2 Iterative methods for linear systems of equations

We now consider what is maybe the most fundamental problem in scientific computing: Find a vector $x \in \mathbb{C}^n$ such that

$$Ax = b, \tag{2.1}$$

where $b \in \mathbb{C}^n$ is a given vector and $A \in \mathbb{C}^{n\times n}$ is a matrix. The matrix A is assumed to be large, sparse and non-singular. This chapter is about methods which are iterative in nature. In our setting this means that the method consists of a loop where, in every iteration, we try to improve an approximate solution to (2.1).

Different applications lead to matrices A with substantially different properties and structures. We cover several methods suitable for different properties and matrix structures.

This course block is about iterative methods. The other most important method class for (2.1) are *direct methods*. In contrast to iterative methods, direct methods are designed to determine an exact solution after a finite number of steps (in exact arithmetic). The most important direct method is Gaussian elimination which you have learned in basic linear algebra courses. Gaussian elimination is also the basis of methods for (2.1) by computing LU-factorizations.

Section 2.1: GMRES - Generalized Minimum Residual method

Section 2.2: CG - Conjugate Gradients method

Section 2.3: CGNE - Conjugate Gradients normal equations

Section 2.4: BiCG - BiConjugate gradients method

In many practical situations, these methods do not have satisfactory performance unless a specialized acceleration technique is applied. We learn about one of the acceleration techniques called *preconditioning* in Section 2.5.

2.1 GMRES - Generalized minimum residual method

The GMRES method is a method based on the idea that if the residual

$$r = A\tilde{x} - b$$

is small, $\tilde{x}$ is probably a good approximation of x. We try to minimize the norm of the residual (residual norm) over an appropriate space.

2.1.1 Derivation of GMRES

It turns out that if restrict our search for an approximation $\tilde{x}$ in a Krylov subspace, the minimizer of the residual norm can be elegantly and efficiently computed as a by-product of the Arnoldi method. We define the approximation x_m generated after m steps of GMRES as minimizers of the residual norm (with respect to the 2-norm) over the Krylov subspace associated with A and the right-hand side b:

$$\|Ax_m - b\|_2 := \min_{x\in\mathcal{K}_m(A,b)} \|Ax-b\|_2. \tag{2.2}$$

Why do we minimize over a Krylov subspace? Short answer: We do what we can, and we know how to compute an Arnoldi factorization. For problems of this size, you sometimes only have access to the matrix A via a matrix-vector product (which is often a complicated program). Arnoldi's method only involves the matrix A by a matrix-vector product.

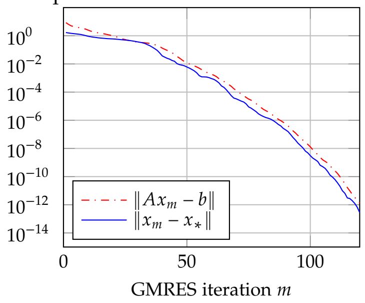
The GMRES-iterates are minimizers of the residual norm with respect to the two-norm over a Krylov subspace. In other methods, which we discuss later, we optimize over other sets, and use other norms.

We have seen earlier in this course that the Arnoldi method produces an Arnoldi factorization

$$AQ_m = Q_{m+1}\underline{H}_m \tag{2.3}$$

where Q_m is an orthogonal matrix and $\underline{H}_m$ a Hessenberg matrix. The following result shows how the solution to (2.2) can be directly computed if we have access to an Arnoldi factorization.

Residual norm vs norm of error: If the residual $A\tilde{x}-b$ is zero, the error $\tilde{x}-x_*$ is zero. Moreover, the relative residual norm is bounded by the relative error times the condition number of A, since $\frac{\|A\tilde{x}-b\|}{\|b\|} \le \|A\|\|A^{-1}\|\frac{\|\tilde{x}-x_*\|}{\|x_*\|}$. However, a small residual does not always imply that the error is small. It is however a common situation.
Example:



Lemma 2.1.1 (Minimization definition of GMRES iterates)**.** *Suppose Q_m and $\underline{H}_m$ satisfy the Arnoldi relation and $q_1 = b/\|b\|$. Then,*

$$\min_{x\in\mathcal{K}_m(A,b)} \|Ax-b\|_2 = \min_{z\in\mathbb{C}^n} \|\underline{H}_m z - \|b\|e_1\|_2. \tag{2.4}$$

Proof. During the proof we need the following property of orthogonal matrices. If $Q\in\mathbb{R}^{m\times k}$ with $m\ge k$ is an orthogonal matrix, then,

$$\|Qz\|_2^2 = z^TQ^TQz = z^Tz = \|z\|_2^2. \tag{2.5}$$

Since $\mathcal{K}_m(A,b) = \operatorname{span}(q_1,\ldots,q_m)$ we can reparameterize the set over which we minimize. The conclusion of the theorem follows from (2.3) and (2.5):

$$\begin{aligned}
\min_{x\in\mathcal{K}_m(A,b)} \|Ax-b\|_2 &= \min_{z\in\mathbb{C}^n} \|AQ_mz-b\|_2 \\
&= \min_{z\in\mathbb{C}^n} \|AQ_mz-\|b\|q_1\|_2 \\
&= \min_{z\in\mathbb{C}^n} \|Q_{m+1}\underline{H}_mz-\|b\|Q_{m+1}e_1\|_2 \\
&= \min_{z\in\mathbb{C}^n} \|Q_{m+1}(\underline{H}_mz-\|b\|e_1)\|_2 \\
&= \min_{z\in\mathbb{C}^n} \|\underline{H}_mz-\|b\|e_1\|_2
\end{aligned}$$

We start iteration with $q_1 = b/\|b\|$

Use the Arnoldi relation (2.3) and that $q_1 = Q_{m+1}e_1$.

Use (2.5) with $Q = Q_{m+1}$.

□

The approximations x_m are computed by solving the linear least-squares problem in the right-hand side of (2.4) and setting $x_m = Q_mz$.

