Matrix iterative methods are especially useful for the solution of linear systems involving large sparse matrices (i.e., many zero entries).

A large class of such methods can be defined as follows: Write $A = N - P$, where $N$ and $P$ are matrices of the same order as $A$, which we shall choose to have appropriate properties. The system $Ax = b$ is then written $Nx = Px + b$ and we define a simple iteration scheme by:

$$Nx^{k+1} = Px^k + b, \quad k = 0, 1, \ldots,$$

where $x^0$ denotes an initial guess. We assume that $\det N \neq 0$, so that the iteration scheme produces a unique sequence of vectors $\{x^k\}$. We also choose the matrix $N$ so that the system of equations $Ny = z$ is easily solved (e.g., $N$ may be diagonal or upper or lower triangular).

To describe some examples of this procedure, we write $A = L + U + D$ where $L$ denotes the matrix whose elements below the main diagonal are equal to those of $A$, with the remaining elements chosen to be zero. The matrix $U$ is an upper triangular matrix that coincides with the upper triangular elements of $A$, and $D$ is a diagonal matrix that coincides with the diagonal entries of $A$.

The Jacobi method (or method of simultaneous displacements) chooses $N = D$, $P = -(L + U)$ so

$$x^{k+1} = -D^{-1}(L + U)x^k + D^{-1}b, \quad k = 0, 1, \ldots,$$

where we have assumed the diagonal entries of $A$ are non-zero (otherwise interchange rows and columns to get an equivalent system with this property).

In terms of components, we have

$$x_i^{k+1} = \frac{1}{a_{ii}} \left[ b_i - \sum_{j=1}^{n} a_{ij} x_j^k \right], \quad i = 1, \ldots, n.$$

We note from these equations that some components of $x^{k+1}$ are known, but not used, while computing the remaining components. The Gauss-Seidel method (or method of successive displacements) is a modification of the Jacobi method in which all the latest components are used, as they are computed. This scheme is:

$$x_i^{k+1} = \frac{1}{a_{ii}} \left[ b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{k+1} - \sum_{j=i+1}^{n} a_{ij} x_j^k \right], \quad i = 1, \ldots, n.$$

The splitting of $A$ that gives this procedure is $N = L + D$, $P = -U$, so that

$$x^{k+1} = -(L + D)^{-1}Ux^k + (L + D)^{-1}b, \quad k = 0, 1, \ldots,$$

To consider the convergence of schemes of this form, i.e.,

$$x^{k+1} = N^{-1}Px^k + N^{-1}b, \quad k = 0, 1, \ldots,$$
we set $M = N^{-1}P$. Since $Nx = Px + b$, we have $x = N^{-1}Px + N^{-1}b$. Define the error vector $e^k = x - x^k$. Then, subtracting equations, we have
\[e^{k+1} = N^{-1}Px^k \equiv Me^k.\]
Iterating this equation, we get
\[e^{k+1} = Me^k = M^2e^{k-1} = \cdots = M^{k+1}e^0,
\]
so $e^k = M^ke^0$. Thus, a sufficient condition for convergence of the iteration schemes, i.e., that $\lim_{k \to \infty} e^k = 0$ is that $\lim_{k \to \infty} M^k = 0$. If the method is to converge for all choices of $e^0$, then this condition is also necessary. A matrix $M$ that satisfies this condition is called a convergent matrix. The basic results characterizing convergent matrices are the following.

**Theorem 5.** The matrix $M$ is convergent if and only if all the eigenvalues of $M$ are less than one in absolute value, i.e., $\rho(M) < 1$.

A sufficient condition for convergence, that is often easier to apply is:

**Theorem 6.** The matrix $M$ is convergent if for any matrix norm, $\|M\| < 1$.

Hence, if $M = (m_{ij})$, then $M$ will be convergent if
\[\|M\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^{n} |m_{ij}| < 1 \quad \text{or} \quad \|M\|_1 = \max_{1 \leq i \leq n} \sum_{j=1}^{n} |m_{ij}| < 1.
\]

A simple application of this result is the following:

**Theorem 7.** If $A$ is strictly diagonally dominant, then Jacobi’s method converges.

**Proof.** By hypothesis,
\[|a_{ii}| > \sum_{j=1}^{n} |a_{ij}|, \quad i = 1, \ldots, n.
\]
Hence,
\[\sum_{j=1}^{n} |a_{ij}|/|a_{ii}| < 1, \quad i = 1, \ldots, n.
\]
Recall that for Jacobi’s method, the iteration matrix $M = -D^{-1}(L+U)$, i.e., $m_{ij} = -a_{ij}/a_{ii}$ when $i \neq j$ and $m_{ij} = 0$ when $i = j$. Hence,
\[\|M\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^{n} |m_{ij}| = \max_{1 \leq i \leq n} \sum_{j=1}^{n} |a_{ij}|/|a_{ii}| < 1.
\]

Although the proof is more complicated (we use an induction argument), one can also show:
Theorem 8. If $A$ is strictly diagonally dominant, then the Gauss-Seidel method converges.

