MATHEMATICS OF COMPUTATION
VOLUME 62, NUMBER 206
APRIL 1994, PAGES 619–644

THE SIGMA-SOR ALGORITHM AND THE OPTIMAL STRATEGY FOR THE UTILIZATION OF THE SOR ITERATIVE METHOD

ZBIGNIEW I. WOŹNICKI

ABSTRACT. The paper describes, discusses, and numerically illustrates the method for obtaining a priori estimates of the optimum relaxation factor in the SOR iteration method. The computational strategy of this method uses the so-called Sigma-SOR algorithm based on the theoretical result proven in the paper. The method presented is especially efficient for problems with slowly convergent iteration process and in this case is strongly competitive with adaptive procedures used for determining dynamically the optimum relaxation factor during the course of the SOR solution.

1. INTRODUCTION

The SOR (Successive Over-Relaxation) method and its line variants are among the most popular and efficient iterative methods used for solving large and sparse linear systems of equations arising in many areas of science and engineering. The popularity of SOR algorithms is in a great measure due to their simplicity from the programming point of view. The rate of convergence of the SOR method depends strongly on the relaxation factor ω; therefore, the main difficulty in the efficient use of this method lies in making a good estimate of the optimum relaxation factor ω_{opt} which maximizes the rate of convergence.

For a large class of matrix problems arising in the discretization of elliptic partial differential equations the coefficient matrices have certain eigenvalue properties allowing us to determine explicitly the optimum relaxation factor ω_{opt}. In the case when the coefficient matrix is 2-cyclic and consistently ordered [1] (this property will be assumed in the remainder), ω_{opt} can be determined by finding the value of the spectral radius $\rho(\mathcal{L}_1)$ for the associated Gauss-Seidel iteration matrix $\mathcal{L}_1$.

However, it is well known that the nature of the dependence of ω_{opt} on $\rho(\mathcal{L}_1)$ indicates the sensitivity of the rate of convergence to the accuracy in determining ω_{opt}, as $\rho(\mathcal{L}_1)$ approaches unity [1, 2]. When $\rho(\mathcal{L}_1)$ is very close to unity, small changes in the estimate of $\rho(\mathcal{L}_1)$ can seriously decrease the rate of convergence, and just in this case the availability of an accurate value of $\rho(\mathcal{L}_1)$ is an essential point for the efficient use of the SOR method.

Received by the editor October 9, 1992.
1991 *Mathematics Subject Classification.* Primary 65B99, 65F10, 65F15, 65F50.
Key words and phrases. SOR iteration method, power method, acceleration of convergence, eigenvalues of iteration matrix, estimation of optimum relaxation factor.

0025-5718/94 $1.00 + $.25 per page

In practice two approaches are used to determine ω_{opt}. One approach proposed in the literature [2, 3, 4] is determining ω_{opt} dynamically, as the SOR iteration proceeds with using some $\omega_i < \omega_{\text{opt}}$. Then by examining certain conditions for quantities derived from current numerical results, ω_i is updated to a new relaxation factor $\omega_{i+1} \leq \omega_{\text{opt}}$ until the assumed tolerance criterion is satisfied.

The second approach for determining ω_{opt} is based on obtaining an a priori estimation of $\rho(\mathscr{L}_1)$, usually by means of the power method or its modifications. As is well known, the rate of convergence of the power method is governed by the ratio of the largest subdominant (in the absolute value) to the dominant eigenvalue. If this ratio is close to unity, the power method will converge very slowly and in such a case determining ω_{opt} may be more time-consuming than the SOR iteration itself.

Basically, there is no general comparison procedure to determine which approach is "best". However in the case of 2-cyclic consistently ordered matrices, an accurate estimate for $\rho(\mathscr{L}_1)$ prior to the SOR iteration solution can be effectively obtained by an appropriate use of power method iterations, and this topic is the main purpose of the paper.

In the next section the SOR iterative method and the power method are briefly described, and well-known basic results are recalled. These basic results are essential in deriving the Sigma-SOR algorithm. The computational strategy for determining the optimum relaxation factor ω_{opt} is described in the third subsection of §2.

The secondary purpose of this paper, discussed in §3, is to give numerical results for a variety of problems presented in the literature in order to illustrate the efficiency of the proposed method for the a priori determination of the optimum relaxation factor ω_{opt}.

2. Formulation

2.1. **The SOR iteration method.** In the iterative solution of the linear system

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{1}$$

the $n \times n$ matrix $\mathbf{A}$ is usually defined by the following decomposition:

$$\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{U}, \tag{2}$$

where $\mathbf{D}$, $\mathbf{L}$, and $\mathbf{U}$ are diagonal, strictly lower triangular and strictly upper triangular matrices, respectively.

The SOR iterative method [1] is defined by

$$\mathbf{D}\mathbf{x}^{(t+1)} = \omega[\mathbf{L}\mathbf{x}^{(t+1)} + \mathbf{U}\mathbf{x}^{(t)} + \mathbf{b}] - (\omega - \mathbf{1})\mathbf{D}\mathbf{x}^{(t)}, \qquad t = 0, 1, 2, \ldots \tag{3}$$

or equivalently, if $\mathbf{D}$ is a nonsingular matrix

$$\mathbf{x}^{(t+1)} = \mathscr{L}_\omega \mathbf{x}^{(t)} + (\mathbf{D} - \omega\mathbf{L})^{-1}\mathbf{b}, \tag{4}$$

where

$$\mathscr{L}_\omega = (\mathbf{D} - \omega\mathbf{L})^{-1}[\omega\mathbf{U} - (\omega - \mathbf{1})\mathbf{D}] \tag{5}$$

is called the *SOR iteration matrix* and ω is the *relaxation factor*. For $\omega = 1$ the SOR method reduces to the classical scheme known as the *Gauss-Seidel iterative method* and

$$\mathscr{L}_1 = (\mathbf{D} - \mathbf{L})^{-1}\mathbf{U} \tag{6}$$

is called the *Gauss-Seidel iteration matrix.*

In the point algorithm, the iteration proceeds for one component of the approximate solution vector at a time. For block or line algorithms, the iteration involves improving simultaneously groups of components, and therefore they are more efficient than the point algorithm. In this case the matrices $\mathbf{D}$, $\mathbf{L}$, and $\mathbf{U}$ have a block structure corresponding to the assumed partitioning of components.

It is well known [1] that in the case of 2-cyclic consistent orderings, when the associated nonnegative *Jacobi iteration matrix*

$$\mathbf{B} = \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U}) \geq 0$$

is convergent (i.e., $\rho(B) < 1$), then $\mathscr{L}_1$ has only nonnegative eigenvalues λ_i such that

$$1 > \rho(\mathscr{L}_1) = \lambda_1 > \lambda_2 > \lambda_3 > \cdots , \tag{7}$$

and the following fundamental relation due to Young (see, for example [1] and the references given therein) holds between λ_i and the corresponding eigenvalues ν_i of $\mathscr{L}_\omega$:

$$\lambda_i = \frac{1}{\nu_i}\left[\frac{\nu_i + \omega - 1}{\omega}\right]^2 . \tag{8}$$

Moreover, $\rho(\mathscr{L}_\omega) = \max_{1\leq i\leq n} |\nu_i| < 1$ for $0 < \omega < 2$, and its minimum value is attained when

$$\omega = \omega_{\text{opt}} = \overline{\omega} = \frac{2}{1 + \sqrt{1 - \rho(\mathscr{L}_1)}} , \tag{9}$$

in which case

$$\rho(\mathscr{L}_{\overline{\omega}}) = \overline{\omega} - 1 = \frac{1 - \sqrt{1 - \rho(\mathscr{L}_1)}}{1 + \sqrt{1 - \rho(\mathscr{L}_1)}}. \tag{10}$$

In the convergence analysis of iterative methods the (*asymptotic*) *rate of convergence*

$$\mathbf{R}(\mathscr{G}) = -\ln \rho(\mathscr{G}) \tag{11}$$

is certainly the simplest practical measure of the speed of convergence for a convergent matrix $\mathscr{G}$. The rate of convergence is especially useful for comparing the efficiency of different iterative methods, because the number of iterations t required for reducing the error norm in a given method by a prescribed factor $\mathbf{f}$ is roughly inversely proportional to the rate of convergence, and is given by

$$t \approx \frac{-\ln \mathbf{f}}{\mathbf{R}(\mathscr{G})} , \tag{12}$$

where $\mathscr{G}$ is the iteration matrix of the method.

Thus, the efficiency of different iterative methods (with a similar arithmetical effort per iteration) can be theoretically evaluated by a comparison of the rate of convergence. The data given in Table 1 (next page) illustrate the efficiency of the SOR method by comparing it with the Gauss-Seidel method, where

$$\mathbf{E}_t = \frac{\mathbf{R}(\mathscr{L}_{\overline{\omega}})}{\mathbf{R}(\mathscr{L}_1)} \tag{13}$$

is the *theoretical coefficient of efficiency* and $\overline{\omega} \equiv \omega_{\text{opt}}$.

TABLE 1

$\rho(\mathscr{L}_1)$	$\rho(\mathscr{L}_\omega)$	$\mathbf{E}_t$
0.9	0.5195	6
0.99	0.8182	20
0.999	0.9387	63
0.9999	0.9802	200

As can be seen from Table 1, the efficiency of the SOR method drastically increases as $\rho(\mathscr{L}_1)$ becomes close to unity. For the case when $\rho(\mathscr{L}_1) = 0.9999$, the SOR method is asymptotically 200 times faster than the Gauss-Seidel method. Since ω_{opt} is a function only of the spectral radius $\rho(\mathscr{L}_1)$, then for any efficient use of the SOR method, computing an accurate value of $\rho(\mathscr{L}_1)$ is needed, and the order of the accuracy in estimating $\rho(\mathscr{L}_1)$ is dependent on the closeness of $\rho(\mathscr{L}_1)$ to unity.

2.2. **The power method.** Usually, an estimate for $\rho(\mathscr{L}_1)$ is obtained by using the ordinary power method [5], which will be used in the analysis presented in this paper. The power method is conceptually and computationally the simplest iterative procedure for approximating the eigenvector corresponding to the dominant (largest in modulus) eigenvalue of a given matrix $\mathscr{G}$. It is defined by the iterative process

$$\mathbf{z}^{(t)} = \mathscr{G}\mathbf{z}^{(t-1)} = \mathscr{G}^2\mathbf{z}^{(t-2)} = \cdots = \mathscr{G}^t\mathbf{z}^{(0)}, \qquad t = 1, 2, \ldots, \tag{14}$$

which converges for almost any randomly chosen nonzero starting vector $\mathbf{z}^{(0)}$.

We assume, throughout this paper, that the $n \times n$ real matrix $\mathscr{G}$ has n linearly independent eigenvectors $\mathbf{u}_i$, and its eigenvalues λ_i will be ordered such that

$$\lambda_1 > |\lambda_2| \geq |\lambda_3| \geq \cdots \geq |\lambda_n|. \tag{15}$$

Since by assumption, $\mathscr{G}$ has a complete set of eigenvectors $\mathbf{u}_i$, an arbitrary nonzero vector $\mathbf{z}^{(0)}$ can be expressed in the form

$$\mathbf{z}^{(0)} = \sum_{i=1}^{n} a_i\mathbf{u}_i, \tag{16}$$

where a_i are scalars not all zero.

Then the sequence (14) has the representation

$$\mathbf{z}^{(t)} = \sum_{i=1}^{n} a_i\lambda_i^t\mathbf{u}_i = \lambda_1^t\left[a_1\mathbf{u}_1 + \sum_{i=2}^{n} a_i(\lambda_i/\lambda_1)^t\mathbf{u}_i\right] = \lambda_1^t[a_1\mathbf{u}_1 + \mathbf{e}^{(t)}]. \tag{17}$$

Since $|\lambda_i/\lambda_1| < 1$ for all $i \geq 2$, it is clear that $\mathbf{z}^{(t)}$ converges to $\mathbf{u}_1$ as $t \to \infty$, provided only that $a_1 \neq 0$.