The overdetermined linear system is of dimension $(m+1)\times m$ which is much smaller than size of the original matrix. We can solve it with any method for small dense systems, such as the one implemented in the backslash operator in MATLAB. Since it is small, computing this is typically much cheaper than other operations in the algorithm. By extending the Arnoldi method with a computation of a least squares solution in every iteration, leads to Algorithm 1.

```
Input: Matrix $A$ and vector $b$
Output: An approximate solution $\tilde{x}$ to the linear system $Ax = b$
Set $q_1 = b/\|b\|$, $H_0$ =empty matrix
for $m = 1, 2, \ldots$ do
    Compute $x = Aq_m$
    Orthogonalize $x$ against $q_1, \ldots, q_m$ by computing $h \in \mathbb{C}^m$ and
      $x_\perp \in \mathbb{C}^m$ such that $Q^T x_\perp = 0$ and
          $x_\perp = x - Qh.$
    Let $\beta = \|x_\perp\|$
    Let $q_{m+1} = x_\perp/\beta$
    Let
          $\underline{H}_m = \begin{bmatrix} \underline{H}_{m-1} & h \\ 0 & \beta \end{bmatrix}$
    Solve the overdetermined linear system by computing
      $z_* \in \mathbb{R}^n$ such that:
          $z_* = \underset{z \in \mathbb{R}^m}{\operatorname{argmin}} \|\underline{H}_m z - e_1 \|b\|\|$
    Compute approximate solution $\tilde{x} = Q_m z_*$
end
```

Algorithm 1: GMRES. Note that all steps except the last two steps are identical to the Arnoldi method.

When $\beta = 0$, GMRES has (so-called) break-down. This turns out to be not as dramatic as one might expect as we shall illustrate later.

2.1.2 *Convergence theory*

Finite termination of GMRES

The definition of a Krylov subspace implies that we add one vector at a time (unless we have break-down which corresponds to $\beta = 0$). After m steps, $\mathcal{K}_m(A, b)$ is therefore of dimension m and $\mathcal{K}_m(A, b) = \mathbb{C}^n$ and we try to minimize over the entire space. This means that after at most m steps, GMRES will terminate with an exact solution. GMRES is a method intended for very large problems, and in most practical situations, m steps of GMRES is computationally infeasable. It is our hope that the method generates a reasonable approximation much earlier than after m iterations.

Non-increasing residual norm

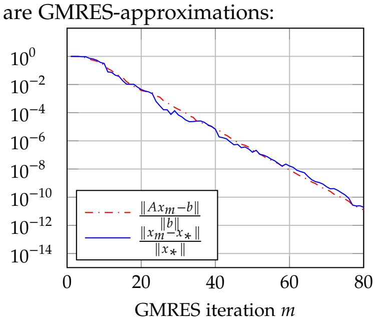
Due to the definition of GMRES-approximations as solutions to the minimization problem (2.2) we have a nice property: The solution can in a certain sense not become worse by further iteration. This is due to the fact that sequence of Krylov subspaces corresponds to an expanding set $\mathcal{K}_m(A,b) \subseteq \mathcal{K}_{m+1}(A,b)$, for any m. Therefore,

$$\|r_{m+1}\| = \min_{x\in\mathcal{K}_{m+1}(A,b)} \|Ax-b\| \leq \min_{x\in\mathcal{K}_m(A,b)} \|Ax-b\| = \|r_m\|.$$

Hence, if x_m is the GMRES-approximation at step m,

the norm of the residual vector $Ax_m - b$ is not increasing.

Note that a non-increasing residual norm does not imply a non-increasing error norm. A typical example where x_m are GMRES-approximations:



Convergence factor bound for diagonalizable matrices

Further analysis of convergence is simplified by the use polymomial sets.

Definition 2.1.2 (Polynomials and 0-normalized polynomials)**.**

$$P_m \quad := \quad \{\textit{polynomials of degree at most } m\} \tag{2.6a}$$
$$P_m^0 \quad := \quad \{p \in P_m : p(0) = 1\} \tag{2.6b}$$

With this polynomial set, we can express the residual corresponding to any Krylov approximation with a normalized polynomial.

Matrix polynomials. We here use the notation of matrix polynomials. If $p(z) = \alpha_0 + \cdots + \alpha_m z^n$ we define

$$p(A) := \alpha_0 I + \alpha_1 A + \cdots + \alpha_m A^m.$$

We will learn more about functions of matrices in block 4 of this course.

Lemma 2.1.3 (Krylov subspace equivalence)**.** *For any $A \in \mathbb{C}^{n\times n}$ and $b \in \mathbb{C}^n$,*

$$\left\{b - Ax : x \in \mathcal{K}_m(A,b)\right\} = \left\{p(A)b : p \in P_m^0\right\}. \tag{2.7}$$

Proof. The proof is based on the fact that

$$\mathcal{K}_m(A,b) = \{\alpha_0 b + \cdots + \alpha_{m-1}A^{m-1}b : \alpha_1,\ldots,\alpha_{m-1} \in \mathbb{C}\} = \{q(A)b : q \in P_{m-1}\}$$

By direct application to the left-hand side of (2.7) we have that

$$\left\{b - Ax : x \in \mathcal{K}_m(A,b)\right\} = \left\{b - Aq(A)b : q \in q \in P_{m-1}\right\}.$$

Note that $r = b - Aq(A)b$ for some $q \in P_{m-1}$ if and only if $r = p(A)b$ for some $p \in P_m^0$ since $p(z) = 1 - zq(z)$. Hence,

$$\left\{b - Aq(A)b : q \in q \in P_{m-1}\right\} = \left\{p(A)b : p \in p \in P_m^0\right\}. \qquad \square$$

Lemma 2.1.3 has a very compact notation. In words: If x is a vector in a Krylov subspace, then the corresponding residual $b - Ax$, can be expressed as $p(A)b$ where p is a normalized polynomial. The converse is also true.

