Mini-course on Stationary Iterative Methods

Lecture 3 – Krylov methods

Felix Kwok
Département de mathématiques et de statistique, Université Laval

June 1, 2021

2021 CRM Summer School: Solving large systems efficiently in multi-physics numerical simulations

Outline

Conjugate gradient method

A minimization problem

Let A be a symmetric, positive definite (spd) matrix, i.e.,

- A is symmetric, $A^T = A$;
- $\mathbf{x}^T A\mathbf{x} > 0$ for all $\mathbf{x} \neq 0$.

If A is spd, then it is invertible. Also, solving $A\mathbf{x} = \mathbf{f}$ is equivalent to solving the minimization problem

$$\min_{\mathbf{x}} J(\mathbf{x}) := \min_{\mathbf{x}} \left[\frac{1}{2}\mathbf{x}^T A\mathbf{x} - \mathbf{x}^T\mathbf{f} \right],$$

since if $A\mathbf{x}^* = \mathbf{f}$, then for any $\mathbf{v} \neq 0$, we have

$$\begin{aligned} J(\mathbf{x}^* + \mathbf{v}) &= \frac{1}{2}(\mathbf{x}^* + \mathbf{v})^T A(\mathbf{x}^* + \mathbf{v}) - (\mathbf{x}^* + \mathbf{v})^T\mathbf{f} \\ &= \frac{1}{2}(\mathbf{x}^*)^T A\mathbf{x}^* + \mathbf{v}^T A\mathbf{x}^* + \frac{1}{2}\mathbf{v}^T A\mathbf{v} - (\mathbf{x}^*)^T\mathbf{f} - \mathbf{v}^T\mathbf{f} \\ &> \frac{1}{2}(\mathbf{x}^*)^T A\mathbf{x}^* - (\mathbf{x}^*)^T\mathbf{f} = J(\mathbf{x}^*). \end{aligned}$$

Steepest descent method

Steepest descent finds the minimum of a function $f(\mathbf{x})$ by generating a sequence of $\mathbf{x}^k$ with successively smaller values of J :

1. Given an initial guess $\mathbf{x}^0$, calculate $\nabla J(\mathbf{x}^0)$.
2. Let $\mathbf{d}^0 = -\nabla J(\mathbf{x}^0)$. Find a step size $\alpha > 0$ that minimizes $J(\mathbf{x}^0 + \alpha \mathbf{d}^0)$.
3. Set $\mathbf{x}^1 = \mathbf{x}^0 + \alpha_0 \mathbf{d}^0$, where α_0 is the optimal step size above.
4. Repeat steps 1–3 until convergence.

Steepest descent method

Steepest descent finds the minimum of a function $f(\mathbf{x})$ by generating a sequence of $\mathbf{x}^k$ with successively smaller values of J :

1. Given an initial guess $\mathbf{x}^0$, calculate $\nabla J(\mathbf{x}^0)$.
2. Let $\mathbf{d}^0 = -\nabla J(\mathbf{x}^0)$. Find a step size $\alpha > 0$ that minimizes $J(\mathbf{x}^0 + \alpha\mathbf{d}^0)$.
3. Set $\mathbf{x}^1 = \mathbf{x}^0 + \alpha_0\mathbf{d}^0$, where α_0 is the optimal step size above.
4. Repeat steps 1–3 until convergence.

When $J(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T A\mathbf{x} - \mathbf{x}^T\mathbf{f}$:

- $\nabla J(\mathbf{x}) = A\mathbf{x} - \mathbf{f} \implies \boxed{\mathbf{d}^0 = \mathbf{f} - A\mathbf{x}^0}$ (i.e., the residual)
- To minimize $J(\mathbf{x}^0 + \alpha\mathbf{d}^0)$:

$$\frac{d}{d\alpha}J(\mathbf{x}^0 + \alpha\mathbf{d}^0) = \nabla J(\mathbf{x}^0 + \alpha\mathbf{d}^0)\cdot\mathbf{d}^0 = 0$$

$$(A(\mathbf{x}^0 + \alpha\mathbf{d}^0) - \mathbf{f})\cdot\mathbf{d}^0 = 0$$

$$\boxed{\alpha = \frac{(\mathbf{f} - A\mathbf{x}^0)\cdot\mathbf{d}^0}{\mathbf{d}^0\cdot A\mathbf{d}^0} = \frac{\mathbf{d}^0\cdot\mathbf{d}^0}{\mathbf{d}^0\cdot A\mathbf{d}^0}}$$