Thus the vector $\mathbf{z}^{(t)}$ is an approximation to an unnormalized eigenvector $\mathbf{u}_1$ belonging to λ_1, which can be considered as accurate if $\|\mathbf{e}^{(t)}\|$ is sufficiently small. Since

$$\mathbf{z}^{(t+1)} = \lambda_1^{t+1}[a_1\mathbf{u}_1 + \mathbf{e}^{(t+1)}],$$

it follows that for any jth component $\mathbf{z}_j$ of the vector $\mathbf{z}$,

$$\frac{\mathbf{z}_j^{(t+1)}}{\mathbf{z}_j^{(t)}} = \lambda_1 \left[\frac{a_1(\mathbf{u}_1)_j + (\mathbf{e}^{(t+1)})_j}{a_1(\mathbf{u}_1)_j + (\mathbf{e}^{(t)})_j} \right] \to \lambda_1 \quad \text{as } t \to \infty. \tag{18}$$

The above result leads to computing the dominant eigenvalue by means of successive approximations of the corresponding eigenvector in the simple power method.

In practice, in order to keep the components of $\mathbf{z}^{(t)}$ within the range of practical calculation, its components are scaled at each iteration step, and (14) is replaced by the pair of equations

$$\mathbf{y}^{(t)} = \mathscr{G}\mathbf{z}^{(t-1)}, \tag{19}$$

$$\mathbf{z}^{(t)} = \mathbf{y}^{(t)} / \|\mathbf{y}^{(t)}\|_p, \tag{20}$$

and in this case,

$$\mathbf{z}^{(t)} \to \mathbf{u}_1 / \|\mathbf{u}_1\|_p \tag{21}$$

and

$$\|\mathbf{y}^{(t)}\|_p \to \lambda_1 \quad \text{as } t \to \infty, \tag{22}$$

where two norms, either the maximum norm $\|\cdot\|_\infty$ or the Euclidean norm $\|\cdot\|_2$, are most commonly used.

The rate of convergence will depend on the constants a_i, but more essentially on the separation of the dominant eigenvalue from the largest subdominant eigenvalues of $\mathscr{G}$, that is, on the ratios $|\lambda_2|/\lambda_1$, $|\lambda_3|/\lambda_1$, ..., and it is evident that the smaller these values, the faster the convergence. However, it may occur that if $\mathbf{z}^{(0)}$ is chosen as almost orthogonal to $\mathbf{u}_1$, then a_1 in (17) will be quite small compared to the other coefficients, and whence for appropriate "small" values of t, $|a_1\lambda_1^t| \ll |a_2\lambda_2^t|$ and the ratio $\mathbf{z}_j^{(t+1)}/\mathbf{z}_j^{(t)}$ will better approximate λ_2 than λ_1, assuming of course that $\lambda_1 > |\lambda_2|$. In the case when $a_1 = 0$, the power method converges theoretically to the second eigenvector. However, in practice rounding errors will introduce small components $\mathbf{u}_1$ into the vector $\mathbf{z}^{(t)}$ and those components will be magnified in subsequent iterations. Whence, convergence is still likely to be to the first eigenvector, although with a larger number of iterations than in the case when a more suitable starting vector $\mathbf{z}^{(0)}$ would be chosen.

In particular, if $|\lambda_2|/\lambda_1$ is close to unity, the accuracy of $\mathbf{z}^{(t)}$ will be proportional to $(|\lambda_2|/\lambda_1)^t$ and the convergence may be intolerably slow, but still to the dominant eigenvalue λ_1. In such cases some practical techniques such as a shift of origin, or Aitken's δ^2-process [5], can be used to speed up the convergence of the simple power method.

In general, when λ_1 is the principal eigenvalue, the ratio

$$\sigma = \max_i \frac{|\lambda_i|}{|\lambda_1|}, \qquad 2 \le i \le n \tag{23}$$

will be called the *subdominance ratio*, which with the assumed ordering of λ_i according to (15) is equivalent to

$$\sigma = |\lambda_2|/\lambda_1. \tag{23a}$$

However, it seems that from the terminology point of view some comments are necessary. In the literature for σ the term "dominance ratio" is usually used by some authors. But it is also interesting to notice that other authors (especially the authors of books dealing with the convergence analysis of eigenvalue problems) do not use the term "dominance ratio" at all. In the author's feeling the term "subdominance ratio" for σ seems to be more appropriate because σ increases with the absolute value of the largest subdominant eigenvalue, and the dominance of the principal eigenvalue decreases.

Since the convergence to the dominant eigenvalue by the power method is geometric in the subdominance ratio σ, then by an analogy to the analysis of iterative methods for solving linear systems of equations one can define the (asymptotic) rate of convergence as

$$\dot{\mathbf{R}}(\mathscr{G}) = -\ln \sigma, \tag{24}$$

which is a useful measure for the speed of convergence to the dominant eigenvalue of a given matrix $\mathscr{G}$ in the power method.

Referring back to the SOR method, we find it convenient to first consider the behavior of the eigenvalues ν_i of $\mathscr{L}_\omega$ as a function of ω for the case of 2-cyclic consistently ordered nonsingular matrices $\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{U}$ of (1), where $\mathbf{D}$, $\mathbf{L}$, and $\mathbf{U}$ are nonsingular diagonal, strictly lower triangular and strictly upper triangular nonnegative matrices, respectively. As is well known [1], the eigenvalues ν_i of $\mathscr{L}_\omega$ are related by (8) to the eigenvalues λ_i of the Gauss-Seidel iteration matrix $\mathscr{L}_1$, the special case of $\mathscr{L}_\omega$ with $\omega = 1$. The matrix $\mathscr{L}_1$ has at least half the eigenvalues equal to zero, and the remaining ones are positive and real, and such that

$$1 > \rho(\mathscr{L}_1) = \lambda_1 > \lambda_2 > \lambda_3 > \cdots. \tag{25}$$

In the analysis of convergence properties of the SOR method, it is very useful to investigate the behavior of the roots of (8),

$$\nu_i^+, \nu_i^- = \frac{1}{2}\left[\omega^2\lambda_i \pm \sqrt{\omega^2\lambda_i[\omega^2\lambda_i - 4(\omega - 1)]}\right] - (\omega - 1). \tag{26}$$

Thus, when $\omega = 1$, it is clear that $\nu_i^+ = \lambda_i$ and $\nu_i^- = 0$. As ω increases from unity, ν_i^+ and ν_i^- are decreasing and increasing functions of ω, respectively, until $\omega^2\lambda_i - 4(\omega - 1) = 0$, which occurs when

$$\omega = \overline{\omega}_i = \frac{2}{1 + \sqrt{1 - \lambda_i}} \tag{27}$$

and both roots coincide with the same value, that is, $\nu_i^+ = \nu_i^- = \overline{\omega}_i - 1$. For $\omega > \overline{\omega}_i$, the roots ν_i^+ and ν_i^- become complex conjugate pairs and increase, the absolute value being $\omega - 1$. It is obvious that, for

$$1 \le \omega \le \overline{\omega}_1 = \frac{2}{1 + \sqrt{1 - \lambda_1}},$$

$\rho(\mathscr{L}_\omega) = \nu_1^+$ is a real and strictly decreasing function of ω while for $\overline{\omega}_1 < \omega \le 2$ one has $\rho(\mathscr{L}_\omega) = |\omega - 1|$.

However, we should add a note about negative eigenvalues ν_i which may exist. The matrix $\mathscr{L}_1$ has s (usually half of n) eigenvalues positive and $n - s$ zero. These positive eigenvalues λ_i give rise to the roots ν_i^+ while the zero

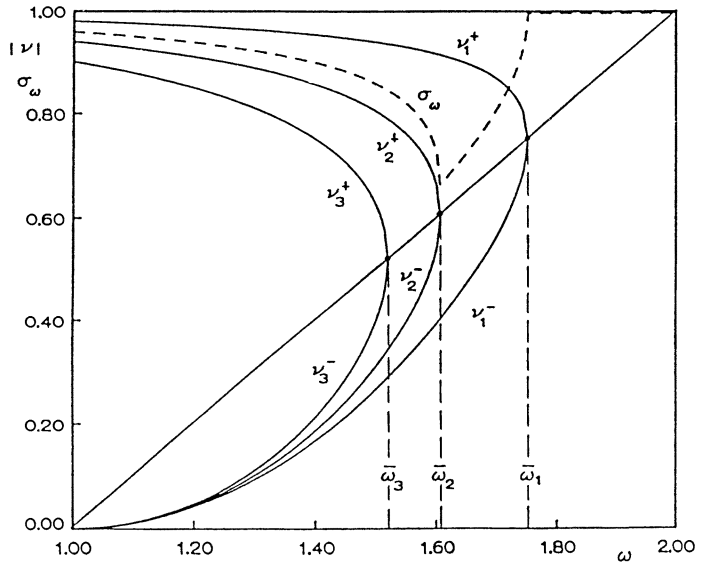


FIGURE 1. The behavior of ν and σ_ω vs. ω

eigenvalues λ_{i+s} give rise to the roots ν_i^- with $\omega = 1$. If $2s < n$, then there are zero eigenvalues λ_j, where $2s < j \leq n$, which satisfy also the relation (8); hence the corresponding eigenvalues $\nu_j = -(\omega - 1)$ are negative for all $\omega > 1$.

The typical behavior of the eigenvalues ν_i of $\mathscr{L}_\omega$ versus ω is shown in Figure 1 for the example in which the three largest eigenvalues of $\mathscr{L}_1$ are $\lambda_1 = 0.98$, $\lambda_2 = 0.94$, and $\lambda_3 = 0.9$, and the subdominance ratio $\sigma_1 = \lambda_2/\lambda_1 = 0.9592$. As can be seen from Figure 1, there exist only two positive eigenvalues ν_1^+ and ν_2^+ for $\overline{\omega}_3 < \omega \leq \overline{\omega}_2$, only one ν_1^+ for $\overline{\omega}_2 < \omega \leq \overline{\omega}_1$, and for $\omega > \overline{\omega}_1$ all eigenvalues ν_i are complex (and negative if they exist) with the absolute value equal to $\omega - 1$.

It is obvious that the subdominance ratio σ_ω for the SOR matrix $\mathscr{L}_\omega$ is a function of ω and $\sigma_\omega = \sigma_1 = \lambda_2/\lambda_1$ when $\omega = 1$. For $1 < \omega \leq \overline{\omega}_2$, $\sigma_\omega = \nu_2/\nu_1$ is a strictly decreasing function as ω increases from unity (because $\nu_1 \equiv \nu_1^+$ decreases much less rapidly than $\nu_2 \equiv \nu_2^+$) and at

$$\omega = \overline{\omega}_2 = \frac{2}{1 + \sqrt{1 - \lambda_2}} \tag{27a}$$

achieves its minimum $\overline{\sigma}_\omega = \overline{\nu}_2/\nu_1 = (\overline{\omega}_2 - 1)/\nu_1$. For $\overline{\omega}_2 < \omega \leq \overline{\omega}_1$, $\sigma_\omega = |\omega - 1|/\nu_1$ is a strictly increasing function of ω and for all $\overline{\omega}_1 \leq \omega \leq 2$, $\sigma_\omega = 1$ because all eigenvalues ν_i have the same absolute value equal to $|\omega - 1|$.

