10 Iterative Methods for Linear Systems

10.1 Introduction

We have seen that a direct algorithm for solving

\[ Ax = b \]

requires \( O(n^3) \) work. This amount of work becomes impractical quite quickly. The basic aim of an iterative method is to produce a method of approximating the action of the inverse of the matrix such that the amount of work required is better than \( O(n^3) \).

To understand the general concept, let us write the matrix \( A \) as 

\[ A = A - B + B \]

with an invertible matrix \( B \) at our disposal. Then, the equation \( Ax = b \) can be reformulated as 

\[ b = Ax = (A - B)x + Bx \]

and hence as

\[ x = B^{-1}(B - A)x + B^{-1}b =: Cx + c =: F(x), \]

so that \( x \) is a fixed point of the mapping \( F \). To calculate this fixed point, we can use the following simple iterative process. We first pick a starting point \( x_0 \) and then form

\[ x_{i+1} := F(x_i), \quad i = 1, 2, 3, \ldots \quad (13) \]

If this sequence converges and if \( F \) is continuous, the limit has to be a fixed point of \( F \).

10.2 Banach’s Fixed Point Theorem

We will now derive a general convergence result for the iteration process (13).

**Definition 10.1** A mapping \( F : \mathbb{R}^n \to \mathbb{R}^n \) is called a contraction mapping with respect to a norm \( \| \cdot \| \) on \( \mathbb{R}^n \) if there is a constant \( 0 < q < 1 \) such that

\[ \| F(x) - F(y) \| \leq q \| x - y \| \]

for all \( x, y \in \mathbb{R}^n \).

A contraction mapping is Lipschitz-continuous with Lipschitz-constant \( q < 1 \).

**Theorem 10.2 (Banach)** If \( F : \mathbb{R}^n \to \mathbb{R}^n \) is a contraction mapping then \( F \) has exactly one fixed point \( x^* \). The sequence \( x_{j+1} := F(x_j) \) converges for every starting point \( x_0 \in \mathbb{R}^n \). Furthermore, we have the error estimates

\[ \| x^* - x_j \| \leq \frac{q^j}{1 - q} \| x_j - x_{j-1} \| \quad (a \ posteriori), \]

\[ \| x^* - x_j \| \leq \frac{q^j}{1 - q} \| x_1 - x_0 \| \quad (a \ priori). \]
If we apply this theorem to our special iteration function \( F(x) = Cx + c \), where \( C \) is the iteration matrix, we see that
\[
\|F(x) - F(y)\| = \|Cx + c - (Cy + c)\| = \|C(x - y)\| \leq \|C\|\|x - y\|,
\]
so that we have convergence if \( \|C\| < 1 \). Unfortunately, this depends on the chosen vector and hence matrix norm, while, since all norms on \( \mathbb{R}^n \) are equivalent, the fact that the sequence converges does not depend on the norm.

In other words, having an induced matrix norm with \( \|C\| < 1 \) is sufficient for convergence but not necessary. A sufficient and necessary condition can be stated using the spectral radius of the iteration matrix.

**Definition 10.3** Let \( A \in \mathbb{R}^{n \times n} \) with eigenvalues \( \lambda_1, \lambda_2, \ldots, \lambda_n \) ordered so that \( |\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n| \). The spectral radius of \( A \) is given by \( \rho(A) := |\lambda_1| \).

Note that, if \( \lambda \) is an eigenvalue of \( A \) with eigenvector \( x \), then \( \lambda^r \) is an eigenvalue of \( A^r \), \( r = 1, 2, 3, \ldots \) with eigenvector \( x \). Hence, \( \rho(A^r) = \rho(A)^r \).

**Theorem 10.4**

1. If \( \|\cdot\| \) is a compatible matrix-norm then \( \rho(A) \leq \|A\| \) for all matrices \( A \in \mathbb{R}^{n \times n} \).

2. For any \( \epsilon > 0 \) there is an induced norm, \( \|\cdot\| \) such that \( \rho(A) \leq \|A\| \leq \rho(A) + \epsilon \).

This allows us to state and prove our main convergence result for iterative processes.

**Theorem 10.5** The iteration \( x_{j+1} = Cx_j + c \) converges for every starting point if and only if \( \rho(C) < 1 \).

