Chapter 3

Stationary iterative methods

In this chapter, we are introducing *iterative methods* to solve the linear system

$$Ax_* = b, \tag{3.1}$$

where $A \in \mathbb{C}^{n\times n}$ is a matrix assumed to be invertible, $x_* \in \mathbb{C}^n$ is the vector of unknowns and $b \in \mathbb{C}^n$ is the right-hand side vector.

The general idea of iterative methods is to define a sequence of vectors $(x^{(k)})_{k\in\mathbb{N}}$ such that $\lim x^{(k)} = x_*$. In this regard, there are two big families of iterative methods that can be distinguished:

1. the stationary iterative methods where $(x^{(k)})$ is defined by

   $$x^{(k+1)} = Gx^{(k)} + v,$$

   where $G \in \mathbb{C}^{n\times n}$ is a certain iteration matrix and $v \in \mathbb{C}^{n\times n}$;

2. Krylov subspace methods where for all n, $x^{(k)} \in \operatorname{Span}_{0\leq j\leq k-1}(A^j b)$.

The advantage of using an iterative solver instead of a Gaussian elimination process relies on the following observation: the Gaussian elimination algorithm requires $\mathcal{O}(n^3)$ operations in order to compute x_*. This quickly becomes untractable. The iterative methods on the other hand only requires matrix-vector multiplication whose cost scales as $\mathcal{O}(n^2)$ for dense matrices. *If* the iterative method converges quickly, an approximate solution can be computed using $\mathcal{O}(kn^2)$ where k is the number of steps of the iterative method. With an efficient iterative method, it is possible to gain a factor n in the resolution of the linear system. Of course, the central question in these iterative methods is the convergence and the speed of convergence of these algorithms.

3.1 Principle of stationary iterative methods

The general framework of this type of methods is to define a splitting of the matrix $A = M - N$, where $M, N \in \mathbb{C}^{n\times n}$ and define the stationary iterative method by

$$\begin{cases} x_0 \in \mathbb{C}^n \\ Mx^{(k+1)} = Nx^{(k)} + b, \quad k \geq 1. \end{cases} \tag{3.2}$$

If the sequence $(x^{(k)})$ converges to a vector x_∞, then $Mx_\infty = Nx_\infty + b$ hence $Ax_\infty = b$. Thus the limit solves the linear system (3.1).

To study the convergence of the sequence $(x^{(k)})$, we see that $Mx_* = Nx_* + b$, so $x^{(k)} - x_*$ satisfies

$$x^{(k+1)} - x_* = M^{-1}Nx^{(k)} - M^{-1}b - x_* = M^{-1}N(x^{(k)} - x_*). \tag{3.3}$$

Hence the convergence of the sequence $(x^{(k)})$ is governed by the spectral properties of the matrix $M^{-1}N$.

Theorem 3.1 (Convergence of stationary iterative methods)**.** *Let $A \in \mathbb{C}^{n\times n}$ be invertible, $b \in \mathbb{C}^n$ and $x_* = A^{-1}b$. The sequence $(x^{(k)})_{k\geq 0}$ defined by Equation* (3.2) *converges to x_* for any $x_0 \in \mathbb{C}^n$ if and only if $\rho(M^{-1}N) < 1$, where $\rho(M^{-1}N) = \max\{|\lambda|, \lambda$ eigenvalue of $M^{-1}N\}$.*

Proof: If $\rho(M^{-1}N) < 1$ then there is an induced matrix norm $\|\cdot\|$ by a vector norm such that $\|M^{-1}N\| < 1$. Thus we have

$$\|x^{(k)} - x_*\| \leq \|M^{-1}N(x^{(k-1)} - x_*)\| \leq \|M^{-1}N\|\|N(x^{(k-1)} - x_*)\| \leq \|M^{-1}N\|^k\|x_0 - x_*\|.$$

Thus $\lim x^{(k)} = x_*$.

On the other hand if $\rho(M^{-1}N) \geq 1$ then there is an eigenvector $y \in \mathbb{C}^n$ of $M^{-1}N$ such that $\|(M^{-1}N)^k y\| = \rho(M^{-1}N)^k\|y\|$ does not converge to 0 as k goes to infinity.

It remains to choose the matrix M in a wise manner, such that at each step the inversion of M has a cost comparable to a matrix-vector product.