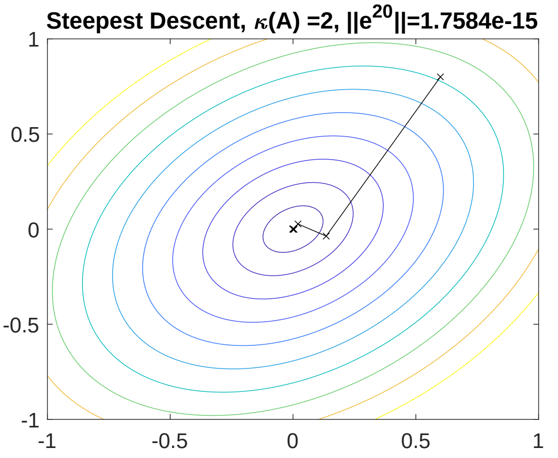
Steepest descent method

To solve : $A\mathbf{x} = \mathbf{f}$, A spd
Given : Initial guess $\mathbf{x}^0$

FOR $k = 0, 1, 2, \ldots,$ DO

1. Set $\mathbf{d}^k = \mathbf{f} - A\mathbf{x}^k$
2. Compute $\alpha_k = \dfrac{\mathbf{d}^k \cdot \mathbf{d}^k}{\mathbf{d}^k \cdot A\mathbf{d}^k}$
3. Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$

END DO



Method is very slow, especially when the “aspect ratio” is large !

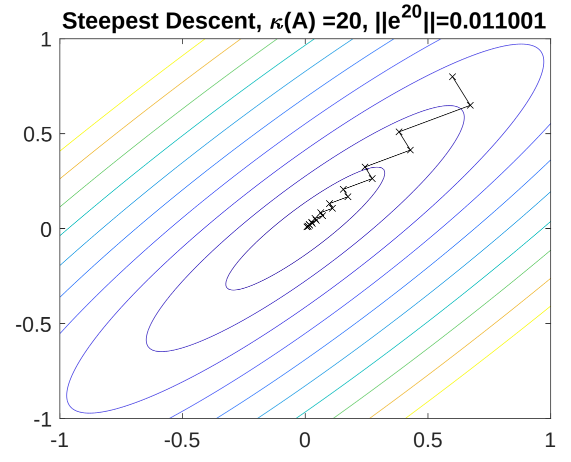
Steepest descent method

To solve : $A\mathbf{x} = \mathbf{f}$, A spd
Given : Initial guess $\mathbf{x}^0$

FOR $k = 0, 1, 2, \ldots,$ DO

1. Set $\mathbf{d}^k = \mathbf{f} - A\mathbf{x}^k$
2. Compute $\alpha_k = \dfrac{\mathbf{d}^k \cdot \mathbf{d}^k}{\mathbf{d}^k \cdot A\mathbf{d}^k}$
3. Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$

END DO



Method is very slow, especially when the "aspect ratio" is large !

Convergence of steepest descent

Define the **condition number** $\kappa(A)$ to be the ratio between the largest and smallest eigenvalues of A, i.e.,

$$\kappa(A) = \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)}$$

Theorem : the error $\mathbf{e}^k := \mathbf{x}^k - \mathbf{x}^*$ of the steepest descent method satisfies

$$(\mathbf{e}^k \cdot A\mathbf{e}^k)^{1/2} \leq \left(\frac{\kappa(A) - 1}{\kappa(A) + 1}\right)^k (\mathbf{e}^0 \cdot A\mathbf{e}^0)^{1/2}.$$

$\kappa(A)$	$(\kappa(A) - 1)/(\kappa(A) + 1)$	# its to reduce error by 10^6
1	0	1
2	0.3333	13
10	0.8182	69
100	0.9802	691
1000	0.9980	6908

How to fix steepest descent

- Problem with steepest descent :
  - the best descent direction is the one directly towards the solution
  - gradients do not generally point in the direction of the solution
  - directions tend to repeat and are not distinct enough
- Look again at the equation for determining step size :

$$\begin{aligned}\nabla J(\mathbf{x}^k + \alpha_k \mathbf{d}^k) \cdot \mathbf{d}^k &= 0 \\ (A(\mathbf{x}^k + \alpha_k \mathbf{d}^k) - \mathbf{f}) \cdot \mathbf{d}^k &= 0 \\ (A\mathbf{x}^{k+1} - A\mathbf{x}^*) \cdot \mathbf{d}^k &= 0 \\ A\mathbf{e}^{k+1} \cdot \mathbf{d}^k &= 0\end{aligned}$$

In 2D, the best direction is the one that's "A-orthogonal" to $\mathbf{d}^k$!