Theorem 2.1.4 (Main convergence theorem of GMRES)**.** *Suppose $A \in \mathbb{C}^{n\times n}$ is an invertible and diagonalizable matrix. Let $A = V\Lambda V^{-1}$ be the Jordan decomposition of A, where Λ is a diagonal matrix. Let x_m, $n = 1,\ldots$ be iterates generated by GMRES. Then,*

$$\frac{\|Ax_m - b\|}{\|b\|} \leq \|V\|\|V^{-1}\| \min_{p\in P_m^0} \max_{i=1,\ldots,n} |p(\lambda_i)|.$$

GMRES convergence is here expressed with a min-max bound over the eigenvalues. There are more accurate min-max-characterizations of the convergence of GMRES, where instead of optimizing in the eigenvalues, the optimization set is the (so-called) pseudospectra.

Proof.

$$\begin{aligned}\|r_m\| &= \min_{x\in\mathcal{K}_m(A,b)} \|b-Ax\| \\ &= \min_{p\in P_m^0} \|p(A)b\| \\ &= \min_{p\in P_m^0} \|p(V\Lambda V^{-1})b\| \\ &= \min_{p\in P_m^0} \|Vp(\Lambda)V^{-1}b\| \\ &\le \min_{p\in P_m^0} \|V\|\|V^{-1}\|\|p(\Lambda)\|\|b\|.\end{aligned}$$

Use Lemma 2.1.3

Use Jordan decomposition

Use that for any polynomial $p(VBV^{-1}) = Vp(B)V^{-1}$.

Norm is submultiplicative

Since Λ is a diagonal matrix we have

$$p(\Lambda) = p(\begin{bmatrix}\lambda_1 & & \\ & \ddots & \\ & & \lambda_n\end{bmatrix}) = \begin{bmatrix}p(\lambda_1) & & \\ & \ddots & \\ & & p(\lambda_n)\end{bmatrix}. \tag{2.8}$$

Moreover, the two-norm of a diagonal matrix can be expressed explicitly. Since

$$\|\begin{bmatrix}\gamma_1 & & \\ & \ddots & \\ & & \gamma_n\end{bmatrix}\|_2^2 = \lambda_{\max}(\begin{bmatrix}\gamma_1 & & \\ & \ddots & \\ & & \gamma_n\end{bmatrix}\begin{bmatrix}\gamma_1 & & \\ & \ddots & \\ & & \gamma_n\end{bmatrix}^T) = \left(\max_{i=1,\ldots,n} |\gamma_i|\right)^2. \tag{2.9}$$

If we combine (2.8) and (2.9) we have

$$\|r_n\| \le \min_{p\in P_m^0} \max_{i=1,\ldots,m} \|V\|\|V^{-1}\||p(\lambda_i)|\|b\|,$$

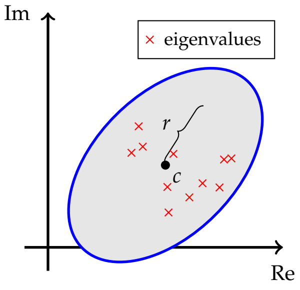
which concludes the proof. □

Corollary 2.1.5 (Single localization disk)**.** *Suppose $A\in\mathbb{C}^{n\times n}$ satisfies the same conditions as in Theorem 2.1.4. Moreover, suppose all eigenvalues are contained in a disk of radius r centered at $c\in\mathbb{C}$,*

$$\lambda_i \in \bar{D}(c,r), \textit{ for } i=1,\ldots,n.$$

Then,

$$\frac{\|Ax_m-b\|}{\|b\|} \le \|V\|\|V^{-1}\|\left(\frac{r}{|c|}\right)^m.$$



Proof. The result follows from Theorem 2.1.4 by considering the polynomial

$$q(z) := \frac{(c-z)^m}{c^m}. \tag{2.10}$$

Since $q\in P_m^0$, we have

$$\min_{p\in P_m^0}\max_{i=1,\ldots,m} |p(\lambda_i)| \le \max_{i=1,\ldots,n} |q(\lambda_i)| = \max_{i=1,\ldots,n} \frac{|c-\lambda_i|^m}{|c|^m} \le \frac{r^m}{|c|^m}.$$

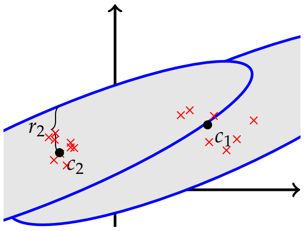
□

The polynomial (2.10) is sometimes called the *Zarantonello polynomial*. It is the minimizing polynomial over a disk in the sense that

$$\min_{p\in P_m^0}\max_{z\in\bar{D}(c,r)} |p(z)| = \max_{z\in\bar{D}(c,r)} |q(z)| = \left(\frac{r}{|c|}\right)^m$$

where $q\in P_m^0$ defined by (2.10).

Corollary 2.1.5 requires that the eigenvalues are contained in a disk and the bound is only useful if the disk does not include the origin. This type of relative localization is only a sufficient condition for fast convergence, and certainly not a necessary condition. For instance, if the eigenvalues are localized in other ways, we can still have fast convergence. The following corollary shows that if the eigenvalues are bounded in two small disks we can also have fast convergence.



Corollary 2.1.6 (Two localization disks)**.** *Suppose $A\in\mathbb{C}^{n\times n}$ satisfies the same conditions as in Theorem 2.1.4. Moreover, suppose all eigenvalues are contained in the union of two disks of radius r_1, r_2 centered at $c_1, c_2\in\mathbb{C}$,*

$$\lambda_i\in\bar{D}(c_1,r_1)\cup\bar{D}(c_2,r_2),\ \textit{for } i=1,\ldots,m.$$

