SIAM J. NUMER. ANAL.
Vol. 26, No. 1, pp. 129–151, February 1989

008

A TWO-LEVEL FOUR-COLOR SOR METHOD*

C.-C. JAY KUO† AND BERNARD C. LEVY‡

Abstract. A two-level four-color SOR method is proposed for the nine-point discretization of the Poisson equation on a square. Instead of examining the Jacobi iteration matrix in the space domain, we consider an equivalent but much simpler four-color iteration matrix in the frequency domain. A two-level SOR method is introduced to increase the convergence rate for the frequency-domain iteration matrix. At the first level, the red and orange points, and then the black and green points are treated as groups, and a block SOR iteration is performed on these two groups. At the second level, another SOR iteration is used to decouple values at the red and orange points, and then at the black and green points. The conventional red/black SOR iteration for a five-point stencil is shown to be a degenerate case of the general two-level four-color SOR method. For the case of the nine-point stencil, a closed-form expression for the optimal relaxation parameters ω_b^* and ω_p^* at the two iteration levels is given, and the efficiency of the resulting method is shown both analytically and numerically.

Key words. successive overrelaxation, nine-point discretization, multicolor SOR, two-level iteration

AMS(MOS) subject classifications. 65N20, 65F10

1. Introduction.

The successive overrelaxation (SOR) method introduced in the late 1940s is an effective scheme for accelerating the Gauss–Seidel iteration [7], [12]. The acceleration effect relies on the properties of a special class of matrices known as *consistently ordered* matrices [13] (or *p-cyclic* matrices [11]). By discretizing elliptic PDEs with finite difference schemes, we often obtain sparse matrix equations where the matrix is consistently ordered. Consequently, the SOR method has a wide range of applications.

Two aspects need to be considered in the study of the SOR iteration: its *definition* and its *analysis*. The SOR scheme is usually defined in the space domain. In this paper, we demonstrate a new way to define an SOR scheme in the Fourier, or frequency, domain. As far as the analysis of the SOR method is concerned, several situations can occur. In some cases, there exists a relation between the eigenvalues of the Jacobi matrix and the eigenvalues of the SOR matrix, and in this situation, the analysis of the SOR method does not require the knowledge of the eigenvectors of the Jacobi and SOR matrices. Therefore this type of SOR analysis relies entirely on matrix eigenvalue analysis. In other cases, it is possible to compute the eigenvectors of the Jacobi matrix and/or the SOR matrix. Since the eigenvectors are usually obtained by Fourier analysis, this form of SOR analysis is performed in the frequency domain. In this paper, to emphasize the important role played by Fourier analysis in the design and analysis of the SOR iteration, studies of the SOR method are divided into three categories, according to whether they rely on *Fourier/Fourier*, *space/Fourier*, or *space/eigenvalue analysis* approaches. In the Fourier/Fourier approach, both the definition and analysis of the SOR scheme are performed in the Fourier domain. In the space/Fourier approach, the SOR scheme is defined in the space domain but analyzed with Fourier

* Received by the editors December 1, 1986; accepted for publication (in revised form) January 25, 1988. This work was supported in part by Army Research Office grants DAAG29-84-K-0005 and DAAL03-86-K-0171, Advanced Research Projects Agency, monitored by the Office of Naval Research contract N00014-81-K-0742, and Air Force Office of Scientific Research contract F49620-84-C-0004. The work was performed when the authors were with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139.

† Department of Mathematics, University of California, Los Angeles, California 90024.

‡ Department of Electrical Engineering and Computer Science, University of California, Davis, California 95616.

techniques. Finally, for the space/eigenvalue analysis approach, the SOR scheme is defined in the space domain and analyzed by relating the eigenvalues of the Jacobi and SOR matrices.

Young's work is a typical example of the space/eigenvalue analysis approach (see [13, Chaps. 5, 6]). This approach starts from an expression for the SOR iteration in the space domain. Then, under some conditions such as consistent ordering and property A, an argument based on matrix algebra is used to find the relationship between the optimal relaxation parameter for the SOR method and the spectral radii of the Jacobi and SOR iteration matrices.