A-orthogonality

Instead of the usual dot product, we can define another **inner product** using A :

$$(\mathbf{u}, \mathbf{v})_A := A\mathbf{u} \cdot \mathbf{v} = \mathbf{u} \cdot A\mathbf{v}.$$

The usual properties of dot products also hold :

- $(\mathbf{u}, \mathbf{v})_A = (\mathbf{v}, \mathbf{u})_A$
- $(\mathbf{u}, \alpha\mathbf{v})_A = \alpha(\mathbf{u}, \mathbf{v})_A = \alpha(\mathbf{u}, \mathbf{v})_A$
- $(\mathbf{u}, \mathbf{v} + \mathbf{w})_A = (\mathbf{u}, \mathbf{v})_A + (\mathbf{u}, \mathbf{w})_A$
- $(\mathbf{u}, \mathbf{v})_A^2 \leq (\mathbf{u}, \mathbf{u})_A \cdot (\mathbf{v}, \mathbf{v})_A$ (Cauchy-Schwarz)
- $(\mathbf{u}, \mathbf{u})_A > 0$ for all $\mathbf{u} \neq 0$, and $\|\mathbf{u}\|_A = (\mathbf{u}, \mathbf{u})_A^{1/2}$ defines a norm

In this language, if we find a direction $\mathbf{d}^{k+1}$ that is orthogonal to $\mathbf{d}^k$ with respect to the A-inner product, then we get a method that converges in two steps in 2D !

An “improved” steepest descent

To solve : $A\mathbf{x} = \mathbf{f}$, A spd
Given : Initial guess $\mathbf{x}^0$

FOR $k = 0, 1, 2, \ldots,$ DO

1. Set $\mathbf{d}^k = \mathbf{f} - A\mathbf{x}^k$
2. Compute $\alpha_k = \dfrac{\mathbf{d}^k \cdot \mathbf{d}^k}{\mathbf{d}^k \cdot A\mathbf{d}^k}$
3. Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$

END DO

To solve : $A\mathbf{x} = \mathbf{f}$, A spd
Given : Initial guess $\mathbf{x}^0$

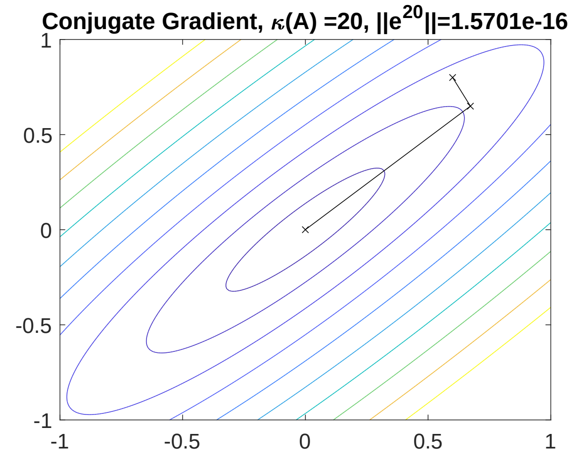
FOR $k = 0, 1, 2, \ldots,$ DO

1. Set $\mathbf{r}^k = \mathbf{f} - A\mathbf{x}^k$
2. Set $\mathbf{d}^k = \mathbf{r}^k - \displaystyle\sum_{j=0}^{k-1} \frac{(\mathbf{r}^k, \mathbf{d}^j)_A}{(\mathbf{d}^j, \mathbf{d}^j)_A} \mathbf{d}^j$ (Gram-Schmidt)
3. Compute $\alpha_k = \dfrac{\mathbf{r}^k \cdot \mathbf{d}^k}{\mathbf{d}^k \cdot A\mathbf{d}^k}$
4. Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$

END DO

“Improved” Steepest Descent = Conjugate Gradients

It works !



