CHAPTER

4

BASIC ITERATIVE METHODS

The first iterative methods used for solving large linear systems were based on *relaxation of the coordinates*. Beginning with a given approximate solution, these methods modify the components of the approximation, one or a few at a time and in a certain order, until convergence is reached. Each of these modifications, called relaxation steps, is aimed at annihilating one or a few components of the residual vector. Now, these techniques are rarely used separately. However, when combined with the more efficient methods described in later chapters, they can be quite successful. Moreover, there are a few application areas where variations of these methods are still quite popular.

4.1 JACOBI, GAUSS-SEIDEL, AND SOR

This chapter begins by reviewing the basic iterative methods for solving linear systems. Given an $n \times n$ real matrix A and a real n-vector b, the problem considered is: Find x belonging to $\mathbb{R}^n$ such that

$$Ax = b \tag{4.1}$$

Equation (4.1) is a *linear system*, A is the *coefficient matrix*, b is the *right-hand side* vector, and x is the *vector of unknowns*. Most of the methods covered in this chapter involve passing from one iterate to the next by modifying one or a few components of an approximate vector solution at a time. This is natural since there are simple criteria when modifying a component in order to improve an iterate. One example is to annihilate some component(s) of the residual vector $b - Ax$. The convergence of these methods is rarely guaranteed for all matrices, but a large body of theory exists for the case where the coefficient matrix arises from the finite difference discretization of Elliptic Partial Differential Equations.

We begin with the decomposition

$$A = D - E - F, \tag{4.2}$$

in which D is the diagonal of A, $-E$ its strict lower part, and $-F$ its strict upper part, as illustrated in Figure 4.1. It is always assumed that the diagonal entries of A are all nonzero.

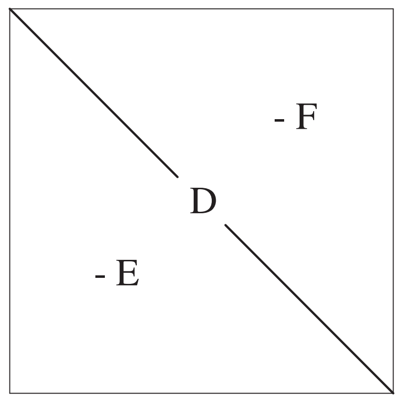

Figure 4.1 *Initial partitioning of matrix A.*

The Jacobi iteration determines the i-th component of the next approximation so as to annihilate the i-th component of the residual vector. In the following, $\xi_i^{(k)}$ denotes the i-th component of the iterate x_k and β_i the i-th component of the right-hand side b. Thus, writing

$$(b - Ax_{k+1})_i = 0, \tag{4.3}$$

in which $(y)_i$ represents the i-th component of the vector y, yields

$$a_{ii}\xi_i^{(k+1)} = -\sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij}\xi_j^{(k)} + \beta_i,$$

or

$$\xi_i^{(k+1)} = \frac{1}{a_{ii}}\left(\beta_i - \sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij}\xi_j^{(k)}\right) \quad i = 1, \ldots, n. \tag{4.4}$$

This is a component-wise form of the Jacobi iteration. All components of the next iterate can be grouped into the vector x_{k+1}. The above notation can be used to rewrite the Jacobi iteration (4.4) in vector form as

$$x_{k+1} = D^{-1}(E + F)x_k + D^{-1}b. \tag{4.5}$$

Similarly, the Gauss-Seidel iteration corrects the i-th component of the current approximate solution, in the order $i = 1, 2, \ldots, n$, again to annihilate the i-th component of the residual. However, this time the approximate solution is updated immediately after the new component is determined. The newly computed components $\xi_i^{(k)}$, $i = 1, 2, \ldots, n$ can be changed within a working vector which is redefined at each relaxation step. Thus, since the order is $i = 1, 2, \ldots$, the result at the i-th step is

$$\beta_i - \sum_{j=1}^{i-1} a_{ij}\xi_j^{(k+1)} - a_{ii}\xi_i^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}\xi_j^{(k)} = 0, \tag{4.6}$$

which leads to the iteration,

$$\xi_i^{(k+1)} = \frac{1}{a_{ii}}\left(-\sum_{j=1}^{i-1} a_{ij}\xi_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}\xi_j^{(k)} + \beta_i\right),\ i = 1, \ldots, n. \tag{4.7}$$

The defining equation (4.6) can be written as

$$b + Ex_{k+1} - Dx_{k+1} + Fx_k = 0,$$

which leads immediately to the vector form of the Gauss-Seidel iteration

$$x_{k+1} = (D - E)^{-1}Fx_k + (D - E)^{-1}b. \tag{4.8}$$

Computing the new approximation in (4.5) requires multiplying by the inverse of the diagonal matrix D. In (4.8) a triangular system must be solved with $D - E$, the lower triangular part of A. Thus, the new approximation in a Gauss-Seidel step can be determined either by solving a triangular system with the matrix $D - E$ or from the relation (4.7).

A *backward* Gauss-Seidel iteration can also be defined as

$$(D - F)x_{k+1} = Ex_k + b, \tag{4.9}$$

which is equivalent to making the coordinate corrections in the order $n, n-1, \ldots, 1$. A Symmetric Gauss-Seidel Iteration consists of a forward sweep followed by a backward sweep.

The Jacobi and the Gauss-Seidel iterations are both of the form

$$Mx_{k+1} = Nx_k + b = (M - A)x_k + b, \tag{4.10}$$

in which

$$A = M - N \tag{4.11}$$

is a *splitting* of A, with $M = D$ for Jacobi, $M = D - E$ for forward Gauss-Seidel, and $M = D - F$ for backward Gauss-Seidel. An iterative method of the form (4.10) can be defined for any splitting of the form (4.11) where M is nonsingular. *Overrelaxation* is based on the splitting

$$\omega A = (D - \omega E) - (\omega F + (1 - \omega)D),$$

and the corresponding *Successive Over Relaxation* (SOR) method is given by the recursion

$$(D - \omega E)x_{k+1} = [\omega F + (1 - \omega)D]x_k + \omega b. \tag{4.12}$$

The above iteration corresponds to the relaxation sequence

$$\xi_i^{(k+1)} = \omega\xi_i^{GS} + (1 - \omega)\xi_i^{(k)}, i = 1, 2, \ldots, n,$$

in which ξ_i^{GS} is defined by the expression in the right-hand side of (4.7). A backward SOR sweep can be defined analogously to the backward Gauss-Seidel sweep (4.9).

A Symmetric SOR (SSOR) step consists of the SOR step (4.12) followed by a backward SOR step,

$$\begin{aligned}(D-\omega E)x_{k+1/2} &= [\omega F+(1-\omega)D]x_k+\omega b\\ (D-\omega F)x_{k+1} &= [\omega E+(1-\omega)D]x_{k+1/2}+\omega b\end{aligned}$$

This gives the recurrence

$$x_{k+1}=G_\omega x_k+f_\omega,$$

where

$$\begin{aligned}G_\omega &= (D-\omega F)^{-1}(\omega E+(1-\omega)D)\ \times\\ &\qquad (D-\omega E)^{-1}(\omega F+(1-\omega)D), \qquad (4.13)\\ f_\omega &= \omega(D-\omega F)^{-1}\left(I+[\omega E+(1-\omega)D](D-\omega E)^{-1}\right)\ b. \qquad (4.14)\end{aligned}$$

Observing that

$$\begin{aligned}[\omega E+(1-\omega)D](D-\omega E)^{-1} &= [-(D-\omega E)+(2-\omega)D](D-\omega E)^{-1}\\ &= -I+(2-\omega)D(D-\omega E)^{-1},\end{aligned}$$

f_ω can be rewritten as

$$f_\omega=\omega(2-\omega)\ (D-\omega F)^{-1}D(D-\omega E)^{-1}b.$$

4.1.1 BLOCK RELAXATION SCHEMES

Block relaxation schemes are generalizations of the “point” relaxation schemes described above. They update a whole set of components at each time, typically a subvector of the solution vector, instead of only one component. The matrix A and the right-hand side and solution vectors are partitioned as follows:

$$A=\begin{pmatrix}A_{11} & A_{12} & A_{13} & \cdots & A_{1p}\\ A_{21} & A_{22} & A_{23} & \cdots & A_{2p}\\ A_{31} & A_{32} & A_{33} & \cdots & A_{3p}\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ A_{p1} & A_{p2} & \cdots & \cdots & A_{pp}\end{pmatrix},\ x=\begin{pmatrix}\xi_1\\ \xi_2\\ \xi_3\\ \vdots\\ \xi_p\end{pmatrix},\ b=\begin{pmatrix}\beta_1\\ \beta_2\\ \beta_3\\ \vdots\\ \beta_p\end{pmatrix}, \qquad (4.15)$$

in which the partitionings of b and x into subvectors β_i and ξ_i are identical and compatible with the partitioning of A. Thus, for any vector x partitioned as in (4.15),

$$(Ax)_i=\sum_{j=1}^{p}A_{ij}\xi_j,$$

in which $(y)_i$ denotes the i-th component of the vector i according to the above partitioning. The diagonal blocks in A are square and assumed nonsingular.