The approach used in [2], [7]–[10] can be viewed as a space/Fourier approach, which adopts the space domain formulation but uses a frequency domain, or Fourier, analysis technique. This approach still starts from a fixed expression for the SOR iteration in the space domain. Then, under the assumption that the PDEs have constant coefficients and are defined on a rectangular domain with Dirichlet or periodic boundary conditions, sinusoidal functions turn out to be eigenfunctions of the discretized system of equations [7], [10]. Hence, the system of equations can be decoupled by using these functions as a basis and, as a consequence, each frequency can be considered separately. This approach, although only rigorous for a restricted class of problems, provides a simple explanation of how the SOR method works.

A common feature of the above two approaches is that an SOR iteration form in the space domain has to be specified a priori. For simple cases such as for a five-point discretization of the Poisson equation, most reasonable SOR iteration forms lead to an analysis in which the optimal relaxation parameter can be determined in closed form. However, for more complicated cases, such as the nine-point stencil case, it is not easy to specify in advance an iteration form whose analysis will be easy [1], [3]. A class of nine-point stencil SOR iteration forms in the space domain was analyzed by Adams, LeVeque, and Young [2]. Since the iteration matrices obtained from these forms are not consistently ordered, the traditional SOR theory cannot be applied for determining the optimal relaxation parameter. Hence, Adams et al. used a separation of variables technique to study the eigenvalues and eigenfunctions of the system of equations, and showed that the optimal relaxation parameter can be determined by solving a quartic equation [2].

In this paper, we study the same problem, i.e., we develop an SOR method for the nine-point discretization of the Poisson equation. However, we use a Fourier/Fourier approach. This approach makes use of the traditional SOR theory for consistently ordered matrices in the *frequency* domain. We first divide grid points into four colors: red, orange, black, and green. By assuming that the PDE has constant coefficients and is defined on a square, we can apply Fourier analysis to each color so that a four-color matrix equation can be obtained in the frequency domain. The four-color matrix is block diagonal with 4×4 matrix blocks along the diagonal. Each of these blocks relates Fourier components of the four colors at a single frequency. If we partition the 4×4 matrix associated to a fixed frequency into four 2×2 blocks, it is consistently ordered with respect to blocks. At the first level, we can use a standard block SOR iteration to accelerate the block Jacobi relaxation. Then, to decouple values of two different colors within the same block, we have to invert a 2×2 matrix. This can be easily accomplished by using a point SOR iteration at the second level. Once the appropriate two-level SOR iteration form is determined in the frequency domain, it is straightforward to transform it back to the space domain.

This procedure yields a new two-level four-color SOR method that is completely different from the single-level SOR method studied in [2]. Suppose that all grid points

are partitioned into two groups G_1 and G_2 that contain, respectively, the red and orange points, and the black and green points. One iteration of the two-level four-color SOR method consists of the following four steps.

Step 1. The first half part of a block SOR iteration between G_1 and G_2 gives an intermediate function defined at points of G_1.

Step 2. With this intermediate function as driving function, several (usually two) point SOR iterations are performed between red and orange points within G_1 to obtain an updated PDE solution at points of G_1.

Step 3. The second half part of a block SOR iteration between G_1 and G_2 yields an intermediate function defined at points of G_2.

Step 4. With this intermediate function as driving function, several point SOR iterations are performed between black and green points within G_2 to obtain an updated solution at points of G_2.

In the above algorithm, Steps 1 and 3 constitute a complete block SOR iteration between groups G_1 and G_2 and define a first level of iteration. Steps 2 and 4 individually consist of several point SOR iteration operations and correspond to the second level of iteration. The values obtained in Steps 1 and 3 are used as driving functions in Steps 2 and 4, respectively. This computational algorithm will be detailed in § 3. The optimal relaxation parameters ω_b^* and ω_p^* at the two iteration levels can be expressed in closed form. The two-level SOR method is easy to implement, and its spectral radius is of the form $1-Ch$, where C is a constant comparable to the one obtained in [2].

The paper is organized as follows. In § 2, we use a simple one-dimensional two-color SOR method to demonstrate the Fourier/Fourier approach. Section 3 describes the main result of this paper, i.e., the two-dimensional two-level four-color SOR algorithm for a nine-point discretization of the Poisson equation. Then, in § 4, we show that the conventional two-dimensional single-level two-color SOR method for the five-point stencil case is a degenerate case of the general two-level four-color scheme. Closed-form formulas for the optimal relaxation parameters ω_b^* and ω_p^* corresponding to the two iteration levels are obtained § 5, where the convergence rate of the two-level SOR method is also analyzed. Finally, some numerical results are presented in § 6.

