

Theorem 7.6. Estimate of the rate of convergence for the CG method. *Let A be symmetric and positive definite with $\lambda_{\min} < \lambda_{\max}$, then*

$$\min_{p_k \in P_k, p_k(0)=1} \|p_k(A)\|_2 \leq 2 \left(\frac{\sqrt{\kappa_2(A)} - 1}{\sqrt{\kappa_2(A)} + 1} \right)^k. \quad (7.9)$$

Consequently, it is for the CG method

$$\frac{\|\underline{x} - \underline{x}^{(k)}\|_A}{\|\underline{x} - \underline{x}^{(0)}\|_A} \leq 2 \left(\frac{\sqrt{\kappa_2(A)} - 1}{\sqrt{\kappa_2(A)} + 1} \right)^k. \quad (7.10)$$

Proof. The idea of the proof consists in constructing a special polynomial which gives the estimate since

$$\min_{p_k \in P_k, p_k(0)=1} \|p_k(A)\|_2 \leq \|p_{k, \text{special}}(A)\|_2.$$

Let λ_{\min} be the smallest and λ_{\max} be the largest eigenvalue of A . Consider the linear function

$$\lambda : \mathbb{R} \rightarrow \mathbb{R}, \quad t \mapsto \frac{\lambda_{\min} + \lambda_{\max}}{2} + \frac{\lambda_{\max} - \lambda_{\min}}{2} t.$$

In particular, the restriction $t \in [-1, 1]$ gives $\lambda \in [\lambda_{\min}, \lambda_{\max}]$. The root of $\lambda(t)$ is denoted by t_0 . It is

$$t_0 = -\frac{\lambda_{\min} + \lambda_{\max}}{\lambda_{\max} - \lambda_{\min}} = -\frac{\kappa_2(A) + 1}{\kappa_2(A) - 1} < -1, \quad (7.11)$$

where one uses that for symmetric positive definite matrices that $\kappa_2(A) = \lambda_{\max}/\lambda_{\min}$. Denoting by $t(\lambda)$ the inverse function, one defines the special polynomial

$$p_k(\lambda) = \frac{T_k(t(\lambda))}{T_k(t(0))} =: \frac{T_k(t)}{T_k(t_0)} \in P_k.$$

Then, it is $p_k(0) = T_k(t_0)/T_k(t_0) = 1$. It is by Lemma 7.4 and since $\lambda \in [\lambda_{\min}, \lambda_{\max}]$ for all eigenvalues of A (the maximum does not decrease if it is searched in a larger set)

$$\begin{aligned} \|p_k(A)\|_2 &= \max_{\lambda \text{ is eigenvalue of } A} |p_k(\lambda)| \leq \max_{\lambda \in [\lambda_{\min}, \lambda_{\max}]} |p_k(\lambda)| = \max_{t \in [-1, 1]} \frac{|T_k(t)|}{|T_k(t_0)|} \\ &= \frac{1}{|T_k(t_0)|} \underbrace{\max_{t \in [-1, 1]} |T_k(t)|}_{\leq 1} \leq \frac{1}{|T_k(t_0)|}. \end{aligned} \quad (7.12)$$

For estimating this term, consider (7.8) since $t_0 < -1$:

$$|T_k(t_0)| = \left| (-1)^k \cosh(k \underbrace{\text{arcosh}(-t_0)}_{\omega_0}) \right| = |\cosh(k\omega_0)| = \frac{e^{k\omega_0} + e^{-k\omega_0}}{2}.$$

One has to estimate this term from below. Consider first the case $k = 1$. Since $-t_0 > 1$, one has, by applying the inverse function,

$$\frac{e^{\omega_0} + e^{-\omega_0}}{2} = \cosh(\omega_0) = \cosh(\text{arcosh}(-t_0)) = -t_0,$$

from what $e^{\omega_0} + e^{-\omega_0} = -2t_0$ follows. This equation is quadratic in e^{ω_0} with the solution

$$e^{\omega_0} = \underbrace{-t_0}_{>1} \pm \sqrt{t_0^2 - 1}.$$

For estimating $|T_k(t_0)|$, one obtains a sharper estimate if the larger one of these two values is considered, see (7.13) below. One gets with (7.11) and the binomial theorem

$$\begin{aligned}
e^{\omega_0} &= -t_0 + \sqrt{t_0^2 - 1} = \frac{\kappa_2(A) + 1}{\kappa_2(A) - 1} + \sqrt{\frac{(\kappa_2(A) + 1)^2 - (\kappa_2(A) - 1)^2}{(\kappa_2(A) - 1)^2}} \\
&= \frac{\kappa_2(A) + 2\sqrt{\kappa_2(A)} + 1}{\kappa_2(A) - 1} = \frac{(\sqrt{\kappa_2(A)} + 1)^2}{(\sqrt{\kappa_2(A)} + 1)(\sqrt{\kappa_2(A)} - 1)} = \frac{\sqrt{\kappa_2(A)} + 1}{\sqrt{\kappa_2(A)} - 1}.
\end{aligned}$$

Now, $|T_k(t_0)|$ is estimated from below

$$|T_k(t_0)| = \frac{e^{k\omega_0} + e^{-k\omega_0}}{2} > \frac{e^{k\omega_0}}{2} = \frac{(e^{\omega_0})^k}{2} = \frac{1}{2} \left(\frac{\sqrt{\kappa_2(A)} + 1}{\sqrt{\kappa_2(A)} - 1} \right)^k. \quad (7.13)$$

Inserting this estimate in (7.12) finishes the proof of (7.9).

