} & \alpha_m \end{bmatrix} \begin{bmatrix} \psi_0 \\ \psi_1 \\ \vdots \\ \vdots \\ \psi_{m-1} \end{bmatrix} + \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ \beta_m \psi_n \end{bmatrix}$$

Roots of $\psi_m(x)$ are the eigenvalues of the Jacobi matrix.

Orthogonal polynomials and Gauss Quadrature

Quadrature formula

$$\int_{\zeta}^{\xi} f(\lambda)\, d\omega(\lambda) \;=\; \sum_{i=1}^{k} w_i f(\nu_i) \;+\; \mathcal{R}_k[f]\,.$$

Gauss Quadrature formula:

- Maximal degree of exactness $2k-1$
- **Weights and nodes** determined by **orthogonal polynomials**
- Computed via Jacobi matrices (Golub-Welsch)

  ν_i . . . eigenvalues

  w_i . . . squared 1st components of the normalized eigenvectors

Back to the conjugate gradient method

- CG is a "**polynomial method**",

$$v \in \mathcal{K}_k(\mathbf{A}, b) \quad \Rightarrow \quad v = \sum_{j=0}^{k-1} \zeta_j \mathbf{A}^j b = q(\mathbf{A}) b\,.$$

- Residuals $r_0, \ldots, r_{k-1}$ are **orthogonal**,

$$0 \quad = \quad r_i^T r_j = b^T q_i(\mathbf{A}) q_j(\mathbf{A}) b\,.$$

- Use the spectral decomposition, $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$, $b = \mathbf{U}\omega\,.$

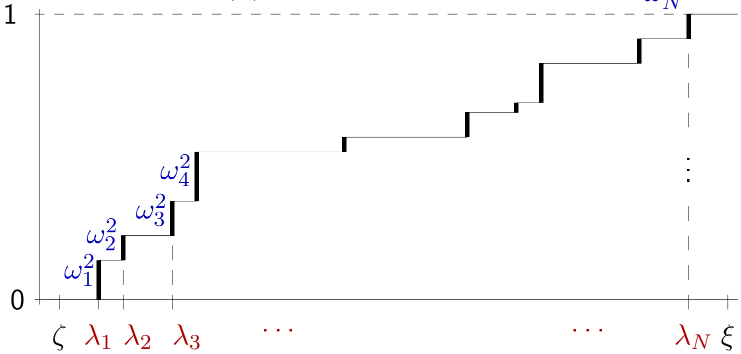
$$0 = \omega^T q_i(\mathbf{\Lambda}) q_j(\mathbf{\Lambda}) \omega = \sum_{\ell=1}^{N} \omega_\ell^2 q_i(\lambda_\ell) q_j(\lambda_\ell) \;\equiv\; \langle q_i, q_j \rangle_{\omega,\Lambda}\,.$$

- CG constructs **a sequence of orthogonal polynomials**.

Distribution function $\omega(\lambda)$

$$\mathbf{A},\ b\ \rightarrow\ \langle\cdot,\cdot\rangle_{\omega,\Lambda}: \qquad \langle f,g\rangle_{\omega,\Lambda} = \sum_{\ell=1}^{N}\omega_\ell^2 f(\lambda_\ell)g(\lambda_\ell)\,.$$

Define the function $\omega(\lambda)$,



Then,

$$\langle f,g\rangle_{\omega,\Lambda} = \int_\zeta^\xi f(\lambda)g(\lambda)\,d\omega(\lambda)\,.$$

CG, Lanczos and Gauss quadrature

At any iteration step k, CG (implicitly) determines **weights** and **nodes** of the k-point Gauss quadrature

$$\int_{\zeta}^{\xi} f(\lambda)\, d\omega(\lambda) \; = \; \sum_{i=1}^{k} \omega_i^{(k)} f(\theta_i^{(k)}) \; + \; \mathcal{R}_k[f]\,.$$