Furthermore, suppose $r_1\ge r_2$ and assume that $\rho<1$ where

$$\rho:=\sqrt{\frac{r_1(r_1+|c_1-c_2|)}{|c_1||c_2|}}.$$

Then,

$$\frac{\|Ax_m-b\|}{\|b\|}\le\|V\|\|V^{-1}\|\,\rho^{m-1}$$

Proof. Let $k\in\mathbb{N}$ be $m/2$ rounded downwards such that $\frac{m-1}{2}\le k\le\frac{m}{2}$. That is, if m is even $k=m/2$ and $k=(m-1)/2$ if m is odd. We can then bound

$$\begin{aligned}\min_{p\in P_m^0}\max_{i=1,\ldots,n}|p(\lambda_i)|&\le\min_{p\in P_{2k}^0}\max_{\lambda\in\lambda(A)}|p(\lambda)|\\&=\min_{p\in P_{2k}^0}\max\left(\max_{\lambda\in\lambda(A)\cup\bar{D}(c_1,r_1)}|p(\lambda)|,\max_{\lambda\in\lambda(A)\cup\bar{D}(c_2,r_2)}|p(\lambda)|\right)\end{aligned}$$

We will bound the minimum with the specific polynomial $q\in P_{2k}^0$:

$$q(z):=\frac{(c_1-z)^k}{c_1}\frac{(c_2-z)^k}{c_2}.$$

Suppose $\lambda_i\in\bar{D}(c_1,r_1)$, then

$$|q(\lambda_i)|\le\frac{r_1^k}{|c_1|^k}\frac{(r_1+|c_1-c_2|)^k}{|c_2|^k}=\left(\frac{r_1(r_1+|c_1-c_2|)}{|c_1||c_2|}\right)^k$$

such that

$$\max_{\lambda\in\lambda(A)\cup D_1}|q(\lambda)|\le\left(\frac{r_1(r_1+|c_1-c_2|)}{|c_1||c_2|}\right)^k=\rho^{2k}.$$

On the other hand, if $\lambda_j\in\bar{D}(c_2,r_2)$, we analogously have that

$$|q(\lambda_j)|\le\left(\frac{r_2(r_2+|c_1-c_2|)}{|c_1||c_2|}\right)^k.$$

such that

$$\max_{\lambda\in\lambda(A)\cup D_1} |q(\lambda)| \le \left(\frac{r_2(r_2+|c_1-c_2|)}{|c_1||c_2|}\right)^k \le \left(\frac{r_1(r_1+|c_1-c_2|)}{|c_1||c_2|}\right)^k = \rho^{2k}.$$

Use theorem assumption: $r_1 \ge r_2$

Hence, by using that $2k > n-1$

$$\min_{p\in P_m^0} \max_{i=1,\dots,n} |p(\lambda_i)| \le \max_{i=1,\dots,n} |q(\lambda_i)| \le \rho^{2k} \le \rho^{n-1}.$$

Use that $\rho^z \le \rho^{z_1}$ when $\rho < 1$ and $z_1 \le z$.

□

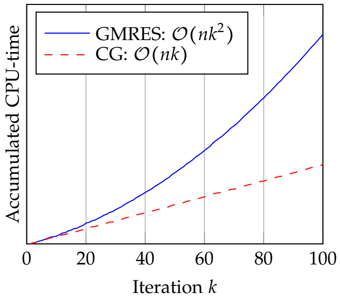
2.2 *Conjugate gradients (CG)*

One of the disadvantages of GMRES (and any method based on the Arnoldi method) is that the computation time associated with the orthogonalization grows with iteration. More precisely, in order to carry out k steps, the accumulated computation time for Gram-Schmidt orthogonalization is

$$t_{\text{GMRES,orth}} = \mathcal{O}(nk^2). \tag{2.11}$$

The quadratic dependence on k, makes it expensive to carry out many iterations. In this section and the following sections, we introduce some other methods based on Krylov subspaces which do not suffer from this problem.

The CG method has the nice feature that the computation-time per iteration is constant such that the accumulated computation-time is linear in the iteration count:



The method we study in this section (Conjugate Gradient method) is derived under the assumption:

We assume that the matrix A is symmetric and positive definite.

This will allow us to avoid the expensive orthogonalization in GMRES.

Definition of CG-iterates with residual minimization with respect to A-norm

The conjugate gradient method is tightly coupled with a somewhat unusual norm. If A is a symmetric positive definite matrix, the matrix can be used to define a weighted two-norm:

$$\|z\|_A = \sqrt{z^T Az}. \tag{2.12}$$

The relation (2.12) only defines a norm if A is symmetric positive definite. Note that A^{-1} is symmetric positive definite if A is symmetric positive definite, such that $\|\cdot\|_{A^{-1}}$ also defines a norm.

Analogous to GMRES, CG is a method which generates iterates that are minimizers of the residual. In contrast to GMRES, the residual norm is measured with respect to the A^{-1}-norm, which we will not need to compute but only use in the definition of the approximation.

Definition 2.2.1 (CG iterates)**.** *The CG-iterates for a matrix A are the minimizers of $\|Ax-b\|_{A^{-1}}$ over the mth Krylov subspace. That is, the CG-iterates $x_1, x_2, \dots$ satisfy*

$$\min_{x\in\mathcal{K}_m(A,b)} \|Ax-b\|_{A^{-1}} = \|Ax_m - b\|_{A^{-1}}, \quad m = 1,2,\dots \tag{2.13}$$

This definition can equivalently be reformulated as an orthogonality condition on the residual.

Lemma 2.2.2 (Optimization)**.** *The following statements are equivalent:*

(i) The approximation x_m is the minimizer of (2.13)

(ii) The residual is orthogonal to $\mathcal{K}_m(A,b)$, such that

$$r_m^T Q = 0 \tag{2.14}$$

where $r_m = b - Ax_m$ for some matrix Q such that $\operatorname{span}(Q) = \mathcal{K}_m(A,b)$.