Estimate (7.10) is obtained by inserting (7.9) in (7.5). ■

Remark 7.7. The case $\lambda_{\min} = \lambda_{\max} = \lambda$. From Remark 2.15, it follows that for $\lambda_{\min} = \lambda_{\max} = \lambda$

$$A = Q^T \lambda I Q = \lambda Q^T Q = \lambda I,$$

i.e., A is a multiple of the identity matrix. In this case, the linear system of equations can be solved directly, without using the CG method. Choosing $p_1(x) = -x/\lambda + 1$, then it is $p_1(0) = 1$, $p_1(\lambda) = 0$ and consequently $\|p_1(A)\|_2 = 0$, see Lemma 7.4. That means, the CG method converges in one iteration. □

Remark 7.8. Connection of the number of iterations and the spectral condition number. To guarantee the reduction of the error by a factor $0 < \eta < 1$ on the basis of (7.5) and estimate (7.9) from Theorem 7.6, the condition

$$2 \left(\frac{\sqrt{\kappa_2(A)} - 1}{\sqrt{\kappa_2(A)} + 1} \right)^k \leq \eta$$

must be satisfied. The number of iterations to achieve this condition is

$$k \geq \frac{|\ln(\eta/2)|}{\left| \ln \left(\frac{\sqrt{\kappa_2(A)} - 1}{\sqrt{\kappa_2(A)} + 1} \right) \right|} = \frac{-\ln(\eta/2)}{\ln \left(\frac{\sqrt{\kappa_2(A)} + 1}{\sqrt{\kappa_2(A)} - 1} \right)}.$$

If $\kappa_2(A)$ is large, then a power series expansion of the logarithm

$$\ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \dots,$$

gives, using only the linear term,

$$\begin{aligned}
\ln \left(\frac{\sqrt{\kappa_2(A)} + 1}{\sqrt{\kappa_2(A)} - 1} \right) &= \ln \left(\frac{1 + \frac{1}{\sqrt{\kappa_2(A)}}}{1 - \frac{1}{\sqrt{\kappa_2(A)}}} \right) \\
&= \ln \left(1 + \frac{1}{\sqrt{\kappa_2(A)}} \right) - \ln \left(1 - \frac{1}{\sqrt{\kappa_2(A)}} \right) \\
&\approx \frac{2}{\sqrt{\kappa_2(A)}}.
\end{aligned}$$

That means, an approximation of the upper bound of the number of iterations to reduce the error by the factor η is

$$k \approx \frac{-\ln(\eta/2)}{2} \sqrt{\kappa_2(A)}.$$

The dependency on $\mathcal{O}\left(\sqrt{\kappa_2(A)}\right)$ can be observed in fact sometimes. However, often the convergence of the CG method is considerably faster than predicted by the upper bound (7.10). One can derive better and sharper bounds if the distribution of all eigenvalues is considered instead only the smallest and the largest one, which are needed for computing the spectral condition number. \square

Remark 7.9. Round-off errors. For studying the behavior of the CG method in practice, one has to take into account in the analysis the round-off errors that are committed due to the finite precision arithmetic. The accumulation of round-off errors might lead to an increasing loss of the property of the computed vectors to be A -conjugate. Then, the computed solution might be only a quite inaccurate approximation of the solution of (1.1). \square

Chapter 8

Preconditioning

8.1 The General Approach

Remark 8.1. Motivation and idea. It was seen in Chapter 7 that the number of iterations might depend on the condition number of the matrix. In order to reduce the number of iterations, one wants to replace the original linear system of equations (1.1) by an equivalent system whose system matrix has a smaller condition number. This strategy is called preconditioning.

The main idea of preconditioning consists in applying the iterative method to the equivalent system

$$M^{-1}Ax = M^{-1}b \quad (\text{preconditioning from left})$$

or

$$AM^{-1}y = b, \quad x = M^{-1}y \quad (\text{preconditioning from right}).$$

The non-singular matrix M is called preconditioner. This matrix should satisfy two requirements:

- The convergence of the iterative method for the system with the matrix $M^{-1}A$ or AM^{-1} , respectively, should be faster than for the original system with the matrix A . That means, M^{-1} should be a good approximation to A^{-1} .
- Linear systems with the matrix M should be solvable with low costs.