The Jacobi matrix available in CG (Lanczos),

$$\mathbf{T}_k = \mathbf{R}_k^T \mathbf{R}_k\,.$$

Understanding the formula: For $f(\lambda) \equiv \lambda^{-1}$ we get

$$\begin{aligned} \left(\mathbf{T}_n^{-1}\right)_{1,1} \quad &= \quad \left(\mathbf{T}_k^{-1}\right)_{1,1} + \mathcal{R}_k[\lambda^{-1}], \\ \|x\|_{\mathbf{A}}^2 \quad &= \quad \sum_{j=0}^{k-1} \gamma_j \|r_j\|^2 + \|x - x_k\|_{\mathbf{A}}^2\,. \end{aligned}$$

Motivation

The normwise backward error

Given x_k, what are the norms of the smallest perturbations $\Delta\mathbf{A}$ of $\mathbf{A}$ and Δb of b (in the relative sense) such that

$$(\mathbf{A} + \Delta\mathbf{A})\, x_k = b + \Delta b\,?$$

We are interested in the quantity

$$\min\left\{\varepsilon :\; (\mathbf{A} + \Delta\mathbf{A})\, x_k = b + \Delta b,\; \frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|} \leq \varepsilon,\; \frac{\|\Delta b\|}{\|b\|} \leq \varepsilon\right\}$$

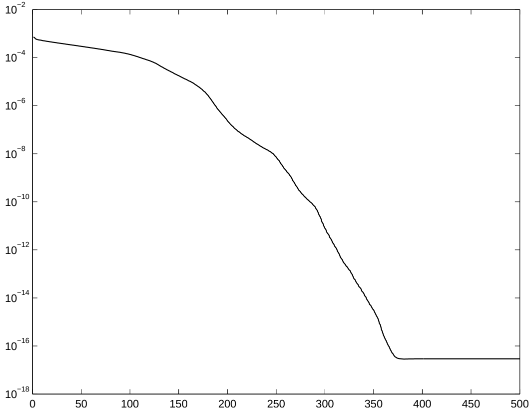
called the **normwise backward error**. It is given by

$$\frac{\|r_k\|}{\|\mathbf{A}\|\|x_k\| + \|b\|}\,.$$

[Rigal, Gaches 1967]

Motivation

Maximum attainable accuracy



$$\|b - \mathbf{A}x_k - r_k\| \leq \varepsilon \, \|\mathbf{A}\| \left(\|x\| + \max_{j \leq k} \|x_j\| \right) \mathcal{O}(k)$$

- κ-bound

$$\frac{\|x - x_k\|_{\mathbf{A}}}{\|x - x_0\|_{\mathbf{A}}} \leq 2 \left(\frac{\sqrt{\kappa(\mathbf{A})} - 1}{\sqrt{\kappa(\mathbf{A})} + 1} \right)^k$$

- $\lambda_{\min}$-bounds

$$\|x - x_k\|_{\mathbf{A}} \leq \frac{\|r_k\|}{\sqrt{\lambda_{\min}}}, \qquad \|x - x_k\| \leq \frac{\|r_k\|}{\lambda_{\min}}$$

- Gauss-Radau quadrature-based bounds

$$\|x - x_k\|_{\mathbf{A}} \leq \sqrt{\gamma_k^{(\mu)}}\, \|r_k\|$$

How to approximate $\lambda_{\min}(\mathbf{A})$ and $\lambda_{\max}(\mathbf{A})$?

$\mathbf{A}$ is symmetric and positive definite

$$\lambda_{\max}(\mathbf{A}) = \|\mathbf{A}\|, \qquad \lambda_{\min}^{-1}(\mathbf{A}) = \|\mathbf{A}^{-1}\|.$$

Important source of information $\rightarrow \mathbf{T}_k = \mathbf{R}_k^T \mathbf{R}_k$.

- Using the **largest** and **smallest** eigenvalue of $\mathbf{T}_k$,
  + can be very accurate
  − solving eigenvalue problems, which k?
  − storing $\mathbf{T}_k$
- Based on **incremental estimation** of $\|\mathbf{T}_k\|$ and $\|\mathbf{T}_k^{-1}\|$
  + accurate enough
  + very cheap
  + no need to store $\mathbf{T}_k$ or some vectors