Now define, similarly to the scalar case, the splitting

$$A=D-E-F$$

with

$$D = \begin{pmatrix} A_{11} & & & \\ & A_{22} & & \\ & & \ddots & \\ & & & A_{pp} \end{pmatrix}, \tag{4.16}$$

$$E = -\begin{pmatrix} O & & & \\ A_{21} & O & & \\ \vdots & \vdots & \ddots & \\ A_{p1} & A_{p2} & \cdots & O \end{pmatrix}, \quad F = -\begin{pmatrix} O & A_{12} & \cdots & A_{1p} \\ & O & \cdots & A_{2p} \\ & & \ddots & \vdots \\ & & & O \end{pmatrix}.$$

With these definitions, it is easy to generalize the previous three iterative procedures defined earlier, namely, Jacobi, Gauss-Seidel, and SOR. For example, the block Jacobi iteration is now defined as a technique in which the new subvectors $\xi_i^{(k)}$ are all replaced according to

$$A_{ii}\xi_i^{(k+1)} = ((E+F)x_k)_i + \beta_i$$

or,

$$\xi_i^{(k+1)} = A_{ii}^{-1}\left((E+F)x_k\right)_i + A_{ii}^{-1}\beta_i, \quad i = 1, \ldots, p,$$

which leads to the same equation as before,

$$x_{k+1} = D^{-1}(E+F)x_k + D^{-1}b,$$

except that the meanings of D, E, and F have changed to their block analogues.

With finite difference approximations of PDEs, it is standard to block the variables and the matrix by partitioning along whole lines of the mesh. For example, for the two-dimensional mesh illustrated in Figure 2.5, this partitioning is

$$\xi_1 = \begin{pmatrix} u_{11} \\ u_{12} \\ u_{13} \\ u_{14} \\ u_{15} \end{pmatrix}, \quad \xi_2 = \begin{pmatrix} u_{21} \\ u_{22} \\ u_{23} \\ u_{24} \\ u_{25} \end{pmatrix}, \quad \xi_3 = \begin{pmatrix} u_{31} \\ u_{32} \\ u_{33} \\ u_{34} \\ u_{35} \end{pmatrix}.$$

This corresponds to the mesh 2.5 of Chapter 2, whose associated matrix pattern is shown in Figure 2.6. A relaxation can also be defined along the vertical instead of the horizontal lines. Techniques of this type are often known as *line relaxation* techniques.

In addition, a block can also correspond to the unknowns associated with a few consecutive lines in the plane. One such blocking is illustrated in Figure 4.2 for a 6×6 grid. The corresponding matrix with its block structure is shown in Figure 4.3. An important difference between this partitioning and the one corresponding to the single-line partitioning is that now the matrices A_{ii} are block-tridiagonal instead of tridiagonal. As a result, solving linear systems with A_{ii} may be much more expensive. On the other hand, the number of iterations required to achieve convergence often decreases rapidly as the block-size increases.

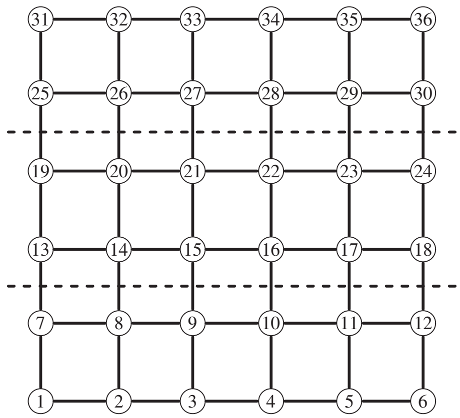

Figure 4.2 *Partitioning of a* 6×6 *square mesh into three subdomains.*

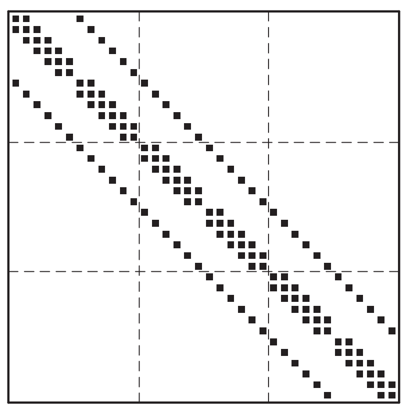

Figure 4.3 *Matrix associated with the mesh of Figure 4.2.*

Finally, block techniques can be defined in more general terms. First, by using blocks that allow us to update arbitrary groups of components, and second, by allowing the blocks to overlap. Since this is a form of the domain-decomposition method which will be seen later, we define the approach carefully. So far, our partition has been based on an actual *set-partition* of the variable set $S = \{1, 2, \ldots, n\}$ into subsets $S_1, S_2, \ldots, S_p$, with the condition that two distinct subsets are disjoint. In set theory, this is called a *partition* of S. More generally, a *set-decomposition* of S removes the constraint of disjointness. In other words it is required that the union of the subsets S_i's be equal to S:

$$S_i \subseteq S, \qquad \bigcup_{i=1,\cdots,p} S_i = S.$$

In the following, n_i denotes the size of S_i and the subset S_i is of the form,

$$S_i = \{m_i(1), m_i(2), \ldots m_i(n_i)\}.$$

A general block Jacobi iteration can be defined as follows. Let V_i be the $n \times n_i$ matrix

$$V_i = [e_{m_i(1)}, e_{m_i(2)}, \dots e_{m_i(n_i)}]$$

and

$$W_i = [\eta_{m_i(1)} e_{m_i(1)}, \eta_{m_i(2)} e_{m_i(2)}, \dots, \eta_{m_i(n_i)} e_{m_i(n_i)}],$$

where each e_j is the j-th column of the $n \times n$ identity matrix, and $\eta_{m_i(j)}$ represents a weight factor chosen so that

$$W_i^T V_i = I.$$

When there is no overlap, i.e., when the S_i's form a partition of the whole set $\{1, 2, \dots, n\}$, then define $\eta_{m_i(j)} = 1$.

Let A_{ij} be the $n_i \times n_j$ matrix

$$A_{ij} = W_i^T A V_j$$

and define similarly the partitioned vectors

$$\xi_i = W_i^T x, \quad \beta_i = W_i^T b.$$

Note that $V_i W_i^T$ is a projector from $\mathbb{R}^n$ to the subspace K_i spanned by the columns $m_i(1)$, $\dots$, $m_i(n_i)$. In addition, we have the relation

$$x = \sum_{i=1}^{s} V_i \xi_i.$$

The n_i-dimensional vector $W_i^T x$ represents the projection $V_i W_i^T x$ of x with respect to the basis spanned by the columns of V_i. The action of V_i performs the reverse operation. That means $V_i y$ is an extension operation from a vector y in K_i (represented in the basis consisting of the columns of V_i) into a vector $V_i y$ in $\mathbb{R}^n$. The operator W_i^T is termed a *restriction operator* and V_i is an *prolongation operator*.

