FEM and Sparse Linear System Solving

Lecture 8, Nov 10, 2017: Steepest descent and conjugate gradient algorithms

http://people.inf.ethz.ch/arbenz/FEM17

Peter Arbenz
Computer Science Department, ETH Zürich
E-mail: arbenz@inf.ethz.ch

Survey on lecture

- The finite element method
- Direct solvers for sparse systems
- Iterative solvers for sparse systems
  - Stationary iterative methods, preconditioning
  - Steepest descent and conjugate gradient methods
  - Krylov space methods, GMRES, MINRES
  - Incomplete factorization preconditioning
  - Multigrid preconditioning
  - Indefinite problems

Today's topic

Today, we restrict ourselves to symmetric positive definite (SPD) problems.

1. Steepest descent minimization of the energy norm
2. Conjugate gradient minimization of the energy norm
3. Chebyshev iteration

Literature

- O. Axelsson & V.A. Barker, *Finite element solution of boundary value problems*, Academic Press, 1984.
  Also: SIAM classics in applied mathematics, 2001
- Saad: *Iterative methods for sparse linear systems*, SIAM, 2nd edition, 2003. Available from
  http://www-users.cs.umn.edu/~saad/books.html

Energy norm minimization

Theorem: For $A\boldsymbol{x} = \boldsymbol{b}$, with A SPD, consider for a real constant γ the functional

$$\varphi(\boldsymbol{x}) := \frac{1}{2}\boldsymbol{x}^T A\boldsymbol{x} - \boldsymbol{x}^T\boldsymbol{b} + \gamma.$$

- φ is continuously differentiable, with $\textbf{grad}\,\varphi(\boldsymbol{x}) = A\boldsymbol{x} - \boldsymbol{b} = -\boldsymbol{r}(\boldsymbol{x})$.
- If $\gamma := \frac{1}{2}\boldsymbol{b}^T A^{-1}\boldsymbol{b}$ (our choice), then

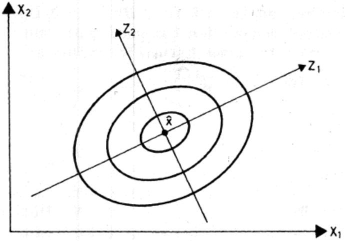
$$\begin{aligned}\varphi(\boldsymbol{x}) &= \frac{1}{2}(A\boldsymbol{x} - \boldsymbol{b})^T A^{-1}(A\boldsymbol{x} - \boldsymbol{b}) \equiv \frac{1}{2}\|\boldsymbol{r}(\boldsymbol{x})\|_{A^{-1}}^2 \\ &= \frac{1}{2}(\boldsymbol{x} - A^{-1}\boldsymbol{b})^T A(\boldsymbol{x} - A^{-1}\boldsymbol{b}) \\ &= \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}^*)^T A(\boldsymbol{x} - \boldsymbol{x}^*) \equiv \frac{1}{2}\|\boldsymbol{e}(\boldsymbol{x})\|_A^2.\end{aligned}$$

- Notice that our choice of γ makes the minimum of φ zero.

Energy norm minimization (cont.)

- φ has a unique minimum satisfying $\boldsymbol{r}(\boldsymbol{x}) = \boldsymbol{0} \Leftrightarrow \boldsymbol{x} = A^{-1}\boldsymbol{b}$.
- Let $A = U\Lambda U^T$ be the spectral decomposition of A. Then

$$\hat{\varphi}(\boldsymbol{z}) := \varphi(U\boldsymbol{z} + \boldsymbol{x}^*) = \frac{1}{2}\|U\boldsymbol{z} + \boldsymbol{x}^* - \boldsymbol{x}^*\|_A^2 = \frac{1}{2}\|U\boldsymbol{z}\|_A^2$$
$$= \frac{1}{2}\boldsymbol{z}^T U^T A U\boldsymbol{z} = \frac{1}{2}\boldsymbol{z}^T\Lambda\boldsymbol{z} = \frac{1}{2}\sum_{i=1}^{n}\lambda_i z_i^2.$$



Level surfaces ($n = 2$) of a quadratic functional φ with a spd A.

Descent directions

Definition: Suppose that for a functional φ and vectors $\boldsymbol{x}$, $\boldsymbol{d}$ there is a α_0 such that

$$\varphi(\boldsymbol{x} + \alpha \boldsymbol{d}) < \varphi(\boldsymbol{x}), \qquad 0 < \alpha \leq \alpha_0.$$

Then, $\boldsymbol{d}$ is a descent direction for φ at $\boldsymbol{x}$.

For our quadratic functional we have

$$\varphi(\boldsymbol{x} + \alpha \boldsymbol{d}) = \varphi(\boldsymbol{x}) - \alpha \, \boldsymbol{r}(\boldsymbol{x})^T \boldsymbol{d} + \frac{\alpha^2}{2} \, \boldsymbol{d}^T A \boldsymbol{d}.$$

Therefore, $\boldsymbol{d}$ is a descent direction if it has a positive component in direction of $-\textbf{grad}\, \varphi(\boldsymbol{x}) = \boldsymbol{r}(\boldsymbol{x})$.

The descent is steepest if $\boldsymbol{d}$ is aligned with $\boldsymbol{r}$.

Local steepest descent minimization of φ

Let $\boldsymbol{x}_k$ be an approximation of $\boldsymbol{x}^*$. We want to improve $\boldsymbol{x}_k$ in the steepest descent direction.

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \mathbf{grad}\, \varphi(\boldsymbol{x}_k) = \boldsymbol{x}_k + \alpha_k \boldsymbol{r}_k$$

We choose α_k such that $\varphi(\boldsymbol{x}_{k+1})$ is minimized (*'local line search'*). From

$$\begin{aligned} 2\varphi(\boldsymbol{x}_{k+1}) &= \|\boldsymbol{e}_{k+1}\|_A^2 = (\boldsymbol{x}^* - \boldsymbol{x}_{k+1})^T A(\boldsymbol{x}^* - \boldsymbol{x}_{k+1}) \\ &= (\boldsymbol{x}^* - \boldsymbol{x}_k - \alpha_k \boldsymbol{r}_k)^T A(\boldsymbol{x}^* - \boldsymbol{x}_k - \alpha_k \boldsymbol{r}_k) \\ &= \|\boldsymbol{e}_k\|_A^2 - 2\alpha_k \boldsymbol{r}_k^T \boldsymbol{r}_k + \alpha_k^2 \boldsymbol{r}_k^T A\, \boldsymbol{r}_k, \qquad A\, \boldsymbol{e}_k = \boldsymbol{r}_k, \end{aligned}$$

it follows that

$$\frac{d\varphi(\alpha_k; \boldsymbol{x}_k, \boldsymbol{r}_k)}{d\alpha_k} \overset{!}{=} 0 \quad \Rightarrow \quad \alpha_k = \frac{\boldsymbol{r}_k^T \boldsymbol{r}_k}{\boldsymbol{r}_k^T A\, \boldsymbol{r}_k}.$$

Steepest descent algorithm

Choose $\boldsymbol{x}_0$.
Set $\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0$ and $\boldsymbol{q}_0 = A\boldsymbol{r}_0$.
$k := 0$.
until convergence **do**
$\quad \alpha_k := \boldsymbol{r}_k^T \boldsymbol{r}_k / \boldsymbol{r}_k^T \boldsymbol{q}_k$.
$\quad \boldsymbol{x}_{k+1} := \boldsymbol{x}_k + \alpha_k \boldsymbol{r}_k$.
$\quad \boldsymbol{r}_{k+1} := \boldsymbol{r}_k - \alpha_k \boldsymbol{q}_k$.
$\quad$ Compute $\boldsymbol{q}_{k+1} = A\boldsymbol{r}_{k+1}$.
$\quad k := k + 1$.
end do

This is very memory efficient, but how fast is the convergence?