Incremental estimation of $\|\mathbf{T}_k\|$ and $\|\mathbf{T}_k^{-1}\|$

- In CG, only $\mathbf{R}_k$ is available, $\mathbf{T}_k = \mathbf{R}_k^T \mathbf{R}_k$, and

$$\|\mathbf{T}_k\| = \|\mathbf{R}_k\|^2 \qquad \|\mathbf{T}_k^{-1}\| = \|\mathbf{R}_k^{-1}\|^2.$$

- **Structure**: $\mathbf{R}_k$ and $\mathbf{R}_k^{-1}$ are
  - **upper triangular**, $\mathbf{R}_k$ bidiagonal,
  - How arises $\mathbf{R}_{k+1}$ from $\mathbf{R}_k$, and $\mathbf{R}_{k+1}^{-1}$ from $\mathbf{R}_k^{-1}$?
  - In both cases, by **adding one column and one row**.

- **Incremental norm estimation**: incrementally improve an approximation of the maximum right singular vector. [Bischof 1990], [Duff, Vömmel 2002], [Duintjer Tebbens, Tůma 2014].

The idea of incremental norm estimation

$\mathbf{U}$ is general, upper triangular

Given $\mathbf{U} \in \mathbb{R}^{k \times k}$ **upper triangular**, and z, $\|z\| = 1$, $\|\mathbf{U}z\| \approx \|\mathbf{U}\|$,

$$\hat{\mathbf{U}} = \begin{bmatrix} \mathbf{U} & v \\ & q \end{bmatrix}, \qquad v \in \mathbb{R}^k, \quad q \in \mathbb{R}.$$

Consider new approximate max. right singular vector in the form

$$\hat{z} = \begin{bmatrix} sz \\ c \end{bmatrix} \quad \to \quad \hat{\mathbf{U}}\hat{z} = \begin{bmatrix} s\,\mathbf{U}z + c\,v \\ c\,q \end{bmatrix}$$

where $s^2 + c^2 = 1$ are chosen such that $\|\hat{\mathbf{U}}\hat{z}\|$ is maximum,

$$\|\hat{\mathbf{U}}\hat{z}\|^2 = \begin{bmatrix} s \\ c \end{bmatrix}^T \begin{bmatrix} \rho & \sigma \\ \sigma & \tau \end{bmatrix} \begin{bmatrix} s \\ c \end{bmatrix}$$

$$\rho = \|\mathbf{U}z\|^2, \qquad \sigma = v^T\mathbf{U}z, \qquad \tau = v^Tv + q^2.$$

$\to$ **maximum eigenvalue and eigenvector** of the 2×2 matrix?

Incremental norm estimation

The algorithm

$$\mathbf{U}_{k+1} = \begin{bmatrix} \mathbf{U}_k & v_k \\ & q_k \end{bmatrix}, \quad v_k \in \mathbb{R}^k, \quad q_k \in \mathbb{R}, \quad \left(z_k \in \mathbb{R}^k \right)$$

1. Compute the entries of the 2×2 matrix and Δ_k,

$$\rho_k = \|\mathbf{U}_k z_k\|^2, \quad \sigma_k = v_k^T \mathbf{U}_k z_k\,, \quad \tau_k = v_k^T v_k + q_k^2\,.$$

2. Compute the new estimate ρ_{k+1} using

$$\Delta_k = (\rho_k - \tau_k)^2 + 4\sigma_k^2\,.$$

$$c_k^2 = \frac{1}{2}\left(1 - \frac{\rho_k - \tau_k}{\sqrt{\Delta_k}}\right), \quad \rho_{k+1} = \rho_k + \sqrt{\Delta_k} c_k^2\,.$$

3. If necessary, compute z_{k+1}

$$s_k = \sqrt{1 - c_k^2}, \quad c_k = |c_k| \operatorname{sign}(\sigma_k)\,, \quad z_{k+1} = \begin{bmatrix} s_k z_k \\ c_k \end{bmatrix}.$$

Specialization to upper bidiagonal matrices

$$\mathbf{B}_{k+1} = \left[\begin{array}{cccc|c} a_1 & b_1 & & & 0 \\ & \ddots & \ddots & & \vdots \\ & & \ddots & b_{k-1} & 0 \\ & & & a_k & b_k \\ \hline & & & & a_{k+1} \end{array}\right]$$