Each component of the Jacobi iteration can be obtained by imposing the condition that the projection of the residual in the span of S_i be zero, i.e.,

$$W_i^T \left[b - A \left(V_i W_i^T x_{k+1} + \sum_{j \neq i} V_j W_j^T x_k \right) \right] = 0.$$

Remember that $\xi_j = W_j^T x$, which can be rewritten as

$$\xi_i^{(k+1)} = \xi_i^{(k)} + A_{ii}^{-1} W_i^T (b - A x_k). \tag{4.17}$$

This leads to the following algorithm:

ALGORITHM 4.1: General Block Jacobi Iteration

1. *For* $k = 0, 1, \dots,$ *until convergence Do:*
2. *For* $i = 1, 2, \dots, p$ *Do:*
3. *Solve* $A_{ii} \delta_i = W_i^T (b - A x_k)$
4. *Set* $x_{k+1} := x_k + V_i \delta_i$
5. *EndDo*
6. *EndDo*

As was the case with the scalar algorithms, there is only a slight difference between the Jacobi and Gauss-Seidel iterations. Gauss-Seidel immediately updates the component to be corrected at step i, and uses the updated approximate solution to compute the residual vector needed to correct the next component. However, the Jacobi iteration uses the same previous approximation x_k for this purpose. Therefore, the block Gauss-Seidel iteration can be defined algorithmically as follows:

ALGORITHM 4.2: General Block Gauss-Seidel Iteration

1. *Until convergence Do:*
2. *For* $i = 1, 2, \ldots, p$ *Do:*
3. *Solve* $A_{ii}\delta_i = W_i^T(b - Ax)$
4. *Set* $x := x + V_i\delta_i$
5. *EndDo*
6. *EndDo*

From the point of view of storage, Gauss-Seidel is more economical because the new approximation can be overwritten over the same vector. Also, it typically converges faster. On the other hand, the Jacobi iteration has some appeal on parallel computers since the second *Do* loop, corresponding to the p different blocks, can be executed in parallel. Although the point Jacobi algorithm by itself is rarely a successful technique for real-life problems, its block Jacobi variant, when using large enough overlapping blocks, can be quite attractive especially in a parallel computing environment.

4.1.2 ITERATION MATRICES AND PRECONDITIONING

The Jacobi and Gauss-Seidel iterations are of the form

$$x_{k+1} = Gx_k + f, \tag{4.18}$$

in which

$$G_{JA}(A) = I - D^{-1}A, \tag{4.19}$$

$$G_{GS}(A) = I - (D - E)^{-1}A, \tag{4.20}$$

for the Jacobi and Gauss-Seidel iterations, respectively. Moreover, given the matrix splitting

$$A = M - N, \tag{4.21}$$

where A is associated with the linear system (4.1), a *linear fixed-point iteration* can be defined by the recurrence

$$x_{k+1} = M^{-1}Nx_k + M^{-1}b, \tag{4.22}$$

which has the form (4.18) with

$$G = M^{-1}N = M^{-1}(M - A) = I - M^{-1}A, \quad f = M^{-1}b. \tag{4.23}$$

For example, for the Jacobi iteration, $M = D, N = A - D$, while for the Gauss-Seidel iteration, $M = D - E, N = M - A = F$.

The iteration $x_{k+1} = Gx_k + f$ can be viewed as a technique for solving the system

$$(I - G)x = f.$$

Since G has the form $G = I - M^{-1}A$, this system can be rewritten as

$$M^{-1}Ax = M^{-1}b.$$

The above system which has the same solution as the original system is called a *preconditioned system* and M is the *preconditioning matrix* or *preconditioner*. In other words, *a relaxation scheme is equivalent to a fixed-point iteration on a preconditioned system.*

For example, for the Jacobi, Gauss-Seidel, SOR, and SSOR iterations, these preconditioning matrices are, respectively,

$$M_{JA} = D, \tag{4.24}$$

$$M_{GS} = D - E, \tag{4.25}$$

$$M_{SOR} = \frac{1}{\omega}(D - \omega E), \tag{4.26}$$

$$M_{SSOR} = \frac{1}{\omega(2 - \omega)}\,(D - \omega E)D^{-1}(D - \omega F). \tag{4.27}$$

Thus, the Jacobi preconditioner is simply the diagonal of A, while the Gauss-Seidel preconditioner is the lower triangular part of A. The constant coefficients in front of the matrices M_{SOR} and M_{SSOR} only have the effect of scaling the equations of the preconditioned system uniformly. Therefore, they are unimportant in the preconditioning context.

Note that the "preconditioned" system may be a full system. Indeed, there is no reason why M^{-1} should be a sparse matrix (even though M may be sparse), since the inverse of a sparse matrix is not necessarily sparse. This limits the number of techniques that can be applied to solve the preconditioned system. Most of the iterative techniques used only require matrix-by-vector products. In this case, to compute $w = M^{-1}Av$ for a given vector v, first compute $r = Av$ and then solve the system $Mw = r$:

$$r = Av,$$
$$w = M^{-1}r.$$

In some cases, it may be advantageous to exploit the splitting $A = M - N$ and compute $w = M^{-1}Av$ as $w = (I - M^{-1}N)v$ by the procedure

$$r = Nv,$$
$$w = M^{-1}r,$$
$$w := v - w.$$

The matrix N may be sparser than A and the matrix-by-vector product Nv may be less expensive than the product Av. A number of similar but somewhat more complex ideas have been exploited in the context of preconditioned iterative methods. A few of these will be examined in Chapter 9.

4.2 CONVERGENCE

All the methods seen in the previous section define a sequence of iterates of the form

$$x_{k+1} = Gx_k + f, \tag{4.28}$$

in which G is a certain *iteration matrix*. The questions addressed in this section are: (a) if the iteration converges, then is the limit indeed a solution of the original system? (b) under which conditions does the iteration converge? (c) when the iteration does converge, how fast is it?

If the above iteration converges, its limit x satisfies

$$x = Gx + f. \tag{4.29}$$

In the case where the above iteration arises from the splitting $A = M - N$, it is easy to see that the solution x to the above system is identical to that of the original system $Ax = b$. Indeed, in this case the sequence (4.28) has the form

$$x_{k+1} = M^{-1}Nx_k + M^{-1}b$$

and its limit satisfies

$$Mx = Nx + b,$$

or $Ax = b$. This answers question (a). Next, we focus on the other two questions.

4.2.1 GENERAL CONVERGENCE RESULT

If $I - G$ is nonsingular then there is a solution x_* to the equation (4.29). Subtracting (4.29) from (4.28) yields

$$x_{k+1} - x_* = G(x_k - x_*) = \cdots = G^{k+1}(x_0 - x_*). \tag{4.30}$$

Standard results seen in Chapter 1 imply that if the spectral radius of the iteration matrix G is less than unity, then $x_k - x_*$ converges to zero and the iteration (4.28) converges toward the solution defined by (4.29). Conversely, the relation

$$x_{k+1} - x_k = G(x_k - x_{k-1}) = \cdots = G^k(f - (I - G)x_0).$$

shows that if the iteration converges for *any* x_0 *and* f then $G^k v$ converges to zero for any vector v. As a result, $\rho(G)$ must be less than unity and the following theorem is proved:

THEOREM 4.1 *Let G be a square matrix such that $\rho(G) < 1$. Then $I - G$ is nonsingular and the iteration (4.28) converges for any f and x_0. Conversely, if the iteration (4.28) converges for for any f and x_0, then $\rho(G) < 1$.*

Since it is expensive to compute the spectral radius of a matrix, sufficient conditions that guarantee convergence can be useful in practice. One such sufficient condition could be obtained by utilizing the inequality, $\rho(G) \le \|G\|$, for any matrix norm.