2. One-dimensional two-color SOR method. In this section, we consider a simple one-dimensional model problem and show how the two-color SOR method can be derived from the Jacobi iteration method by first transforming the problem to the frequency domain and then introducing the relaxation parameter ω inside the frequency-domain iteration matrix. Although the final result is well known, the approach we are taking is new and provides some new insight. The same approach will be used to develop a two-level iteration method in the next section.

2.1. Problem formulation. Consider the discrete one-dimensional Poisson equation on $[0, 1]$ with grid spacing h

$$\frac{1}{h^2}(u_{j-1}-2u_j+u_{j+1})=f_j, \qquad j=1,2,\cdots,N-1,$$

where u_0, u_N are given, and $N=1/h$. Suppose we divide the problem domain into red and black points corresponding, respectively, to points with even and odd indices. With this partitioning, the Jacobi iteration method takes the form

$$u_j^{n+1}=\tfrac{1}{2}(u_{j-1}^n+u_{j+1}^n-2h^2f_j) \qquad j \text{ even},$$
$$u_j^{n+1}=\tfrac{1}{2}(u_{j-1}^n+u_{j+1}^n-2h^2f_j) \qquad j \text{ odd}.$$

Denote the exact solution by $\bar{u}_j$ and define the error as $e_j^n = u_j^n - \bar{u}_j$. Then, the error equations can be written as

$$\begin{aligned} e_j^{n+1} &= \tfrac{1}{2}(e_{j-1}^n + e_{j+1}^n) \qquad j \text{ even}, \\ e_j^{n+1} &= \tfrac{1}{2}(e_{j-1}^n + e_{j+1}^n) \qquad j \text{ odd}, \end{aligned} \tag{2.1}$$

with $e_0 = e_N = 0$.

Since (2.1) is a system of linear constant-coefficient equations with homogeneous boundary conditions, the eigenfunctions of this system are given by $\sin(\xi\pi jh)$, where $\xi = 1, 2, \cdots, (N-1)$. These functions form a basis, so that

$$\begin{aligned} r_j^n &= \sum_{\xi=1}^{N-1} \hat{r}_\xi^n \sin(\xi\pi jh), \qquad 1 \leqq j \leqq N-1, \\ b_j^n &= \sum_{\xi=1}^{N-1} \hat{b}_\xi^n \sin(\xi\pi jh), \qquad 1 \leqq j \leqq N-1, \end{aligned} \tag{2.2}$$

where the coefficients $\hat{r}_\xi^n$ and $\hat{b}_\xi^n$ are chosen such that

$$\begin{aligned} r_j^n &= e_j^n \qquad j \text{ even}, \\ b_j^n &= e_j^n \qquad j \text{ odd}. \end{aligned} \tag{2.3}$$

In other words, r_j^n and b_j^n are two sequences which coincide with the errors at red and black points, respectively. They can be viewed as interpolations of the errors at the red and black points to all grid points. Note that there are $2(N-1)$ undetermined coefficients in (2.2) and only $N-1$ constraints in (2.3). Since (2.2) and (2.3) form an undetermined system of equations, there are many ways to choose $\hat{r}_\xi^n$ and $\hat{b}_\xi^n$. However, the actual values of these coefficients are not important. We are primarily concerned with how they evolve as the iteration proceeds.

Consider the error dynamics relating r_j and b_j,

$$\begin{aligned} r_j^{n+1} &= \tfrac{1}{2}(b_{j-1}^n + b_{j+1}^n), \qquad 1 \leqq j \leqq N-1, \\ b_j^{n+1} &= \tfrac{1}{2}(r_{j-1}^n + r_{j+1}^n), \qquad 1 \leqq j \leqq N-1. \end{aligned} \tag{2.4}$$

Note that (2.4) has twice as many equations and variables as (2.1) has. However, as shown in the following analysis, all the information contained in (2.1) is simply duplicated by (2.4), and consequently the dynamic behavior of (2.1) can be obtained by studying the dynamic behavior of (2.4). Conceptually, (2.4) is easier to analyze than (2.1) since it is a spatially invariant system for both red and black colors.