Proof. We square both sides and reformulate the problem

$$\|Ax_m - b\|_{A^{-1}}^2 = \min_{x\in\mathcal{K}(A,b)} \|Ax-b\|_{A^{-1}}^2 = \min_{x\in\mathcal{K}(A,b)} (Ax-b)^T A^{-1}(Ax-b) =$$
$$\min_{z\in\mathbb{R}^m} (AQz-b)^T A^{-1}(AQz-b) = \min_{z\in\mathbb{R}^m} z^T Q^T AQz - 2b^T Qz + b^T A^{-1} b. \tag{2.15}$$

This is an unconstrained quadratic optimization problem. The matrix $Q^T AQ$ is symmetric positive definite since A is symmetric positive definite. The local optimality condition (corresponding to zero gradient), is therefore also the global optimality condition:

$$0 = (AQz-b)^T Q = r_m^T Q. \tag{2.16}$$

□

The derivation of (2.16) from (2.15) is based on the fact that the minimizer p of $c(p) = p^T B^T Bp + \alpha$ with respect to p for any α for any $B \in \mathbb{R}^{n\times m}$ with full columns span satisfies $Bp = 0$. This stems from the fact that the Hessian of $c(p)$ is $2Bp$.

CG orthogonality example

Before diving into the technical derivation of Algorithm 2 we illustrate the orthogonality of CG. If we generate a basis with the Arnoldi and let Q in (2.16) be the Q-matrix forming a basis of a Krylov subspace, the matrix Q is orthogonal to the residual.

The orthogonality of the residual against the Krylov subspace is in (2.16) is one of the main reasons to use the A^{-1}-norm, and will allow us to derive an efficient algorithm in the following section.

```
>> A=gallery('wathen',10,10); m=length(A);
>> b=ones(m,1);
>> m=5; % number of iterations
>> [x]=cg(A,b,m);    % Run n steps of CG
>> [Q,H]=arnoldi(A,b,m);
>> norm(Q(:,1:(m-1))'*(b-A*x)) % should vanish
ans =
   1.5249e-14
```

Derivation of CG from a low-term recurrence ansatz

Now let p_m denote a correction direction at step m and let α_m denote a scaling of the correction direction. In formulas,

$$x_m - x_{m-1} = \alpha_m p_{m-1}. \tag{2.17}$$

We shall later uniquely specify the scaling α_m.

The residual plays an important role in our derivation and we denote the residual associated with x_m:

$$r_m := b - Ax_m. \tag{2.18}$$

The correction of x_m in terms of p_m in (2.17) can also be interpreted as correction of the residual since $r_m = b - Ax_m = b - A(x_{m-1} + \alpha_m p_{m-1})$ such that

$$r_m = r_{m-1} - \alpha_m A p_{m-1}. \tag{2.19}$$

Our derivation is based on an ansatz. We make the following assumption on x_m and p_m which leads to an algorithm. The algorithm generates unique approximations x_m which we later show are minimizers in the sense of (2.13) by applying Lemma 2.2.2, thereby showing that the assumption is valid.

Assumption 2.2.3 (Short-term recurrence ansatz)**.** *We assume that there is a sequence of scalars α_m and β_m such that the search direction vector p_m is a linear combination of the previous search direction vector and the residual*

$$p_m = \beta_m p_{m-1} + r_m. \tag{2.20}$$

where r_m is defined by (2.18) and x_m defined by Definition 2.2.1

Note that except for the (not-yet-specified) scalars, α_m and β_m, the equations (2.17), (2.20), and (2.19) form an iteration if x_0 is given and we set $r_0 = b$. Moreover, at any point in the execution of the algorithm, only three vectors need to be stored: x_m, p_m and r_m. The algorithm is said to be a three-term recurrence method.

We now need to determine α_m and β_m. In order to simplify our notation, we define matrices with columns consisting of the vectors introduced above:

$$X := [x_1, \ldots, x_m], \quad R := [r_0, \ldots, r_{m-1}], \quad P := [p_0, \ldots, p_{m-1}]$$

The update formulas can be expressed with X, P and R by using the following transformation matrices

$$T := \begin{bmatrix} 1 & -1 & & \\ & \ddots & \ddots & \\ & & \ddots & -1 \\ & & & 1 \end{bmatrix}, \quad B := \begin{bmatrix} 1 & -\beta_1 & & \\ & \ddots & \ddots & \\ & & \ddots & -\beta_{m-1} \\ & & & 1 \end{bmatrix}, \quad D := \begin{bmatrix} \alpha_1 & & \\ & \ddots & \\ & & \alpha_m \end{bmatrix}.$$

The CG-method is commonly used in the field of optimization. The solution to a linear system $Ax = b$ where A is symmetric positive definite is equivalent to finding the (global) minimizer of the quadratic functional $c(q) = q^T A q - b^T q + \beta$. In that context the update vectors p_m are usually referred to as the gradient.

One justification in our reasoning for Assumption 2.2.3 is that we want a three-term recurrence algorithm, which means that we do not have to store more than three vectors at any point in time. With this assumption we reach an algorithm which only invoves x_m, r_m and p_m. In this course we learned about the Lanczos method, whic is another three-term recurrence method (but not directly applicable to linear systems).

The CG method is a short-term recurrence Krylov method. There are other short-term recurrence Krylov methods for symmetric matrices such as MINRES and SYMMLQ (not covered in this course). Among these methods CG is the most common choice, often justified by the fact that these methods have similar convergence, and CG requires the least number of floating point operations per iteration.

The relations (2.17), (2.20) and (2.19) are correspondingly

$$
\begin{aligned}
XT &= PD && (2.21)\\
PB &= R && (2.22)\\
APD &= RT^T - r_m e_m^T && (2.23)
\end{aligned}
$$

These relations, imply directly that each sequence of vectors form a basis of a Krylov subspace.

Lemma 2.2.4 (Krylov subspace span)**.** *Suppose $\alpha_1,\dots,\alpha_m$, $\beta_m,\dots,\beta_m$ are non-zero. Let $x_1,\dots,x_m$, $p_0,\dots,p_{m-1}$ $r_0,\dots,r_{m-1}$, be the vectors generated by (2.17), (2.20) and (2.19) with $x_0 = 0$ and r_0. Then,*

$$
\begin{aligned}
\mathcal{K}_m(A,b) &= \operatorname{span}(b, Ab, \dots, A^{m-1}b) && (2.24a)\\
&= \operatorname{span}(x_1,\dots,x_m) && (2.24b)\\
&= \operatorname{span}(p_0,\dots,p_{m-1}) && (2.24c)\\
&= \operatorname{span}(r_0,\dots,r_{m-1}). && (2.24d)
\end{aligned}
$$

Proof. From (2.22) and (2.21), we have directly that the columns of P, R and X span the same subspace, since B, T and D are non-singular matrices.