COROLLARY 4.1 *Let G be a square matrix such that $\|G\| < 1$ for some matrix norm $\|.\|$. Then $I - G$ is nonsingular and the iteration (4.28) converges for any initial vector x_0.*

Apart from knowing that the sequence (4.28) converges, it is also desirable to know *how fast* it converges. The error $d_k = x_k - x_*$ at step k satisfies

$$d_k = G^k d_0.$$

The matrix G can be expressed in the Jordan canonical form as $G = XJX^{-1}$. Assume for simplicity that there is only one eigenvalue of G of largest modulus and call it λ. Then

$$d_k = \lambda^k X \left(\frac{J}{\lambda}\right)^k X^{-1} d_0.$$

A careful look at the powers of the matrix J/λ shows that all its blocks, except the block associated with the eigenvalue λ, converge to zero as k tends to infinity. Let this Jordan block be of size p and of the form

$$J_\lambda = \lambda I + E,$$

where E is nilpotent of index p, i.e., $E^p = 0$. Then, for $k \geq p$,

$$J_\lambda^k = (\lambda I + E)^k = \lambda^k (I + \lambda^{-1} E)^k = \lambda^k \left(\sum_{i=0}^{p-1} \lambda^{-i} \binom{k}{i} E^i \right).$$

If k is large enough, then for any λ the dominant term in the above sum is the last term, i.e.,

$$J_\lambda^k \approx \lambda^{k-p+1} \binom{k}{p-1} E^{p-1}.$$

Thus, the norm of $d_k = G^k d_0$ has the asymptotical form

$$\|d_k\| \approx C \times |\lambda^{k-p+1}| \binom{k}{p-1},$$

where C is some constant. The *convergence factor* of a sequence is the limit

$$\rho = \lim_{k \to \infty} \left(\frac{\|d_k\|}{\|d_0\|} \right)^{1/k}.$$

It follows from the above analysis that $\rho = \rho(G)$. The *convergence rate* τ is the (natural) logarithm of the inverse of the convergence factor

$$\tau = -\ln \rho.$$

The above definition depends on the initial vector x_0, so it may be termed a *specific* convergence factor. A *general* convergence factor can also be defined by

$$\phi = \lim_{k \to \infty} \left(\max_{x_0 \in \mathbb{R}^n} \frac{\|d_k\|}{\|d_0\|} \right)^{1/k}.$$

This factor satisfies

$$\phi = \lim_{k\to\infty} \left(\max_{d_0 \in \mathbb{R}^n} \frac{\|G^k d_0\|}{\|d_0\|} \right)^{1/k}$$
$$= \lim_{k\to\infty} \left(\|G^k\| \right)^{1/k} = \rho(G).$$

Thus, the global asymptotic convergence factor is equal to the spectral radius of the iteration matrix G. The *general* convergence rate differs from the *specific* rate only when the initial error does not have any components in the invariant subspace associated with the dominant eigenvalue. Since it is hard to know this information in advance, the *general* convergence factor is more useful in practice.

Example 4.1 Consider the simple example of *Richardson's Iteration*,

$$x_{k+1} = x_k + \alpha \left(b - Ax_k \right), \tag{4.31}$$

where α is a nonnegative scalar. This iteration can be rewritten as

$$x_{k+1} = (I - \alpha A)x_k + \alpha b. \tag{4.32}$$

Thus, the iteration matrix is $G_\alpha = I - \alpha A$ and the convergence factor is $\rho(I - \alpha A)$. Assume that the eigenvalues $\lambda_i, i = 1, \ldots, n$, are all real and such that,

$$\lambda_{min} \le \lambda_i \le \lambda_{max}.$$

Then, the eigenvalues μ_i of G_α are such that

$$1 - \alpha\lambda_{max} \le \mu_i \le 1 - \alpha\lambda_{min}.$$

In particular, if $\lambda_{min} < 0$ and $\lambda_{max} > 0$, at least one eigenvalue is > 1, and so $\rho(G_\alpha) > 1$ for any α. In this case the method will always diverge for some initial guess. Let us assume that all eigenvalues are positive, i.e., $\lambda_{min} > 0$. Then, the following conditions must be satisfied in order for the method to converge:

$$1 - \alpha\lambda_{min} < 1,$$
$$1 - \alpha\lambda_{max} > -1.$$

The first condition implies that $\alpha > 0$, while the second requires that $\alpha \le 2/\lambda_{max}$. In other words, the method converges for any scalar α which satisfies

$$0 < \alpha < \frac{2}{\lambda_{max}}.$$

The next question is: What is the best value α_{opt} for the parameter α, i.e., the value of α which minimizes $\rho(G_\alpha)$? The spectral radius of G_α is

$$\rho(G_\alpha) = \max\{|1 - \alpha\lambda_{min}|,\ |1 - \alpha\lambda_{max}|\}.$$

This function of α is depicted in Figure 4.4. As the curve shows, the best possible α is reached at the point where the curve $|1 - \lambda_{max}\alpha|$ with positive slope crosses the curve $|1 - \lambda_{min}\alpha|$ with negative slope, i.e., when

$$-1 + \lambda_{max}\alpha = 1 - \lambda_{min}\alpha.$$

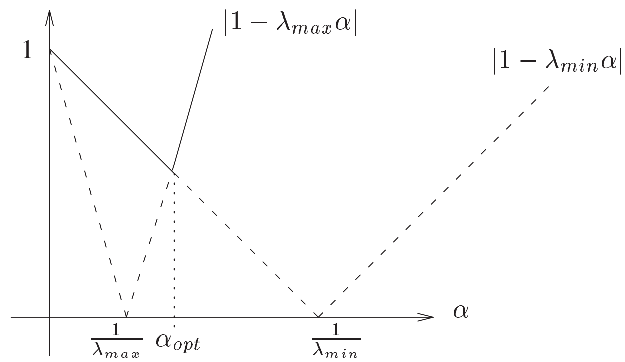

Figure 4.4 *The curve $\rho(G_\alpha)$ as a function of α.*

This gives

$$\alpha_{opt} = \frac{2}{\lambda_{min} + \lambda_{max}}. \tag{4.33}$$

Replacing this in one of the two curves gives the corresponding optimal spectral radius

$$\rho_{opt} = \frac{\lambda_{max} - \lambda_{min}}{\lambda_{max} + \lambda_{min}}.$$

This expression shows the difficulty with the presence of small and large eigenvalues. The convergence rate can be extremely small for realistic problems. In addition, to achieve good convergence, eigenvalue estimates are required in order to obtain the optimal or a near-optimal α, and this may cause difficulties. Finally, since λ_{max} can be very large, the curve $\rho(G_\alpha)$ can be extremely sensitive near the optimal value of α. These observations are common to many iterative methods that depend on an acceleration parameter.

4.2.2 REGULAR SPLITTINGS

DEFINITION 4.1 *Let A, M, N be three given matrices satisfying $A = M - N$. The pair of matrices M, N is a regular splitting of A, if M is nonsingular and M^{-1} and N are nonnegative.*

With a regular splitting, we associate the iteration

$$x_{k+1} = M^{-1}Nx_k + M^{-1}b. \tag{4.34}$$

The question asked is: Under which conditions does such an iteration converge? The following result, which generalizes Theorem 1.15, gives the answer.

THEOREM 4.2 *Let M, N be a regular splitting of a matrix A. Then $\rho(M^{-1}N) < 1$ if and only if A is nonsingular and A^{-1} is nonnegative.*

Proof. Define $G = M^{-1}N$. From the fact that $\rho(G) < 1$, and the relation

$$A = M(I - G) \tag{4.35}$$

it follows that A is nonsingular. The assumptions of Theorem 1.15 are satisfied for the matrix G since $G = M^{-1}N$ is nonnegative and $\rho(G) < 1$. Therefore, $(I - G)^{-1}$ is nonnegative as is $A^{-1} = (I - G)^{-1} M^{-1}$.

To prove the sufficient condition, assume that A is nonsingular and that its inverse is nonnegative. Since A and M are nonsingular, the relation (4.35) shows again that $I - G$ is nonsingular and in addition,

$$\begin{aligned} A^{-1}N &= \left(M(I - M^{-1}N)\right)^{-1} N \\ &= (I - M^{-1}N)^{-1} M^{-1}N \\ &= (I - G)^{-1} G. \end{aligned} \tag{4.36}$$

Clearly, $G = M^{-1}N$ is nonnegative by the assumptions, and as a result of the Perron-Frobenius theorem, there is a nonnegative eigenvector x associated with $\rho(G)$ which is an eigenvalue, such that

$$Gx = \rho(G)x.$$

From this and by virtue of (4.36), it follows that

$$A^{-1}Nx = \frac{\rho(G)}{1 - \rho(G)} x.$$

Since x and $A^{-1}N$ are nonnegative, this shows that

$$\frac{\rho(G)}{1 - \rho(G)} \geq 0$$

and this can be true only when $0 \leq \rho(G) \leq 1$. Since $I - G$ is nonsingular, then $\rho(G) \neq 1$, which implies that $\rho(G) < 1$. ■

This theorem establishes that the iteration (4.34) always converges, if M, N is a regular splitting and A is an M-matrix.