By substituting (2.2) inside (2.4), for $\xi = 1, 2, \cdots, N-1$, we have

$$\begin{pmatrix} \hat{r}_\xi^{n+1} \\ \hat{b}_\xi^{n+1} \end{pmatrix} = B(\xi) \begin{pmatrix} \hat{r}_\xi^n \\ \hat{b}_\xi^n \end{pmatrix}, \tag{2.5}$$

where

$$B(\xi) = \begin{bmatrix} 0 & \cos(\xi\pi h) \\ \cos(\xi\pi h) & 0 \end{bmatrix} \tag{2.6}$$

is called the *Jacobian iteration matrix* for the frequency $\xi\pi$, which has two eigenvalues

$$\mu_\xi = \pm\cos(\xi\pi h).$$

Observe that $\mu_\xi = -\mu_{N-\xi}$ so we only have to consider $\mu_\xi = \cos(\xi\pi h)$, $\xi = 1, 2, \cdots, N-1$. Intuitively speaking, we use the fact that the sinusoidal functions are

eigenfunctions of the linear system (2.4) so that, by changing the coordinate from the space domain to the frequency domain, we are able to decompose the loosely coupled system (2.4) into a decoupled system that is a block diagonal matrix containing many 2×2 matrices along the diagonal.

Since the spectral radius of $B(\xi)$ is less than one for any ξ, the iteration (2.5) converges. Consequently, the asymptotic values $\hat{r}_\xi^\infty$ and $\hat{b}_\xi^\infty$ obtained by this iteration procedure are $\hat{r}_\xi^\infty = \hat{b}_\xi^\infty = 0$, and (2.5) can be viewed as obtained by solving the linear system

$$A(\xi)\begin{pmatrix}\hat{r}_\xi^\infty\\ \hat{b}_\xi^\infty\end{pmatrix} = \begin{pmatrix}0\\0\end{pmatrix} \quad \text{with } A(\xi) = \begin{bmatrix}1 & -\cos(\xi\pi h)\\ -\cos(\xi\pi h) & 1\end{bmatrix},$$

by the Jacobi iteration in the frequency domain. In order to increase the convergence rate of (2.5), we have to reduce the spectral radius of $B(\xi)$.

2.2. Point SOR iteration. The key idea of this paper is that instead of considering the SOR method for the large matrix corresponding to (2.1), we can study the SOR scheme for each small 2×2 matrix given by (2.6) separately and, then, seek the best SOR scheme for all of them. Once the SOR scheme is obtained in the frequency domain, we transform the problem back to the space domain so that the corresponding spatial SOR iteration can be determined.

It is important to observe in this context that $A(\xi)$ and $B(\xi)$ are *consistently ordered.* Since the SOR theory was originally developed to accelerate the convergence rate of consistently ordered matrices, the SOR method can be applied directly to the iteration (2.5). The definitions of consistent ordering, and the details of the SOR theory are all presented in [11] and [13].

Since this is a standard procedure, we only summarize the result here. Let

$$A(\xi) = I - L(\xi) - U(\xi)$$

where $L(\xi)$ and $U(\xi)$ are lower and upper triangular matrices, respectively. Then, for a fixed frequency $\xi\pi$, the Jacobi iteration matrix is

$$B(\xi) = L(\xi) + U(\xi)$$

and the SOR iteration matrix associated with the frequency $\xi\pi$ is

$$G_\omega(\xi) = (I - \omega L(\xi))^{-1}\{(1-\omega)I + \omega U(\xi)\}. \tag{2.7}$$

In addition, the eigenvalues λ_ξ of $G_\omega(\xi)$ and the eigenvalues μ_ξ of $B(\xi)$ are related by [11, p. 106]

$$(\lambda_\xi + \omega_\xi - 1)^2 = \lambda_\xi \omega_\xi^2 \mu_\xi^2.$$

Hence,

$$\lambda_\xi = \left(\frac{\omega_\xi\mu_\xi \pm \sqrt{\Delta}}{2}\right)^2 \quad \text{where } \Delta = \omega_\xi^2\mu_\xi^2 - 4(\omega_\xi - 1),$$

and the spectral radius of (2.7) is

$$\rho_\xi = \begin{cases}\left(\dfrac{|\omega_\xi\mu_\xi| \pm \sqrt{\Delta}}{2}\right)^2 & \text{if } \Delta > 0,\\ \omega_\xi - 1 & \text{if } \Delta \leqq 0.\end{cases}$$

The above quantity can be minimized for all ξ by choosing

$$\omega^* = \frac{2}{1+[1-\mu_{\max}^2]^{1/2}} \quad \text{where} \quad \mu_{\max} = \max_{1\leqq\xi\leqq N-1} |\mu_\xi| = \cos(\pi h), \tag{2.8}$$

and the resulting spectral radius is

$$\rho^* = \omega^* - 1 \approx 1 - 2\sin(\pi h) = 1 - 2\pi h.$$

In particular, since the SOR method is applied to $A(\xi)$ partitioned with 1×1 diagonal submatrices, we call it the *point* SOR method.