Krylov subspaces

To understand the behaviour of conjugate gradients, we study the following subspace :

Definition : Let $\mathbf{x}^0$ be given and $\mathbf{r}^0 = \mathbf{f} - A\mathbf{x}^0$. The $k+1$st **Krylov subspace** as

$$\mathcal{K}_{k+1}(\mathbf{r}^0) = \mathrm{Span}\{\mathbf{r}^0, A\mathbf{r}^0, \ldots, A^k\mathbf{r}^0\}.$$

Assuming that $\mathbf{d}^0, \ldots, \mathbf{d}^{k-1} \neq 0$, we can show recursively that $\mathbf{r}^k, \mathbf{d}^k \in \mathcal{K}_{k+1}(\mathbf{r}^0)$ using the updating formulas

$$\begin{aligned}
\mathbf{r}^k &= \mathbf{f} - A\mathbf{x}^k = \mathbf{f} - A(\mathbf{x}^{k-1} + \alpha_{k-1}\mathbf{d}^{k-1}) = \mathbf{r}^{k-1} - \alpha_{k-1}A\mathbf{d}^{k-1} \\
\mathbf{d}^k &= \mathbf{r}^k - \sum_{j=0}^{k-1} c_{kj}\mathbf{d}^j.
\end{aligned}$$

Moreover, Gram-Schmidt ensures that $\{\mathbf{d}^0, \ldots, \mathbf{d}^{k-1}\}$ is an A-orthogonal basis for $\mathcal{K}_k(\mathbf{r}^0)$.

Optimality

Theorem : Suppose $\mathbf{d}^0, \dots, \mathbf{d}^{k-1} \neq 0$. Then the $\mathbf{x}^k$ produced by Conjugate Gradients (CG) minimizes the error $\|\mathbf{e}^k\|_A = \|\mathbf{x}^k - \mathbf{x}^0\|_A$ over all elements of the set $\mathbf{x}^0 + \mathcal{K}_k(\mathbf{r}^0)$.

Proof :

1. $\mathbf{x}^k \in \mathbf{x}^0 + \mathcal{K}_k(\mathbf{r}^0)$, since $\mathbf{x}^k = \mathbf{x}^0 + \alpha_0 \mathbf{d}^0 + \cdots + \alpha_{k-1}\mathbf{d}^{k-1}$.
2. We show that $\mathbf{r}^k \cdot \mathbf{d}^j = 0$ for $j = 0, \dots, k-1$. For $j = k-1$, we have

$$\begin{aligned} \mathbf{r}^k \cdot \mathbf{d}^{k-1} &= (\mathbf{r}^{k-1} - \alpha_{k-1} A\mathbf{d}^{k-1}) \cdot \mathbf{d}^{k-1} \\ &= \mathbf{r}^{k-1} \cdot \mathbf{d}^{k-1} - \frac{\mathbf{r}^{k-1} \cdot \mathbf{d}^{k-1}}{\mathbf{d}^{k-1} \cdot A\mathbf{d}^{k-1}} (A\mathbf{d}^{k-1} \cdot \mathbf{d}^{k-1}) = 0. \end{aligned}$$

For $j \leq k-2$, we have

$$\begin{aligned} \mathbf{r}^k \cdot \mathbf{d}^j &= (\mathbf{r}^{k-1} - \alpha_{k-1} A\mathbf{d}^{k-1}) \cdot \mathbf{d}^j \\ &= \underbrace{\mathbf{r}^{k-1} \cdot \mathbf{d}^j}_{=0 \text{ by induction}} - \alpha_{k-1} \underbrace{A\mathbf{d}^{k-1} \cdot \mathbf{d}^j}_{=0 \text{ by } A\text{-orthog.}} = 0. \end{aligned}$$

Proof of optimality (cont'd.)

3. If $\hat{\mathbf{x}}$ is any other element in $\mathbf{x}^0 + \mathcal{K}_k(\mathbf{r}^0)$, then $\hat{\mathbf{x}} - \mathbf{x}^k \in \mathcal{K}_k(\mathbf{r}^0)$. Then

$$\begin{aligned}\|\hat{\mathbf{x}} - \mathbf{x}^*\|_A^2 &= \|\mathbf{e}^k + (\hat{\mathbf{x}} - \mathbf{x}^k)\|_A^2 \\ &= (\mathbf{e}^k, \mathbf{e}^k)_A + 2(\mathbf{e}^k, \hat{\mathbf{x}} - \mathbf{x}^k)_A + (\hat{\mathbf{x}} - \mathbf{x}^k, \hat{\mathbf{x}} - \mathbf{x}^k)_A \\ &= (\mathbf{e}^k, \mathbf{e}^k)_A + 2A\mathbf{e}^k \cdot (\hat{\mathbf{x}} - \mathbf{x}^k) + (\hat{\mathbf{x}} - \mathbf{x}^k, \hat{\mathbf{x}} - \mathbf{x}^k)_A.\end{aligned}$$

But $A\mathbf{e}^k = A(\mathbf{x}^k - \mathbf{x}^*) = A\mathbf{x}^k - \mathbf{f} = -\mathbf{r}^k$; since $\hat{\mathbf{x}} - \mathbf{x}^k$ is a linear combination of $\mathbf{d}^0, \ldots, \mathbf{d}^{k-1}$, the inner product in red vanishes. Thus, we have

$$\|\hat{\mathbf{x}} - \mathbf{x}^*\|_A^2 = \|\mathbf{e}^k\|_A^2 + \|\hat{\mathbf{x}} - \mathbf{x}^k\|_A^2 > \|\mathbf{e}^k\|_A^2.$$

So the error is minimized for $\mathbf{x}^k$.