Convergence of steepest descent

Lemma: (Kantorovich inequality) Let A be any real SPD matrix and $\lambda_{\min}$, $\lambda_{\max}$ its smallest and largest eigenvalues. Then

$$\frac{(\mathbf{x}^T A\mathbf{x})(\mathbf{x}^T A^{-1}\mathbf{x})}{(\mathbf{x}^T\mathbf{x})^2} \leq \frac{(\lambda_{\max}+\lambda_{\min})^2}{4\lambda_{\max}\lambda_{\min}} \qquad \text{for all } \mathbf{x} \neq \mathbf{0}.$$

For a proof see Saad, p. 138.

Remark: From this we have

$$\frac{(\mathbf{x}^T\mathbf{x})^2}{(\mathbf{x}^T A\mathbf{x})(\mathbf{x}^T A^{-1}\mathbf{x})} \geq \frac{4\lambda_{\max}\lambda_{\min}}{(\lambda_{\max}+\lambda_{\min})^2}$$

and therefore

$$1 - \frac{(\mathbf{x}^T\mathbf{x})^2}{(\mathbf{x}^T A\mathbf{x})(\mathbf{x}^T A^{-1}\mathbf{x})} \leq 1 - \frac{4\lambda_{\max}\lambda_{\min}}{(\lambda_{\max}+\lambda_{\min})^2} = \frac{(\lambda_{\max}-\lambda_{\min})^2}{(\lambda_{\max}+\lambda_{\min})^2}$$

Convergence of steepest descent (cont.)

Theorem: For the errors $\boldsymbol{e}_k = \boldsymbol{x}^* - \boldsymbol{x}_k$ in steepest descent we have

$$\|\boldsymbol{e}_{k+1}\|_A \le \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} \|\boldsymbol{e}_k\|_A.$$

Proof.

$$\begin{aligned}
\|\boldsymbol{e}_{k+1}\|_A^2 &= \boldsymbol{e}_{k+1}^T A\, \boldsymbol{e}_{k+1} = \boldsymbol{r}_{k+1}^T A^{-1}\, \boldsymbol{r}_{k+1} && (A\boldsymbol{e}_{k+1} = \boldsymbol{r}_{k+1}) \\
&= (\boldsymbol{r}_k - \alpha_k A\boldsymbol{r}_k)^T A^{-1} (\boldsymbol{r}_k - \alpha_k A\boldsymbol{r}_k) \\
&= \boldsymbol{r}_k^T A^{-1} (I - \alpha_k A)^2 \boldsymbol{r}_k \\
&= \boldsymbol{r}_k^T A^{-1}\, \boldsymbol{r}_k - (\boldsymbol{r}_k^T \boldsymbol{r}_k)^2 / \boldsymbol{r}_k^T A\, \boldsymbol{r}_k && \left(\alpha_k = \frac{\boldsymbol{r}_k^T\, \boldsymbol{r}_k}{\boldsymbol{r}_k^T A\, \boldsymbol{r}_k}\right) \\
&= \boldsymbol{r}_k^T A^{-1}\, \boldsymbol{r}_k \left(1 - \frac{(\boldsymbol{r}_k^T \boldsymbol{r}_k)^2}{(\boldsymbol{r}_k^T A\, \boldsymbol{r}_k)(\boldsymbol{r}_k^T A^{-1}\, \boldsymbol{r}_k)}\right) && \square
\end{aligned}$$

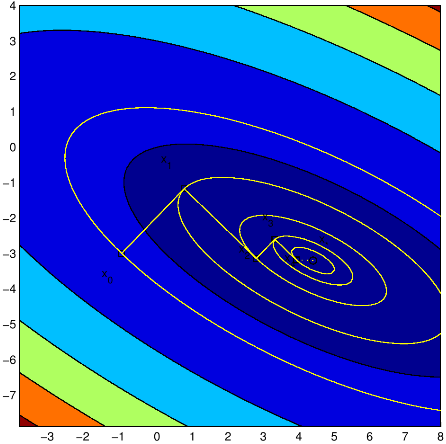
Convergence of steepest descent (cont.)

From the theorem we have

$$\|\boldsymbol{e}_{k+1}\|_A \leq \frac{\kappa(A)-1}{\kappa(A)+1}\,\|\boldsymbol{e}_k\|_A \leq \left(\frac{\kappa(A)-1}{\kappa(A)+1}\right)^{k+1} \|\boldsymbol{e}_0\|_A.$$

- We always have convergence; but it can be very slow if $\kappa(A) \gg 1$. Then the contour ellipsoids are very elongated , that is stretched or squished $\Longleftrightarrow$ some eigenvalues of A are much smaller/larger than others.
- Steepest descent minimizes φ only *locally* (greedy algorithm).

Slow convergence of steepest descent method



Picture: Martin Gutknecht

Iteration steps

How many iteration steps $p = p(\varepsilon)$ are needed such that

$$\|\boldsymbol{e}_p\|_A \leq \varepsilon \|\boldsymbol{e}_0\|_A?$$

We write

$$\frac{1}{\varepsilon} \approx \left(\frac{1 + 1/\kappa(A)}{1 - 1/\kappa(A)} \right)^p$$

Taking logarithms and using
$\log[(1+s)/(1-s)] = 2(s + s^3/3 + \cdots)$ we get

$$p(\varepsilon) \approx \frac{1}{2} \kappa(A) \log(1/\varepsilon).$$

Preconditioning steepest descent

Let M be SPD with Cholesky factorization $M = LL^T$
Standard: instead of solving $A\boldsymbol{x} = \boldsymbol{b}$, apply steepest descent to the SPD problem

$$L^{-1}AL^{-T}\boldsymbol{y} = L^{-1}\boldsymbol{b}, \quad L^{-T}\boldsymbol{y} = \boldsymbol{x}$$

The above equation is obtained as follows: Since $M^{-1} = L^{-T}L^{-1}$ we have

$$\begin{aligned} M^{-1}A\boldsymbol{x} = M^{-1}\boldsymbol{b} &\iff L^{-T}L^{-1}A\boldsymbol{x} = L^{-T}L^{-1}\boldsymbol{b} \\ &\iff L^{-1}A\boldsymbol{x} = L^{-1}\boldsymbol{b} \\ &\iff L^{-1}AL^{-T}L^{T}\boldsymbol{x} = L^{-1}\boldsymbol{b} \\ &\iff L^{-1}AL^{-T}\boldsymbol{y} = L^{-1}\boldsymbol{b}, \qquad \boldsymbol{y} = L^{T}\boldsymbol{x}. \end{aligned}$$

Preconditioning steepest descent (cont.)