The remaining problem is to transform the SOR iteration matrix (2.7) back to the space domain. By using the correspondence,

$$\cos(\xi \pi l h) = \tfrac{1}{2}(e^{i\xi\pi lh} + e^{-i\xi\pi lh}) \leftrightarrow \tfrac{1}{2}(E^l + E^{-l}) \quad l = 1, 2, \cdots,$$

where E^l and E^{-l} are the lth order forward and backward shift operators defined as $E^l u_j = u_{j+l}$ and $E^{-l} u_j = u_{j-l}$, we find that the SOR iteration for r_j^n and b_j^n becomes

$$\begin{aligned} r_j^{n+1} &= (1-\omega^*) r_j^n + \frac{\omega^*}{2}(b_{j-1}^n + b_{j+1}^n), \\ b_j^{n+1} &= (1-\omega^*) b_j^n + \frac{\omega^*}{2}(r_{j-1}^{n+1} + r_{j+1}^{n+1}). \end{aligned} \tag{2.9}$$

It is straightforward to reconstruct the SOR iteration from (2.9), i.e.,

$$u_j^{n+1} = (1-\omega^*) u_j^n + \frac{\omega^*}{2}(u_{j-1}^n + u_{j+1}^n - h^2 f_j) \qquad j \text{ even},$$

$$u_j^{n+1} = (1-\omega^*) u_j^n + \frac{\omega^*}{2}(u_{j-1}^{n+1} + u_{j+1}^{n+1} - h^2 f_j) \qquad j \text{ odd},$$

which is consistent with the conventional SOR method with red/black partitioning.

3. Two-dimensional two-level four-color SOR method.

3.1. Problem formulation. The one-dimensional two-color SOR scheme discussed in the previous section can be naturally generalized to the two-dimensional case by using four colors.

Consider the following discretized system with uniform grid spacing h,

$$\begin{aligned} &\frac{1}{h^2}\{q_1(u_{j+1,k} + u_{j-1,k}) + q_2(u_{j,k+1} + u_{j,k-1}) \\ &\qquad + q_3(u_{j+1,k+1} + u_{j+1,k-1} + u_{j-1,k+1} + u_{j-1,k-1}) - q u_{j,k}\} = f_{j,k} \\ &j, k = 1, 2, \cdots, N-1, \end{aligned} \tag{3.1}$$

where

$$q = 2q_1 + 2q_2 + 4q_3,$$

and $N = 1/h$, and where q_1, q_2, and q_3 are nonnegative and not all zeros. It is also assumed that values at all boundary points are given. The system (3.1) can be viewed as obtained from a five-point or nine-point stencil discretization [5] of the equation

$$q_1' \frac{\partial^2 u(x,y)}{\partial x^2} + q_2' \frac{\partial^2 u(x,y)}{\partial y^2} = f(x,y) \quad \text{where } q_1', q_2' > 0, \tag{3.2}$$

on the unit square $[0,1]^2$ with Dirichlet boundary conditions. In particular, when $q_1' = q_2'$, (3.2) becomes the Poisson equation. This section presents a Fourier approach for the design of a two-level four-color SOR method to solve (3.1). Several concrete examples will then be examined in §§ 4–6.

We can divide the grid points into four groups, say, red, black, green, and orange. A grid point is red if both j and k are even, black if j is odd and k is even, green if j is even and k is odd, and orange if both j and k are odd, as shown in Fig. 1. Following the procedure described in the previous section, to understand the error dynamics of the error associated with the Jacobi iteration for the system (3.1), we examine the dynamics of the four two-dimensional sequences

$$r_{j,k}^n = \sum_{\xi=1}^{N-1} \sum_{\eta=1}^{N-1} \hat{r}_{\xi,\eta}^n \sin(\xi\pi jh) \sin(\eta\pi kh), \qquad 1 \leqq j, k \leqq N-1,$$

$$b_{j,k}^n = \sum_{\xi=1}^{N-1} \sum_{\eta=1}^{N-1} \hat{b}_{\xi,\eta}^n \sin(\xi\pi jh) \sin(\eta\pi kh), \qquad 1 \leqq j, k \leqq N-1,$$