General property: If $W = VZ$ where Z is non-singular, then $\operatorname{span}(w_1,\dots,w_p) = \operatorname{span}(v_1,\dots,v_p)$.

In order to show that they span a Krylov subspace, suppose that the conclusion is satisfied for $j = 1,\dots,n-1$ and

$$\operatorname{span}(b, Ab, \dots, A^{m-1}b) = \operatorname{span}(R) = \operatorname{span}(P).$$

Then there exists an upper triangular matrix U such that $[b,\dots,A^{m-1}b] = PDU$, since D is non-singular. Hence, from (2.23) we have

$$
\begin{aligned}
&[b,\dots,A^m b] = [b, A[b,\dots,A^{m-1}b]] =\\
&\qquad [b, APDU] = [b, (RT^T - r_m e_m^T)U] = [R, r_m][e_1, (\underline{T}^T - e_{m+1}e_m^T)U]
\end{aligned}
$$

The matrix $[e_1, (\underline{T}^T - e_{m+1}e_m^T)U]$ is non-singular since it is upper triangular with non-zero diagonal elements. □

Orthogonality properties: Since x_m are defined as minimizers, the r_m vectors must satisfy the property (2.14). Moreover, the span of $r_0,\dots,r_{j-1}$ is Krylov subspace (due to equation (2.24d)), $r_i^T r_j$ for $j = 0,\dots,i-1$, or in matrix notation R^TR is a diagonal matrix

$$
R^TR = \begin{bmatrix} r_0^T r_0 & & \\ & \ddots & \\ & & r_{m-1}^T r_{m-1} \end{bmatrix}. \qquad (2.25)
$$

Hence

the residual vectors of CG are orthogonal.

By multiplying (2.23) from the left with P^T we have

$$P^TRT^T - P^Tr_me_m^T = P^TAPD$$

and therefore (from $P^Tr_m = 0$)

$$P^TRT^TD^{-1} = P^TAP. \tag{2.26}$$

From (2.25) and (2.22) we find that $P^TR = (R^TP)^T = (R^TRB^{-1})^T$ is a lower triangular matrix. Hence, the left-hand side of (2.26) is a product of lower triangular matrices (which is again an upper triangular matrix) and the right-hand side is a symmetric matrix (since A is symmetric). Therefore the matrix in (2.26) must be a diagonal matrix:

$$P^TAP = \begin{bmatrix} p_0^TAp_0 & & \\ & \ddots & \\ & & p_{m-1}^TAp_{m-1} \end{bmatrix} \tag{2.27}$$

The fact that the update vectors of the conjugate gradients satisfy an A-conjugacy condition (2.27) is the justification for its name. The p_m vectors (gradients) are *A-conjugate* (equivalently A-orthogonal).

In other words, $p_m^TAp_i = 0$ for $i = 0,\ldots,m-1$, and which in words is said

the update-vectors of CG are A-orthogonal.

Derivation of orthogonality and formulas for α_m and β_m. With some further analysis we can now establish explicit conditions on α_m and β_m.

Lemma 2.2.5 (Orthogonality of CG)**.** *Suppose x_m, p_m and r_m are generated by (2.17), (2.20) and (2.19), with scalar coefficients α_m and β_m which are selected such that for all m we have*

$$0 = r_{m-1}^Tr_{m-1} - \alpha_m p_{m-1}^TAp_{m-1} \tag{2.28a}$$
$$0 = p_{m-1}^TAp_{m-1}\alpha_m\beta_m - r_m^Tr_m \tag{2.28b}$$

and suppose $\alpha_1,\ldots,\alpha_m \neq 0$. Then, (2.25) and (2.27) are satisfied. That is, R^TR and P^TAP are diagonal.

Proof. The proof is done by induction, essentially by using the update relations for x_m, p_m and r_m in matrix notation. Suppose R^TR diagonal and P^TAP diagonal. We show that $R^Tr_m = 0$ and $P^TAp_m = 0$:

$$0 = R^Tr_m = R^Tr_{m-1} - \alpha_mR^TAp_{m-1} \tag{2.29a}$$
$$= e_mr_{m-1}^Tr_{m-1} - \alpha_mB^TP^TAp_{m-1} \tag{2.29b}$$
$$= e_mr_{m-1}^Tr_{m-1} - \alpha_me_mp_{m-1}^TAp_{m-1} \tag{2.29c}$$

Moreover,

$$\begin{aligned} 0 = P^TAp_m &= P^TAp_{m-1}\beta_m + P^TAr_m \\ &= e_mp_{m-1}^TAp_{m-1}\beta_m + (AP)^Tr_m \\ &= e_mp_{m-1}^TAp_{m-1}\beta_m + D^{-1}TR^Tr_m - D^{-1}e_mr_m^Tr_m \\ &= e_mp_{m-1}^TAp_{m-1}\beta_m - \frac{1}{\alpha_m}e_mr_m^Tr_m \quad \square \end{aligned}$$

The relations for α_m and β_m can be made explicit as follows. By solving (2.28a) for α_m we have

$$\alpha_m = \frac{r_{m-1}^T r_{m-1}}{p_{m-1}^T A p_{m-1}}. \tag{2.30}$$

Similarly, from (2.28b),

$$\beta_m = \frac{r_m^T r_m}{\alpha_m p_{m-1}^T A p_{m-1}} \tag{2.31a}$$

$$= \frac{r_m^T r_m}{r_{m-1}^T r_{m-1}} \tag{2.31b}$$

These choices of α_m and β_m can be combined into an algorithm which is commonly called the conjugate gradient method (Algorithm 2).