**Proof:** Assume first that \( \rho(C) < 1 \). Then, we can pick an \( \epsilon > 0 \) such that \( \rho(C) + \epsilon < 1 \) and, by Theorem 10.4, we can find an induced matrix norm \( \|\cdot\| \) such that \( \|C\| \leq \rho(C) + \epsilon < 1 \), which gives convergence.

Assume now that the iteration converges to \( x^* \) for every starting point \( x_0 \). If we pick the starting point such that \( x = x_0 - x^* \) is an eigenvector of \( C \) with eigenvalue \( \lambda \), then
\[
x_j - x^* = F(x_{j-1}) - F(x^*) = C(x_{j-1} - x^*) = \ldots = C^j(x_0 - x^*) = \lambda^j(x_0 - x^*).
\]
Since the expression on the left hand side tends to zero for \( j \to \infty \), so does the expression on the right hand side. This, however, is only possible if \( |\lambda| < 1 \). Since \( \lambda \) was an arbitrary eigenvalue of \( C \), this shows that \( \rho(C) < 1 \). \( \square \)
After this general discussion, we return to the question on how to pick the iteration matrix $C$. Our initial approach yields

$$C = B^{-1}(B - A) = I - B^{-1}A,$$

with a matrix $B$, which should be sufficiently close to $A$ but also easily invertible. From now on, we will assume that the diagonal elements of $A$ are all nonzero. This can be achieved by exchanging rows and/or columns as long as $A$ is nonsingular.

Next, we decompose $A$ in its lower-left sub-diagonal part, its diagonal part and its upper-right sup-diagonal part, i.e.

$$A = L + D + R.$$

The simplest possible approximation to $A$ is then given by picking its diagonal part $D$ for $B$ so that the iteration matrix becomes

$$C_J = I - B^{-1}A = I - D^{-1}(L + D + R) = -D^{-1}(L + R),$$

with entries

$$c_{ik} = \begin{cases} -a_{ik}/a_{ii}, & \text{if } i \neq k, \\ 0 & \text{else.} \end{cases}$$

(14)

Hence, we can write the iteration

$$x^{(j+1)} = -D^{-1}(L + R)x^{(j)} + D^{-1}b$$

component-wise as

$$x_i^{(j+1)} = \frac{1}{a_{ii}} \left( b_i - \sum_{k=1 \atop k \neq i}^{n} a_{ik}x_k^{(j)} \right), \quad 1 \leq i \leq n,$$

(15)

where, from now on, we will write the iteration index as an upper index.

**Definition 10.6** The iteration defined by (15) is called Jacobi method.

Obviously, one can expect convergence of the Jacobi method if the original matrix $A$ resembles a diagonal matrix.

**Definition 10.7** A matrix $A$ is called strongly row diagonally dominant if

$$\sum_{k=1 \atop k \neq i}^{n} |a_{ik}| < |a_{ii}|, \quad 1 \leq i \leq n.$$
Theorem 10.8 The Jacobi method converges for every starting point if the matrix $A$ is strongly row diagonally dominant.

Proof: We use the row sum norm to calculate the norm of the iteration matrix $C$:

$$
\|C\|_\infty = \max_{1 \leq i \leq n} \sum_{k=1}^{n} |c_{ik}| = \max_{1 \leq i \leq n} \sum_{k=1, k \neq i}^{n} |a_{ik}| / |a_{ii}| < 1.
$$

Hence, we have convergence. □

A closer inspection of the method (15) shows that the computation of $x_i^{(j+1)}$ is independent of any other $x_k^{(j+1)}$. This means that, on a parallel or vector computer all components of the new iteration $x^{(j+1)}$ can be computed simultaneously.

However, it also gives us the possibility to improve the process. For example, to calculate $x_2^{(j+1)}$ we could already employ the newly computed $x_1^{(j+1)}$. Then, for computing $x_3^{(j+1)}$ we could use $x_1^{(j+1)}$ and $x_2^{(j+1)}$ and so on.

This leads to the following iteration scheme.