Conclusion : CG always finds the best solution within the search space ! In particular, if the exact solution $\mathbf{x}^*$ lies in $\mathbf{x}^0 + \mathcal{K}_k(\mathbf{r}^0)$, then the method converges in k iterations.

Breakdown

Problem : CG cannot continue if $\mathbf{d}^k = 0$.

- If $\mathbf{d}^k = 0$, then

$$\mathbf{r}^k = \sum_{j=0}^{k-1} c_{kj}\mathbf{d}^j.$$

- But $\mathbf{r}^k \cdot \mathbf{d}^j = 0$ for $j = 0, \ldots, k-1$, so

$$\mathbf{r}^k \cdot \mathbf{r}^k = 0 \implies \mathbf{r}^k = 0.$$

- Breakdown only happens when $\mathbf{x}^k = \mathbf{x}^*$, i.e., when the method converges !

To solve : $A\mathbf{x} = \mathbf{f}$, A spd
Given : Initial guess $\mathbf{x}^0$

FOR $k = 0, 1, 2, \ldots,$ DO

1. Set $\mathbf{r}^k = \mathbf{f} - A\mathbf{x}^k$
2. Set $\mathbf{d}^k = \mathbf{r}^k - \sum_{j=0}^{k-1} \frac{(\mathbf{r}^k, \mathbf{d}^j)_A}{(\mathbf{d}^j, \mathbf{d}^j)_A}\mathbf{d}^j$
3. Compute $\alpha_k = \frac{\mathbf{r}^k \cdot \mathbf{d}^k}{\mathbf{d}^k \cdot A\mathbf{d}^k}$
4. Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$

END DO

Short recurrence

Problem : Gram-Schmidt is expensive.

To solve : $A\mathbf{x} = \mathbf{f}$, A spd
Given : Initial guess $\mathbf{x}^0$

FOR $k = 0, 1, 2, \ldots,$ DO

1. Set $\mathbf{r}^k = \mathbf{f} - A\mathbf{x}^k$
2. Set $\mathbf{d}^k = \mathbf{r}^k - \sum_{j=0}^{k-1} \frac{(\mathbf{r}^k, \mathbf{d}^j)_A}{(\mathbf{d}^j, \mathbf{d}^j)_A} \mathbf{d}^j$
3. Compute $\alpha_k = \dfrac{\mathbf{r}^k \cdot \mathbf{d}^k}{\mathbf{d}^k \cdot A\mathbf{d}^k}$
4. Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$

END DO

Short recurrence

Problem : Gram-Schmidt is expensive.

- We calculate

$$(\mathbf{r}^k, \mathbf{d}^j)_A = \mathbf{r}^k \cdot \underbrace{A\mathbf{d}^j}_{\in \mathcal{K}_{j+2}(\mathbf{r}^0)} = \mathbf{r}^k \cdot \sum_{\ell=0}^{j+1} \beta_\ell \mathbf{d}^\ell .$$

To solve : $A\mathbf{x} = \mathbf{f}$, A spd

Given : Initial guess $\mathbf{x}^0$

FOR $k = 0, 1, 2, \ldots,$ DO

1. Set $\mathbf{r}^k = \mathbf{f} - A\mathbf{x}^k$
2. Set $\mathbf{d}^k = \mathbf{r}^k - \sum_{j=0}^{k-1} \frac{(\mathbf{r}^k, \mathbf{d}^j)_A}{(\mathbf{d}^j, \mathbf{d}^j)_A} \mathbf{d}^j$
3. Compute $\alpha_k = \frac{\mathbf{r}^k \cdot \mathbf{d}^k}{\mathbf{d}^k \cdot A\mathbf{d}^k}$
4. Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$

END DO

Short recurrence

Problem : Gram-Schmidt is expensive.