Corollary 2.2.6 (Ansatz is correct)**.** *If α_m and β_m are finite, the approximation x_m is the minimizer in sense of Definition 2.2.1.*

Proof. When selecting α_m and β_m according to (2.30) and (2.31) we clearly have that (2.29) is satisfied. Therefore, $Q^T r_m = 0$ with $Q = R$. The conclusion follows from Lemma 2.2.2, with $Q = R$ which satisfies $\mathcal{K}_m(A, b) = \operatorname{span}(R)$ according to Lemma 2.2.4 □

```
$x_0 = 0$, $r_0 = b$, $p_0 = r_0$
for $m = 1, 2, \ldots$ do
    $\alpha_m = \frac{r_{m-1}^T r_{m-1}}{p_{m-1}^T A p_{m-1}}$
    $x_m = x_{m-1} + \alpha_m p_{m-1}$
    $r_m = r_{m-1} - \alpha_m A p_{m-1}$
    $\beta_m = \frac{r_m^T r_m}{r_{m-1}^T r_{m-1}}$
    $p_m = r_m + \beta_m p_{m-1}$
end
```

Algorithm 2: Conjugate Gradient method (Hestenes and Stiefel variant)

Convergence of CG

Read TB pages 298-301. The proof of theorem TB Thm 38.5 is not a part of the course.

2.3 *Conjugate gradients normal equations (CGNE)*

In the previous section we illustrated that under the assumption that the matrix A is symmetric and positive definite, we can derive an algorithm which in a certain sense is better than GMRES. The attractive

CGNE is described in TB pages 304-305

feature is the short-term recurrence, and that the computation time per iteration is constant.

With the success of CG in mind, we now (in this section and the next section) approach the natural next question:

Are there short-term recurrence methods also for matrices which are not symmetric positive definite?

The short answer is yes. As usual, there is no free lunch and it comes at a price.

The idea of CGNE is simple. If we multiply $Ax = b$ with A^T we obtain

$$A^TAx = A^Tb \tag{2.32}$$

which is a linear system of equations $Bx = c$ with

$$B = A^TA$$

and $c = A^Tb$. Note that B is symmetric and positive definite. We just learned about CG which is a method for symmetric positive definite problems, and CGNE is based on applying CG on (2.32). The algorithm is identical to Algorithm 2, but $r_0 = b$ replaced with $r_0 = A^Tb$ and Ap_m replaced by $A^T(Ap_m)$.

2.3.1 *Computation cost CGNE*

Note that CGNE is a short-term recurrence method. The matrix vector products are often the computationally dominating part with short-term recurrence methods. Since, CGNE requires two matrix multiplications, CGNE is in a certain sense twice as expensive as CG.

2.3.2 *Relationship GMRES and CGNE*

We have seen that CG minimizes the residual with respect to a particular norm. Since CGNE is equivalent to CG with a particular matrix, CGNE also satisfies a minimization property. From (2.13) we find that the CGNE iterates satisfy

$$\|Bx_m - c\|_{B^{-1}} = \min_{x\in\mathcal{K}_m(B,c)} \|Bx - c\|_{B^{-1}}$$

The objective function in the right-hand side can be simplified further as follows

$$\begin{aligned}
\|Bx - c\|_{B^{-1}}^2 &= \|x - x_*\|_B^2 \\
&= (x - x_*)^TA^TA(x - x_*) \\
&= (Ax - Ax_*)^T(Ax - Ax_*) \\
&= (Ax - b)^T(Ax - b) = \|Ax - b\|_2^2.
\end{aligned}$$

Hence, CGNE minimizes the residual with respect to the standard Euclidean norm. Note that GMRES also minimizes the residual with respect to the standard Euclidean norm. However, a big difference is that GMRES minimizes the residual over the subspace

$$\mathcal{K}_m(A,b)$$

whereas CGNE minimizes the residual over the subspace

$$\mathcal{K}_m(A^TA,A^Tb).$$

These subspaces have very different approximation properties and most of the time CGNE subspace has worse approximation properties.

GMRES and CGNE both minimize the residual with respect to the standard Euclidean norm, but over different subspaces.

2.3.3 *Convergence of CGNE*

We can directly apply the convergence results for CG to CGNE and obtain the min-max bound

$$\frac{\|e_n\|_B}{\|e_0\|_B}\le\min_{p\in P_m^0}\max_{i=1,\ldots,n}|p(\lambda_i(A^TA))|.$$

Recall definition of singular values: $\sqrt{\lambda(A^TA)}$.

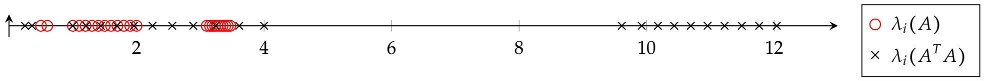
The only difference (in the right-hand side) in relation to CG, is that the maximization is with respect to the eigenvalues of A^TA instead of the eigenvalues of A. The square root of the eigenvalues of the matrix A^TA are also known as the singular values, and are often more spread out than the eigenvalues.

Except for some bounds such as $\max(|\lambda_i(A)|)<\max(|\lambda_i(A^TA)|)$, there are few relationships between the eigenvalues of the matrix and the singular values. Still, often the singular values squared are often more spread than the eigenvalues.

Example: Singular values vs eigenvalues

Although CGNE should not be applied to symmetric postive definite matrices we now do so to illustrate the difference. Suppose the matrix A is symmetric positive definite with eigenvalues as in the figure below. The eigenvalues squared are also given in the figure below. Clearly, the singular values squared are more spread out (less clustered) than the eigenvalues and CGNE therefore is expected to have slower convergence.

For symmmetric matrices, $\lambda_i(A^TA)=\lambda_i(A^2)=\lambda_i(A)^2$, such that the singular values squared equals the eigenvalues squared.