Definition 10.9 The Gauss-Seidel method is given by the iteration scheme

$$
x_i^{(j+1)} = \frac{1}{a_{ii}} \left( b_i - \sum_{k=1}^{i-1} a_{ik}x_k^{(j+1)} - \sum_{k=i+1}^{n} a_{ik}x_k^{(j)} \right), \quad 1 \leq i \leq n. \quad (16)
$$

To analyse the convergence of this scheme, we have to find the iteration matrix $C = I - B^{-1}A$. To this end, we rewrite (16) as

$$
a_{ii}x_i^{(j+1)} + \sum_{k=1}^{i-1} a_{ik}x_k^{(j+1)} = b_i - \sum_{k=i+1}^{n} a_{ik}x_k^{(j)},
$$

which translates into

$$(L + D)x^{(j+1)} = -Rx^{(j)} + b.$$

Thus, the iteration matrix of the Gauss-Seidel method is given by

$$
C_G = -(L + D)^{-1}R.
$$

Later on, we will prove a more general version of the following theorem.

Theorem 10.10 If $A = A^T$ is positive definite then the Gauss-Seidel method converges.
10.4 Relaxation

A further improvement of both methods can be achieved by Relaxation. We start by looking at the Jacobi method. Here, the iterations can be written as

\[ x^{(j+1)} = D^{-1}b - D^{-1}(L + R)x^{(j)} = x^{(j)} + D^{-1}b - D^{-1}(L + R + D)x^{(j)} = x^{(j)} + D^{-1}(b - Ax^{(j)}). \]

The latter equality shows that the new iteration \( x^{(j+1)} \) is given by the old iteration \( x^{(j)} \) corrected by the \( D^{-1} \)-multiple of the residual \( b - Ax \). In practice, one often notice that the correction term is off the correct correction term by a fixed factor. Hence, it makes sense to introduce a relaxation parameter \( \omega \) and to form the new iteration as

\[ x^{(j+1)} = x^{(j)} + \omega D^{-1}(b - Ax^{(j)}), \tag{17} \]

which gives the following component-wise scheme:

**Definition 10.11** The Jacobi Relaxation is given by

\[ x_i^{(j+1)} = x_i^{(j)} + \frac{\omega}{a_{ii}} \left( b_i - \sum_{k=1}^{n} a_{ik}x_k^{(j)} \right), \quad 1 \leq i \leq n. \]

Of course, the relaxation parameter should be chosen such that the convergence improves compared to the original Jacobi method. The iteration matrix follows from

\[ x^{(j+1)} = x^{(j)} + \omega D^{-1}b - \omega D^{-1}(L + D + R)x^{(j)} = [(1 - \omega)I - \omega D^{-1}(L + R)]x^{(j)} + \omega D^{-1}b \]

to be

\[ C_J(\omega) = [(1 - \omega)I - \omega D^{-1}(L + R)] = (1 - \omega)I + \omega C_J, \]

which shows that \( C_J(1) = C_J \) corresponds to the classical Jacobi method.

**Theorem 10.12** Assume that \( C_J = -D^{-1}(L + R) \) has only real eigenvalues \( \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n < 1 \) with corresponding eigenvectors \( z^{(1)}, \ldots, z^{(n)} \). Then, \( C(\omega) \) has the same eigenvectors \( z^{(1)}, \ldots, z^{(n)} \), but with eigenvalues \( \mu_j = 1 - \omega + \omega \lambda_j \) for \( 1 \leq j \leq n \). The spectral radius of \( C(\omega) \) is minimised by choosing

\[ \omega^* = \frac{2}{2 - \lambda_1 - \lambda_n}. \tag{18} \]

In the case of \( \lambda_1 \neq -\lambda_n \) Relaxation converges faster then the Jacobi method.
Figure 2: Determination of the relaxation parameter

**Proof:** For every eigenvector $z^{(j)}$ of $C_J$ it follows that

$$C(\omega)z^{(j)} = (1 - \omega)z^{(j)} + \omega \lambda_j z^{(j)} = (1 - \omega + \omega \lambda_j)z^{(j)},$$

i.e. $z^{(j)}$ is eigenvector of $C(\omega)$ for the eigenvalue $1 - \omega + \omega \lambda_j =: \mu_j(\omega)$. Thus, the spectral radius of $C(\omega)$ is given by

$$\rho(C(\omega)) = \max_{1 \leq j \leq n} |\mu_j(\omega)| = \max_{1 \leq j \leq n} |1 - \omega + \omega \lambda_j|,$$

which should be minimised. For a fixed $\omega$ let us have a look at the function $f_\omega(\lambda) := 1 - \omega + \omega \lambda$, which is, as a function of $\lambda$, a straight line with $f_\omega(1) = 1$.