4.2.3 DIAGONALLY DOMINANT MATRICES

We begin with a few standard definitions.

DEFINITION 4.2 *A matrix A is*

- *(weakly) diagonally dominant if*

$$|a_{jj}| \geq \sum_{\substack{i=1 \\ i \neq j}}^{i=n} |a_{ij}|, \quad j = 1, \ldots, n.$$

- *strictly diagonally dominant if*

$$|a_{jj}| > \sum_{\substack{i=1\\ i\neq j}}^{i=n} |a_{ij}|, \quad j = 1, \ldots, n.$$

- *irreducibly diagonally dominant if A is irreducible, and*

$$|a_{jj}| \geq \sum_{\substack{i=1\\ i\neq j}}^{i=n} |a_{ij}|, \quad j = 1, \ldots, n.$$

  with strict inequality for at least one j.

Often the term diagonally dominant is used instead of *weakly* diagonally dominant.

Diagonal dominance is related to an important result in Numerical Linear Algebra known as Gershgorin's theorem. This theorem allows rough locations for all the eigenvalues of A to be determined. In some situations, it is desirable to determine these locations in the complex plane by directly exploiting some knowledge of the entries of the matrix A. The simplest such result is the bound

$$|\lambda_i| \leq \|A\|$$

for any matrix norm. Gershgorin's theorem provides a more precise localization result.

THEOREM 4.3 ***(Gershgorin)*** *Any eigenvalue λ of a matrix A is located in one of the closed discs of the complex plane centered at a_{ii} and having the radius*

$$\rho_i = \sum_{\substack{j=1\\ j\neq i}}^{j=n} |a_{ij}|.$$

In other words,

$$\forall\ \lambda\ \in \sigma(A), \quad \exists\, i \quad \text{such that} \quad |\lambda - a_{ii}| \leq \sum_{\substack{j=1\\ j\neq i}}^{j=n} |a_{ij}|. \tag{4.37}$$

Proof. Let x be an eigenvector associated with an eigenvalue λ, and let m be the index of the component of largest modulus in x. Scale x so that $|\xi_m| = 1$, and $|\xi_i| \leq 1$, for $i \neq m$. Since x is an eigenvector, then

$$(\lambda - a_{mm})\xi_m = -\sum_{\substack{j=1\\ j\neq m}}^{n} a_{mj}\xi_j,$$

which gives

$$|\lambda - a_{mm}| \leq \sum_{\substack{j=1\\ j\neq m}}^{n} |a_{mj}||\xi_j| \leq \sum_{\substack{j=1\\ j\neq m}}^{n} |a_{mj}| = \rho_m. \tag{4.38}$$

This completes the proof. ∎

Since the result also holds for the transpose of A, a version of the theorem can also be formulated based on column sums instead of row sums.

The n discs defined in the theorem are called Gershgorin discs. The theorem states that the union of these n discs contains the spectrum of A. It can also be shown that if there are m Gershgorin discs whose union S is disjoint from all other discs, then S contains exactly m eigenvalues (counted with their multiplicities). For example, when one disc is disjoint from the others, then it must contain exactly one eigenvalue.

An additional refinement which has important consequences concerns the particular case when A is irreducible.

THEOREM 4.4 *Let A be an irreducible matrix, and assume that an eigenvalue λ of A lies on the boundary of the union of the n Gershgorin discs. Then λ lies on the boundary of all Gershgorin discs.*

Proof. As in the proof of Gershgorin's theorem, let x be an eigenvector associated with λ, with $|\xi_m| = 1$, and $|\xi_i| \leq 1$, for $i \neq m$. Start from equation (4.38) in the proof of Gershgorin's theorem which states that the point λ belongs to the m-th disc. In addition, λ belongs to the boundary of the union of all the discs. As a result, it cannot be an interior point to the disc $D(\lambda, \rho_m)$. This implies that $|\lambda - a_{mm}| = \rho_m$. Therefore, the inequalities in (4.38) both become equalities:

$$|\lambda - a_{mm}| = \sum_{\substack{j=1\\ j\neq m}}^{n} |a_{mj}||\xi_j| = \sum_{\substack{j=1\\ j\neq m}}^{n} |a_{mj}| = \rho_m. \tag{4.39}$$

Let j be any integer $1 \leq j \leq n$. Since A is irreducible, its graph is connected and, therefore, there exists a path from node m to node j in the adjacency graph. Let this path be

$$m, m_1, m_2, \ldots, m_k = j.$$

By definition of an edge in the adjacency graph, $a_{m,m_1} \neq 0$. Because of the equality in (4.39), it is necessary that $|\xi_j| = 1$ for any nonzero ξ_j. Therefore, $|\xi_{m_1}|$ must be equal to one. Now repeating the argument with m replaced by m_1 shows that the following equality holds:

$$|\lambda - a_{m_1,m_1}| = \sum_{\substack{j=1\\ j\neq m_1}}^{n} |a_{m_1,j}||\xi_j| = \sum_{\substack{j=1\\ j\neq m_1}}^{n} |a_{m_1,j}| = \rho_{m_1}. \tag{4.40}$$

The argument can be continued showing each time that

$$|\lambda - a_{m_i,m_i}| = \rho_{m_i}, \tag{4.41}$$

and this is valid for $i = 1, \ldots, k$. In the end, it will be proved that λ belongs to the boundary of the j-th disc for an arbitrary j. ∎

An immediate corollary of the Gershgorin theorem and the above theorem follows.

COROLLARY 4.2 *If a matrix A is strictly diagonally dominant or irreducibly diagonally dominant, then it is nonsingular.*

Proof. If a matrix is strictly diagonally dominant, then the union of the Gershgorin disks excludes the origin, so $\lambda = 0$ cannot be an eigenvalue. Assume now that it is only irreducibly diagonal dominant. Then if it is singular, the zero eigenvalue lies on the boundary of the union of the Gershgorin disks. In this situation, according to the previous theorem, this eigenvalue should lie on the boundary of all the disks. This would mean that

$$|a_{jj}| = \sum_{\substack{i=1\\ i\neq j}}^{n} |a_{ij}| \quad \text{for} \quad j = 1, \ldots, n,$$

which contradicts the assumption of irreducible diagonal dominance. ■

The following theorem can now be stated.

THEOREM 4.5 *If A is a strictly diagonally dominant or an irreducibly diagonally dominant matrix, then the associated Jacobi and Gauss-Seidel iterations converge for any x_0.*

Proof. We first prove the results for strictly diagonally dominant matrices. Let λ be the dominant eigenvalue of the iteration matrix $M_J = D^{-1}(E + F)$ for Jacobi and $M_G = (D - E)^{-1}F$ for Gauss-Seidel. As in the proof of Gershgorin's theorem, let x be an eigenvector associated with λ, with $|\xi_m| = 1$, and $|\xi_i| \leq 1$, for $i \neq 1$. Start from equation (4.38) in the proof of Gershgorin's theorem which states that for M_J,

$$|\lambda| \leq \sum_{\substack{j=1\\ j\neq m}}^{n} \frac{|a_{mj}|}{|a_{mm}|}|\xi_j| \leq \sum_{\substack{j=1\\ j\neq m}}^{n} \frac{|a_{mj}|}{|a_{mm}|} < 1.$$

This proves the result for Jacobi's method.