- We calculate

$$(\mathbf{r}^k, \mathbf{d}^j)_A = \mathbf{r}^k \cdot \underbrace{A\mathbf{d}^j}_{\in \mathcal{K}_{j+2}(\mathbf{r}^0)} = \mathbf{r}^k \cdot \sum_{\ell=0}^{j+1} \beta_\ell \mathbf{d}^\ell.$$

- But $\mathbf{r}^k \cdot \mathbf{d}^0 = \cdots = \mathbf{r}^k \cdot \mathbf{d}^{k-1} = 0$. So if $j \leq k-2$, then $(\mathbf{r}^k, \mathbf{d}^j)_A = 0$.
- So there is only one non-zero orthogonalization term in the Gram-Schmidt !

To solve : $A\mathbf{x} = \mathbf{f}$, A spd
Given : Initial guess $\mathbf{x}^0$

FOR $k = 0, 1, 2, \ldots,$ DO

1. Set $\mathbf{r}^k = \mathbf{f} - A\mathbf{x}^k$
2. Set
$$\mathbf{d}^k = \mathbf{r}^k - \frac{(\mathbf{r}^k, \mathbf{d}^{k-1})_A}{(\mathbf{d}^{k-1}, \mathbf{d}^{k-1})_A} \mathbf{d}^{k-1}$$
3. Compute $\alpha_k = \dfrac{\mathbf{r}^k \cdot \mathbf{d}^k}{\mathbf{d}^k \cdot A\mathbf{d}^k}$
4. Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$

END DO

Standard form of CG

Further simplifications lead to the "standard form" (Hestenes & Stiefel 1952) :

To solve : $A\mathbf{x} = \mathbf{f}$, A spd
Given : Initial guess $\mathbf{x}^0$

Initialize : $\mathbf{d}^{-1} = 0$
FOR $k = 0, 1, 2, \ldots,$ DO

1. Set $\mathbf{r}^k = \mathbf{f} - A\mathbf{x}^k$
2. Set $\mathbf{d}^k = \mathbf{r}^k - \dfrac{(\mathbf{r}^k, \mathbf{d}^{k-1})_A}{(\mathbf{d}^{k-1}, \mathbf{d}^{k-1})_A}\mathbf{d}^{k-1}$
3. Compute $\alpha_k = \dfrac{\mathbf{r}^k \cdot \mathbf{d}^k}{\mathbf{d}^k \cdot A\mathbf{d}^k}$
4. Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$

END DO

$\Longrightarrow$

To solve : $A\mathbf{x} = \mathbf{f}$, A spd
Given : Initial guess $\mathbf{x}^0$

Initialize : $\mathbf{r}^0 = \mathbf{f} - A\mathbf{x}^0$, $\mathbf{d}^{-1} = 0$
FOR $k = 0, 1, 2, \ldots,$ DO

1. Set $\mathbf{d}^k = \mathbf{r}^k + \dfrac{\mathbf{r}^k \cdot \mathbf{r}^k}{\mathbf{r}^{k-1} \cdot \mathbf{r}^{k-1}}\mathbf{d}^{k-1}$
2. Compute $\alpha_k = \dfrac{\mathbf{r}^k \cdot \mathbf{r}^k}{\mathbf{d}^k \cdot A\mathbf{d}^k}$
3. Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$
4. Set $\mathbf{r}^{k+1} = \mathbf{r}^k - \alpha_k A\mathbf{d}^k$

END DO

Cost of CG

Each iteration of CG requires :

- One matrix-vector multiplication $A\mathbf{d}^k$
- Two inner products $\mathbf{r}^k \cdot \mathbf{r}^k$, $\mathbf{d}^k \cdot A\mathbf{d}^k$

In some applications, the multiplication $A\mathbf{d}^k$ can be done "matrix-free".

To solve : $A\mathbf{x} = \mathbf{f}$, A spd

Given : Initial guess $\mathbf{x}^0$

Initialize : $\mathbf{r}^0 = \mathbf{f} - A\mathbf{x}^0$, $\mathbf{d}^{-1} = 0$

FOR $k = 0, 1, 2, \ldots,$ DO

1. Set $\mathbf{d}^k = \mathbf{r}^k + \dfrac{\mathbf{r}^k \cdot \mathbf{r}^k}{\mathbf{r}^{k-1} \cdot \mathbf{r}^{k-1}} \mathbf{d}^{k-1}$
2. Compute $\alpha_k = \dfrac{\mathbf{r}^k \cdot \mathbf{r}^k}{\mathbf{d}^k \cdot A\mathbf{d}^k}$
3. Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$
4. Set $\mathbf{r}^{k+1} = \mathbf{r}^k - \alpha_k A\mathbf{d}^k$