In the example shown in Figure 1, the dashed curve illustrates the behavior of σ_ω versus ω, where the minimum $\overline{\sigma}_\omega = 0.6639$ occurs at $\overline{\omega}_2 = 1.6065$. In terms of the rate of convergence the *theoretical coefficient of efficiency*

$$\dot{\mathbf{E}}_t = \frac{\dot{\mathbf{R}}(\overline{\sigma}_\omega)}{\dot{\mathbf{R}}(\sigma_1)} \tag{28}$$

is equal to 9.84. Thus for this example the computation of $\rho(\mathscr{L}_\omega)$ by means of the power method with $\omega = \overline{\omega}_2$ is asymptotically about 10 times faster than with $\omega = 1$.

2.3. **The Sigma-SOR algorithm and computational strategy.** The observations in the previous subsection show the existence of the minimum value $\overline{\sigma}_\omega < \sigma_1$ and moreover they allow us to precisely identify its locality which occurs at $\omega = \overline{\omega}_2$ minimizing the value of the subdominant eigenvalue ν_2. The question now arises whether there exists a mathematical basis for determining the value of $\overline{\sigma}_\omega$ in dependence on $\sigma_1 = \lambda_2/\lambda_1$. The following theorem gives an answer to this question.

Theorem. *Let* ν_i *be the eigenvalues of the* $n \times n$ *SOR iteration matrix*

$$\mathscr{L}_\omega = (\mathbf{D} - \omega\mathbf{L})^{-1}[\omega\mathbf{U} - (\omega - 1)\mathbf{D}]$$

and let λ_i *be the eigenvalues of the Gauss-Seidel iteration matrix*

$$\mathscr{L}_1 = (\mathbf{D} - \mathbf{L})^{-1}\mathbf{U}.$$

If the eigenvalues of both matrices are related by

$$\lambda_i = \frac{1}{\nu_i}\left[\frac{\nu_i + \omega - 1}{\omega}\right]^2 \tag{29}$$

and $\mathscr{L}_1$ *has only nonnegative real eigenvalues such that*

$$1 > \lambda_1 > \lambda_2 > \lambda_3 > \cdots, \tag{30}$$

then the subdominance ratio $\sigma_\omega = \nu_2/\nu_1$ *of* $\mathscr{L}_\omega$ *achieves its minimum* $\overline{\sigma}_\omega = \overline{\nu}_2/\nu_1$ *with*

$$\omega = \overline{\omega}_2 = \frac{2}{1 + \sqrt{1 - \lambda_2}} = \frac{2}{1 + \sqrt{1 - \sigma_1\lambda_1}} \tag{31}$$

and is defined by the following formula:

$$\overline{\sigma}_\omega = \frac{2}{1 + \sqrt{1 - \sigma_1}} - 1 = \frac{1 - \sqrt{1 - \sigma_1}}{1 + \sqrt{1 - \sigma_1}}, \tag{32}$$

where $\sigma_1 = \lambda_2/\lambda_1$.

Proof. By using (29), one obtains that

$$\sigma_1 = \frac{\lambda_2}{\lambda_1} = \frac{\nu_1}{\nu_2}\left[\frac{\nu_2 + \omega - 1}{\nu_1 + \omega - 1}\right]^2 = \sigma_\omega\left[\frac{1 + \frac{\omega - 1}{\nu_2}}{1 + \sigma_\omega\frac{\omega - 1}{\nu_2}}\right]^2 \tag{33}$$

or equivalently

$$\sigma_1 = \frac{1}{\sigma_\omega}\left[\frac{\sigma_\omega + \frac{\omega - 1}{\nu_1}}{1 + \frac{\omega - 1}{\nu_1}}\right]^2. \tag{33a}$$

The proof follows immediately from a close inspection of (33). As was already stated, σ_ω is minimized when $\omega = \overline{\omega}_2$ and its value is $\sigma_\omega = \overline{\sigma}_\omega = \overline{\nu}_2/\nu_1$, where $\overline{\nu}_2 = \overline{\omega}_2 - 1$. Hence, for $\omega = \overline{\omega}_2$, (33) reduces to

$$\sigma_1 = \overline{\sigma}_\omega\left[\frac{2}{1 + \overline{\sigma}_\omega}\right]^2 \tag{34}$$

and has the solution

$$\overline{\sigma}_\omega = \frac{2}{1 + \sqrt{1 - \sigma_1}} - 1 = \frac{1 - \sqrt{1 - \sigma_1}}{1 + \sqrt{1 - \sigma_1}}.$$

This completes the proof of the theorem. □

It is necessary, however, to make some comments on the above result, because (34) has two roots $\overline{\sigma}_\omega^+ < 1$ (corresponding to the above result) and $\overline{\sigma}_\omega^- > 1$. Since with $\omega = \overline{\omega}_2$ the matrix $\mathscr{L}_\omega$ has only four real eigenvalues (see Figure 1) such that $\nu_1^+ > \nu_2^+ = \nu_2^- = \overline{\nu}_2 = \overline{\omega}_2 - 1 > \nu_1^- > 0$, then for $\sigma_1 < 1$

$$\overline{\sigma}_\omega^+ = \frac{\overline{\nu}_2}{\nu_1^+} = \frac{2}{1+\sqrt{1-\sigma_1}} - 1 < 1 \quad \text{and} \quad \overline{\sigma}_\omega^- = \frac{\overline{\nu}_2}{\nu_1^-} = \frac{2}{1-\sqrt{1-\sigma_1}} - 1 > 1.$$

But both $\overline{\nu}_2$ and ν_1^- are subdominant eigenvalues and therefore the fact that $\overline{\sigma}_\omega^- > 1$ has no practical significance.

The most interesting conclusion from this theorem is the fact that the minimum values of both spectral radius and subdominance ratio for the SOR iteration matrix are governed by the same formula (see (10) and (32)). In other words, for the same values of both $\rho(\mathscr{L}_1)$ and σ_1 the quantities $\rho(\mathscr{L}_\omega)$ and σ_ω achieve the same minimum value but with different values of ω. It is evident that replacing $\rho(\mathscr{L}_1)$, $\rho(\mathscr{L}_{\overline{\omega}})$, and $\mathbf{E}_t$ in Table 1 by σ_1, $\overline{\sigma}_\omega$, and $\dot{\mathbf{E}}_t$ (defined by (28)), respectively, the data of this table illustrate also the efficiency of the power method in the asymptotic range as in the case of the SOR method.

Thus, the result of this theorem is of fundamental importance in the computational strategy for a "rapid" estimate of an "accurate" value of the optimum relaxation factor ω_{opt} in the SOR method.

The algorithm for determining ω_{opt}, called the *Sigma*-SOR *algorithm*, is based on the following computational strategy. Assume that λ^* and σ^*, appropriate estimates for $\lambda_1 \equiv \rho(\mathscr{L}_1)$ and σ_1, respectively, are known. Using

$$\omega^* = \frac{2}{1+\sqrt{1-\sigma^*\lambda^*}}, \tag{35a}$$

we can obtain $\nu^* \equiv \rho(\mathscr{L}_{\omega^*})$ by the power method iteration until a required convergence criterion is satisfied. Then from the relation (29) one obtains

$$\lambda_1 = \frac{1}{\nu^*}\left[\frac{\nu^* + \omega^* - 1}{\omega^*}\right]^2, \tag{35b}$$

which allows us to determine

$$\overline{\omega}_1 = \frac{2}{1+\sqrt{1-\lambda_1}}, \tag{35c}$$

an a priori "accurate" estimate for ω_{opt}. Thus, the accuracy of ω_{opt} is conditional to the computation of an accurate value of ν^*.

As is demonstrated in numerical experiments given in the next section, the above algorithm, even with crude approximations λ^* and σ^*, is very efficient and strongly competitive with the SOR adaptive procedure [1] when $\rho(\mathscr{L}_1)$ is very close to unity $(0.999 < \rho(\mathscr{L}_1) < 1)$.

Estimates for σ^* approximating σ_1 can be obtained by observing the decay rate of some quantities, for instance

$$\sigma^{(t+1)} = \frac{|\lambda^{(t+1)} - \lambda^{(t)}|}{|\lambda^{(t)} - \lambda^{(t-1)}|}, \tag{36}$$

or ratios of differences between the components of successive eigenvectors in the iteration process of the power method (19), (20), using a suitable norm (see, for example, [4], where the term dominance ratio is used for σ). As follows from

(18), for each jth nonzero component $\mathbf{z}_j$ of $\mathbf{z}$ approximating the eigenvector corresponding to the dominant eigenvalue in the power method, we have that

$$\lambda^{(t+1)} = \frac{\mathbf{z}_j^{(t+1)}}{\mathbf{z}_j^{(t)}} = \lambda_1 \left[\frac{a_1(\mathbf{u}_1)_j + (\mathbf{e}^{(t+1)})_j}{a_1(\mathbf{u}_1)_j + (\mathbf{e}^{(t)})_j} \right] \to \lambda_1 \quad \text{as } t \to \infty, \tag{37}$$

where

$$(\mathbf{e}^{(t)})_j = \sum_{i=2}^{n} \left(\frac{\lambda_i}{\lambda_1} \right)^t a_i(\mathbf{u}_i)_j. \tag{38}$$

Substituting (37) into (36), one obtains

$$\sigma^{(t+1)} = \frac{a_1(\mathbf{u}_1)_j + (\mathbf{e}^{(t+1)})_j}{a_1(\mathbf{u}_1)_j + (\mathbf{e}^{(t)})_j} \left[\frac{(\mathbf{e}^{(t+1)})_j - 2(\mathbf{e}^{(t)})_j + (\mathbf{e}^{(t-1)})_j}{(\mathbf{e}^{(t)})_j - 2(\mathbf{e}^{(t-1)})_j + (\mathbf{e}^{(t-2)})_j} \right],$$

and for t sufficiently large, $a_1(\mathbf{u}_1)_j \gg (\mathbf{e}^{(t)})_j$, so that

$$\sigma^{(t+1)} \approx \frac{(\mathbf{e}^{(t+1)})_j - 2(\mathbf{e}^{(t)})_j + (\mathbf{e}^{(t-1)})_j}{(\mathbf{e}^{(t)})_j - 2(\mathbf{e}^{(t-1)})_j + (\mathbf{e}^{(t-2)})_j}. \tag{39}$$

Assume now that for any $t' \geq 1$

$$\left(\frac{\lambda_2}{\lambda_1} \right)^{t'} a_2(\mathbf{u}_2)_j > \left(\frac{\lambda_i}{\lambda_1} \right)^{t'} a_i(\mathbf{u}_i)_j \quad \text{for all } 3 \leq i \leq n. \tag{40}$$

Equation (38) can be written in the form

$$(\mathbf{e}^{(t)})_j = \left(\frac{\lambda_2}{\lambda_1} \right)^t a_2(\mathbf{u}_2)_j + (\tilde{\mathbf{e}}^{(t)})_j, \tag{41}$$

where

$$(\tilde{\mathbf{e}}^{(t)})_j = \sum_{i=3}^{n} \left(\frac{\lambda_i}{\lambda_1} \right)^t a_i(\mathbf{u}_i)_j. \tag{42}$$

Substituting (41) into (39) yields

$$\sigma^{(t+1)} \approx \frac{\frac{\lambda_2}{\lambda_1} \left[\left(\frac{\lambda_2}{\lambda_1} \right)^t - 2 \left(\frac{\lambda_2}{\lambda_1} \right)^{t-1} + \left(\frac{\lambda_2}{\lambda_1} \right)^{t-2} \right] a_2(\mathbf{u}_2)_j + (\tilde{\mathbf{e}}^{(t+1)})_j - 2(\tilde{\mathbf{e}}^{(t)})_j + (\tilde{\mathbf{e}}^{(t-1)})_j}{\left[\left(\frac{\lambda_2}{\lambda_1} \right)^t - 2 \left(\frac{\lambda_2}{\lambda_1} \right)^{t-1} + \left(\frac{\lambda_2}{\lambda_1} \right)^{t-2} \right] a_2(\mathbf{u}_2)_j + (\tilde{\mathbf{e}}^{(t)})_j - 2(\tilde{\mathbf{e}}^{(t-1)})_j + (\tilde{\mathbf{e}}^{(t-2)})_j}.$$

But when $t \gg t'$, the relation (40) implies that $(\tilde{\mathbf{e}}^{(t)})_j$ becomes sufficiently small, and it can be concluded that

$$\sigma^{(t+1)} \to \frac{\lambda_2}{\lambda_1} = \sigma_1. \tag{43}$$

In the calculation of $\rho(\mathscr{L}_1)$ (or $\rho(\mathscr{L}_\omega)$) by means of the algorithm of the power method defined by (19)–(22), the notation $\lambda_M \equiv \rho(\mathscr{L}_1)$ (or $\nu_M \equiv \rho(\mathscr{L}_\omega)$) corresponds to using the maximum norm $\|\cdot\|_\infty$, and $\lambda_E \equiv \rho(\mathscr{L}_1)$ (or $\nu_E \equiv \rho(\mathscr{L}_\omega)$) corresponds to using the Euclidean norm $\|\cdot\|_2$ in the scaling procedure.