Clever: replace all Euclidean inner products by M-inner product $\langle \mathbf{x}, \mathbf{y} \rangle_M := \mathbf{x}^T M \mathbf{y}$ and solve system $M^{-1}A\mathbf{x} = M^{-1}\mathbf{b}$.

Note: $M^{-1}A$ is SPD *with respect to the M-inner product*

$$\begin{aligned}\langle \mathbf{x}, M^{-1}A\mathbf{y} \rangle_M &= \mathbf{x}^T M(M^{-1}A\mathbf{y}) = \mathbf{x}^T A\mathbf{y} \\ &= \mathbf{x}^T AM^{-1}M\mathbf{y} = (M^{-1}A\mathbf{x})^T M\mathbf{y} \\ &= \langle M^{-1}A\mathbf{x}, \mathbf{y} \rangle_M\end{aligned}$$

Preconditioning steepest descent (cont.)

So, we can apply the steepest descent algorithm to

$$M^{-1}A\boldsymbol{x} = M^{-1}\boldsymbol{b}$$

and replace the ordinary Euklidian inner product with the M-inner product.

We want to retain the notation $\boldsymbol{r}_k = \boldsymbol{b} - A\boldsymbol{x}_k$ for the residual that we actually care for. We define additionally the preconditioned residual by

$$\boldsymbol{z}_k = M^{-1}\boldsymbol{r}_k = M^{-1}(\boldsymbol{b} - A\boldsymbol{x}_k).$$

Preconditioning steepest descent (cont.)

So, in the steepest descent algorithm we simply have to replace

- A by $M^{-1}A$,
- $\boldsymbol{r}_k$ by $\boldsymbol{z}_k$, and
- Euklidian inner products by M-inner products.

So, $\boldsymbol{q}_k = M^{-1}A\boldsymbol{z}_k$ and

$$\alpha_k = \frac{\boldsymbol{z}_k^T M \boldsymbol{z}_k}{\boldsymbol{z}_k^T M \boldsymbol{q}_k} = \frac{\boldsymbol{z}_k^T \boldsymbol{r}_k}{\boldsymbol{z}_k^T A \boldsymbol{z}_k} = \frac{\boldsymbol{z}_k^T \boldsymbol{r}_k}{\boldsymbol{z}_k^T \tilde{\boldsymbol{q}}_k}$$

with $\tilde{\boldsymbol{q}}_k = A\boldsymbol{z}_k$.
We actually do not need the auxiliary vectors $\boldsymbol{q}_k$ but only the $\tilde{\boldsymbol{q}}_k$.
Therefore, we omit the tilde in the following algorithm.

Preconditioned steepest descent algorithm

Choose $\boldsymbol{x}_0$.
Set $\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0$. Solve $M\boldsymbol{z}_0 = \boldsymbol{r}_0$. Set $\boldsymbol{q}_0 = A\boldsymbol{z}_0$.
$k := 0$.
until convergence **do**
    $\alpha_k := \boldsymbol{z}_k^T \boldsymbol{r}_k / \boldsymbol{z}_k^T \boldsymbol{q}_k$.
    $\boldsymbol{x}_{k+1} := \boldsymbol{x}_k + \alpha_k \boldsymbol{z}_k$.
    $\boldsymbol{r}_{k+1} := \boldsymbol{r}_k - \alpha_k \boldsymbol{q}_k$.
    Solve $M\boldsymbol{z}_{k+1} = \boldsymbol{r}_{k+1}$.
    Compute $\boldsymbol{q}_{k+1} = A\boldsymbol{z}_{k+1}$.
    $k := k + 1$.
end do

Per iteration step we have to multiply a vector by A and solve a linear system with M.

Convergence of the Preconditioned Steepest Descent Algorithm

We can easily adapt the convergence theorem to obtain

$$\|\boldsymbol{e}_{k+1}\|_A \leq \left(\frac{\kappa(M^{-1}A) - 1}{\kappa(M^{-1}A) + 1}\right) \|\boldsymbol{e}_k\|_A.$$

(Notice that a direct translation of norms gives

$$\boldsymbol{e}_k^T M(M^{-1}A\boldsymbol{e}_k) = \boldsymbol{e}_k^T A\boldsymbol{e}_k = \|\boldsymbol{e}_k\|_A^2.)$$

The preconditioner M again has the function to reduce the condition of the original system matrix A.

The conjugate gradient (cg) method

Problem: Solve $A\boldsymbol{x} = \boldsymbol{b}$ with symmetric positive definite (spd) A.

The steepest descent algorithm is a greedy algorithm. It solves the problem of minimization of φ only *locally*.

cg is also a descent method: In each iteration step we look for the minimum of φ along a line $\boldsymbol{x}_k + \alpha \boldsymbol{p}_k$. Here, $\boldsymbol{x}_k$ is the current approximation of the solution $\boldsymbol{x}^*$ and $\boldsymbol{p}_k$ is the search direction. In order that the method is a descent method, $\boldsymbol{p}_k$ has to have a component in the direction of the residual $\boldsymbol{r}_k = \boldsymbol{r}(\boldsymbol{x}_k)$, meaning

$$\boldsymbol{p}_k^T \boldsymbol{r}_k > 0.$$

We determine

$$\boxed{\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k} \qquad (1)$$

such that $\varphi(\boldsymbol{x}_{k+1})$ is minimal. As

$$\varphi(\boldsymbol{x}_{k+1}) = \alpha_k \boldsymbol{p}_k^T A \boldsymbol{x}_k + \frac{1}{2}\alpha_k^2 \boldsymbol{p}_k^T A \boldsymbol{p}_k - \alpha_k \boldsymbol{p}_k^T \boldsymbol{b} + \text{const}$$

is a quadratic polynomial in α_k with positive second derivative, its minimum is unique.

$$0 \stackrel{!}{=} \frac{\partial \varphi(\boldsymbol{x}_k)}{\partial \alpha_k} = \boldsymbol{p}_k^T (A\boldsymbol{x}_k - \boldsymbol{b}) + \alpha_k \boldsymbol{p}_k^T A \boldsymbol{p}_k = -\boldsymbol{p}_k^T \boldsymbol{r}_k + \alpha_k \boldsymbol{p}_k^T A \boldsymbol{p}_k.$$

Thus

$$\alpha_k = \frac{\boldsymbol{p}_k^T \boldsymbol{r}_k}{\boldsymbol{p}_k^T A \boldsymbol{p}_k} \qquad (2)$$

The residual can be computed recursively,

$$\boxed{\mathbf{r}_{k+1} = \mathbf{b} - A\mathbf{x}_{k+1} = \mathbf{b} - A\mathbf{x}_k - \alpha_k A\mathbf{p}_k = \mathbf{r}_k - \alpha_k A\mathbf{p}_k} \qquad (3)$$

Notice that $A\mathbf{p}_k$ is needed already in the computation of α_k in (2). Multiplying (3) by $\mathbf{p}_k^T$ gives

$$\mathbf{p}_k^T \mathbf{r}_{k+1} = \mathbf{p}_k^T \mathbf{r}_k - \alpha_k \mathbf{p}_k^T A\mathbf{p}_k \stackrel{(2)}{=} 0. \qquad (4)$$

That is, the new residual is orthogonal to the previous search direction. This was true already with the steepest descent method.