In general, one has to find a compromise between these two requirements.

Usually, left and right preconditioning lead to different methods which might behave sometimes quite differently. \square

Remark 8.2. Some preconditioners. An easy way to construct preconditioners consists in starting with the decomposition $A = D + L + U$, see Section 3.2, and using parts of this decomposition which are easily invertible:

- $M = D$, diagonal preconditioner, Jacobi preconditioner,
- $M = D + L$, forward Gauss–Seidel preconditioner,
- $M = D + U$, backward Gauss–Seidel preconditioner,
- $M = (D + L)D^{-1}(D + U)$, symmetric Gauss–Seidel preconditioner.

Damped versions of the classical iterative schemes can be also used. A more advanced preconditioner will be presented in Section 8.3.

Note that M or M^{-1} do not need to be known explicitly. They can also stand for some numerical (iterative) method for solving linear systems of equations. Then, M^{-1} means that this method should be applied to a vector. \square

Remark 8.3. Change in algorithms for general matrices if the preconditioner is applied. In algorithms for general matrices A , preconditioning from left consists in replacing A by $M^{-1}A$ and $\underline{r}^{(k)}$ by $M^{-1}\underline{r}^{(k)}$ in the algorithms. Then, e.g., GMRES computes the iterate

$$\underline{x}^{(k)} \in \underline{x}^{(0)} + K_k \left(M^{-1} \underline{r}^{(0)}, M^{-1} A \right)$$

such that $\left\| M^{-1} \underline{r}^{(k)} \right\|_2$ becomes minimal. □

8.2 Symmetric Matrices

Remark 8.4. A difficulty and its solution. A problem occurs if the matrix A is symmetric and the iterative method wants to exploit this property, e.g., using short recurrences, since in general neither $M^{-1}A$ nor AM^{-1} are symmetric. This problem can be solved by constructing the orthonormal basis of the Krylov subspace with respect to an appropriate inner product.

Let H be a Hilbert¹ space with the inner product $(\cdot, \cdot)_H$ and $\mathcal{L} : H \rightarrow H$ be a linear map. This map is called self-adjoint with respect to $(\cdot, \cdot)_H$ if

$$(\mathcal{L}v, w)_H = (v, \mathcal{L}w)_H \quad \forall v, w \in H.$$

In the case $H = \mathbb{R}^n$ equipped with the standard Cartesian basis and the Euclidean inner product (\cdot, \cdot) , a linear map, which is represented by a matrix A , is self-adjoint if

$$(A\underline{x}, \underline{y}) = (\underline{x}, A\underline{y}) \quad \forall \underline{x}, \underline{y} \in \mathbb{R}^n.$$

This condition is equivalent to A being symmetric.

If the preconditioner M is symmetric and positive definite, then

$$(\underline{x}, \underline{y})_M = (\underline{x}, M\underline{y}), \quad \forall \underline{x}, \underline{y} \in \mathbb{R}^n$$

defines an inner product in \mathbb{R}^n . The induced norm is given by $\|\underline{x}\|_M = (\underline{x}, \underline{x})_M^{1/2}$.

Consider for the remainder of this section preconditioning from left. The matrix $M^{-1}A$ is self-adjoint with respect to this inner product since

$$(M^{-1}A\underline{x}, \underline{y})_M = (M^{-1}A\underline{x}, M\underline{y}) = (A\underline{x}, \underline{y}) = (\underline{x}, A\underline{y}) = (\underline{x}, M^{-1}A\underline{y})_M$$

for all $\underline{x}, \underline{y} \in \mathbb{R}^n$.

Now, one can generate an orthonormal basis with respect to the inner product $(\cdot, \cdot)_M$ of $K_k \left(M^{-1} \underline{r}^{(0)}, M^{-1} A \right)$ by an appropriate modification of the Lanczos algorithm. □

Algorithm 8.5. Preconditioned Lanczos algorithm for symmetric matrices. Given a symmetric matrix $A \in \mathbb{R}^{n \times n}$, a symmetric positive definite matrix $M \in \mathbb{R}^{n \times n}$, and $\underline{r}^{(0)} \in \mathbb{R}^n$.

1. $\underline{z} = M^{-1} \underline{r}^{(0)}$
2. $\underline{q}_1 = \frac{\underline{z}}{(\underline{r}^{(0)}, \underline{z})^{1/2}}$
3. $\beta_0 = 0$
4. $\underline{q}_0 = \underline{0}$
5. **for** $j = 1 : k$
6. $\underline{s} = A\underline{q}_j$
7. $\underline{z} = M^{-1} \underline{s}$
8. $\alpha_j = (\underline{s}, \underline{q}_j)$
9. $\underline{z} = \underline{z} - \alpha_j \underline{q}_j - \beta_{j-1} \underline{q}_{j-1}$

¹ David Hilbert (1862 – 1943)