With these notations,

$$\sigma_{\mathrm{M}}^{(t+1)} = \frac{|\lambda_{\mathrm{M}}^{(t+1)} - \lambda_{\mathrm{M}}^{(t)}|}{|\lambda_{\mathrm{M}}^{(t)} - \lambda_{\mathrm{M}}^{(t-1)}|}, \tag{44a}$$

$$\sigma_{\mathrm{E}}^{(t+1)} = \frac{|\lambda_{\mathrm{E}}^{(t+1)} - \lambda_{\mathrm{E}}^{(t)}|}{|\lambda_{\mathrm{E}}^{(t)} - \lambda_{\mathrm{E}}^{(t-1)}|}. \tag{44b}$$

Usually, the convergence behavior of both λ_{M} and σ_{M} have a monotone decreasing character, whereas for λ_{E} and σ_{E} it was observed that they first increase and then (mainly for λ_{E}) slowly decrease as the number of iterations increases.

In the case of using the Euclidean norm for scaling purposes, the following two additional measures for σ can be used:

$$\sigma_{\mathrm{EM}}^{(t+1)} = \frac{|\,\|\mathbf{y}_{\mathrm{E}}^{(t+1)}\|_\infty - \|\mathbf{y}_{\mathrm{E}}^{(t)}\|_\infty|}{|\,\|\mathbf{y}_{\mathrm{E}}^{(t)}\|_\infty - \|\mathbf{y}_{\mathrm{E}}^{(t-1)}\|_\infty|} \tag{44c}$$

and

$$\sigma_{\mathrm{EE}}^{(t+1)} = \frac{|\,\|\mathbf{y}_{\mathrm{E}}^{(t+1)} - \mathbf{y}_{\mathrm{E}}^{(t)}\|_2 - \|\mathbf{y}_{\mathrm{E}}^{(t)} - \mathbf{y}_{\mathrm{E}}^{(t-1)}\|_2|}{|\,\|\mathbf{y}_{\mathrm{E}}^{(t)} - \mathbf{y}_{\mathrm{E}}^{(t-1)}\|_2 - \|\mathbf{y}_{\mathrm{E}}^{(t-1)} - \mathbf{y}_{\mathrm{E}}^{(t-2)}\|_2|}, \tag{44d}$$

where the successive eigenvectors $\mathbf{y}_{\mathrm{E}}^{(t+1)}$, $\mathbf{y}_{\mathrm{E}}^{(t)}$, $\mathbf{y}_{\mathrm{E}}^{(t-1)}$, and $\mathbf{y}_{\mathrm{E}}^{(t-2)}$ are generated by (19)–(22) with using the Euclidean norm for scaling.

As demonstrated in numerical experiments, the most rapid convergence is observed for σ_{EE} with a monotone increasing character, which provides certain values estimating the true σ_1 from below.

As can be seen from Figure 1 the behavior of σ_ω near $\overline{\omega}_2$ is similar in nature to the behavior of $\rho(\mathscr{L}_\omega)$ near $\overline{\omega}_1$. From an inspection of the slope of the curve for σ_ω near $\overline{\omega}_2$, it follows that errors with underestimating $\overline{\omega}_2$ give larger values of σ_ω than errors (comparable in size) with overestimating $\overline{\omega}_2$. In the range $1 \le \omega \le \overline{\omega}_2$, the value of σ_ω can be determined from (33) in dependence on σ_1 and $(\omega - 1)/\nu_2$ (or $(\omega - 1)/\nu_1$ in the case of (33a)), and in the range $\overline{\omega}_2 < \omega \le \overline{\omega}_1$, it is defined by $|\omega - 1|/\nu_1$.

Thus, from the viewpoint of obtaining the maximum rate of convergence in the power method, overestimating $\overline{\omega}_2$ is less dangerous than underestimating $\overline{\omega}_2$ by the same amount, but as σ_1 approaches unity, this becomes a more important problem because underestimating $\overline{\omega}_2$ drastically decreases the rate of convergence.

On the other hand, however, underestimating $\overline{\omega}_2$ may be attractive for accelerating convergence by the use of the Aitken δ^2-process [5]. This procedure, known also under the name of Aitken extrapolation, is a useful tool for improving convergence, and can be used for any process converging linearly (i.e., as in (14), $\mathbf{z}^{(t)} = \mathscr{G}\mathbf{z}^{(t-1)}$). In the case of the simple power method, the convergent sequence $\{\lambda^{(t)}\}$ for the dominant eigenvalue can be transformed into a more rapidly convergent sequence $\{\tilde{\lambda}^{(t)}\}$ by using

$$\tilde{\lambda}^{(t)} = \lambda^{(t-2)} - \frac{(\lambda^{(t-2)} - \lambda^{(t-1)})^2}{\lambda^{(t-2)} - 2\lambda^{(t-1)} + \lambda^{(t)}}. \tag{45}$$

This process will be most effective if both eigenvalues $\nu_1 \equiv \nu_1^+$ and $\nu_2 \equiv \nu_2^+$ are real and well separated from $\nu_3 \equiv \nu_3^+$. As can be easily concluded from Figure 1,

this occurs when ω is close to $\overline{\omega}_3$, which minimizes ν_3 for all $1 \leq \omega \leq \overline{\omega}_3$ and provides the best separation of ν_1 and ν_2 from ν_3. The distance of separation is a decreasing function as ω increases for $\overline{\omega}_3 \leq \omega \leq \overline{\omega}_2$ and vanishes for $\overline{\omega}_2 < \omega \leq \omega_1$ because in this region all subdominant eigenvalues have the same absolute value. Thus, the use of σ_{EE}, providing an underestimated value of σ_1, can give some advantages in the form of an increased rate of convergence when the Aitken extrapolation is applied. This aspect will be discussed and illustrated by numerical results in the next section.

In conclusion it should be stated that in the efficient use of the power method for determining an accurate value of the optimum relaxation factor in the SOR iterative method, the relaxation factors $\overline{\omega}_2$ and $\overline{\omega}_3$ play an important role; $\overline{\omega}_2$ maximizes the rate of convergence in the simple power method, whereas $\overline{\omega}_3$, providing the best separation of two dominant eigenvalues from the remaining subdominant eigenvalues of the SOR iteration matrix, maximizes the rate of convergence of the Aitken extrapolation used as a practical technique for improving the convergence of the power method.

3. Numerical Experiments

In this section the results of numerical experiments are presented for the numerical solution of a two-dimensional elliptic equation of the form

$$-D(x, y)\left[\frac{\partial \varphi^2}{\partial x^2} + \frac{\partial \varphi^2}{\partial y^2}\right] + \Sigma(x, y)\varphi = \mathbf{s}(x, y) \quad \text{for } x, y \in \Omega \tag{46}$$

with

$$\varphi(x, y) = \mathbf{g}(x, y) \quad \text{or} \quad \frac{\partial \varphi}{\partial n} = \mathbf{g}(x, y) \quad \text{for } x, y \in \partial\Omega,$$

where Ω is an open bounded region with boundary $\partial\Omega$, n is the exterior normal, $D(x, y) > 0$, and $\Sigma(x, y) \geq 0$.

The standard finite difference discretization of (46) in a spatial mesh imposed on Ω leads to a system of linear equations of the form

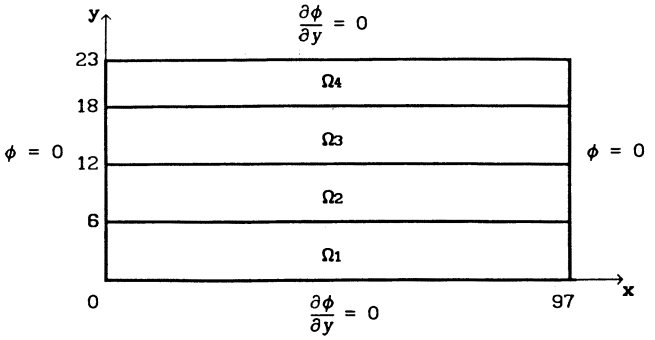
$$\mathbf{A}\phi = \mathbf{b}, \tag{47}$$

where the components of ϕ approximate the values of φ at each mesh point (x, y). In the case of the natural ordering of mesh points for the standard five-point difference operator, the $n \times n$ coefficient matrix $\mathbf{A}$ has only five nonzero diagonals forming a tridiagonal block structure suitable for the implementation of the 1-line SOR algorithm, and is 2-cyclic consistently ordered [1].

Five test problems taken from the literature [6, 7] are considered with discontinuous coefficients D and Σ, but chosen to be constant in each subregion Ω_k, and different boundary conditions on $\partial\Omega$ for uniform and nonuniform mesh structures.

Test Problem 1. This example, obtained by assuming $D = 1$ and $\Sigma = 0$ in Ω, the unit square $(0, 1) \times (0, 1)$, the Dirichlet boundary conditions $\varphi = 0$ on $\partial\Omega$, is usually used as a model problem in the analysis of numerical solutions of elliptic-type problems. A square mesh with width $h = \frac{1}{N+1}$ yields $n = N^2$ mesh points, which is also the order of $\mathbf{A}$. We assume $n = 48 \times 48 = 2304$, as in Problem A in [6].

Test Problem 2. In this problem (Problem B in [6]), whose domain and coefficients are depicted in Figure 2 (the numbers on the x-axis and y-axis in this

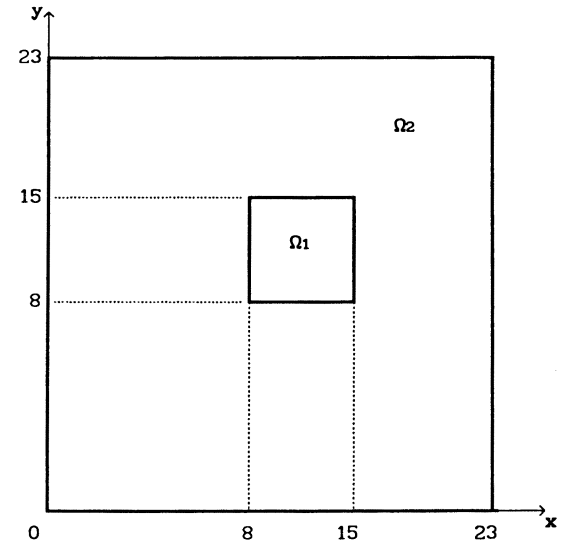


Ω	D	Σ
1	5.	0.
2	4.	0.
3	9.	0.
4	1.	0.