How should we choose the $\boldsymbol{p}_k$?

Method of steepest descent: $\boldsymbol{p}_k = -\mathbf{grad}\,\varphi(\boldsymbol{x}_k) = \boldsymbol{r}_k$.
This choice leads to slow convergence if $\varphi(\boldsymbol{x})$ form long narrow ellipsoids (i.e. if A has a big condition number).

We set

$$\boldsymbol{p}_0 = \boldsymbol{r}_0 \qquad (= \boldsymbol{b} \quad \text{if } \boldsymbol{x}_0 = \mathbf{0}).$$

(as we do not have anything better) and

$$\boldsymbol{p}_k = \boldsymbol{r}_k + \beta_{k-1}\boldsymbol{p}_{k-1}, \qquad k = 1, 2, \ldots \tag{5}$$

This makes $\boldsymbol{p}_k$ a descent direction independent of the choice of β_{k-1} because (4) implies $\boldsymbol{r}_k^T\boldsymbol{p}_k \overset{(5)}{=} \boldsymbol{r}_k^T\boldsymbol{r}_k > 0$.
(Steepest descent: $\beta_{k-1} = 0$.)

Strategy for determining β_{k-1}: Minimize error $\|\boldsymbol{x}-\boldsymbol{x}^*\|_A = \|\boldsymbol{r}\|_{A^{-1}}$ over some k dimensional subspace of $\mathbb{R}^n$.

We have $\boldsymbol{r}_1 \overset{(3)}{=} \boldsymbol{r}_0 - \alpha_0 A\boldsymbol{p}_0 = \boldsymbol{r}_0 - \alpha_0 A\boldsymbol{r}_0$.

$$\begin{aligned}
\boldsymbol{r}_2 &\overset{(3)}{=} \boldsymbol{r}_1 - \alpha_1 A\boldsymbol{p}_1 \\
&\overset{(5)}{=} \boldsymbol{r}_1 - \alpha_1 A(\boldsymbol{r}_1 + \beta_0\boldsymbol{p}_0) \\
&\overset{(3)}{=} (\boldsymbol{r}_0 - \alpha_0 A\boldsymbol{r}_0) - \alpha_1 A(\boldsymbol{r}_0 - \alpha_0 A\boldsymbol{r}_0) - \alpha_1\beta_0 A\boldsymbol{r}_0 \\
&= \boldsymbol{r}_0 - (\alpha_0 + \alpha_1 + \alpha_1\beta_0)A\boldsymbol{r}_0 + \alpha_0\alpha_1 A^2\boldsymbol{r}_0 \\
&\vdots \\
\boldsymbol{r}_k &= \boldsymbol{r}_0 + \mu_1^{(k)} A\boldsymbol{r}_0 + \mu_2^{(k)} A^2\boldsymbol{r}_0 + \cdots + \mu_k^{(k)} A^k\boldsymbol{r}_0, \quad \mu_k^{(k)} = \pm\prod_{j=0}^{k-1}\alpha_j \neq 0.
\end{aligned}$$

So, $\boldsymbol{r}_k$ is element in a k-dimensional affine subspace.

Let $S_k = \text{span}\{A\boldsymbol{r}_0, A^2\boldsymbol{r}_0, \ldots, A^k\boldsymbol{r}_0\}$ be a k-dim. subspace of $\mathbb{R}^n$.

Let $T_k = \boldsymbol{r}_0 + S_k = \{\boldsymbol{r} \in \mathbb{R}^n | \boldsymbol{r} = \boldsymbol{r}_0 + \boldsymbol{h}, \boldsymbol{h} \in S_k\}$.

Then $\boldsymbol{r}_k \in T_k$.

Theorem: If we impose the condition

$$\|\boldsymbol{r}_k\|_{A^{-1}} = \min_{\boldsymbol{r}\in T_k} \|\boldsymbol{r}\|_{A^{-1}}$$

then

$$\beta_k = -\frac{\boldsymbol{r}_{k+1}^T A\boldsymbol{p}_k}{\boldsymbol{p}_k^T A\boldsymbol{p}_k}.$$

Furthermore,

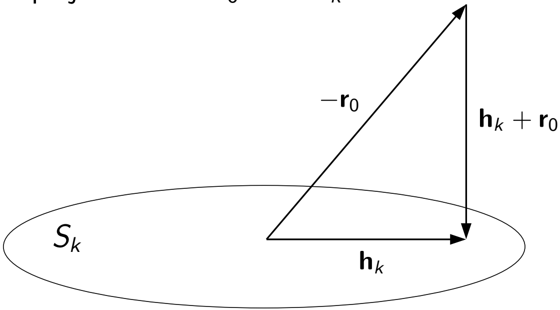
$$\boldsymbol{r}_k^T \boldsymbol{r}_\ell = 0, \qquad k \neq \ell. \tag{6}$$

$$\boldsymbol{p}_k^T A\boldsymbol{p}_\ell = 0, \qquad k \neq \ell. \tag{7}$$

Proof: $\boldsymbol{r} \in T_k = \boldsymbol{r}_0 + S_k \implies \boldsymbol{r} = \boldsymbol{r}_0 + \boldsymbol{h},\ \boldsymbol{h} \in S_k$

$$\|\boldsymbol{r}_k\|_{A^{-1}} \equiv \|\boldsymbol{r}_0 + \boldsymbol{h}_k\|_{A^{-1}} = \min_{\boldsymbol{h} \in S_k} \|\boldsymbol{r}_0 + \boldsymbol{h}\|_{A^{-1}}$$

Interpretation: To have small norm, $\boldsymbol{h}_k$ must be close to $-\boldsymbol{r}_0$.
Let $\boldsymbol{h}_k$ be the projection of $-\boldsymbol{r}_0$ onto S_k.



Note that the projection is with the A^{-1} inner product.