The direct application of the condition number bound for CG leads to a condition number bound for CGNE,

$$\frac{\|e_m\|_B}{\|e_0\|_B}\le 2\left(\frac{\sqrt{K(B)}-1}{\sqrt{K(B)}+1}\right)^m \tag{2.33}$$

where $K(B)$ is the condition number $K(B) := \|B\|\|B^{-1}\|$. From the relationship between singular values and the norm of a matrix one can show that

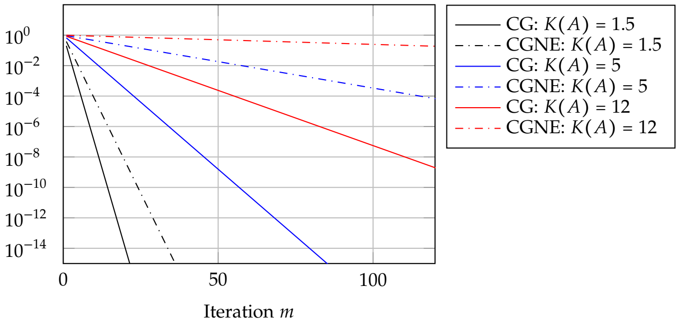
$$K(B) = \|B^{-1}\|_2\|B\|_2 = (\|A\|_2\|A^{-1}\|_2)^2 = K(A)^2 \tag{2.34}$$

The proof of (2.34) is based on the fact that Euclidean norm of a matrix is $\|A\|_2 = \sigma_{\max}(A)$ and $\|A^{-1}\|_2 = 1/\sigma_{\min}(A)$ where $\sigma_{\min}$ and $\sigma_{\max}$ are the smallest and largest singular value.

Therefore, (2.33) becomes

$$\frac{\|e_m\|_B}{\|e_0\|_B} \leq 2\left(\frac{K(A)-1}{K(A)+1}\right)^m$$

Note that since $K(A) > 1$ the condition number bound for CG will always be smaller than the condition number bound for CGNE. In fact, it is often considerably smaller. The impact of difference in the bounds can be seen in the figure below. Clearly, CGNE can only be expected to be competitive when $K(A)$ is not too large.



The improvement of the Biconjugate gradients method named BiCG-stab is the most used method to solve large and sparse linear systems of equations. The paper where it was published is the most cited paper in the field in the '90ies.

2.4 Biconjugate gradients method (BiCG)

GMRES is defined via a minimization problem. There is no minimization viewpoint of the algorithm we present now. However, GMRES can equivalently be stated using an orthogonality property, which we can appropriately modify to get a short-term recurrence method.

Theorem 2.4.1. *Suppose GMRES does not break-down at step m or earlier. Then, iterate m is the unique vector*

$$x_m \in \mathcal{K}_m(A,b)$$

such that

$$r_m^T A Q_m = 0 \tag{2.35}$$

where $r_m = b - Ax_m$.

Equation (2.35) is indeed an orthogonality condition. It can be directly written as an orthogonality to a Krylov subspace since Q_m is a basis of a Krylov subspace:

$$r_m \perp A\mathcal{K}_m(A,b).$$

Proof. We start by parameterizing the minimization definition of GMRES with the Q_m-matrix:

$$\min_{x\in\mathcal{K}_m(A,b)} \|Ax-b\| = \min_{z\in\mathbb{R}^m} \|AQz-b\|.$$

Now note that the right-hand side is a linear least squares problem, whose solution is explicitly given by the normal equations

$$(AQ_m)^T AQ_m z = (AQ_m)^T b$$

By rearranging the terms we obtain

$$(AQ_m)^T(AQ_m z - b) = 0$$

or equivalently

$$r_m^T AQ_m = 0.$$

Uniqueness follows from the fact that the linear least squares problem has a unique solution. □

```
$x_0 = 0$, $r_0 = b$, $p_0 = r_0$, $\hat{x}_0 = 0$, $\hat{r}_0 = b$, $\hat{p}_0 = \hat{r}_0$
for $m = 1, 2, \ldots$ do
    $\alpha_m = \frac{\hat{r}_m^T r_m}{\hat{p}_m^T A p_m}$
    $x_{m+1} = x_m + \alpha_m p_m$
    $\hat{x}_{m+1} = \hat{x}_m + \alpha_m \hat{p}_m$
    $r_{m+1} = r_m - \alpha_m A p_m$
    $\hat{r}_{m+1} = \hat{r}_m - \alpha_m A^T \hat{p}_m$
    $\beta_m = \frac{\hat{r}_{m+1}^T r_{m+1}}{\hat{r}_m^T r_m}$
    $p_{m+1} = r_{m+1} + \beta_m p_m$
    $\hat{p}_{m+1} = \hat{r}_{m+1} + \beta_m \hat{p}_m$
end
```

Algorithm 3: Bi-conjugate Gradient method

The Bi-Conjugate gradient method can be derived from orthogonality relations similar to the theorem above. However, we use two subspaces

$$\mathcal{K}_m(A, r_0) = \operatorname{span}(r_0, Ar_0, \ldots, A^{m-1}r_0) \quad (2.36)$$

$$\mathcal{K}_m(A^T, \hat{r}_0) = \operatorname{span}(\hat{r}_0, A^T\hat{r}_0, \ldots, A^{m-1}\hat{r}_0) \quad (2.37)$$

In general the form a basis of a Krylov subspace and

$$\operatorname{span}(r_0, \ldots, r_{m-1}) = \mathcal{K}_m(A, r_0) \quad (2.38)$$

$$\operatorname{span}(\hat{r}_0, \ldots, \hat{r}_{m-1}) = \mathcal{K}_m(A^T, \hat{r}_0) \quad (2.39)$$

However, the iterates are not orthogonal, but instead bi-orthogonal

$$\hat{r}_i^T r_j = 0 \quad (2.40a)$$

$$\hat{p}^T A p_j = 0 \quad (2.40b)$$

Read TB pages 305-309

2.5 Preconditioning

Read TB pages 313-314, and video on preconditioning.

Further reading

- GMRES and CG can be initiated with a different starting vector.
- Different convergence bounds based on pseudospectra, etc
- Flexible GMRES provides a way to use different preconditioners in each step
- Floating point arithmetic has substantial impact on the convergence of CG
- Many problem-specific ways to carry out preconditioning