3.2 Classical iterative methods

To define the methods, we introduce the following notation $D, E, F \in \mathbb{C}^{n\times n}$ such that $A = D - E - F$ with

$$D = \begin{bmatrix} a_{11} & 0 & \dots & 0 \\ 0 & a_{22} & \ddots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & a_{nn} \end{bmatrix}, \quad -E = \begin{bmatrix} 0 & 0 & \dots & 0 \\ a_{21} & 0 & \ddots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ a_{n1} & \dots & a_{n,n-1} & 0 \end{bmatrix}, \text{ and } -F = \begin{bmatrix} 0 & a_{12} & \dots & a_{1n} \\ 0 & 0 & \ddots & a_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 0 \end{bmatrix}.$$

Jacobi method

For the Jacobi method, we set $M = D$ and $N = E + F$. In that case, the i-th entry of the vector $x^{(k)}$ is given by

$$x_i^{(k)} = \frac{b_i - \sum_{j\neq i} a_{ij}x_j^{(k-1)}}{a_{ii}},$$

hence this can be updated in parallel.

Proposition 3.2. *If A is row-wise diagonally dominant,* i.e. *for each $1 \leq i \leq n$, $|a_{ii}| > \sum_{j\neq i} |a_{ij}|$, then the Jacobi method converges.*

Proof: We simply need to check that the spectral radius $\rho(M^{-1}N) < 1$. For $y \in \mathbb{C}^n$, we have

$$\begin{aligned} |(M^{-1}Ny)_i| &= \Big| \sum_{j \neq i} \frac{a_{ij}}{a_{ii}} y_j \Big| \\ &< \|y\|_\infty, \end{aligned}$$

thus $\|M^{-1}N\|_\infty < 1$ and $\rho(M^{-1}N) < 1$.

Gauss-Seidel method

For the Gauss-Seidel method, we set $M = D - E$ and $N = F$.

In terms of number of operations, the Gauss-Seidel algorithm requires the inversion of a triangular system which scales as $\mathcal{O}(n^2)$ if the matrix is dense, but as $\mathcal{O}(n)$ if the matrix is sparse.

In that case, the i-th entry of the vector $x^{(k)}$ is given by

$$x_i^{(k)} = \frac{b_i - \sum_{j<i} a_{ij} x_j^{(k)} - \sum_{j>i} a_{ij} x_j^{(k-1)}}{a_{ii}}.$$

Once $x_i^{(k)}$ is computed, $x_i^{(k-1)}$ is not useful anymore. The update can be implemented in place. Contrary to the Jacobi method, the Gauss-Seidel algorithm is hardly parallelisable.

We have the same convergence theorem as previously.

Proposition 3.3. *If A is row-wise diagonally dominant,* i.e. *for each $1 \leq i \leq n$, $|a_{ii}| > \sum_{j \neq i} |a_{ij}|$, then the Gauss-Seidel method converges.*

Proof: Let $y, z \in \mathbb{C}^n$ such that $z = M^{-1}Ny$. Then we have $Mz = Ny$ so

$$a_{ii} z_i = \sum_{j>i} a_{ij} y_j + \sum_{j<i} a_{ij} z_j.$$

Let i_0 such that $|z_{i_0}| = \|z\|_\infty$. Then

$$|a_{i_0 i_0} z_{i_0}| \leq \sum_{j<i_0} |a_{i_0 j}| \|z\|_\infty + \sum_{j>i_0} |a_{i_0 j}| \|y\|_\infty$$

but since A is diagonally dominant

$$|a_{i_0 i_0}| - \sum_{j<i_0} |a_{i_0 j}| > \sum_{j>i_0} |a_{i_0 j}|,$$

thus

$$|z_{i_0}| < \|x\|_\infty.$$

This shows that $\rho(M^{-1}N) < 1$.

Successive over relaxation (SOR) method

For the SOR method, we have a positive parameter ω and we set $M = \frac{1}{\omega}D - E$ and $N = (\frac{1}{\omega} - 1)D + F$.

The SOR method also involves the inversion of a triangular matrix, as such, the cost at each step of the algorithm scales as $\mathcal{O}(n^2)$ for a dense matrix and $\mathcal{O}(n)$ for a sparse matrix.

We also have the same type of convergence result.

Proposition 3.4. *If A is row-wise diagonally dominant,* i.e. *for each $1 \leq i \leq n$, $|a_{ii}| > \sum_{j \neq i} |a_{ij}|$, and if $0 < \omega \leq 1$ then the SOR method converges.*

Proof: Exercise.