$\boldsymbol{r}_0 + \boldsymbol{h}_k$ is A^{-1}-orthogonal on S_k

$$\Longleftrightarrow (\boldsymbol{r}_0 + \boldsymbol{h}_k)^T A^{-1} \boldsymbol{h} = 0, \qquad \forall \boldsymbol{h} \in S_k.$$

Let $\boldsymbol{r} \in T_{k-1}$. Then $\boldsymbol{h} := A\boldsymbol{r} \in S_k$.

$$\Longrightarrow (\boldsymbol{r}_0 + \boldsymbol{h}_k)^T A^{-1} A\boldsymbol{r} = \boldsymbol{r}_k \boldsymbol{r} = 0, \qquad \forall \boldsymbol{r} \in T_{k-1}.$$

From this we have (6) since $\boldsymbol{r}_\ell \in T_\ell \subset T_{k-1}$ for all $\ell < k$.

We now want to establish (7). Let $\ell < k$. Then

$$\begin{aligned} \boldsymbol{p}_k^T A \boldsymbol{p}_\ell = (A\boldsymbol{p}_k)^T \boldsymbol{p}_\ell &\overset{(3)}{=} -\alpha_k^{-1} (\boldsymbol{r}_{k+1} - \boldsymbol{r}_k)^T \boldsymbol{p}_\ell \\ &= -\alpha_k^{-1} (\boldsymbol{r}_{k+1} - \boldsymbol{r}_k)^T (\boldsymbol{r}_\ell + \beta_{\ell-1} \boldsymbol{p}_{\ell-1}) \\ &\overset{(6)}{=} -\alpha_k^{-1} \beta_{\ell-1} (\boldsymbol{r}_{k+1} - \boldsymbol{r}_k)^T \boldsymbol{p}_{\ell-1}. \end{aligned}$$

Hence, by induction,

$$\boldsymbol{p}_k^T A \boldsymbol{p}_\ell = -\alpha_k^{-1} \left(\prod_{i=0}^{\ell-1} \beta_i \right) (\boldsymbol{r}_{k+1} - \boldsymbol{r}_k)^T \underbrace{\boldsymbol{p}_0}_{\boldsymbol{r}_0} \overset{(6)}{=} 0.$$

Now, the value of β_{k-1} follows by setting $\ell = k - 1$:

$$\begin{aligned} 0 \overset{!}{=} \boldsymbol{p}_{k-1}^T A \boldsymbol{p}_k &= \boldsymbol{p}_{k-1}^T A (\boldsymbol{r}_k + \beta_{k-1} \boldsymbol{p}_{k-1}) \\ \Rightarrow \beta_{k-1} &= -\frac{\boldsymbol{p}_{k-1}^T A \boldsymbol{r}_k}{\boldsymbol{p}_{k-1}^T A \boldsymbol{p}_{k-1}} \end{aligned}$$

□

The rest is cosmetics. We want to get nicer expressions for some of the quantities involved.

Multiplying (5) by $\boldsymbol{r}_k^T$ gives

$$\boldsymbol{r}_k^T \boldsymbol{p}_k = \boldsymbol{r}_k^T \boldsymbol{r}_k + \beta_{k-1} \underbrace{\boldsymbol{r}_k^T \boldsymbol{p}_{k-1}}_{\overset{(4)}{=}0}.$$

Thus,

$$\boxed{\alpha_k = \frac{\boldsymbol{p}_k^T \boldsymbol{r}_k}{\boldsymbol{p}_k^T A \boldsymbol{p}_k} = \frac{\boldsymbol{r}_k^T \boldsymbol{r}_k}{\boldsymbol{p}_k^T A \boldsymbol{p}_k} = \frac{\|\boldsymbol{r}_k\|^2}{\boldsymbol{p}_k^T A \boldsymbol{p}_k} > 0.} \tag{8}$$

Furthermore, we have

$$\boldsymbol{r}_{k+1}^T A \boldsymbol{p}_k \overset{(3)}{=} \boldsymbol{r}_{k+1}^T \left(\frac{1}{\alpha_k} (\boldsymbol{r}_k - \boldsymbol{r}_{k+1}) \right) \overset{(6)}{=} -\frac{\|\boldsymbol{r}_{k+1}\|^2}{\alpha_k}.$$

Thus,

$$\beta_k = -\frac{\boldsymbol{r}_{k+1}^T A \boldsymbol{p}_k}{\boldsymbol{p}_k^T A \boldsymbol{p}_k} = \frac{\|\boldsymbol{r}_{k+1}\|^2}{\alpha_k \boldsymbol{p}_k^T A \boldsymbol{p}_k} \overset{(8)}{=} \frac{\|\boldsymbol{r}_{k+1}\|^2}{\|\boldsymbol{r}_k\|^2}$$

whence

$$\boxed{\beta_k = \frac{\rho_{k+1}}{\rho_k}, \qquad \rho_k = \|\boldsymbol{r}_k\|^2.} \tag{9}$$

Algorithm CG

The conjugate gradient algorithm thus becomes

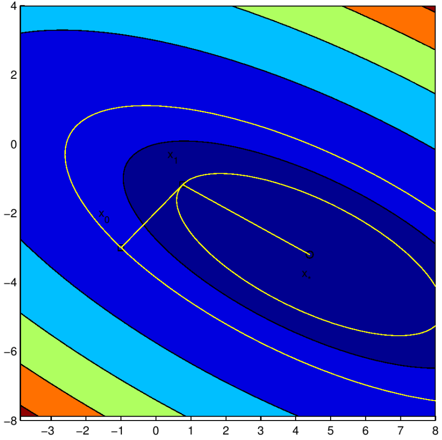
Choose $\boldsymbol{x}_0$, set $\boldsymbol{p}_0 = \boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0$ and $\rho_0 = \|\boldsymbol{r}_0\|_2^2$
for $k = 0, 1, \ldots$ **do**
$\quad \boldsymbol{q}_k = A\boldsymbol{p}_k.$
$\quad \alpha_k = \boldsymbol{r}_k^T \boldsymbol{r}_k / \boldsymbol{p}_k^T \boldsymbol{q}_k.$
$\quad \boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k.$
$\quad \boldsymbol{r}_{k+1} = \boldsymbol{r}_k - \alpha_k \boldsymbol{q}_k.$
$\quad \rho_{k+1} = \|\boldsymbol{r}_{k+1}\|_2^2.$
$\quad$ if $\rho_{k+1} < \varepsilon$ exit.
$\quad \beta_k = \rho_{k+1}/\rho_k.$
$\quad \boldsymbol{p}_{k+1} = \boldsymbol{r}_{k+1} + \beta_k \boldsymbol{p}_k.$
endfor

Remark: Here we used the auxiliary vector $\boldsymbol{q}_k = A\boldsymbol{p}_k$.

Finite termination property

Theorem: The cg method applied to a spd n-by-n matrix A finds the solution after at most n iteration steps.

Proof. Direct consequence of (6).