Inverse

$$\mathbf{B}_{k+1}^{-1} = \begin{bmatrix} \mathbf{B}_k^{-1} & -w_k \frac{b_k}{a_{k+1}} \\ & \frac{1}{a_{k+1}} \end{bmatrix}$$

where w_k is the last column of the matrix $\mathbf{B}_k^{-1}$, i.e.,

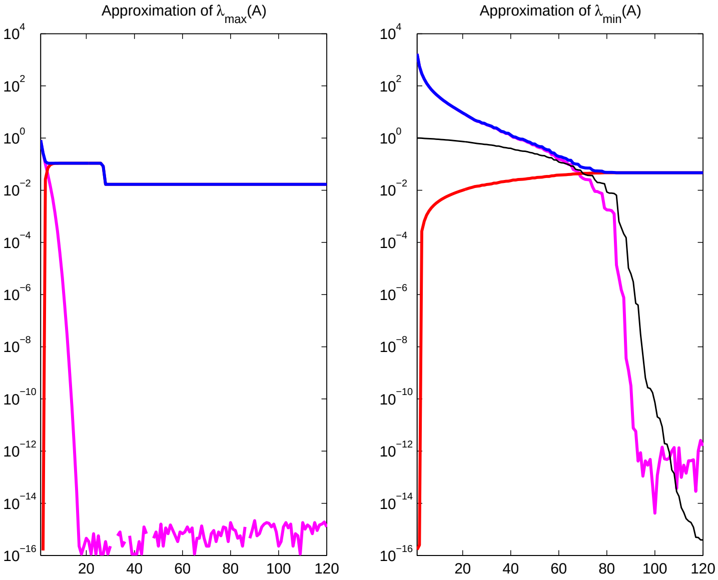
$$w_{k+1} = \begin{bmatrix} -w_k \frac{b_k}{a_{k+1}} \\ \frac{1}{a_{k+1}} \end{bmatrix}.$$

CG with incremental estimation of $\|\mathbf{A}\|$

$\mathbf{R}_k$ in CG have the entries $a_k = \frac{1}{\sqrt{\gamma_{k-1}}}$, $b_k = \sqrt{\frac{\delta_k}{\gamma_{k-1}}}$, $k \geq 1$.

input $\mathbf{A}$, b, x_0
$r_0 = b - \mathbf{A}x_0$, $p_0 = r_0$
for $k = 1, \ldots,$ **do**
  CG iteration(k) $\rightarrow \gamma_{k-1}$, x_k, r_k, δ_k, p_k
  if $k = 1$ **then**
    $c_0^2 = 1$, $\rho_1 = \gamma_0^{-1}$
  end if
  $\sigma_k = \frac{\sqrt{\delta_k}}{\gamma_{k-1}}\, c_{k-1}$
  $\tau_k = \frac{\delta_k}{\gamma_{k-1}} + \frac{1}{\gamma_k}$
  $\Delta_k = (\rho_k - \tau_k)^2 + 4\sigma_k^2$
  $c_k^2 = \frac{1}{2}\left(1 - \frac{\rho_k - \tau_k}{\sqrt{\Delta_k}}\right)$
  $\rho_{k+1} = \rho_k + \sqrt{\Delta_k}\, c_k^2$
end for

strakos48 matrix, $n = 48$



Estimates of the extreme eigenvalues (summary)

- $\mathbf{T}_k = \mathbf{R}_k^T \mathbf{R}_k$ represents an important **source of information**.
- We developed **cheap estimators** of $\lambda_{\min}(\mathbf{A})$ and $\lambda_{\max}(\mathbf{A})$, based on incremental estimation of $\|\mathbf{R}_k\|$ and $\|\mathbf{R}_k^{-1}\|$.
- The reached **relative accuracy** of estimates is usually between 10^{-1} and 10^{-2}.
- These estimates **can be used**, e.g., to approximate
  - the normwise backward error,
  - condition number of $\mathbf{A}$,
  - attainable level of accuracy,
  - $\mathbf{A}$-norm of the error.