$\Omega = (0,97)\times(0,23)$

FIGURE 2. Test Problem 2

Interval	.1633	.3033	.2562	.17	.1	.17	.2562	.25275	.1633
To vertical line	1	6	7	8	15	16	17	22	23



Ω	D	Σ
1	500.	0.05
2	2.	0.05

$\Omega = (0,4.65925)\times(0,4.65925)$

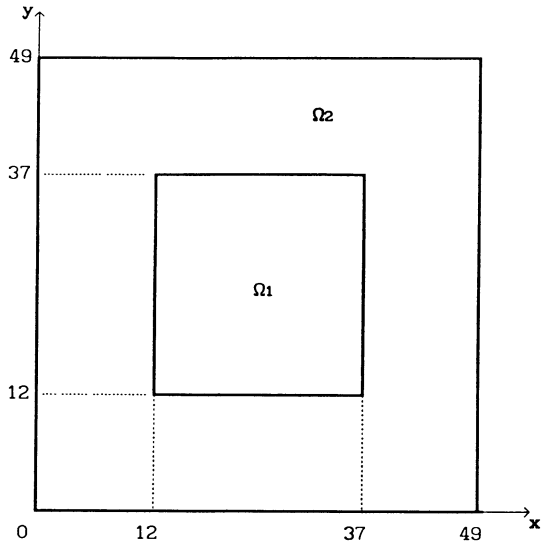
$\frac{\partial \phi}{\partial n} = 0$ on $\partial\Omega$

FIGURE 3. Test Problem 3

and subsequent figures are indices of mesh lines, not values of x and y), there is a discontinuity of coefficients in the vertical direction, and mixed boundary conditions are used on $\partial\Omega$ as shown in Figure 2. The number of mesh points is $n = 96 \times 24 = 2304$, where $h = 1$ is assumed in both horizontal and vertical direction.

Test Problem 3. In this problem (Problem C in [6]), with $n = 24 \times 24 = 576$ and discontinuous coefficients, a nonuniform mesh is used. The mesh division, assumed the same in both horizontal and vertical direction, corresponds to the mesh division used in Problem 5 given in Reference 7 of [6]. The domain, coefficients and the mesh division are depicted in Figure 3.

Test Problem 4. This problem, taken from [7] (and analyzed in [8]), has a strongly discontinuous D, and $n = 48 \times 48 = 2304$ in the square mesh shown in Figure 4.



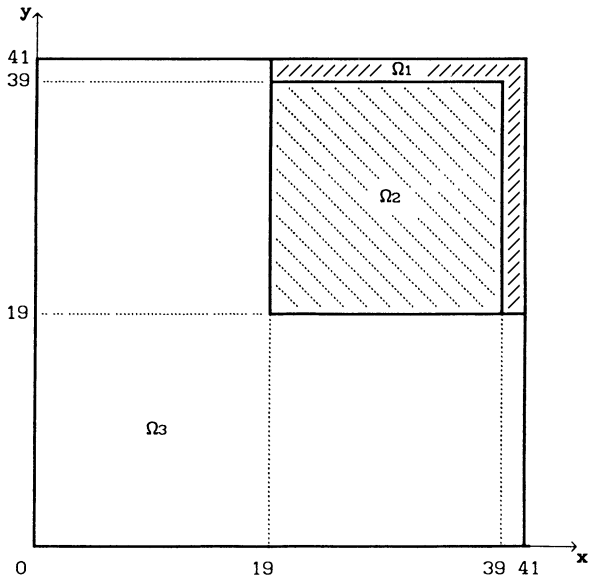
Ω	D	Σ
1	1000.	0.
2	1.	0.

$\Omega = (0,1)\times(0,1)$

$\phi = 0$ on $\partial\Omega$

FIGURE 4. Test Problem 4

Interval	.05263158	.05
To vertical line	19	41



Ω	D	Σ
1	1.	0.02
2	2.	0.03
3	3.	0.05

$\Omega = (0,2.1)\times(0,2.1)$

$\frac{\partial\phi}{\partial n} = 0$ on $\partial\Omega$

FIGURE 5

Test Problem 5. This problem, also taken from [7] (and analyzed in [8]), has a slightly modified mesh division, giving $n = 42 \times 42 = 1764$, in order to keep the number of horizontal lines divisible by 2 for convenient use of 2-line SOR algorithms. The domain, coefficients and mesh division (assumed the same in both directions) are depicted in Figure 5. In [7, 8] a uniform mesh with $h = 0.05$ was used, giving the number of mesh points $n = 43 \times 43 = 1849$.

For solving (47) in the above five test problems, the following line algorithms of the SOR iterative method are used [1, 6]:

1. SLOR—1-line system,
2. S2LOR—2-line system,
3. S2LCROR—2-line cyclically reduced system.

In our computations for each problem it was assumed that $s(x, h) \equiv 0$ in (46), so that the unique solution of each discrete problem is the null vector. All components of the starting vector $\phi^{(0)}$ were equal to unity, and computations for each iterative method were continued until the maximum absolute value of all components of the iterate $\phi^{(t)}$ was less than a prescribed number ε. Thus, the stopping criterion

$$\varepsilon^{(t)} = \|\phi^{(t)}\|_\infty \leq \varepsilon \tag{48}$$

can be considered as the most reliable measure of the error vector in estimating the accuracy of the solution.

All computations were carried out on a PC computer in single-precision FORTRAN for the SOR iteration (including the calculation of the coefficient matrices **A**), and in double-precision FORTRAN for the power iteration. The results of computation are shown in Table 2 (next page).

The accurate value of λ_1 was obtained with $\omega = 1$ when the stabilization to nine significant figures of λ_E was observed in the power method ((19)–(22), using the Euclidean norm); I_E and I_A are the numbers of iterations observed in the power method without and with using the Aitken extrapolation (45), respectively; I_S is the number of SOR iterations required to satisfy the stopping criterion (48) for two successive iterations with $\overline{\omega}_1$ as the optimum relaxation factor and for two values $\varepsilon = 10^{-6}$ and $\varepsilon = 10^{-8}$.

The results obtained when using the SOR adaptive subroutine [1, pp. 368–372] are shown under items 6, 7, and 8 of Table 2.

The data given in items 9–15 are related to computing the accurate value of ν_1 (with stabilization to nine significant figures) in the power method with the value of $\omega = \overline{\omega}_2$ determined from (27a) where $\lambda_2 = \{\sigma_1[\text{accur}]\} \times \lambda_1$ and $\sigma_1[\text{accur}]$, approximated by σ_{EE} (defined by (44d)), was obtained with the calculation of λ_1 in item 1 for $\omega = 1$. Hence, by (8) and (27), the accurate value of $\overline{\omega}_1$ can be found. Provided σ_1 is known, the accurate value of the optimum relaxation factor $\omega_{opt} \equiv \overline{\omega}_1$ can thus be efficiently computed. Comparison of the number of iterations I_E (or I_A) given in items 2 and 13 allows us to illustrate the efficiency of the power method used in the case when $\omega = \overline{\omega}_2$ for each test problem. The values of $\sigma_{\overline{\omega}}$ given in items 14 and 15, and computed from $(\overline{\omega}_2 - 1)/\nu_1$ and (32), respectively, indicate the consistency of the results in all cases, except for Test Problem 5 solved by the SLOR iterative method, where $\sigma_1 = 0.9944$ was found to only four significant figures.

The results obtained for the Sigma-SOR algorithm are given in items 16–27. The subdominance ratio σ_1, approximated by σ_{EE} is estimated once the stopping criterion

$$\delta^{(t)} = |\sigma_{EE}^{(t)} - \sigma_{EE}^{(t-1)}| \leq \delta = 10^{-3} \tag{49}$$

has been satisfied in two successive iterations in all test problems; I_{EE} is the

TABLE 2. Computational results

	Test Problem 1			Test Problem 2			Test Problem 3			Test Problem 4			Test Problem 5		
	SLOR	S2LOR	S2LCROR	SLOR	S2LOR	S2LCROR	SLOR	S2LOR	S2LCROR	SLOR	S2LOR	S2LCROR	SLOR	S2LOR	S2LCROR
1. λ_1[accur]	.991815239	.983729344	.978344547	.998951986	.997887384	.997029824	.999961143	.999909653	.999878524	.999983580	.999968430	.999956693	.999956430	.999910469	.999882183
2. I_E, I_A	984, 650	551, 277	430, 226	826, 462	481, 188	330, 235	1327, 571	644, 300	479, 211	485, 329	306, 173	210, 132	592, 145	372, 180	320, 139
3. $\tilde{\omega}_1$	1.83407	1.77375	1.74344	1.93728	1.91211	1.89663	1.98761	1.98117	1.97820	1.99193	1.98883	1.98692	1.98689	1.98125	1.97852
4. $I_S[\varepsilon=10^{-6}]$	106	72	64	269	189	160	1347	882	757	2048	1486	1275	1281	893	777
5. $I_S[\varepsilon=10^{-8}]$	132	91	80	343	241	204	1739	1138	975	2634	1914	1645	1651	1152	1001
6. ω_{Adap}	1.83328	1.78273	1.76228	1.93587	1.91437	1.89755	1.98765	1.98128	1.97823	1.99186	1.98878	1.98698	1.98700	1.98127	1.97891
7. $I_S[\varepsilon=10^{-6}]$	127	83	70	343	208	195	1853	1132	997	3090	2047	1705	1738	1154	925
8 $I_S[\varepsilon=10^{-8}]$	155	98	90	431	275	234	2234	1370	1215	3743	2498	2058	2074	1412	1088
9. σ_1[accur]	.98770	.97572	.96760	.99167	.98334	.9766	.98976	.97768	.96666	.98430	.97126	.95979	.9944	.98897	.98520
10. λ_2	.97962	.95984	.94665	.99063	.98126	.9737	.98972	.97759	.96654	.98428	.97123	.95974	.9944	.98888	.98508
11. $\tilde{\omega}_2$	1.7501	1.6661	1.6247	1.8235	1.7592	1.7209	1.8159	1.7396	1.6907	1.7772	1.7100	1.6658	1.8608	1.8092	1.7823
12. ν_1	.937281201	.912047890	.898987883	.988910479	.984221808	.981300669	.999615978	.999395998	.999335582	.999869000	.999813813	.999784103	.999416594	.999149680	.999033961
13. I_E, I_A	120, 101	82, 80	70, 73	102, 101	63, 64	67, 68	108, 103	68, 63	62, 52	91, 92	65, 67	55, 57	112, 122	90, 83	73, 69
14. $\sigma_{\tilde{\omega}}$ [comp]	.8003	.7303	.6949	.8327	.7714	.7346	.8162	.7400	.6912	.7773	.7101	.6659	.8613	.8099	.7831
15. $\sigma_{\tilde{\omega}}$ [theor]	.8003	.7304	.6949	.8327	.7714	.7346	.8161	.7400	.6912	.7773	.7101	.6659	.8608	.8099	.7831
16. $\sigma_{EE}[\delta=10^{-3}]$	.96170	.93618	.91761	.96239	.96689	.96676	.95238	.97633	.98773	.95804	.89515	.85906	.93316	.96876	.96610
17. I_{EE}	39	27	23	46	47	44	22	39	30	25	28	25	22	40	57
18. λ_2[est]	.95385	.92101	.89789	.96121	.96480	.96387	.95205	.97613	.96755	.95845	.89508	.85901	.93312	.96867	.96599
19. $\tilde{\omega}_2$[est]	1.646	1.561	1.516	1.671	1.684	1.681	1.641	1.732	1.695	1.661	1.511	1.454	1.589	1.699	1.689
20. $\nu_E[\delta=10^{-8}]$	.960797729	.940410621	.930273746	.994653474	.988626551	.984126949	.999822172	.999415687	.999324477	.999919651	.999902541	.999884748	.999831803	.999494535	.999359879
21. I_E	183	120	107	121	61	45	199	61	44	78	58	55	29	67	65
22. λ_E	.991785846	.983729365	.978344567	.998951966	.997887376	.997029820	.999961101	.999909647	.999878522	.999983602	.999968460	.999956722	.999956499	.999910493	.999882201
23. ω_E	1.83380	1.77375	1.74344	1.93728	1.91211	1.89663	1.98760	1.98117	1.97820	1.99193	1.98883	1.98693	1.98690	1.98126	1.97853
24. $\nu_A[\delta=10^{-8}]$	.960797526	.940410145	.930273647	.994653637	.988626583	.984126976	.999822354	.999415721	.999324484	.999919534	.999902457	.999884668	.999831767	.999494432	.999359775
25. I_A	100	55	59	67	44	43	76	54	45	69	41	37	27	40	42
26. λ_A	.991815225	.983729240	.978344538	.998951998	.997887382	.997029825	.999961141	.999909652	.999878523	.999983578	.999968433	.999956692	.999956489	.999910475	.999882181
27. ω_A	1.83407	1.77375	1.74344	1.93728	1.91211	1.89663	1.98761	1.98117	1.97820	1.99193	1.98883	1.98692	1.98689	1.98125	1.97852
28. ω_B	1.83704	1.77765	1.74777	1.93847	1.91376	1.89855	1.98785	1.98154	1.97862	1.99209	1.98905	1.98717	1.98715	1.98161	1.97894
29. $I_B[\varepsilon=10^{-6}]$	99	66	58	229	160	136	1139	740	634	1736	1284	1103	1077	759	654
30. ω_B	1.83557	1.77572	1.74563	1.93788	1.91295	1.89760	1.98773	1.98135	1.97841	1.99201	1.98894	1.98705	1.98702	1.98143	1.97873
31. $I_S[\varepsilon=10^{-8}]$	125	86	75	310	217	184	1573	1028	872	2331	1719	1489	1489	1047	903