END DO

Convergence of CG

- By the optimality property, we know that CG chooses $\mathbf{x}^k \in \mathbf{x}^0 + \mathcal{K}_k(\mathbf{r}^0)$ that minimizes the error $\|\mathbf{x}^k - \mathbf{x}^*\|_A$.
- If $\mathbf{x}^* \in \mathbf{x}^0 + \mathcal{K}_k(\mathbf{r}^0)$, then CG converges in k iterations.
- Assuming no breakdown until $k = n$, we have $\dim \mathcal{K}_n(\mathbf{r}^0) = n$, so $\mathcal{K}_n(\mathbf{r}^0) = \mathbb{R}^n \implies$ CG converges in at most n iterations (in exact arithmetic) !
- Another interpretation : at step k, CG minimizes

$$
\begin{aligned}
\|\mathbf{e}^k\|_A &= \|\mathbf{x}^0 - \mathbf{x}^* + (\text{lin. comb. of } \mathbf{d}^0, \mathbf{d}^1, \dots, \mathbf{d}^{k-1})\|_A \\
&= \|\mathbf{e}^0 + (\text{lin. comb. of } \mathbf{r}^0, A\mathbf{r}^0, \dots, A^{k-1}\mathbf{r}^0)\|_A \\
&= \|\mathbf{e}^0 + (\text{lin. comb. of } A\mathbf{e}^0, A^2\mathbf{e}^0, \dots, A^k\mathbf{e}^0)\|_A \\
&= \|p_k(A)\mathbf{e}^0\|_A,
\end{aligned}
$$

where $p_k(x)$ is a polynomial of degree $\leq k$ with $p_k(0) = 1$.

Convergence of CG

- In other words, CG chooses the best degree-k polynomial $p_k(x)$ in order to minimize $\|p_k(A)\mathbf{e}^0\|_A$!
- One can estimate the convergence of k steps of CG by choosing another (sub-optimal) polynomial $\hat{p}_k$ which is small inside the interval $[\lambda_{\min}, \lambda_{\max}]$:

Theorem : If $\mathbf{e}^k = \mathbf{x}^k - \mathbf{x}^*$ is the error of CG after k steps, then

$$\|\mathbf{e}^k\|_A \leq 2 \left(\frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} + 1} \right)^k \|\mathbf{e}^0\|_A,$$

where $\kappa(A) = \lambda_{\max}/\lambda_{\min}$ is the condition number of A.

Steepest Descent vs CG

Steepest Descent :

$\kappa(A)$	$(\kappa(A)-1)/(\kappa(A)+1)$	# its to reduce error by 10^6
1	0	1
2	0.3333	13
10	0.8182	69
100	0.9802	691
1000	0.9980	6908

Conjugate Gradients :

$\kappa(A)$	$(\sqrt{\kappa(A)}-1)/(\sqrt{\kappa(A)}+1)$	# its to reduce error by 10^6
1	0	1
2	0.1716	8
10	0.5195	21
100	0.8182	69
1000	0.9387	218

What about non-spd matrices ?

- Symmetric but indefinite : MINRES (minimum residual, Paige & Saunders 1975)
  - Short recurrence, constant cost per iteration
  - Minimizes residual with respect to the 2-norm
- Non-symmetric : GMRES (Generalized MINRES, Saad & Schultz 1986)
  - Minimizes residual with respect to the 2-norm
  - No short recurrence !
- Other solvers : BiCG(Stab), QMR, CGS, . . .
- For convergence results, see Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed., SIAM 2003.

Preconditioning

Preconditioning

- The convergence rates of CG, GMRES, etc. all depend on the condition number $\kappa(A)$.
- Idea of preconditioning : transform $A\mathbf{x} = \mathbf{f}$ into an equivalent problem :
  - Left preconditioning : $M^{-1}A\mathbf{x} = M^{-1}\mathbf{f}$
  - Right preconditioning :

$$AM^{-1}\mathbf{y} = \mathbf{f} \qquad \text{(with the substitution } M\mathbf{x} = \mathbf{y}\text{)}$$

  - Two-sided preconditioning :

$$M_L^{-1}AM_R^{-1}\mathbf{y} = M_L^{-1}\mathbf{f} \qquad \text{(with the substitution } M_R\mathbf{x} = \mathbf{y}\text{)}$$

- M is called the **preconditioner**
- Goal : transform the spectrum of A into something more friendly to CG, GMRES, etc.
- Ideally, $M \approx A$, so that $M^{-1}A \approx I$ has a small condition number
- M must be easy to solve $\implies$ trade-off !