Estimating the $\mathbf{A}$-norm of the error

A brief history

- The function $(x - x_k, \mathbf{A}(x - x_k))$ can be used as a **measure of the "goodness"** of x_k as an estimate of x. [Hestenes, Stiefel 1952]
- **Gene Golub** and collaborators: [Dahlquist, Golub, Nash 1978], [Golub, Meurant 1994] relate error bounds to **Gauss quadrature**.
- **Idea of estimating errors** in CG, behavior in finite precision arithmetic [Golub, Strakoš 1994], the CGQL algorithm [Golub, Meurant, 1997], intensively studied in many later papers.
- **Numerical stability** of the estimates based on Gauss quadrature [Strakoš, T. 2002], [Strakoš, T. 2005].
- Summary in the book [Golub, Meurant 2010].
- Improvement [Meurant, T. 2013] $\rightarrow$ the **CGQ algorithm**.

At any iteration step k, CG determines (through $\mathbf{T}_k$) **weights** and **nodes** of the k-point **Gauss quadrature**. For $f(\lambda) = \lambda^{-1}$:

$$\|x\|_{\mathbf{A}}^2 \;=\; \sum_{j=0}^{k-1} \gamma_j \|r_j\|^2 + \|x - x_k\|_{\mathbf{A}}^2 \,.$$

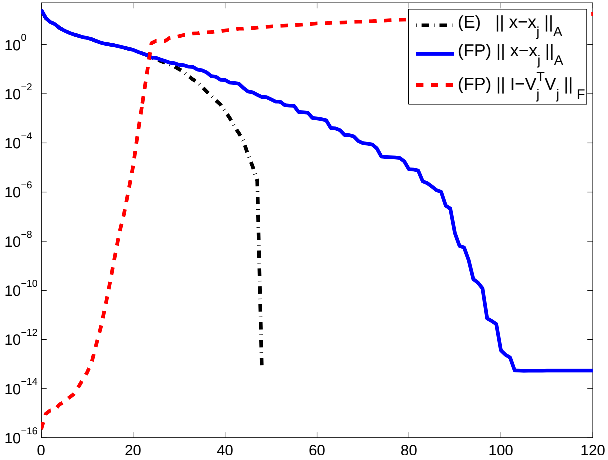
Considering the same formula for some $k + d$ we obtain

$$\|x - x_k\|_{\mathbf{A}}^2 \;=\; \sum_{j=k}^{k+d-1} \gamma_j \|r_j\|^2 + \|x - x_{k+d}\|_{\mathbf{A}}^2$$

We have a **lower bound**. E.g., if $\frac{\|x - x_{k+d}\|_{\mathbf{A}}}{\|x - x_k\|_{\mathbf{A}}} < 0.8$, then

$$\sqrt{\sum_{j=k}^{k+d-1} \gamma_j \|r_j\|^2} < \|x - x_k\|_{\mathbf{A}} < 2\sqrt{\sum_{j=k}^{k+d-1} \gamma_j \|r_j\|^2} \,.$$

Orthogonality is lost, convergence is delayed!



Identities need not hold in finite precision arithmetic!

$$\|x - x_k\|_{\mathbf{A}}^2 = \sum_{j=k}^{k+d-1} \gamma_j \|r_j\|^2 + \|x - x_{k+d}\|_{\mathbf{A}}^2.$$

Practically relevant questions:

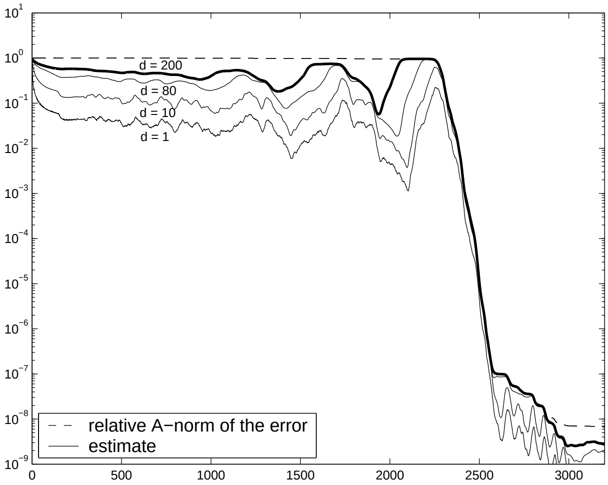
+ We can compute it almost **for free**.
+ It is **numerically stable** [Strakoš, T. 2002, 2005]:

– How to **control the quality** of the bound?
  - How to choose d adaptively?
  - How to detect a decrease of the $\mathbf{A}$-norm of the error?