respective number of iterations required. As can be seen in Table 2, the above stopping test provides an underestimation of σ_1 in all cases except for the S2LCROR method in Test Problem 3, for which σ_{EE} gives a slight overestimation of σ_1. In computing $\overline{\omega}_2$[est] according to (27a) it was assumed that $\lambda_2[\text{est}] = \sigma_{EE}\lambda_1^{(t)}$, where $\lambda_1^{(t)}$ is the approximation of λ_1 obtained at iteration $t = I_{EE}$ and using Aitken extrapolation. In item 20 the value ν_E approximating ν_1 with $\omega = \overline{\omega}_2$[est] is obtained by satisfying the stopping criterion

$$\delta^{(t)} = |\nu_E^{(t)} - \nu_E^{(t-1)}| \leq \delta = 10^{-8}, \tag{50}$$

which is achieved after I_E iterations without Aitken extrapolation The corresponding values of λ_E and ω_E are given in items 22 and 23. In items 24–27 the same quantities are given when Aitken extrapolation is used. For the SLOR method in Test Problem 1 there is a small difference between ω_E and $\omega_A = \overline{\omega}_1$, but in all remaining cases it is observed that $\omega_E = \omega_A = \overline{\omega}_1$ and I_A is smaller than I_E, as $\overline{\omega}_2$ is more underestimated by $\overline{\omega}_2$[est], because in this case the separation of λ_1 and λ_2 from the remaining eigenvalues increases and Aitken extrapolation becomes more efficient. In the case where the ω used is close to the true value of $\overline{\omega}_2$ (item 11), this separation of λ_1 and λ_2 from the remaining eigenvalues disappears and the numbers of iterations I_E and I_A are comparable (item 13).

Thus, with the choice $\delta = 10^{-3}$ for σ_{EE} and $\delta = 10^{-8}$ for ν_A and with the use of Aitken extrapolation, the Sigma-SOR algorithm provides an estimate for $\omega_A = \overline{\omega}_1 \equiv \omega_{opt}$ to six significant figures in all considered test problems, with $I_{EE} + I_A$ (items 17 and 25) being the number of iterations required for obtaining this estimate.

In all eigenvalue calculations carried out by means of the power method, all components of the starting vector $\mathbf{z}^{(0)}$ were taken to be unity.

The behavior of σ_E, σ_M, σ_{EE}, and σ_{EM} (defined by (44a, b, c, d)), representing different measures for σ_1, versus the number of iterations is depicted in Figures 6–10 (see pp. 636–638) for all five test problems solved by means of the SLOR iterative method. As can be seen in these figures, σ_{EE} converges most rapidly to σ_1. (The true value of σ_1 given in item 9 of Table 2 is marked in the figures by a straight line parallel to the x-axis.) In the initial phase of the iteration process, σ_{EE} provides estimates of σ_1 from below, which are helpful in using the Aitken extrapolation.

In the convergence behavior of σ_M, the decreasing character is observed as the number of iterations is increasing, but there are strong local variations (occurring sometimes also for σ_{EM}) visible in all figures, except for Test Problem 2 depicted in Figure 7. In the case of Test Problems 1 and 4 (Figures 6 and 9), it can be observed that for our starting vector $\mathbf{z}^{(0)}$, all of whose components are equal to unity, all measures considered for σ_1 tend first to λ_3/λ_1 and then to $\sigma_1 = \lambda_2/\lambda_1$ as the number of iterations increases. This is due to the fact that for the assumed starting vector $\mathbf{z}^{(0)}$ the inequality $a_3 \gg a_2$ in the representation (16) implies that in spite of $\lambda_2 > \lambda_3$, the inequality $|a_3\lambda_3^t| \gg |a_2\lambda_2^t|$ holds for appropriate "small" values of t, so that the inequality (40) is not satisfied because $t < t'$ (where t may not necessarily be very small if t' is very large, as occurs in the case of Test Problem 4) and $\sigma^{(t)}$ will converge to λ_3/λ_1, the dominant term in this range of t-values.

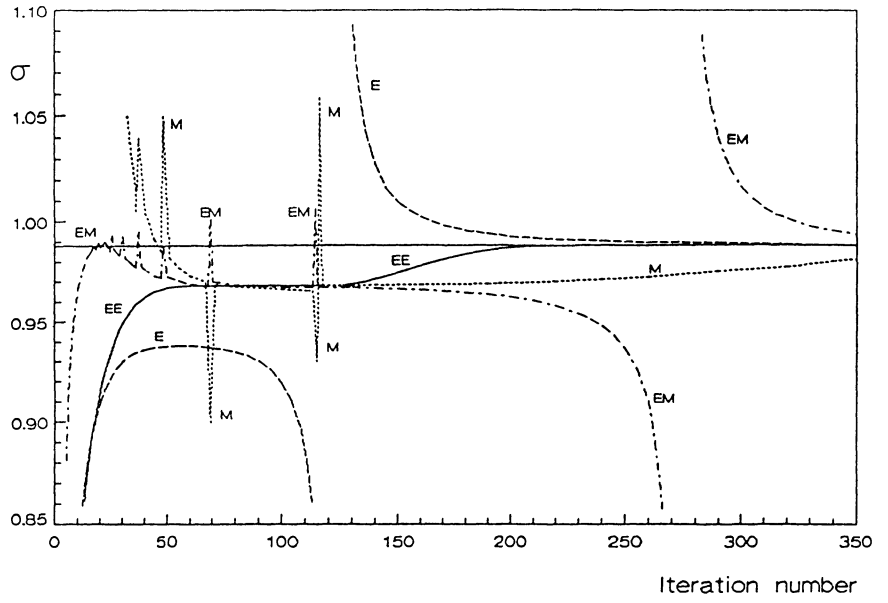


FIGURE 6. Test Problem 1

M: σ_M (eq. (44a)); E: σ_E (eq. (44b)); EM: σ_{EM} (eq. (44c)); EE: σ_{EE} (eq. (44d))

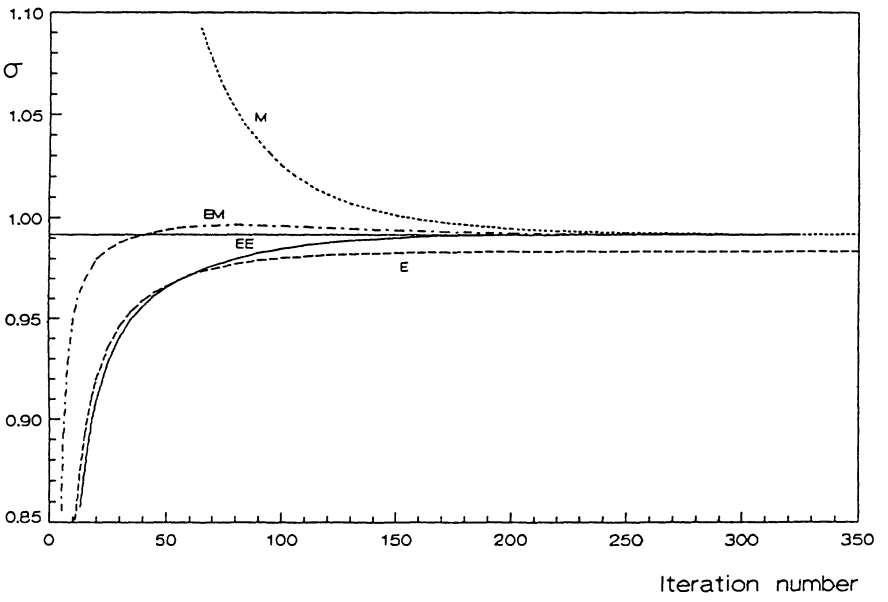


FIGURE 7. Test Problem 2

M: σ_M (eq. (44a)); E: σ_E (eq. (44b)); EM: σ_{EM} (eq. (44c)); EE: σ_{EE} (eq. (44d))

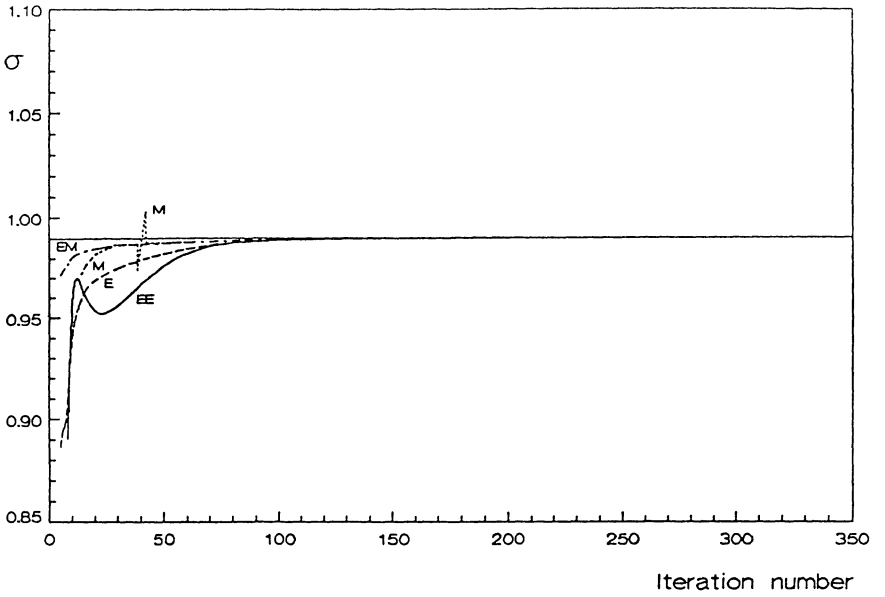


FIGURE 8. Test Problem 3

M: σ_M (eq. (44a)); E: σ_E (eq. (44b)); EM: σ_{EM} (eq. (44c)); EE: σ_{EE} (eq. (44d))