Some other convergence results for these methods are:

Theorem 9. (i) If $A$ is Hermitian and positive definite, then the Gauss-Seidel method converges. (ii) If $A$ is Hermitian and $A$ and $2D - A$ are positive definite, then Jacobi’s method converges. (iii) If $A$ is irreducible and weakly diagonally dominant, then the Gauss-Seidel method and Jacobi’s method converge. (iv) If $A$ is an $L$-matrix (i.e., $a_{ii} > 0$, $i = 1, \ldots, n$ and $a_{ij} \leq 0$, $i \neq j$, $i, j = 1, \ldots, n$, then the Gauss-Seidel method converges if and only if the Jacobi method converges. If both converge, then the Gauss-Seidel method converges faster, i.e., $\rho(GS) < \rho(J)$, where $\rho(A)$ denotes the spectral radius of the matrix $A$.

We next consider a method for accelerating the convergence of iterative methods. We define the iteration:

$$x_{i}^{k+1} = (1 - \omega)x_{i}^{k} + \frac{\omega}{a_{ii}}\left[b_{i} - \sum_{j=1}^{i-1} a_{ij}x_{j}^{k+1} - \sum_{j=i+1}^{n} a_{ij}x_{j}^{k}\right], \quad i = 1, \ldots, n,$$

where $\omega$ is a real parameter called the relaxation factor. Note $\omega = 1$ gives the Gauss-Seidel method. The choice $\omega < 1$ is called under-relaxation, while $\omega > 1$ is called over-relaxation. The usual strategy is to choose $\omega > 1$ and the resulting method is called SOR (successive over-relaxation). Note that we may also write these equations as:

$$a_{ii}x_{i}^{k+1} + \omega \sum_{j=1}^{i-1} a_{ij}x_{j}^{k+1} = a_{ii}(1 - \omega)x_{i}^{k} + \omega\left[b_{i} - \sum_{j=i+1}^{n} a_{ij}x_{j}^{k}\right], \quad i = 1, \ldots, n,$$

In matrix form, we have

$$(D + L\omega)x^{k+1} = (1 - \omega)Dx^{k} + \omega[b - Ux^{k}],$$

which we may rewrite as

$$x^{k+1} = (D + L\omega)^{-1}[(1 - \omega)D - \omega U]x^{k} + \omega(D + L\omega)^{-1}b.$$

The motivation for this method comes from the proof of convergence of the general iteration scheme. Recall, we showed that $e^{k+1} = Me^{k}$, where $M$ is the iteration matrix. Hence, $\|e^{k+1}\| \leq \|M\|\|e^{k}\|$, so we would like $\|M\|$ as small as possible to reduce the error as much as possible at each iteration.

When $M$ is symmetric and $\| \cdot \| = \| \cdot \|_{2}$, then $\|M\| = \rho(M) = \max_{i} |\lambda_{i}|$. In this case, we would like to choose $\omega$ to minimize $\rho(M)$. However, the best choice of $\omega$ depends on $A$ and is difficult to calculate except in some special cases. However, there are some known convergence results.

Theorem 10. If SOR converges, then $0 < \omega < 2$.

Proof. The iteration matrix for SOR is given by

$$M = (D + L\omega)^{-1}[(1 - \omega)D - \omega U] = (D[I + D^{-1}L\omega])^{-1}[(1 - \omega)D - \omega U]$$

$$= (I + D^{-1}L\omega)^{-1}D^{-1}[(1 - \omega)D - \omega U] = (I + D^{-1}L\omega)^{-1}[(1 - \omega)I - \omega D^{-1}U].$$
We will need the following facts about determinants.

\[ \det(AB) = \det A \det B, \quad \det A = \lambda_1 \lambda_2 \cdots \lambda_n, \quad \det(L + D) = \det(D + U) = d_{11} \cdots d_{nn}. \]

Now

\[ \rho(M) = \max_i |\lambda_i| \geq |\lambda_1 \lambda_2 \cdots \lambda_n|^{1/n} = |\det M|^{1/n}. \]

But

\[ \det M = \det[(I + D^{-1}L\omega)^{-1}] \cdot \det[(1 - \omega)I - \omega D^{-1}U] \]

\[ = \frac{\det[(1 - \omega)I - \omega D^{-1}U]}{\det[(I + D^{-1}L\omega)]} = 1 \cdot (1 - \omega)^n = (1 - \omega)^n, \]

where we have used the fact that \((I + D^{-1}L\omega)\) and \([(1 - \omega)I - \omega D^{-1}U]\) are triangular matrices. So

\[ \rho(M) \geq |\det M|^{1/n} = |(1 - \omega)^n|^{1/n} = |1 - \omega|. \]

Hence \(\rho(M) > 1\) unless \(0 < \omega < 2\) and so if SOR converges, then \(0 < \omega < 2\). \(\square\)

**Theorem 11.** If \(0 < \omega < 2\) and \(A\) is real and positive definite, then SOR converges.

The usual choice is \(1 < \omega < 2\).

We can also define symmetric versions of the Jacobi, Gauss-Seidel, and SOR methods. For example, if we first define a backward version of the Gauss-Seidel method, i.e.,

\[ x^{k+1} = -(U + D)^{-1}Lx^k + (U + D)^{-1}b, \]

then a symmetric version can be defined by combining the forward and backward versions as follows.

\[ x^{k+1/2} = -(L + D)^{-1}Ux^k + (L + D)^{-1}b, \quad x^{k+1} = -(U + D)^{-1}Lx^{k+1/2} + (U + D)^{-1}b. \]

Eliminating \(x^{k+1/2}\), we get

\[ x^{k+1} = (U + D)^{-1}L(L + D)^{-1}Ux^k + (U + D)^{-1}[I - L(L + D)^{-1}]b. \]

Symmetric versions of the other methods are defined in a similar way.