Picture: Martin Gutknecht

Intermezzo: Chebyshev polynomials

- Family of orthogonal polynomials on $[-1, 1]$ with respect to weight function

$$w(x) = \frac{1}{\sqrt{1 - x^2}}$$

- Define

$$T_j(x) = \cos(j \arccos(x)) = \cos(j\vartheta), \qquad x = \cos(\vartheta).$$

- Clearly, the larger j the more oscillatory T_j.
- Orthogonality:

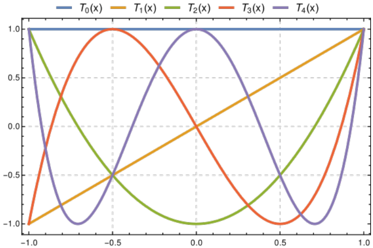
$$\int_{-1}^{1} w(x) T_j(x) T_k(x)\, dx = \begin{cases} 0, & j \neq k, \\ \frac{\pi}{2}, & j = k > 0, \\ \pi, & j = k = 0. \end{cases}$$

Intermezzo: Chebyshev polynomials (cont.)

- Simple 3-term recurrence relation[1]

$$T_0(x) = 1, \qquad T_1(x) = x,$$
$$T_{j+1}(x) = 2xT_j(x) - T_{j-1}(x), \qquad j \geq 1.$$

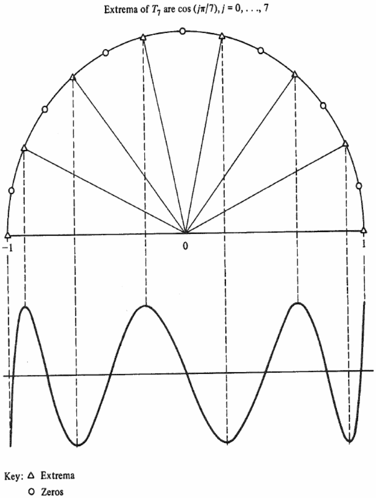
- Polynomials satisfy $|T_j(x)| \leq 1, \qquad -1 \leq x \leq 1.$



Chebyshev polynomials $T_0, \ldots, T_5$ in interval $[-1, 1]$
Source: https://en.wikipedia.org/wiki/Chebyshev_polynomials

[1] Remember: $\cos(j+1)\vartheta + \cos(j-1)\vartheta = 2\cos\vartheta\cos j\vartheta$

Intermezzo: Chebyshev polynomials (cont.)



Intermezzo: Chebyshev polynomials (cont.)

Theorem (Chebyshev polynomials)

Of all polynomials $p \in \mathbb{P}_n$ with a coefficient 1 in the highest (x^n) term the Chebyshev polynomial $T_n(x)/2^{n-1}$ has the smallest maximum norm in the interval $[-1, 1]$.

Theorem (Chebyshev polynomials on interval (α, β))

Let $(\alpha, \beta) \subset \mathbb{R}$ be nonempty and let $\gamma \in \mathbb{R}$ be outside $[\alpha, \beta]$. Then,

$$\min_{p \in \mathbb{P}_n, p(\gamma)=1} \max_{\alpha < x < \beta} |p(x)|$$

is attained by the shifted Chebyshev polynomial

$$\hat{T}_n(x) = \frac{T_n(1 + 2\frac{x-\beta}{\beta-\alpha})}{T_n(1 + 2\frac{\gamma-\beta}{\beta-\alpha})}.$$

Shifted Chebyshev polynomials

Let us now look at shifted Chebyshev polynomials p_k that correspond to the interval $[\lambda_1, \lambda_n]$ instead of $[-1, 1]$.

We assume that $0 < \lambda_1$ and normalize the polynomials: $p_k(0) = 1$.

We define the map

$$[\lambda_1, \lambda_n] \ni \lambda \longmapsto x = \frac{\lambda_1 + \lambda_n - 2\lambda}{\lambda_n - \lambda_1} \in [-1, 1].$$

and

$$\vartheta \equiv \frac{\lambda_1 + \lambda_n}{2}, \qquad \delta \equiv \frac{\lambda_n - \lambda_1}{2}, \qquad \sigma_k \equiv p_k(0) = T_k\left(\frac{\vartheta}{\delta}\right)$$

3-term recurrence for σ's

$$\sigma_{k+1} = 2\frac{\vartheta}{\delta}\sigma_k - \sigma_{k-1}, \qquad \sigma_0 = 1, \quad \sigma_1 = \frac{\vartheta}{\delta}. \tag{10}$$

Three-term recurrence for p_k

Then,

$$p_k(\lambda) = \frac{1}{\sigma_k} T_k\left(\frac{\vartheta - \lambda}{\delta}\right), \quad p_0(\lambda) = 1, \quad p_1(\lambda) = \frac{\delta}{\vartheta}\frac{\vartheta - \lambda}{\delta} = 1 - \frac{\lambda}{\vartheta}.$$

The 3-term recurrence for p_k's is given by

$$\begin{aligned} p_{k+1}(\lambda) &= \frac{1}{\sigma_{k+1}}\left[2\frac{\vartheta - \lambda}{\delta} T_k\left(\frac{\vartheta - \lambda}{\delta}\right) - T_{k-1}\left(\frac{\vartheta - \lambda}{\delta}\right)\right] \\ &= \frac{1}{\sigma_{k+1}}\left[2\frac{\vartheta - \lambda}{\delta}\sigma_k p_k(\lambda) - \sigma_{k-1} p_{k-1}(\lambda)\right] \\ &= \frac{\sigma_k}{\sigma_{k+1}}\left[2\frac{\vartheta - \lambda}{\delta} p_k(\lambda) - \frac{\sigma_{k-1}}{\sigma_k} p_{k-1}(\lambda)\right]. \end{aligned}$$

Three-term recurrence for p_k (cont.)

Defining

$$\rho_k = \frac{\sigma_k}{\sigma_{k+1}}, \qquad k = 1, 2, \ldots$$

The 3-term recurrence (10) for the σ's gives

$$\rho_k = \frac{1}{2\sigma_1 - \rho_{k-1}}.$$

The 3-term recurrence for p_k's then becomes

$$p_{k+1}(\lambda) = \rho_k \left[2\left(\sigma_1 - \frac{\lambda}{\delta}\right) p_k(\lambda) - \rho_{k-1} p_{k-1}(\lambda) \right],$$

$$p_0(\lambda) = 1, \qquad p_1(\lambda) = 1 - \frac{\lambda}{\vartheta}.$$

Convergence of CG

Convergence rate:

$$\|\boldsymbol{x} - \boldsymbol{x}_k\|_A \leq \|\boldsymbol{x} - \boldsymbol{x}_0\|_A \left(\frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} + 1} \right)^k . \qquad (11)$$

The number of iterations $p(\varepsilon)$ to reduce the error $\|\boldsymbol{x} - \boldsymbol{x}_k\|_A$ by a factor ε is

$$p(\varepsilon) \approx \frac{1}{2}\sqrt{\kappa(A)}\log(2/\varepsilon).$$

This is in general a huge reduction in comparison with steepest descent.