The choice of d

R. Kouhia: Cylindrical shell (Matrix Market), matrix s3dkt3m2

PCG, $\kappa(\mathbf{A}) = 3.62e+11$, $n = 90499$, cholinc$(\mathbf{A}, 0)$.



CG and Gauss-Radau quadrature

- Modification of Gauss quadrature rule.
- **Prescribe** μ such that $0 < \mu \leq \lambda_{\min}$.

Gauss-Radau quadrature rule can be written algebraically as

$$\|x\|_{\mathbf{A}}^2 \;=\; \sum_{j=0}^{k-2} \gamma_j \|r_j\|^2 + \gamma_{k-1}^{(\mu)} \|r_{k-1}\|^2 + \mathcal{R}_k^{(R)}$$

where $\mathcal{R}_k^{(R)} < 0$, $\gamma_0^{(\mu)} = \mu^{-1}$, and

$$\gamma_{j+1}^{(\mu)} \;=\; \frac{\left(\gamma_j^{(\mu)} - \gamma_j\right)}{\mu\left(\gamma_j^{(\mu)} - \gamma_j\right) + \delta_{j+1}}\,.$$

[Golub, Meurant, 1997], [Meurant, T. 2013]

How to construct an **upper bound** on $\|x - x_k\|_{\mathbf{A}}$?

CG and Gauss-Radau quadrature

Upper bound

Using **Gauss** rule for k and **Gauss-Radau** rule for $k+d$,

$$\|x - x_k\|_{\mathbf{A}}^2 \quad = \quad \sum_{j=k}^{k+d-2} \gamma_j \|r_j\|^2 + \gamma_{k+d-1}^{(\mu)} \|r_{k+d-1}\|^2 + \mathcal{R}_{k+d}^{(R)}$$

and $\mathcal{R}_{k+d}^{(R)} < 0$.

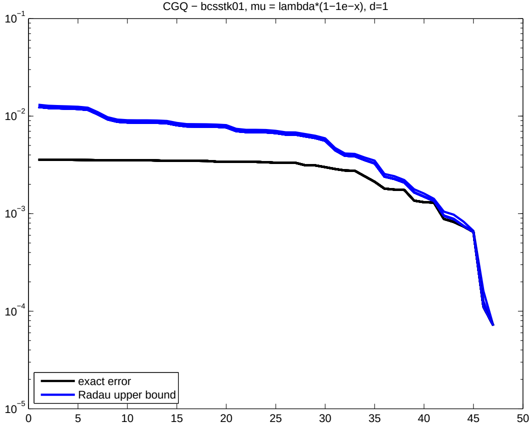
Practically relevant questions:

+ We can compute it almost **for free**.
- How to get μ?
- **Sensitivity** of the bound on μ?
- Numerical **behavior**?
- **Quality** of the bound?

Gauss-Radau upper bound, exact arithmetic

various values of μ, bcsstk01 matrix, $n = 48$

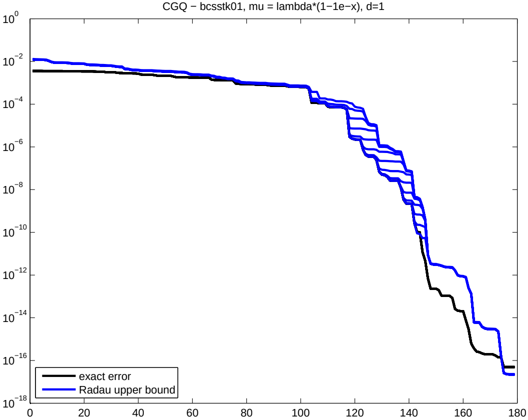
$$\mu = \lambda_{\min}(1 - 10^{-m}), \quad m = 1, \dots, 14$$



Gauss-Radau upper bound, finite precision arithmetic

various values of μ, bcsstk01 matrix, $n = 48$

$$\mu = \lambda_{\min}(1 - 10^{-m}), \quad m = 1, \ldots, 14$$



An upper bound on the upper bound

Find an envelope for all curves that corresponds to various μ's.