$$g_{j,k}^n = \sum_{\xi=1}^{N-1} \sum_{\eta=1}^{N-1} \hat{g}_{\xi,\eta}^n \sin(\xi\pi jh) \sin(\eta\pi kh), \qquad 1 \leqq j, k \leqq N-1,$$

$$o_{j,k}^n = \sum_{\xi=1}^{N-1} \sum_{\eta=1}^{N-1} \hat{o}_{\xi,\eta}^n \sin(\xi\pi jh) \sin(\eta\pi kh), \qquad 1 \leqq j, k \leqq N-1,$$

where the coefficients $\hat{r}_{\xi,\eta}^n$, $\hat{b}_{\xi,\eta}^n$, $\hat{g}_{\xi,\eta}^n$, $\hat{o}_{\xi,\eta}^n$ are chosen such that

$$r_{j,k}^n = e_{j,k}^n \quad j \text{ even} \quad k \text{ even}, \qquad b_{j,k}^n = e_{j,k}^n \quad j \text{ odd} \quad k \text{ even},$$

$$g_{j,k}^n = e_{j,k}^n \quad j \text{ even} \quad k \text{ odd}, \qquad o_{j,k}^n = e_{j,k}^n \quad j \text{ odd} \quad k \text{ odd},$$

where $e_{j,k}^n = u_{j,k}^n - \bar{u}_{j,k}$ is the nth iteration error at grid point (jh, kh).

As shown before, we can transform the Jacobi iteration for $r_{j,k}^n$, $b_{j,k}^n$, $g_{j,k}^n$, and $o_{j,k}^n$, or equivalently for the errors in the space domain at the red, black, green, and orange points, into an equivalent set of iterations for the Fourier coefficients $\hat{r}_{\xi,\eta}^n$, $\hat{b}_{\xi,\eta}^n$, $\hat{g}_{\xi,\eta}^n$, and $\hat{o}_{\xi,\eta}^n$ in the frequency domain. With respect to the frequency domain vector $(\hat{r}_{\xi,\eta}^n, \hat{o}_{\xi\ \eta}^n, \hat{b}_{\xi,\eta}^n, \hat{g}_{\xi,\eta}^n)^T$, these iterations can be viewed as solving the system

$$A(\xi, \eta) \begin{pmatrix} \hat{r}_{\xi,\eta}^{\infty} \\ \hat{o}_{\xi,\eta}^{\infty} \\ \hat{b}_{\xi,\eta}^{\infty} \\ \hat{g}_{\xi,\eta}^{\infty} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{3.3a}$$

where $A(\xi, \eta)$ is the frequency domain *coefficient* matrix for the frequency $(\xi\pi, \eta\pi)$

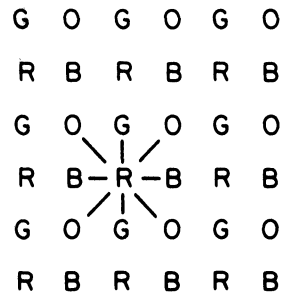

FIG. 1. *Four-color partitioning for the nine-point stencil discretization.*

and has the form

$$A(\xi, \eta) = \begin{bmatrix} 1 & -\alpha_3 & -\alpha_1 & -\alpha_2 \\ -\alpha_3 & 1 & -\alpha_2 & -\alpha_1 \\ -\alpha_1 & -\alpha_2 & 1 & -\alpha_3 \\ -\alpha_2 & -\alpha_1 & -\alpha_3 & 1 \end{bmatrix}, \tag{3.3b}$$

where

$$\alpha_1 = \frac{2q_1 \cos(\xi\pi h)}{q}, \quad \alpha_2 = \frac{2q_2 \cos(\eta\pi h)}{q}, \quad \alpha_3 = \frac{4q_3 \cos(\xi\pi h)\cos(\eta\pi h)}{q}. \tag{3.4}$$

Notice that the coefficient matrix $A(\xi, \eta)$ in (3.3) is symmetric and diagonally dominant with positive diagonal elements. The application of the Jacobi, block Jacobi, SOR, and block SOR iterations with $0 < \omega < 2$ to the system of equations (3.3) is expected to converge [11], [13].