For the Gauss-Seidel iteration, write the m-th row of the equation $Fx = \lambda(D - E)x$ in the form

$$\sum_{j<m} a_{mj}\xi_j = \lambda \left(a_{mm}\xi_m + \sum_{j>m} a_{mj}\xi_j \right),$$

which yields the inequality

$$|\lambda| \leq \frac{\sum_{j<m} |a_{mj}||\xi_j|}{|a_{mm}| - \sum_{j>m} |a_{mj}||\xi_j|} \leq \frac{\sum_{j<m} |a_{mj}|}{|a_{mm}| - \sum_{j>m} |a_{mj}|}.$$

The last term in the above equation has the form $\sigma_2/(d - \sigma_1)$ with d, σ_1, σ_2 all nonnegative and $d - \sigma_1 - \sigma_2 > 0$. Therefore,

$$|\lambda| \leq \frac{\sigma_2}{\sigma_2 + (d - \sigma_2 - \sigma_1)} < 1.$$

In the case when the matrix is only irreducibly diagonally dominant, the above proofs only show that $\rho(M^{-1}N) \leq 1$, where $M^{-1}N$ is the iteration matrix for either Jacobi or Gauss-Seidel. A proof by contradiction will be used to show that in fact $\rho(M^{-1}N) < 1$. Assume that λ is an eigenvalue of $M^{-1}N$ with $|\lambda| = 1$. Then the matrix $M^{-1}N - \lambda I$ would be singular and, as a result, $A' = N - \lambda M$ would also be singular. Since $|\lambda| = 1$, it is clear that A' is also an irreducibly diagonally dominant matrix. This would contradict Corollary 4.2. ■

4.2.4 SYMMETRIC POSITIVE DEFINITE MATRICES

It is possible to show that when A is Symmetric Positive Definite, then SOR will converge for any ω in the open interval $(0, 2)$ and for any initial guess x_0. In fact, the reverse is also true under certain assumptions.

THEOREM 4.6 *If A is symmetric with positive diagonal elements and for $0 < \omega < 2$, SOR converges for any x_0 if and only if A is positive definite.*

4.2.5 PROPERTY A AND CONSISTENT ORDERINGS

A number of properties which are related to the graph of a finite difference matrix are now defined. The first of these properties is called Property A. A matrix has Property A if its graph is *bipartite*. This means that the graph is two-colorable in the sense defined in Chapter 3: Its vertices can be partitioned in two sets in such a way that no two vertices in the same set are connected by an edge. Note that, as usual, the self-connecting edges which correspond to the diagonal elements are ignored.

DEFINITION 4.3 *A matrix has Property A if the vertices of its adjacency graph can be partitioned in two sets S_1 and S_2, so that any edge in the graph links a vertex of S_1 to a vertex of S_2.*

In other words, nodes from the first set are connected only to nodes from the second set and vice versa. This definition is illustrated in Figure 4.5.

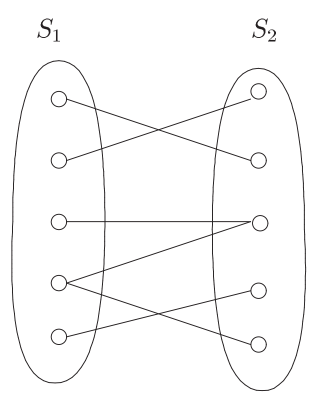

Figure 4.5 *Graph illustration of Property A.*

An alternative definition is that a matrix has Property A if it can be permuted into a matrix with the following structure:

$$A' = \begin{pmatrix} D_1 & -F \\ -E & D_2 \end{pmatrix}, \tag{4.42}$$

where D_1 and D_2 are diagonal matrices. This structure can be obtained by first labeling all the unknowns in S_1 from 1 to n_1, in which $n_1 = |S_1|$ and the rest from $n_1 + 1$ to n. Note that the Jacobi iteration matrix will have the same structure except that the D_1, D_2 blocks will be replaced by zero blocks. These Jacobi iteration matrices satisfy an important property stated in the following proposition.

PROPOSITION 4.1 *Let B be a matrix with the following structure:*

$$B = \begin{pmatrix} O & B_{12} \\ B_{21} & O \end{pmatrix}, \tag{4.43}$$

and let L and U be the lower and upper triangular parts of B, respectively. Then

1. *If μ is an eigenvalue of B, then so is $-\mu$.*
2. *The eigenvalues of the matrix*

$$B(\alpha) = \alpha L + \frac{1}{\alpha} U$$

defined for $\alpha \neq 0$ are independent of α.

Proof. The first property is shown by simply observing that if $\binom{x}{v}$ is an eigenvector associated with μ, then $\binom{x}{-v}$ is an eigenvector of B associated with the eigenvalue $-\mu$.

Consider the second property. For any α, the matrix $B(\alpha)$ is similar to B, i.e., $B(\alpha) = XBX^{-1}$ with X defined by

$$X = \begin{pmatrix} 1 & O \\ O & \alpha \end{pmatrix}.$$

This proves the desired result ∎

A definition which generalizes this important property is *consistently ordered matrices*. Varga [213] calls a consistently ordered matrix one for which the eigenvalues of $B(\alpha)$ are independent of α. Another definition given by Young [232] considers a specific class of matrices which generalize this property. We will use this definition here. Unlike Property A, the consistent ordering property depends on the initial ordering of the unknowns.

DEFINITION 4.4 *A matrix is said to be consistently ordered if the vertices of its adjacency graph can be partitioned in p sets $S_1, S_2, \ldots, S_p$ with the property that any two adjacent vertices i and j in the graph belong to two consecutive partitions S_k and $S_{k'}$, with $k' = k - 1$, if $j < i$, and $k' = k + 1$, if $j > i$.*

It is easy to show that consistently ordered matrices satisfy property A: the first color is made up of all the partitions S_i with odd i and the second with the partitions S_i with even i.

Example 4.2 Block tridiagonal matrices of the form

$$T = \begin{pmatrix} D_1 & T_{12} & & & \\ T_{21} & D_2 & T_{23} & & \\ & T_{32} & D_3 & \ddots & \\ & & \ddots & \ddots & T_{p-1,p} \\ & & & T_{p,p-1} & D_p \end{pmatrix}$$

whose diagonal blocks D_i are diagonal matrices are called T-matrices. Clearly, such matrices are consistently ordered. Note that matrices of the form (4.42) are a particular case with $p = 2$.

Consider now a general, consistently ordered matrix. By definition, there is permutation π of $\{1, 2, \ldots, n\}$ which is the union of p disjoint subsets

$$\pi = \pi_1 \bigcup \pi_2 \ldots \bigcup \pi_p \tag{4.44}$$

with the property that if $a_{ij} \neq 0$, $j \neq i$ and i belongs to π_k, then j belongs to $\pi_{k\pm1}$ depending on whether $i < j$ or $i > j$. This permutation π can be used to permute A symmetrically. If P is the permutation matrix associated with the permutation π, then clearly

$$A' = P^T A P$$

is a T-matrix.

Not every matrix that can be symmetrically permuted into a T-matrix is consistently ordered. The important property here is that the partition $\{\pi_i\}$ *preserves the order of the indices* i, j *of nonzero elements*. In terms of the adjacency graph, there is a partition of the graph with the property that an oriented edge i, j from i to j always points to a set with a larger index if $j > i$, or a smaller index otherwise. In particular, a very important consequence is that edges corresponding to the lower triangular part will remain so in the permuted matrix. The same is true for the upper triangular part. Indeed, if a nonzero element in the permuted matrix is $a'_{i',j'} = a_{\pi^{-1}(i),\pi^{-1}(j)} \neq 0$ with $i' > j'$, then by definition of the permutation $\pi(i') > \pi(j')$, or $i = \pi(\pi^{-1}(i)) > j = \pi(\pi^{-1}(j))$. Because of the order preservation, it is necessary that $i > j$. A similar observation holds for the upper triangular part. Therefore, this results in the following proposition.

PROPOSITION 4.2 *If a matrix A is consistently ordered, then there exists a permutation matrix P such that $P^T AP$ is a T-matrix and*

$$(P^T AP)_L = P^T A_L P, \quad (P^T AP)_U = P^T A_U P \tag{4.45}$$

in which X_L represents the (strict) lower part of X and X_U the (strict) upper part of X.

With the above property it can be shown that for consistently ordered matrices the eigenvalues of $B(\alpha)$ as defined in Proposition 4.1 are also invariant with respect to α.

PROPOSITION 4.3 *Let B be the Jacobi iteration matrix associated with a consistently ordered matrix A, and let L and U be the lower and upper triangular parts of B, respec-*

tively. Then the eigenvalues of the matrix

$$B(\alpha) = \alpha L + \frac{1}{\alpha} U$$

defined for $\alpha \neq 0$ do not depend on α.