Convergence of CG (cont.)

$$\|\boldsymbol{r}_k\|_{A^{-1}} = \min_{\boldsymbol{r}\in T_k} \|\boldsymbol{r}\|_{A^{-1}}$$

A typical element of T_k has the form

$$\boldsymbol{r} = \boldsymbol{r}_0 + \sum_{j=1}^{k} \mu_j A^j \boldsymbol{r}_0 = p_k(A)\boldsymbol{r}_0,$$

where $p_k \in \mathbb{P}'_k$ is a polynomial of degree k normalized such that $p_k(0) = 1$. So,

$$\|\boldsymbol{r}_k\|_{A^{-1}} = \min_{p_k\in\mathbb{P}'_k} \|p_k(A)\boldsymbol{r}_0\|_{A^{-1}} = \min_{p_k\in\mathbb{P}'_k} [\boldsymbol{r}_0^T A^{-1} p_k(A)^2 \boldsymbol{r}_0]^{1/2}.$$

Convergence of CG (cont.)

Let again $A = U\Lambda U^T$ be the spectral decomposition of A and let $\boldsymbol{z} = U^T\boldsymbol{r}_0$. Then

$$\|\boldsymbol{r}_k\|_{A^{-1}} = \min_{p_k \in \mathbb{P}'_k} [\boldsymbol{r}_0^T A^{-1} p_k(A)^2 \boldsymbol{r}_0]^{1/2} = \min_{p_k \in \mathbb{P}'_k} \sum_{i=1}^{n} z_i^2 \lambda_i^{-1} p_k(\lambda_i)^2.$$

If $|p_k(\lambda_i)| \leq M$ for all eigenvalues λ_i then

$$\|\boldsymbol{x}_k - \boldsymbol{x}^*\|_A = \|\boldsymbol{r}_k\|_{A^{-1}} \leq M \left(\sum_{i=1}^{n} z_i^2 \lambda_i^{-1} \right)^{1/2} = M \, \|\boldsymbol{r}_0\|_{A^{-1}}$$

To get at a value for M, we now select a set S that contains all the eigenvalues and seek a polynomial $\tilde{p}_k(\lambda)$ such that $M := \max_{\lambda \in S} |\tilde{p}_k(\lambda)|$ is small.

Convergence of CG (cont.)

It is straightforward to set $S = [\lambda_1, \lambda_n] = [\lambda_{\min}, \lambda_{\max}]$. Then we look for a polynomial $\tilde{p}_k(\lambda)$ with the property that

$$\max_{\lambda_1 \le \lambda \le \lambda_n} |\tilde{p}_k(\lambda)| = \min_{p_k \in \mathbb{P}'_k} \max_{\lambda_1 \le \lambda \le \lambda_n} |p_k(\lambda)|.$$

The solution to this problem is known to be the shifted Chebyshev polynomial of degree k

$$\tilde{p}_k(\lambda) = \frac{T_k((\lambda_n + \lambda_1 - 2\lambda)/(\lambda_n - \lambda_1))}{T_k((\lambda_n + \lambda_1)/(\lambda_n - \lambda_1))}$$

that increases rapidly outside the interval S. We have that

$$\max_{\lambda_1 \le \lambda \le \lambda_n} |\tilde{p}_k(\lambda)| = \frac{1}{T_k(\frac{\lambda_n + \lambda_1}{\lambda_n - \lambda_1})}$$

Convergence of CG (cont.)

From a particular representation of Chebyshev polynomials one obtains (see Saad, p. 204f)

$$\frac{1}{T_k(\frac{\lambda_n+\lambda_1}{\lambda_n-\lambda_1})} = \frac{1}{T_k(\eta)} \qquad \eta = \frac{\lambda_n + \lambda_1}{\lambda_n - \lambda_1}$$

$$\leq 2\left(\frac{1}{\eta + \sqrt{\eta^2+1}}\right)^k = \left(\eta - \sqrt{\eta^2 - 1}\right)^k$$

$$= \left(\frac{\lambda_n + \lambda_1 - 2\sqrt{\lambda_1\lambda_n}}{\lambda_n - \lambda_1}\right)^k$$

$$= \left(\frac{\sqrt{\lambda_n} + \sqrt{\lambda_1}}{\sqrt{\lambda_n} - \sqrt{\lambda_1}}\right)^k \qquad \square$$

Representative values for $T_k(1+2\gamma)$

γ	10^{-4}	10^{-3}	10^{-2}	10^{-1}
$T_{10}(1+2\gamma)$	1.02	1.21	3.75	252
$T_{100}(1+2\gamma)$	3.76	$2.79 \cdot 10^{2}$	$2.35 \cdot 10^{8}$	$5.34 \cdot 10^{26}$
$T_{200}(1+2\gamma)$	27.3	$1.55 \cdot 10^{5}$	$1.10 \cdot 10^{17}$	$5.71 \cdot 10^{53}$
$T_{1000}(1+2\gamma)$	$2.43 \cdot 10^{8}$	$1.45 \cdot 10^{27}$	$2.59 \cdot 10^{86}$	$9.72 \cdot 10^{269}$

$$\gamma = \frac{\lambda_1}{\lambda_n - \lambda_1}$$

$$\frac{\lambda_n + \lambda_1}{\lambda_n - \lambda_1} = \frac{\lambda_n - \lambda_1 + 2\lambda_1}{\lambda_n - \lambda_1} = 1 + \frac{2\lambda_1}{\lambda_n - \lambda_1} = 1 + 2\gamma$$

Preconditioning CG with SPD M

As with steepest descent we apply the conjugate gradient algorithm to the SPD problem

$$M^{-1}A\boldsymbol{x} = M^{-1}\boldsymbol{b}$$

and replace the ordinary Euklidian inner product with the M-inner product.

Here is how the crucial equations (1), (3), (8), and (9) change. On the left are the formulae with the straightforward changements, on the right you see how we actually use them.

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k \Longrightarrow \boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k \qquad \text{unchanged}$$

Preconditioning CG with SPD M (cont.)

We want to retain the notation $\boldsymbol{r}_k = \boldsymbol{b} - A\boldsymbol{x}_k$ for the residual that we actually care for, so we define the preconditioned residual by $\boldsymbol{z}_k = M^{-1}\boldsymbol{r}_k$.

$$\boldsymbol{z}_{k+1} = \boldsymbol{z}_k - \alpha_k M^{-1} A\boldsymbol{p}_k \Longrightarrow \boldsymbol{r}_{k+1} = \boldsymbol{r}_k - \alpha_k A\boldsymbol{p}_k,$$

$$\alpha_k = \frac{\boldsymbol{z}_k^T M \boldsymbol{z}_k}{\boldsymbol{p}_k^T M M^{-1} A\boldsymbol{p}_k} \Longrightarrow \alpha_k = \frac{\boldsymbol{r}_k^T \boldsymbol{z}_k}{\boldsymbol{p}_k^T A\boldsymbol{p}_k}$$

$$\rho_k = \boldsymbol{z}_k^T M \boldsymbol{z}_k \Longrightarrow \rho_k = \boldsymbol{r}_k^T \boldsymbol{z}_k$$

Notice that there is just one additional set of vectors, $\{\boldsymbol{z}_k\}$.