Preconditioned CG

- Requires **spd** preconditioner M
- Example : for $A = D + L + L^T$,
  - Jacobi : $M = D$ ✓
  - Gauss-Seidel : $M = D + L$ ✗
  - *Symmetric* Gauss-Seidel :
    $M = (D+L)D^{-1}(D+L)^T$ ✓
- *Mathematically equivalent* to applying CG to

$$M^{-1/2}AM^{-1/2}\mathbf{y} = M^{-1/2}\mathbf{f}, \quad M^{1/2}\mathbf{y} = \mathbf{x}.$$

- Convergence depends on eigenvalues of $M^{-1/2}AM^{-1/2}$.
- No need to calculate $M^{-1/2}$! Only need to solve linear systems of the form $M\mathbf{z}^k = \mathbf{r}^k$.

To solve : $A\mathbf{x} = \mathbf{f}$, A spd
Given : Initial guess $\mathbf{x}^0$

Initialize : $\mathbf{r}^0 = \mathbf{f} - A\mathbf{x}^0$, $\mathbf{d}^{-1} = 0$
FOR $k = 0, 1, 2, \ldots,$ DO

1. Solve $M\mathbf{z}^k = \mathbf{r}^k$.
2. Set $\mathbf{d}^k = \mathbf{z}^k + \dfrac{\mathbf{r}^k \cdot \mathbf{z}^k}{\mathbf{r}^{k-1} \cdot \mathbf{z}^{k-1}}\mathbf{d}^{k-1}$
3. Compute $\alpha_k = \dfrac{\mathbf{r}^k \cdot \mathbf{z}^k}{\mathbf{d}^k \cdot A\mathbf{d}^k}$
4. Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$
5. Set $\mathbf{r}^{k+1} = \mathbf{r}^k - \alpha_k A\mathbf{d}^k$

END DO

Acceleration

- **Recall :** For a stationary method of the form

$$M\mathbf{x}^{k+1} = N\mathbf{x}^k + \mathbf{f},$$

  The error satisfies $\mathbf{e}^k = (M^{-1}N)^k\mathbf{e}^0 = (I - M^{-1}A)^k\mathbf{e}^0$.
- Letting $\tilde{\mathbf{e}}^k = M^{1/2}\mathbf{e}^k$, we see that the stationary method generates iterates that satisfy

$$\tilde{\mathbf{e}}^k = p_k(M^{-1/2}AM^{-1/2})\tilde{\mathbf{e}}^0,$$

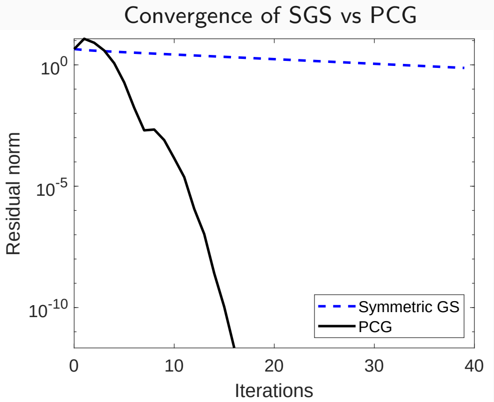
  where $p_k(x) = (1-x)^k$.
- But CG generates the **best** polynomial p_k that minimizes $\|\mathbf{e}^k\|_A$!
- Therefore, PCG is an **accelerator** for the basic stationary method defined by M.
- The same holds for GMRES, or any other method with an optimality property.

Stationary vs PCG acceleration

- Solve 20×20 system $A\mathbf{x} = \mathbf{f}$ with

$$A = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}$$

  and $\mathbf{f} = (1, 1, \ldots, 1)^T$
- Stationary : Symmetric Gauss-Seidel
- Krylov : CG preconditioned with SGS



Final remarks

- If you have a good stationary method, using it as a preconditioner within Krylov methods (PCG, GMRES, etc.) will make the convergence even faster.
- PCG and GMRES do not require $\rho(M^{-1}N) < 1$ to converge
- Some popular preconditioners (e.g. additive Schwarz with overlap) cannot be used as stationary methods (because $\rho(M^{-1}N) > 1$), but are still effective as preconditioners
- Preconditioner design is as much an art as it is a science, especially for multiphysics problems !