Proof. First transform $B(\alpha)$ into a T-matrix using the permutation π in (4.44) provided by the previous proposition

$$P^T B(\alpha) P = \alpha P^T L P + \frac{1}{\alpha} P^T U P.$$

From the previous proposition, the lower part of $P^T B P$ is precisely $L' = P^T L P$. Similarly, the upper part is $U' = P^T U P$, the lower and upper parts of the associated T-matrix. Therefore, we only need to show that the property is true for a T-matrix.

In this case, for any α, the matrix $B(\alpha)$ is similar to B. This means that $B(\alpha) = X B X^{-1}$ with X being equal to

$$X = \begin{pmatrix} 1 & & & & \\ & \alpha I & & & \\ & & \alpha^2 I & & \\ & & & \ddots & \\ & & & & \alpha^{p-1} I \end{pmatrix},$$

where the partitioning is associated with the subsets $\pi_1, \ldots, \pi_p$ respectively. ∎

Note that T-matrices and matrices with the structure (4.42) are two particular cases of matrices which fulfill the assumptions of the above proposition. There are a number of well known properties related to Property A and consistent orderings. For example, it is possible to show that,

- Property A is invariant under symmetric permutations.
- A matrix has Property A if and only if there is a permutation matrix P such that $A' = P^{-1} A P$ is consistently ordered.

Consistently ordered matrices satisfy an important property which relates the eigenvalues of the corresponding SOR iteration matrices to those of the Jacobi iteration matrices. The main theorem regarding the theory for SOR is a consequence of the following result proved by Young [232]. Remember that

$$\begin{aligned} M_{SOR} &= (D - \omega E)^{-1} (\omega F + (1 - \omega) D) \\ &= (I - \omega D^{-1} E)^{-1} \left(\omega D^{-1} F + (1 - \omega) I \right). \end{aligned}$$

THEOREM 4.7 *Let A be a consistently ordered matrix such that $a_{ii} \neq 0$ for $i = 1, \ldots, n$, and let $\omega \neq 0$. Then if λ is a nonzero eigenvalue of the SOR iteration matrix M_{SOR}, any scalar μ such that*

$$(\lambda + \omega - 1)^2 = \lambda \omega^2 \mu^2 \tag{4.46}$$

is an eigenvalue of the Jacobi iteration matrix B. Conversely, if μ is an eigenvalue of the Jacobi matrix B and if a scalar λ satisfies (4.46), then λ is an eigenvalue of M_{SOR}.

Proof. Denote $D^{-1}E$ by L and $D^{-1}F$ by U, so that

$$M_{SOR} = (I - \omega L)^{-1}(\omega U + (1 - \omega)I)$$

and the Jacobi iteration matrix is merely $L + U$. Writing that λ is an eigenvalue yields

$$\det\left(\lambda I - (I - \omega L)^{-1}(\omega U + (1 - \omega)I)\right) = 0$$

which is equivalent to

$$\det\left(\lambda(I - \omega L) - (\omega U + (1 - \omega)I)\right) = 0$$

or

$$\det\left((\lambda + \omega - 1)I - \omega(\lambda L + U)\right) = 0.$$

Since $\omega \neq 0$, this can be rewritten as

$$\det\left(\frac{\lambda + \omega - 1}{\omega}I - (\lambda L + U)\right) = 0,$$

which means that $(\lambda + \omega - 1)/\omega$ is an eigenvalue of $\lambda L + U$. Since A is consistently ordered, the eigenvalues of $\lambda L + U$ which are equal to $\lambda^{1/2}(\lambda^{1/2}L + \lambda^{-1/2}U)$ are the same as those of $\lambda^{1/2}(L + U)$, where $L + U$ is the Jacobi iteration matrix. The proof follows immediately. ■

This theorem allows us to compute an optimal value for ω, which can be shown to be equal to

$$\omega_{opt} = \frac{2}{1 + \sqrt{1 - \rho(B)^2}}. \tag{4.47}$$

A typical SOR procedure starts with some ω, for example, $\omega = 1$, then proceeds with a number of SOR steps with this ω. The convergence rate for the resulting iterates is estimated providing an estimate for $\rho(B)$ using Theorem 4.7. A better ω is then obtained from the formula (4.47), and the iteration restarted. Further refinements of the optimal ω are calculated and retrofitted in this manner as the algorithm progresses.

4.3 ALTERNATING DIRECTION METHODS

The Alternating Direction Implicit (ADI) method was introduced in the mid-1950s by Peaceman and Rachford [162] specifically for solving equations arising from finite difference discretizations of elliptic and parabolic Partial Differential Equations. Consider a partial differential equation of elliptic type

$$\frac{\partial}{\partial x}\left(a(x,y)\frac{\partial u(x,y)}{\partial x}\right) + \frac{\partial}{\partial y}\left(b(x,y)\frac{\partial u(x,y)}{\partial y}\right) = f(x,y) \tag{4.48}$$

on a rectangular domain with Dirichlet boundary conditions. The equations are discretized with centered finite differences using $n + 2$ points in the x direction and $m + 2$ points in the y direction, This results in the system of equations

$$Hu + Vu = b, \tag{4.49}$$

in which the matrices H and V represent the three-point central difference approximations to the operators

$$\frac{\partial}{\partial x}\left(a(x,y)\frac{\partial}{\partial x}\right) \quad \text{and} \quad \frac{\partial}{\partial y}\left(b(x,y)\frac{\partial}{\partial y}\right),$$

respectively. In what follows, the same notation is used to represent the discretized version of the unknown function u.

The ADI algorithm consists of iterating by solving (4.49) in the x and y directions alternatively as follows.

ALGORITHM 4.3: Peaceman-Rachford (PR) ADI

1. *For* $k = 0., 1, \ldots,$ *until convergence Do:*
2. *Solve:* $(H + \rho_k I)u_{k+\frac{1}{2}} = (\rho_k I - V)u_k + b$
3. *Solve:* $(V + \rho_k I)u_{k+1} = (\rho_k I - H)u_{k+\frac{1}{2}} + b$
4. *EndDo*

Here, $\rho_k, k = 1, 2, \ldots$, is a sequence of positive acceleration parameters.

The specific case where ρ_k is chosen to be a constant ρ deserves particular attention. In this case, we can formulate the above iteration in the usual form of (4.28) with

$$G = (V + \rho I)^{-1}(H - \rho I)(H + \rho I)^{-1}(V - \rho I), \tag{4.50}$$

$$f = (V + \rho I)^{-1}\left[I - (H - \rho I)(H + \rho I)^{-1}\right] b \tag{4.51}$$

or, when $\rho > 0$, in the form (4.22) with

$$M = \frac{1}{2\rho}(H + \rho I)(V + \rho I), \quad N = \frac{1}{2\rho}(H - \rho I)(V - \rho I). \tag{4.52}$$

Note that (4.51) can be rewritten in a simpler form; see Exercise 5.

The ADI algorithm is often formulated for solving the time-dependent Partial Differential Equation

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(a(x,y)\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial y}\left(b(x,y)\frac{\partial u}{\partial y}\right) \tag{4.53}$$

on the domain $(x, y, t) \in \Omega \times [0, T] \equiv (0, 1) \times (0, 1) \times [0, T]$. The initial and boundary conditions are:

$$u(x, y, 0) = x_0(x, y), \ \forall(x, y) \in \Omega, \tag{4.54}$$

$$u(\bar{x}, \bar{y}, t) = g(\bar{x}, \bar{y}, t), \ \forall(\bar{x}, \bar{y}) \in \partial\Omega, \ t > 0, \tag{4.55}$$

where $\partial\Omega$ is the boundary of the unit square Ω. The equations are discretized with respect to the space variables x and y as before, resulting in a system of Ordinary Differential Equations:

$$\frac{du}{dt} = Hu + Vu, \tag{4.56}$$

in which the matrices H and V have been defined earlier. The Alternating Direction Implicit algorithm advances the relation (4.56) forward in time alternately in the x and y directions as follows:

$$(I - \frac{1}{2}\Delta t\ H)u_{k+\frac{1}{2}} = (I + \frac{1}{2}\Delta t\ V)u_k\ ,$$

$$(I - \frac{1}{2}\Delta t\ V)u_{k+1} = (I + \frac{1}{2}\Delta t\ H)u_{k+\frac{1}{2}}.$$

The acceleration parameters ρ_k of Algorithm 4.3 are replaced by a natural time-step.