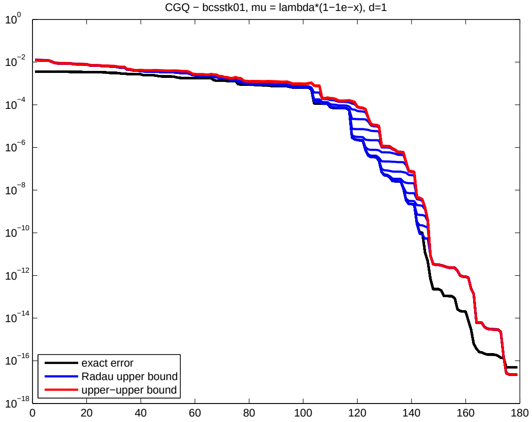
Multiply the updating formula for $\gamma_{j+1}^{(\mu)}$ by μ

$$
\begin{aligned}
\mu\gamma_{j+1}^{(\mu)} &= \frac{\mu\left(\gamma_j^{(\mu)} - \gamma_j\right)}{\mu\left(\gamma_j^{(\mu)} - \gamma_j\right) + \delta_{j+1}} \\
&< \frac{\mu\,\gamma_j^{(\mu)}}{\mu\,\gamma_j^{(\mu)} + \delta_{j+1}} \\
&\leq \frac{\frac{\|r_j\|^2}{\|p_j\|^2}}{\frac{\|r_j\|^2}{\|p_j\|^2} + \delta_{j+1}} = \frac{\|r_{j+1}\|^2}{\|p_{j+1}\|^2}
\end{aligned}
$$

An upper bound on the upper bound

In summary, if $\mu \leq \lambda_{\min}$, then

$$\|x - x_k\|_{\mathbf{A}}^2 \leq \gamma_k^{(\mu)} \|r_k\|^2 < \frac{\|r_k\|^2}{\mu} \frac{\|r_k\|^2}{\|p_k\|^2}.$$



CG and Gauss-Radau quadrature

Upper bound on the upper bound

If $\mu \leq \lambda_{\min}$, then

$$\|x - x_k\|_{\mathbf{A}}^2 \quad < \quad \sum_{j=k}^{k+d-1} \gamma_j \|r_j\|^2 + \frac{\|r_{k+d}\|^2}{\mu} \frac{\|r_{k+d}\|^2}{\|p_{k+d}\|^2}$$

Practically relevant questions:

+ We can compute it almost **for free**.
+ No too much **sensitive** to the choice of μ.
+ Heuristics $\rightarrow$ we can use it even if $\mu > \lambda_{\min}$ (e.g., use the estimate of the smallest Ritz value).
+ Numerical **behavior**? Looks OK, so far no analysis.

– **Quality** of the bound. Not so tight.

Conclusions and questions

- $\mathbf{T}_k = \mathbf{R}_k^T \mathbf{R}_k$ represents an important **source of information** that can be used for estimating the CG convergence.
- The estimation of the **$\mathbf{A}$-norm of the error** should be based on the numerical stable **lower bound**.
- How to **detect a decrease** of the $\mathbf{A}$-norm of the error? (How to choose d adaptively?).
- We found an **upper bound** on the Gauss-Radau upper bound that is insensitive to the choice of μ, and hope to be useful in the adaptive choice of d for the lower bound.

Related papers

- G. Meurant and P. Tichý, [Practical estimation of the A-norm of the error in CG, in preparation, 2017]
- G. Meurant and P. Tichý, [On computing quadrature-based bounds for the $\mathbf{A}$-norm of the error in conjugate gradients, Numer. Algorithms, 62 (2013), pp. 163-191]
- G. H. Golub and G. Meurant, [Matrices, moments and quadrature with applications, Princeton University Press, USA, 2010.]
- Z. Strakoš and P. Tichý, [On error estimation in the conjugate gradient method and why it works in finite precision computations, Electron. Trans. Numer. Anal., 13 (2002), pp. 56–80.]
- G. H. Golub and Z. Strakoš, [Estimates in quadratic formulas, Numer. Algorithms, 8 (1994), pp. 241–268.]

Thank you for your attention!