Algorithm PCG

Choose $\boldsymbol{x}_0$, set $\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0$. Solve $M\boldsymbol{z}_0 = \boldsymbol{r}_0$. $\rho_0 = \boldsymbol{z}_0^T\boldsymbol{r}_0$. Set $\boldsymbol{p}_0 = \boldsymbol{z}_0$.
for $k = 0, 1, \ldots$ **do**
 $\boldsymbol{q}_k = A\boldsymbol{p}_k$.
 $\alpha_k = \boldsymbol{z}_k^T\boldsymbol{r}_k/\boldsymbol{p}_k^T\boldsymbol{q}_k$.
 $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k\boldsymbol{p}_k$.
 $\boldsymbol{r}_{k+1} = \boldsymbol{r}_k - \alpha_k\boldsymbol{q}_k$.
 Solve $M\boldsymbol{z}_{k+1} = \boldsymbol{r}_{k+1}$.
 $\rho_{k+1} = \boldsymbol{z}_{k+1}^T\boldsymbol{r}_{k+1}$.
 if $\rho_{k+1} < \varepsilon$ exit.
 $\beta_k = \rho_{k+1}/\rho_k$.
 $\boldsymbol{p}_{k+1} = \boldsymbol{z}_{k+1} + \beta_k\boldsymbol{p}_k$.
endfor

There is one new statement in the algorithm in which the preconditioned residual $\boldsymbol{z}_k$ is computed.

Chebyshev iteration

- In each iteration step, CG determines $\boldsymbol{x}_k$ such that $\|\boldsymbol{e}_k\|_A = \|\boldsymbol{x}^* - \boldsymbol{x}_k\|_A$ or $\|\boldsymbol{r}_k\|_{A^{-1}}$ is minimized in some k-dimensional subspace of $\mathbb{R}^n$.
- In the proof of convergence we employ Chebyshev polynomials to establish some upper bounds for $\|\boldsymbol{r}_k\|_{A^{-1}}$,

$$\|\boldsymbol{r}_k\|_{A^{-1}} \leq M\|\boldsymbol{r}_0\|_{A^{-1}}, \qquad M = \max_{\lambda_1 \leq \lambda \leq \lambda_n} |\tilde{p}_k(\lambda)| = \frac{1}{T_k(\frac{\lambda_n+\lambda_1}{\lambda_n-\lambda_1})}$$

where

$$\tilde{p}_k(\lambda) = \frac{T_k((\lambda_n + \lambda_1 - 2\lambda)/(\lambda_n - \lambda_1))}{T_k((\lambda_n + \lambda_1)/(\lambda_n - \lambda_1))}$$

is the shifted Chebyshev polynomial of degree k corresponding to the interval $[\lambda_1, \lambda_n]$ normalized such that $\tilde{p}_k(0) = 1$.

Chebyshev iteration (cont.)

- The upper bound is "realistic". Therefore it seems to be natural to define the Chebyshev iteration such that

$$\boldsymbol{r}_k = \tilde{p}_k(A)\boldsymbol{r}_0, \qquad k > 0.$$

- Because of the 3-term recurrence for Chebyshev polynomials, this iteration can be executed efficiently.
- Why should this be useful? CG gives the best we can hope for. We can avoid computing the coefficients α_k and β_k at the expense of upper/lower bounds for the spectrum.
- The computation of α_k and β_k requires inner products which may be costly (in particular in a parallel computation).
- Bounds for $|\tilde{p}_k(A)|$ independent of $\boldsymbol{r}_0$.

Chebyshev iteration (cont.)

How do we get at $\boldsymbol{x}_k$?
Try to find a recurrence relation for $\boldsymbol{x}_k$.

$$\boldsymbol{r}_k = p_k(A)\boldsymbol{r}_0, \quad p \in \mathbb{P}'_k \iff p_k(\lambda) = 1 + \lambda s_k(\lambda), \quad s \in \mathbb{P}_{k-1}.$$

So,

$$\boldsymbol{r}_{k+1} - \boldsymbol{r}_k = A(\boldsymbol{e}_{k+1} - \boldsymbol{e}_k) = -A(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k), \quad \boldsymbol{e}_j = \boldsymbol{x}^* - \boldsymbol{x}_j.$$

$$\iff p_{k+1}(\lambda) - p_k(\lambda) = -\lambda(s_{k+1}(\lambda) - s_k(\lambda)).$$

Chebyshev iteration (cont.)

Using the 3-term recurrence for the p_k's and $1 = \rho_k(2\sigma_1 - \rho_{k-1})$ (see slide 37) gives

$$\begin{aligned} p_{k+1}(\lambda) - p_k(\lambda) &= p_{k+1}(\lambda) - \rho_k(2\sigma_1 - \rho_{k-1})p_k(\lambda) \\ &= \rho_k\left[-\frac{2\lambda}{\delta}p_k(\lambda) + \rho_{k-1}(p_k(\lambda) - p_{k-1}(\lambda))\right] \end{aligned}$$

and after division by $-\lambda$

$$s_{k+1}(\lambda) - s_k(\lambda) = \rho_k\left[\rho_{k-1}(s_k(\lambda) - s_{k-1}(\lambda)) + \frac{2}{\delta}p_k(\lambda)\right].$$

Defining $\boldsymbol{d}_k = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k$ with get

$$\boldsymbol{d}_k = \rho_k\left[\rho_{k-1}\boldsymbol{d}_{k-1} + \frac{2}{\delta}\boldsymbol{r}_k\right].$$

Algorithm: Chebyshev iteration

Choose $\boldsymbol{x}_0$, set $\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0$; $\sigma_1 = \vartheta/\delta$.
$\rho_0 = 1/\sigma_1$; $\boldsymbol{d}_0 = \frac{1}{\vartheta}\boldsymbol{r}_0$.
for $k = 0, 1, \ldots$ until convergence **do**
$\quad \boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{d}_k$.
$\quad \boldsymbol{r}_{k+1} = \boldsymbol{r}_k - A\boldsymbol{d}_k$.
$\quad \rho_{k+1} = (2\sigma_1 - \rho_k)^{-1}$.
$\quad \boldsymbol{d}_{k+1} = \rho_{k+1}\rho_k\boldsymbol{d}_k + \frac{2\rho_{k+1}}{\delta}\boldsymbol{r}_{k+1}$.
endfor

No inner products but knowledge of (bounds for) λ_1 and λ_n required.

For details see Saad, Section 12.3.2.

See also
https://en.wikipedia.org/wiki/Chebyshev_iteration.

Exercise 8:

http://people.inf.ethz.ch/arbenz/FEM17/pdfs/ex8.pdf