3.3 Richardson iteration

For Richardson iteration, the method corresponds to taking $M = \frac{1}{\alpha}\operatorname{id}$ and $N = \frac{1}{\alpha}\operatorname{id} - A$:

$$x^{(k+1)} = (\operatorname{id} - \alpha A)x^{(k)} + \alpha b. \tag{3.4}$$

Proposition 3.5. *Assume that $A \in \mathbb{C}^{n \times n}$ is invertible and diagonalisable with eigenvalues $\lambda_1, \dots, \lambda_n$. Then the Richardson iteration converges if and only if $0 < \alpha < 2\frac{\min \Re\lambda_j}{|\lambda_j|^2}$ or $2\frac{\min \Re\lambda_j}{|\lambda_j|^2} < \alpha < 0$.*

Proof: Again we need to study the spectral radius of $M^{-1}N = (\operatorname{id} - \alpha A)$. The eigenvalues of $\operatorname{id} - \alpha A$ are simply $1 - \alpha\lambda_j, j = 1 \dots n$. Hence $\rho(M^{-1}N) < 1 \iff \forall 1 \leq j \leq n, |1 - \alpha\lambda_j|^2 < 1$. But $|1 - \alpha\lambda_j|^2 = 1 - 2\alpha\Re\lambda_j + \alpha^2|\lambda_j|^2 < 1$ thus the condition is

$$\forall 1 \leq j \leq n, \alpha^2 < 2\alpha\Re\lambda_j.$$

This can be satisfied only if α has the same sign as all the λ_j and we find the result.

Suppose now that the matrix A is diagonalisable and has only positive eigenvalues $0 < \lambda_1 < \cdots < \lambda_n$. We can wonder for which value α, the spectral radius of the iteration matrix $\operatorname{id} - \alpha A$ is the smallest:

$$\rho(\operatorname{id} - \alpha A) = \max(|1 - \alpha\lambda_1|, \dots, |1 - \alpha\lambda_n|) = \max(|1 - \alpha\lambda_1|, |1 - \alpha\lambda_n|).$$

Proposition 3.6. *The spectral radius of the iteration matrix $\operatorname{id} - \alpha A$ is minimal is minimal for*

$$\alpha = \frac{2}{\lambda_1 + \lambda_n},$$

and for this value we have

$$\rho(\operatorname{id} - \alpha A) = \frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1}.$$

Proof: Graphically, we see that the minimal value of $\rho(\operatorname{id} - \alpha A)$ is attained when

$$-1 + \alpha\lambda_n = 1 - \alpha\lambda_1,$$

which gives $\alpha = \frac{2}{\lambda_1 + \lambda_n}$.

Remark 3.7. *For a Hermitian positive definite matrix, this can be rewritten as*

$$\rho(\mathrm{id} - \alpha A) = \frac{\mathrm{cond}_2(A) - 1}{\mathrm{cond}_2(A) + 1}.$$

If the condition number $\mathrm{cond}_2(A)$ *is large, the spectral radius of the iteration matrix is close to 1.*

Interpretation as a gradient descent method

Suppose that $A \in \mathbb{R}^{n \times n}$ is a symmetric positive-definite matrix and consider the functional F

$$F(x) = \frac{1}{2}\langle x, Ax\rangle - \langle b, x\rangle, \tag{3.5}$$

where $\langle \cdot, \cdot \rangle$ denotes the scalar product of $\mathbb{R}^n$. Since A is positive-definite, the functional F is convex. Moreover, $\lim_{\|x\| \to \infty} F(x) = \infty$, hence F has a unique minimum satisfying $\nabla F(x_*) = Ax_* - b = 0$.

Hence to solve the linear problem $Ax = b$, we can use a minimisation algorithm to the functional F. A simple fixed-step gradient algorithm is thus

$$x^{(k+1)} = x^{(k)} - \alpha \nabla F(x^{(k)}) = (\mathrm{id} - \alpha A)x^{(k)} + \alpha b,$$

which is simply the Richardson iteration of the previous subsection.

Steepest descent

The parameter α can also be chosen adaptively, a natural choice being to minimise at each iteration the function $f : \alpha \mapsto F(x^{(k)} + \alpha p^{(k)})$ where $p^{(k)} = b - Ax^{(k)}$.