For different choices of $\omega$ we have this way a collection of such lines (see Figure 2) and it follows that the maximum in the definition of $\rho(C(\omega))$ can only be attained for the indices $j = 1$ and $j = n$. Moreover, it follows that $\omega$ is optimally chosen if $f_\omega(\lambda_1) = -f_\omega(\lambda_n)$ or

$$1 - \omega + \omega \lambda_1 = -(1 - \omega + \omega \lambda_n).$$

This gives (18). Finally, we have the Jacobi method if and only if $\omega^* = 1$, which is equivalent to $\lambda_1 = -\lambda_n$.

An alternative interpretation of the relaxation can be derived from

$$x^{(j+1)} = (1 - \omega)x^{(j)} + \omega C_J x^{(j)} + \omega D^{-1} b$$

$$= (1 - \omega)x^{(j)} + \omega (C_J x^{(j)} + D^{-1} b).$$

Hence, if we define $z^{(j+1)} = C_J x^{(j)} + D^{-1} b$, which is one step of the classical Jacobian method, the next iteration of the Jacobi Relaxation method is

$$x^{(j+1)} = (1 - \omega)x^{(j)} + \omega z^{(j+1)},$$

which is a linear interpolation between the old iteration and the new Jacobian iteration.
This idea can be used to introduce relaxation for the Gauss-Seidel method as well. We start by looking at $Dx^{(j+1)} = b - Lx^{(j+1)} - Rx^{(j+1)}$ and replace the iteration on the left hand side by $z^{(j+1)}$ and then use linear interpolation again. Hence, we set

$$Dz^{(j+1)} = b - Lx^{(j+1)} - Rx^{(j+1)}.$$ 

Multiplying the second equation with $D$ and inserting the first one yields

$$Dx^{(j+1)} = (1 - \omega)x^{(j)} + \omega z^{(j+1)}.$$ 

and hence

$$C_G(\omega) = (D + \omega L)^{-1}[(1 - \omega)D - \omega R].$$ 

We can rewrite this component-wise.

**Definition 10.13** The Gauss-Seidel Relaxation or SOR (successive over-relaxation) method is given by

$$x_i^{(j+1)} = x_i^{(j)} + \frac{\omega}{a_{ii}} \left( b_i - \sum_{k=1}^{i-1} a_{ik}x_k^{(j+1)} - \sum_{k=i}^{n} a_{ik}x_k^{(j)} \right), \quad 1 \leq i \leq n.$$ 

Again, we have to deal with the question on how to choose the relaxation parameter.

**Theorem 10.14** The spectral radius of the iteration matrix $C_G(\omega)$ of SOR satisfies

$$\rho(C_G(\omega)) \geq |\omega - 1|.$$ 

Hence, convergence is only possible if $\omega \in (0, 2)$.

**Proof:** The iteration matrix $C_G(\omega)$ can be written in the form

$$C_G(\omega) = (I + \omega D^{-1}L)^{-1}[(1 - \omega)I - \omega D^{-1}R].$$ 

The first matrix in this product is a normalised lower triangular matrix and the second matrix is an upper triangular matrix with diagonal entries all equal to $1 - \omega$. Since the determinant of a matrix equals the product of its eigenvalues, we have

$$|1 - \omega|^n = |\det C_G(\omega)| \leq \rho(C_G(\omega))^n,$$

which gives the result. 

We will now show that for a positive definite matrix $\omega \in (0, 2)$ is also sufficient for convergence. Since $\omega = 1$ gives the classical Gauss-Seidel method, we also cover Theorem 10.10.
Theorem 10.15 Let $A \in \mathbb{R}^{n \times n}$ be symmetric and positive definite. Then, the SOR method converges for every relaxation parameter $\omega \in (0, 2)$.