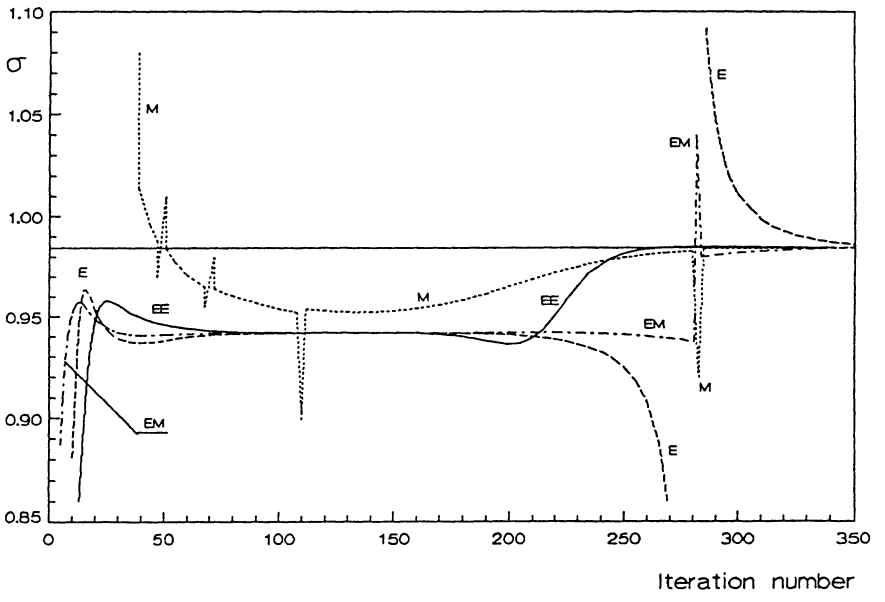


FIGURE 9. Test Problem 4

M: σ_M (eq. (44a)); E: σ_E (eq. (44b)); EM: σ_{EM} (eq. (44c)); EE: σ_{EE} (eq. (44d))

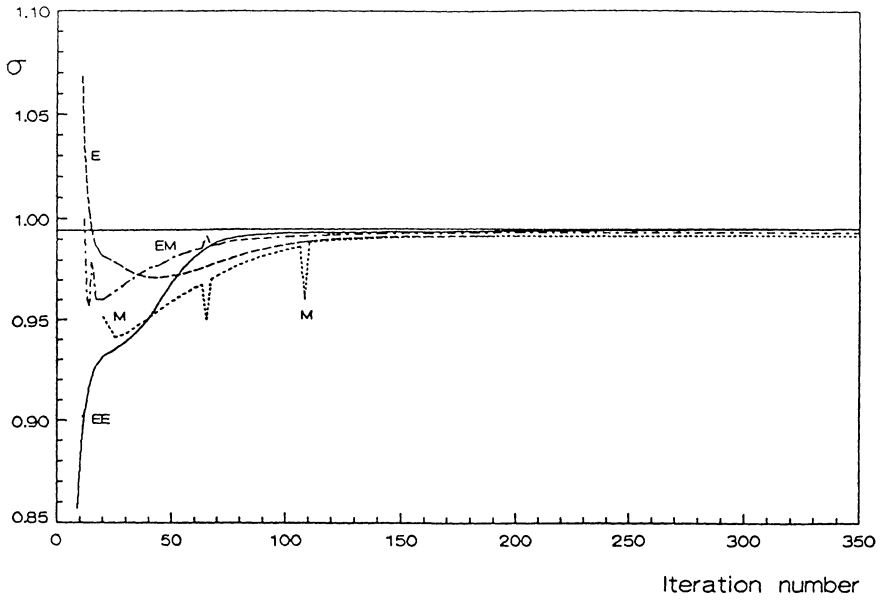


FIGURE 10. Test Problem 5

M: σ_M (eq. (44a)); E: σ_E (eq. (44b)); EM: σ_{EM} (eq. (44c)); EE: σ_{EE} (eq. (44d))

Moreover, it is interesting to notice that the convergence behavior of σ_{EE} and σ_M has a continuous character when passing from convergence to λ_3/λ_1 to convergence to λ_2/λ_1, whereas for σ_E and σ_{EM} strong deviations similar to discontinuities are observed.

It is a well-known fact that for the SOR iterative method the optimum relaxation factor $\omega_{opt} \equiv \overline{\omega}_1$ which maximizes theoretically the rate of convergence does not provide the best results. In practice, one observes the existence of a best relaxation factor ω_B (slightly greater then ω_{opt}) which minimizes the number of iterations for the required accuracy of the solution. Unfortunately, there is no rigorous analysis in the literature explaining the reasons for this ω_B and predicting its value. From numerical experience, it can be concluded that ω_B is a function of ω_{opt} and the required degree of accuracy of the solution. One observes the following empirical formula:

$$\ln(\omega_B - 1) = \frac{1}{c}\ln(\omega_{opt} - 1), \tag{51}$$

where the correction coefficient $c = 1.02$ when using $\varepsilon = 10^{-6}$, and $c = 1.01$ when using $\varepsilon = 10^{-8}$, provides a quite satisfactory estimate for ω_B. The use of ω_B obtained from the above formula allows us to improve the convergence. Usually, the number of iterations obtained with ω_B is about 15% less than that obtained with ω_{opt} for slowly convergent problems. The results obtained with ω_B for two different stopping criteria are given in items 28–31.

The deterioration in the rate of convergence resulting from using an inaccurate value of ω_{opt} is strongly dependent on the closeness of $\rho(\mathscr{L}_1)$ to unity, and it seems to be reasonable that this dependence should be taken in consideration when estimating ω_{opt} a priori. The nature of calculating $\rho(\mathscr{L}_1)$ by

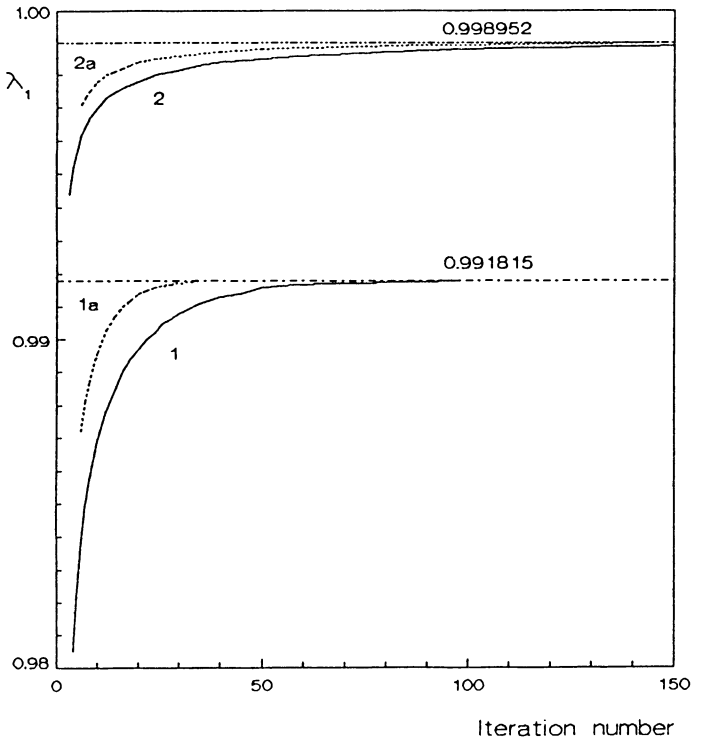


FIGURE 11. Test Problems 1 and 2

means of the power method is such that the first few significant figures of $\rho(\mathscr{L}_1)$ are rapidly fixed at the beginning of the power iterations, whereas convergence to the next figures begins to be governed by the subdominance ratio σ_1. The behavior of $\rho(\mathscr{L}_1)$ versus the number of power iterations for Test Problems 1 and 2 is depicted in Figure 11 where the dashed curves (denoted by 1a and 2a) correspond to using Aitken extrapolation for accelerating the convergence in the power method.

In the determination of ω_{opt} based on a priori estimates for $\rho(\mathscr{L}_1)$, the application of the stopping criterion

$$\delta^{(t)} = |\lambda_A^{(t)} - \lambda_A^{(1-t)}| \leq \bar{\delta} = 10^{-3}|(1 - \lambda_A^{(t)})|, \tag{52}$$

where $\lambda_A^{(t)}$ is an approximation of $\lambda_1 \equiv \rho(\mathscr{L}_1)$ in the power iteration t using the Aitken extrapolation, yields results strongly competitive with the SOR adaptive procedure [1] when the values of $\rho(\mathscr{L}_1)$ are close to unity.

In items 32–35 of Table 3 (next page) results are given for all test problems solved by the SLOR method in which the estimate of ω_{opt} is based on the computation of $\lambda_1 \equiv \rho(\mathscr{L}_1)$ by using the stopping criterion (52); the remaining items quoted from Table 2 are given for comparison purposes.

Table 4 summarizes the results obtained for different computational strategies implemented in four programs used for solving the test problems. The data given in this table represent the numbers of iterations required to obtain the solution which the stopping criterion $\|\phi^{(t)}\|_\infty \leq 10^{-6}$ satisfied for two successive

TABLE 3. Results obtained with using the "dynamic" stopping criterion (52)

	Test Problem 1	Test Problem 2	Test Problem 3	Test Problem 4	Test Problem 5
1. λ_1 [accur]	.991815239	.998951986	.999961143	.999983580	.999956430
2. I_A	650	462	571	329	145
3. $\bar{\omega}_1$	1.83407	1.93728	1.98761	1.99193	1.98689
4. $I_S[\varepsilon=10^{-6}]$	106	269	1347	2048	1281
6. ω_{Adap}	1.83328	1.93587	1.98765	1.99186	1.98700
7 $I_S[\varepsilon=10^{-6}]$	127	343	1853	3090	1738
17. I_{EE}	39	46	22	25	22
25. I_A	100	67	76	69	27
28. ω_B	1.83704	1.93847	1.98785	1.99209	1.98715
29. $I_B[\varepsilon=10^{-6}]$	99	229	1139	1736	1077
32. $\lambda_1[\bar{\delta}=10^{-3}(1-\lambda_1)]$	.991816463	.998929054	.999960952	.999983490	.999956396
33. I_A	35	96	155	101	18
34. $\bar{\omega}_{Est}$	1.83408	1.93662	1.98758	1.99191	1.98688
35. $I_S[\varepsilon=10^{-6}]$	106	283	1365	2077	1286

TABLE 4. Comparison of computational strategies

Method	Program No.	Test Problem 1	Test Problem 2	Test Problem 3	Test Problem 4	Test Problem 5
1-line	A1	127	343	1853	3090	1738
	B1	106 (35)	283 (96)	1365 (155)	2077 (101)	1286 (18)
	C1	99 (139)	229 (113)	1139 (98)	1736 (84)	1077 (49)
2-line	A2	83	208	1132	2047	1154
	B2	72 (21)	193 (58)	866 (74)	1501 (55)	890 (8)
	C2	66 (82)	160 (91)	740 (93)	1284 (69)	759 (80)
	D2	61	169	752	-	-
2-line cyclically reduced	A3	70	195	997	1705	925
	B3	63 (17)	162 (45)	733 (70)	1270 (43)	775 (4)
	C3	58 (82)	136 (87)	634 (75)	1103 (62)	654 (99)
	D3	52	145	681	-	-

iterations. The numbers given in parentheses correspond to the number of iterations required to compute the relaxation factor ω for a given strategy.

The **A** program uses the SOR adaptive procedure [1]. In the **B** program the estimate of ω_{opt} is based on computing $\lambda_1 \equiv \rho(\mathscr{L}_1)$ by using the stopping criterion (52) and Aitken extrapolation as an acceleration procedure. The **C** program uses the Sigma-SOR algorithm for computing ω_B. The numbers attached to the programs correspond to the applied solution methods, which are specified in the first column of the table. In addition, the results from [6] are quoted under the **D2** program, which uses the 2-line cyclic Chebyshev method

applied to the original system, and the **D3** program, which uses the 2-line cyclic Chebyshev method applied to the cyclically reduced system. Both these programs were used in [6] for solving Test Problems 1, 2, and 3 only; the results from these programs for Test Problems 4 and 5 were not available.