By composition, the function f is convex, hence the minimum is attained where the derivative vanishes. First we have

$$F(x^{(k)} + \alpha p^{(k)}) = \frac{1}{2}\langle x^{(k)}, Ax^{(k)}\rangle + \frac{1}{2}\langle p^{(k)}, Ap^{(k)}\rangle + \alpha\langle p^{(k)}, Ax^{(k)}\rangle - \langle x^{(k)}, b\rangle - \alpha\langle p^{(k)}, b\rangle,$$

thus

$$f'(\alpha) = \alpha\langle p^{(k)}, Ap^{(k)}\rangle + \langle p^{(k)}, Ax^{(k)}\rangle - \langle p^{(k)}, b\rangle.$$

Thus the parameter α_k such that $f'(\alpha_k) = 0$ is given by

$$\alpha_k = \frac{\langle p^{(k)}, Ax^{(k)} - b\rangle}{\langle p^{(k)}, Ap^{(k)}\rangle} = \frac{\langle p^{(k)}, p^{(k)}\rangle}{\langle p^{(k)}, Ap^{(k)}\rangle}. \tag{3.6}$$

Since A is symmetric, positive-definite, the bilinear form $(x, y) \mapsto \langle x, Ay\rangle$ defines a scalar product. Let us denote the associated norm by $\| \cdot \|_A$. Note that

$$\begin{aligned}
\frac{1}{2}\langle x - x_*, A(x - x_*)\rangle &= \frac{1}{2}\langle x, Ax\rangle - \langle x, Ax_*\rangle + \frac{1}{2}\langle x_*, Ax_*\rangle \\
&= \frac{1}{2}\langle x, Ax\rangle - \langle x, b\rangle + \frac{1}{2}\langle x_*, Ax_*\rangle \\
&= F(x) + \frac{1}{2}\langle x_*, Ax_*\rangle.
\end{aligned}$$

Thus minimising F is the same thing as minimising $\|x - x_*\|_A$.

With this observation, we can prove the following theorem on the convergence of the steepest descent algorithm.

Algorithm 6 Steepest descent gradient

```
function SteepestDescent(A, b, ε_tol)
    x = 0
    p = b
    while ‖p‖ > ε_tol do
        α = ‖p‖² / ⟨p, Ap⟩
        x = x + αp
        p = p − αAp
    end while
    return x
end function
```

Theorem 3.8. *Assume that A is a symmetric, positive-definite matrix. Denote by $(x^{(k)})$ the sequence by Algorithm 6. Then we have for all $k \geq 0$*

$$\|x^{(k)} - x_*\|_A \leq \Big(\frac{\mathrm{cond}_2(A) - 1}{\mathrm{cond}_2(A) + 1}\Big)^k \|x^{(0)} - x_*\|. \tag{3.7}$$

Proof: By definition of $x^{(k)}$, recalling that $\alpha_{\mathrm{opt}} = \frac{2}{\lambda_{\min}+\lambda_{\max}}$ and we have

$$\begin{aligned}
\|x^{(k)} - x_*\|_A &= \min_{\alpha\in\mathbb{R}} \|x^{(k-1)} - x_* + \alpha p^{(k-1)}\|_A \\
&\leq \|x^{(k-1)} - x_* + \alpha_{\mathrm{opt}} p^{(k-1)}\|_A \\
&\leq \|x^{(k-1)} - x_* + \alpha_{\mathrm{opt}} (b - Ax^{(k-1)})\|_A \\
&\leq \|x^{(k-1)} - x_* + \alpha_{\mathrm{opt}} (Ax_* - Ax^{(k-1)})\|_A \\
&\leq \|(\mathrm{id} - \alpha_{\mathrm{opt}} A)(x^{(k-1)} - x_*)\|_A \\
&\leq \rho(\mathrm{id} - \alpha_{\mathrm{opt}} A)\|x^{(k-1)} - x_*\|_A,
\end{aligned}$$

where we have used that if G and A commute

$$\langle Gy, AGy\rangle = \langle A^{1/2}Gy, A^{1/2}Gy\rangle = \langle GA^{1/2}y, GA^{1/2}y\rangle \leq \rho(G)^2\langle A^{1/2}y, A^{1/2}y\rangle = \rho(G)^2\|y\|_A^2.$$

The result follows from $\rho(\mathrm{id} - \alpha_{\mathrm{opt}}A) = \frac{\mathrm{cond}_2(A)-1}{\mathrm{cond}_2(A)+1}$.

Again an ill-conditioned matrix impedes the speed of convergence of the steepest descent algorithm.