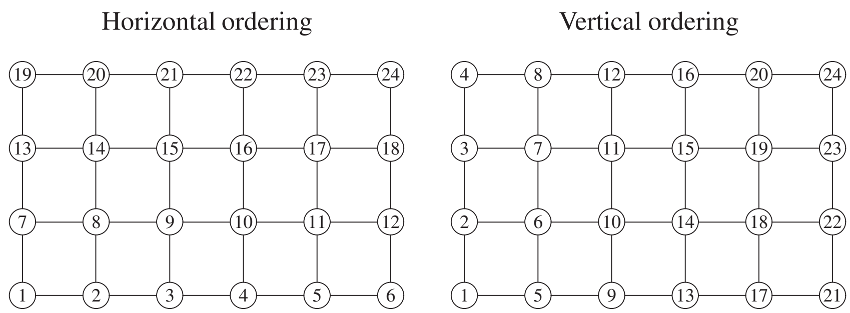

Figure 4.6 *The horizontal and vertical orderings for the unknowns in ADI.*

Assuming that the mesh-points are ordered by lines in the x-direction, then the first step of Algorithm 4.3 constitutes a set of m independent tridiagonal linear systems of size n each. However, the second step constitutes a large tridiagonal system whose three diagonals are offset by $-m$, 0, and m, respectively. This second system can also be rewritten as a set of n independent tridiagonal systems of size m each by reordering the grid points by lines, this time in the y direction. The natural (horizontal) and vertical orderings are illustrated in Figure 4.6. Whenever moving from one half step of ADI to the next, we must implicitly work with the transpose of the matrix representing the solution on the $n \times m$ grid points. This data operation may be an expensive task on parallel machines and often it is cited as one of the drawbacks of Alternating Direction Methods in this case.

ADI methods were extensively studied in the 1950s and 1960s for the particular case of positive definite systems. For such systems, H and V have real eigenvalues and the following is a summary of the main results in this situation. First, when H and V are Symmetric Positive Definite, then the stationary iteration ($\rho_k = \rho > 0$, for all k) converges. For the model problem, the asymptotic rate of convergence of the stationary ADI iteration using the optimal ρ is the same as that of SSOR using the optimal ω. However, each ADI step is more expensive than one SSOR step. One of the more important results in the ADI theory is that the rate of convergence of ADI can be increased appreciably by using a cyclic sequence of parameters, ρ_k. A theory for selecting the best sequence of ρ_k's is well understood in the case when H and V commute [26]. For the model problem, the parameters can be selected so that the time complexity is reduced to $O(n^2 \log n)$, for details see [162].

EXERCISES

1. Consider an $n \times n$ tridiagonal matrix of the form

$$T_\alpha = \begin{pmatrix} \alpha & -1 & & & & \\ -1 & \alpha & -1 & & & \\ & -1 & \alpha & -1 & & \\ & & -1 & \alpha & -1 & \\ & & & -1 & \alpha & -1 \\ & & & & -1 & \alpha \end{pmatrix}, \tag{4.57}$$

where α is a real parameter.

a. Verify that the eigenvalues of T_α are given by

$$\lambda_j = \alpha - 2\cos(j\theta) \quad j = 1, \ldots, n,$$

where

$$\theta = \frac{\pi}{n+1}$$

and that an eigenvector associated with each λ_j is

$$q_j = [\sin(j\theta), \sin(2j\theta), \ldots, \sin(nj\theta)]^T.$$

Under what condition on α does this matrix become positive definite?

b. Now take $\alpha = 2$. How does this matrix relate to the matrices seen in Chapter 2 for one-dimensional problems?

i. Will the Jacobi iteration converge for this matrix? If so, what will its convergence factor be?

ii. Will the Gauss-Seidel iteration converge for this matrix? If so, what will its convergence factor be?

iii. For which values of ω will the SOR iteration converge?

2. Prove that the iteration matrix G_ω of SSOR, as defined by (4.13), can be expressed as

$$G_\omega = I - \omega(2-\omega)(D - \omega F)^{-1} D (D - \omega E)^{-1} A.$$

Deduce the expression (4.27) for the preconditioning matrix associated with the SSOR iteration.

3. Let A be a matrix with a positive diagonal D.

a. Obtain an expression equivalent to that of (4.13) for G_ω but which involves the matrices $S_E \equiv D^{-1/2} E D^{-1/2}$ and $S_F \equiv D^{-1/2} F D^{-1/2}$.

b. Show that

$$D^{1/2} G_\omega D^{-1/2} = (I - \omega S_F)^{-1} (I - \omega S_E)^{-1} (\omega S_E + (1-\omega)I)(\omega S_F + (1-\omega)I)$$

c. Now assume that in addition to having a positive diagonal, A is symmetric. Prove that the eigenvalues of the SSOR iteration matrix G_ω are real and nonnegative.

4. Let

$$A = \begin{pmatrix} D_1 & -F_2 & & & \\ -E_2 & D_2 & -F_3 & & \\ & -E_3 & D_3 & \ddots & \\ & & \ddots & \ddots & -F_m \\ & & & -E_m & D_m \end{pmatrix},$$

where the D_i blocks are nonsingular matrices which are not necessarily diagonal.

a. What are the *block Jacobi* and *block Gauss-Seidel* iteration matrices?

b. Show a result similar to that in Proposition 4.3 for the Jacobi iteration matrix.

c. Show also that for $\omega = 1$ (1) the block Gauss-Seidel and block Jacobi iterations either both converge or both diverge, and (2) when they both converge, then the block Gauss-Seidel iteration is (asymptotically) twice as fast as the block Jacobi iteration.

5. According to formula (4.23), the f vector in iteration (4.22) should be equal to $M^{-1}b$, where b is the right-hand side and M is given in (4.52). Yet, formula (4.51) gives a different expression for f. Reconcile the two results, i.e., show that the expression (4.51) can also be rewritten as

$$f = 2\rho(V + \rho I)^{-1}(H + \rho I)^{-1}b.$$

6. Show that a matrix has Property A if and only if there is a permutation matrix P such that $A' = P^{-1}AP$ is consistently ordered.

7. Consider a matrix A which is consistently ordered. Show that the asymptotic convergence rate for Gauss-Seidel is double that of the Jacobi iteration.

8. A matrix of the form

$$B = \begin{pmatrix} 0 & E & 0 \\ 0 & 0 & F \\ H & 0 & 0 \end{pmatrix}$$

is called a three-cyclic matrix.

a. What are the eigenvalues of B? (Express them in terms of eigenvalues of a certain matrix which depends on E, F, and H.)

b. Assume that a matrix A has the form $A = D + B$, where D is a nonsingular diagonal matrix, and B is three-cyclic. How can the eigenvalues of the Jacobi iteration matrix be related to those of the Gauss-Seidel iteration matrix? How does the asymptotic convergence rate of the Gauss-Seidel iteration compare with that of the Jacobi iteration matrix in this case?

c. Answer the same questions as in (2) for the case when SOR replaces the Gauss-Seidel iteration.

d. Generalize the above results to p-cyclic matrices, i.e., matrices of the form

$$B = \begin{pmatrix} 0 & E_1 & & & \\ & 0 & E_2 & & \\ & & 0 & \ddots & \\ & & & 0 & E_{p-1} \\ E_p & & & & 0 \end{pmatrix}.$$

NOTES AND REFERENCES. Two good references for the material covered in this chapter are Varga [213] and and Young [232]. Although relaxation-type methods were very popular up to the 1960s, they are now mostly used as preconditioners, a topic which will be seen in detail in Chapters 9 and 10. One of the main difficulties with these methods is finding an optimal relaxation factor for general matrices. Theorem 4.4 is due to Ostrowski. For details on the use of Gershgorin's theorem in eigenvalue problems, see [180]. The original idea of the ADI method is described in [162] and those results on the optimal parameters for ADI can be found in [26]. A comprehensive text on this class of techniques can be found in [220]. Not covered in this book is the related class of multigrid methods; see the reference [115] for a detailed exposition. Closely related to the multigrid approach is the Aggregation-Disaggregation technique which is popular in Markov chain modeling. A recommended book for these methods and others used in the context of Markov chain modeling is [203]. ■