4. Concluding Remarks

From the practical point of view, the best solution method is one that for the required accuracy provides the solution with the minimum total arithmetical effort, which is what mainly determines the cost of computations. In the case of the SOR iterative method, the arithmetical effort is roughly proportional to the number of SOR iterations required for obtaining the solution with a given degree of accuracy, and the number of power iterations required for estimating the appropriate relaxation factor ω. Since the number of arithmetical operations per iteration in both SOR and power methods are comparable (the power method defined by (19)–(22) needs a few additional arithmetical operations for computing the Euclidean norm and for division by this norm), the efficiency of the assumed solution method can be measured in terms of the total number of iterations. Moreover, this total number of iterations, as well as the fraction of both SOR and power iterations, may change from problem to problem.

The number of SOR iterations is roughly inversely proportional to the rate of convergence where the deterioration of the convergence rate resulting from using an inaccurate value of ω_{opt} is strongly dependent on the closeness of $\rho(\mathcal{L}_1)$ to unity. The speed of convergence in the power method is governed by the value of the subdominance ratio σ_ω, which determines the rate of convergence, similarly as $\rho(\mathcal{L}_\omega)$ does in the SOR method, and the number of power iterations is also strongly dependent on the closeness of σ_1 to unity or on the degree of separation of two dominant eigenvalues from the remaining ones, if the Aitken extrapolation is used. Thus, it seems that the selection and application of the iterative strategy for solving different problems should be based more on the analysis of results obtained in practice than on theoretical considerations.

In the test problems considered in this work and representing a class of nuclear engineering problems, we have

$$0.978 < \rho(\mathcal{L}_1) < 0.99999 \quad \text{and} \quad 0.96 < \sigma_1 < 0.995,$$

so that the analysis of numerical results obtained for these problems should also be conclusive with solving large-scale scientific problems.

It seems that in the selection of computational strategy in solving elliptic-type problems, the SOR adaptive technique (implemented in the **A1**, **A2**, and **A3** programs) is favored in the literature [1, 2, 3, 4, and 6] as a more efficient solution method in comparison with the computational strategy based on a priori estimate of ω_{opt}. However, the numerical experiments on all test problems considered here show that the **B2**, **B2**, and **B3** programs, in which an a priori estimate for ω_{opt} is obtained by calculating $\lambda_1 \equiv \rho(\mathcal{L}_1)$ with the power method accelerated by Aitken extrapolation and using the stopping criterion (52), are competitive with the **A1**, **A2**, and **A3** programs, especially when $\rho(\mathcal{L}_1)$ is close to unity.

As can be seen in Table 4, in the case of Test Problem 1 the **B1** program

needs 14 iterations more (that is, about 10% more) than the **A1** program. But for Test Problem 4 the difference is equal to 912 iterations in favor of the **B1** program, which corresponds to about 40% more iterations in the **A1** program. Since both test problems have the same size (2304 mesh points), the advantages resulting from solving Test Problem 4 by the **B1** program in comparison to the **A1** program can be estimated by this difference of iterations, which in this case is about seven times greater than the total number of iterations required for solving Test Problem 1 by the **A1** or **B1** programs.

Suppose that both problems are solved with an a priori estimate for ω_{opt} based on using the accurate value of $\rho(\mathscr{L}_1)$ given in item 1 of Table 3 and obtained with 650 and 329 iterations (item 2 of Table 3) for Test Problems 1 and 4, respectively. Then, in the case of Test Problem 1 the solution is obtained with 106 iterations (the same number of iterations as for the B1 solution), but the total number of iterations is increased to 755, that is, 615 iterations more than for the **B1** solution given in Table 4. For Test Problem 4 the total number of iterations (accompanied by a small decrease of SOR iterations) is increased to 2377, that is, 199 iterations more than for the **B1** solution but still much less than for the **A1** solution. A similar behavior can be observed when comparing the results of Table 4 given for the **A2** and **A3** programs with those given for the **B2** and **B3** programs, respectively.

From the above comparisons, it is apparent that in the solution method based on a priori estimates for ω_{opt}, the main difficulty lies in the choice of the degree of accuracy appropriate for estimating $\rho(\mathscr{L}_1)$ in a given problem; it is probably for this reason that a priori estimates for ω_{opt} are given less attention in the literature. However, as can be concluded from the results given in Table 4 for the **B1**, **B2**, and **B3** programs, the simple trick of using the stopping criterion (52) conditioned by the closeness of $\rho(\mathscr{L}_1)$ to unity allows us in some sense to avoid this main difficulty and to make a priori estimation of ω_{opt} a more useful computational technique and competitive with the solution method based on using the SOR adaptive procedure [1], especially for problems in which the values of $\rho(\mathscr{L}_1)$ are very close to unity. In the range $0.98 \leq \rho(\mathscr{L}_1) \leq 0.999$, represented by Test Problems 1 and 2, the SOR adaptive procedure discussed extensively and illustrated numerically in [1] just for this range of values of $\rho(\mathscr{L}_1)$, provides solutions with a smaller number of iterations than in the case of using a priori estimates for ω_{opt} based on the stopping test (52). But as was demonstrated above for Test Problem 1, the advantages resulting from decreasing the total number of iterations have no practical significance because in this range of spectral radii, the deterioration of the convergence rate caused by using an inaccurate value of ω_{opt} does not strongly change the number of iterations. For the class of problems with $0.999 < \rho(\mathscr{L}_1) < 0.99999$, represented by Test Problems 3, 4, and 5, the efficiency of solution becomes more sensitive to the accurate value of ω_{opt} as $\rho(\mathscr{L}_1)$ approaches unity, and the computational strategy based on determining an accurate value of ω_{opt} prior to the SOR solution is much superior than the SOR adaptive technique, as can be seen in Table 4. In this case, the last estimate for ω_{opt} in the SOR adaptive technique is most time-consuming because σ_ω becomes close to unity (see Figure 1). It is interesting to note that in the case of Test Problem 5 extremely small numbers of iterations are required to a priori estimate ω_{opt} in the **B1**, **B2**, and **B3** programs.

In the **C1**, **C2**, and **C3** programs, the Sigma-SOR algorithm defined by (35a)–(35c) is used for the a priori determination of ω_{opt}, whose value to six significant figures was computed with the choice of $\delta = 10^{-3}$ for approximating σ_1 by σ_{EE} and $\delta = 10^{-8}$ for approximating ν^* by ν_{A}, and using the Aitken extrapolation. The detailed results are given in items 16–27 in Table 2. In SOR iterations the best relaxation factor ω_B is used which is computed from the relation (51) and is given in item 28 of Table 2. As can be seen in Table 4, the Sigma-SOR algorithm needs about 100 iterations for computing ω_{opt} to six significant figures in all test problems. For Test Problem 1 the number of iterations required to obtain this accurate estimate for ω_{opt} exceeds the number of SOR iterations, so that the total number of iterations in the **C1**, **C2**, and **C3** programs is about two times greater than in the **A1**, **A2**, and **A3** programs, respectively. However, as $\rho(\mathscr{L}_1)$ becomes close to unity in the next test problems, the efficiency of the computational strategy with the Sigma-SOR algorithm is strongly improving in comparison to the former solution methods. Moreover, it is observed that in the case of Test Problems 3, 4, and 5 solved by the **C1**, **C2**, and **C3** programs, the total number of iterations (needed for estimating ω_B and obtaining the solution) is smaller than the number of SOR iterations observed when using the accurate value of $\omega_{\text{opt}} = \overline{\omega}_1$ (items 3 and 4 in Table 2).

The results for Test Problems 1, 2, and 3 obtained in [6] by means of the **D2** program, using the 2-line cyclic Chebyshev method applied to the original system, and the **D3** program, using the 2-line cyclic Chebyshev method applied to the cyclically reduced system, are given additionally in Table 4. From an inspection of these results, it is apparent that the solution efficiency of the **D2** and **D3** programs, which is the best in the case of Test Problem 1, decreases when going to Test Problems 2 and 3 in comparison to the convergence behavior of the **C2** and **C3** programs, respectively. For Test Problem 3, the **C2** and **C3** programs provide solutions with the total number of iterations somewhat greater than in the **D2** and **D3** programs. However, as follows from an exact calculation of the number of arithmetical operations for the obtained solutions, the **C2** and **C3** programs need somewhat less total arithmetical effort than the **D2** and **D3** programs, respectively. This is due to the fact that in each iteration of the **D2** and **D3** programs, except for the arithmetical operations related with the solution, additional arithmetical operations are required for the computation of the Euclidean norm, whereas in the **C2** and **C3** programs only about 10% of the number of iterations (the numbers given in parentheses in Table 4) are related to those additional computations.

Thus, it can be concluded from the results obtained for our test problems, that the Sigma-SOR algorithm based on the important theoretical result given by (32) is a useful computational tool for the calculation of an accurate a priori estimate of ω_{opt}, which in turn allows to determine the best relaxation factor ω_B from (51) when solving problems for which $0.999 < \rho(\mathscr{L}_1) < 1$. In comparison to the SOR adaptive procedure, the efficiency of the Sigma-SOR algorithm increases as $\rho(\mathscr{L}_1)$ and σ_1 become closer to unity; and it seems that for the range $0.999 < \sigma_1 < 1$, the Sigma-SOR algorithm should be extremely efficient. In the case when the matrix problem (47) is to be solved many times for different vectors $\mathbf{b}$, the advantages resulting from using ω_B obtained by means of the Sigma-SOR algorithm are obvious.

Finally, it should be mentioned that the subsequent updated values of ω_i in the SOR adaptive technique are underestimated with respect to ω_{opt}, but this underestimation drastically decreases the rate of convergence as $\rho(\mathscr{L}_1)$ becomes close to unity, and therefore the efficiency of the SOR adaptive procedure also decreases when $\rho(\mathscr{L}_1)$ approaches unity.

ACKNOWLEDGMENT

The author would like to thank Drs. J. Kubowski and K. Pytel for their useful discussions and comments, as well as M. Sci. P. Jarzembowski for his expert programming assistance. Thanks are also due to the Editor, Professor W. Gautschi, for his significant contribution with revising the manuscript.

BIBLIOGRAPHY

1. L. A. Hageman and D. Young, *Applied iterative methods*, Academic Press, New York, 1981.
2. B. A. Carré, *The determination of the optimum accelerating factor for successive over-relaxation*, Comput. J. **4** (1961), 73–78.
3. H. E. Kulsrud, *A practical technique for the determination of the optimum relaxation factor of the successive over-relaxation method*, Comm. ACM **4** (1961), 184–187.
4. L. A. Hageman and R. B. Kellogg, *Estimating optimum relaxation factors for use in the successive overrelaxation and the Chebyshev polynomial methods of iteration*, WAPD-TM-592, 1966.
5. J. H. Wilkinson, *The algebraic eigenvalue problem*, Oxford Univ. Press, London, 1965.
6. L. A. Hageman and R. S. Varga, *Block iterative methods for cyclically reduced matrix equations*, Numer. Math. **6** (1964), 106–119.
7. P. Concus, G. H. Golub, and G. Meurant, *Block preconditioning for the conjugate gradient method*, SIAM J. Sci. Statist. Comput. **6** (1985), 220–252.
8. Z. I. Woźnicki, *On numerical analysis of conjugate gradient method*, Japan J. Indust. Appl. Math. **10** (1993), 487–519.

INSTITUTE OF ATOMIC ENERGY, 05-400 OTWOCK-SWIERK, POLAND
E-mail address: r05zw@cx1.cyf.gov.pl