Proof: We have to show that $\rho(C_G(\omega)) < 1$. To this end, we rewrite the iteration matrix $C_G(\omega)$ in the form

\[
C_G(\omega) = (D + \omega L)^{-1}[D + \omega L - \omega(L + D + R)]
= I - \omega(D + \omega L)^{-1}A = I - \left(\frac{1}{\omega}D + L\right)^{-1}A
= I - B^{-1}A,
\]

with $B = \frac{1}{\omega}D + L$. Let $\lambda \in \mathbb{C}$ be an eigenvalue of $C_G(\omega)$ with corresponding eigenvector $x \in \mathbb{C}^n$, which we assume to be normalised by $\|x\|_2 = 1$. Then, we have $C_G(\omega)x = (I - B^{-1}A)x = \lambda x$ or $A x = (1 - \lambda)Bx$. Since $A$ is positive definite, we must have $\lambda \neq 1$ such that we can conclude

\[
\frac{1}{1 - \lambda} = \frac{x^T B x}{x^T A x}.
\]

Since $A$ is symmetric, we can conclude that $B + B^T = (\frac{2}{\omega} - 1)D + A$, such that the real part of $1/(1 - \lambda)$ satisfies

\[
\Re\left(\frac{1}{1 - \lambda}\right) = \frac{1}{2} \frac{x^T (B + B^T)x}{x^T A x} = \frac{1}{2} \left\{ \frac{2}{\omega} - 1 \right\} \frac{x^T D x}{x^T A x} + 1 > \frac{1}{2},
\]

because, on account of $\omega \in (0, 2)$, the expression $2/\omega - 1$ is positive, as well as $x^T D x / x^T A x$. The latter follows since the diagonal entries of a positive definite matrix have to be positive. If we write $\lambda = u + iv$ then we can conclude that

\[
\frac{1}{2} < \Re\left(\frac{1}{1 - \lambda}\right) = \frac{1 - u}{(1 - u)^2 + v^2}
\]

and hence $|\lambda|^2 = u^2 + v^2 < 1$. $\square$

Example 10.16 Suppose we wish to solve the system $Ax = b$ where

\[
A = \begin{bmatrix}
1 & 0 & 0.25 & 0.25 \\
0 & 1 & 0 & 0.25 \\
0.25 & 0 & 1 & 0 \\
0.25 & 0.25 & 0 & 1
\end{bmatrix}, \quad b = \begin{bmatrix}
0.25 \\
0.5 \\
0.75 \\
1.0
\end{bmatrix}
\]

Note that $A$ is symmetric and diagonally dominant.
Using Jacobi we have

$$
\begin{array}{cccc}
  x_0 & x_1^j & x_2^j & x_3^j \\
  0 & 0.25 & -0.1875 & -0.125 \\
  0 & 0.5 & 0.25 & 0.2969 \\
  0 & 0.75 & 0.6875 & 0.7969 \\
  0 & 1.0 & 0.8125 & 0.9844 \\
\end{array}
$$

$$
\|Ax_j - b\|_2 = 1.3693 \quad 0.5413 \quad 0.2182 \quad 0.0882
$$

Using Gauss-Seidel we have

$$
\begin{array}{cccc}
  x_0 & x_G^1 & x_G^2 & x_G^3 \\
  0 & 0.25 & -0.125 & -0.1846 \\
  0 & 0.5 & 0.2969 & 0.2607 \\
  0 & 0.6875 & 0.7812 & 0.7961 \\
  0 & 0.8125 & 0.9570 & 0.9810 \\
\end{array}
$$

$$
\|Ax_G - b\|_2 = 1.3693 \quad 0.4265 \quad 0.0697 \quad 0.0114
$$

Using SOR with $\omega = 1.05$

$$
\begin{array}{cccc}
  x_0 & x_S^1 & x_S^2 & x_S^3 \\
  0 & 0.2625 & -0.1606 & -0.1943 \\
  0 & 0.5250 & 0.2774 & 0.2546 \\
  0 & 0.7186 & 0.7937 & 0.7988 \\
  0 & 0.8433 & 0.9772 & 0.9853 \\
\end{array}
$$

$$
\|Ax_S - b\|_2 = 1.3693 \quad 0.4699 \quad 0.0394 \quad 0.0020
$$