3.2. Block SOR iteration. The matrix $A(\xi, \eta)$ partitioned such that its diagonal submatrices are all 1×1 matrices is *not* a consistently ordered matrix. However, if $A(\xi, \eta)$ is partitioned with 2×2 block diagonal submatrices, it is consistently ordered with respect to blocks. Hence, a *block* SOR iteration can be applied to $A(\xi, \eta)$ with this kind of partitioning.

The matrix A can be written as

$$A(\xi, \eta) = D(\xi, \eta) - E(\xi, \eta) - F(\xi, \eta)$$

where

$$D(\xi, \eta) = \begin{bmatrix} 1 & -\alpha_3 & 0 & 0 \\ -\alpha_3 & .1 & 0 & 0 \\ 0 & 0 & 1 & -\alpha_3 \\ 0 & 0 & -\alpha_3 & 1 \end{bmatrix}, \qquad E(\xi, \eta) = \begin{bmatrix} 0 & 0 & \alpha_1 & \alpha_2 \\ 0 & 0 & \alpha_2 & \alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

and $F(\xi, \eta) = E^T(\xi, \eta)$. In addition, we can define $L(\xi, \eta) \equiv D^{-1}(\xi, \eta)E(\xi, \eta)$ and $U(\xi, \eta) \equiv D^{-1}(\xi, \eta)F(\xi, \eta)$, so that

$$L(\xi, \eta) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \beta_1 & \beta_2 & 0 & 0 \\ \beta_2 & \beta_1 & 0 & 0 \end{bmatrix}, \qquad U(\xi, \eta) = \begin{bmatrix} 0 & 0 & \beta_1 & \beta_2 \\ 0 & 0 & \beta_2 & \beta_1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

where

$$\beta_1 = \frac{\alpha_1 + \alpha_2\alpha_3}{1 - \alpha_3^2}, \qquad \beta_2 = \frac{\alpha_2 + \alpha_1\alpha_3}{1 - \alpha_3^2}.$$

Then, the block Jacobi iteration matrix is

$$B(\xi, \eta) = L(\xi, \eta) + U(\xi, \eta), \tag{3.5}$$

and the corresponding block SOR iteration matrix is

$$G_\omega(\xi, \eta) = (I - \omega L(\xi, \eta))^{-1}\{(1 - \omega)I + \omega U(\xi, \eta)\}. \tag{3.6}$$

It is easy to find that the eigenvalues of $B(\xi, \eta)$ are double roots at

$$\mu_{\xi,\eta} = \beta_1 \pm \beta_2 = \frac{\alpha_1 + \alpha_2}{1 - \alpha_3}, \quad \frac{\alpha_1 - \alpha_2}{1 + \alpha_3}. \tag{3.7}$$

Note that the eigenvalue $(\alpha_1+\alpha_2)/(1-\alpha_3)$ at the frequency $(\xi\pi, \eta\pi)$ has the same value as the eigenvalue $(\alpha_1-\alpha_2)/(1+\alpha_3)$ at the frequency $(\xi\pi, (N-\eta)\pi)$ and, hence, we only have to consider the eigenvalues $(\alpha_1+\alpha_2)/(1-\alpha_3)$, $\xi, \eta = 1, 2, \cdots, N-1$.

The eigenvalues $\mu_{\xi,\eta}$ of the Jacobi iteration matrix $B(\xi, \eta)$ and the eigenvalues $\lambda_{\xi,\eta}$ of the SOR iteration matrix $G_\omega(\xi, \eta)$ are related by

$$(\lambda_{\xi,\eta}+\omega_{\xi,\eta}-1)^2 = \lambda_{\xi,\eta}\omega_{\xi,\eta}^2\mu_{\xi,\eta}^2.$$

Hence, if we proceed as in the one-dimensional case, except for a change of subscript from the one-dimensional index ξ to the two-dimensional index (ξ, η), we find that

$$\lambda_{\xi,\eta} = \left(\frac{\omega_{\xi,\eta}\mu_{\xi,\eta} \pm \sqrt{\Delta}}{2}\right)^2 \quad \text{where } \Delta = \omega_{\xi,\eta}^2\mu_{\xi,\eta}^2 - 4(\omega_{\xi,\eta}-1)$$

and the spectral radius of $G_\omega(\xi, \eta)$ is

$$\rho_{\xi,\eta} = \begin{cases} \left(\dfrac{|\omega_{\xi,\eta}\mu_{\xi,\eta}| \pm \sqrt{\Delta}}{2}\right)^2 & \text{if } \Delta > 0, \\ \omega_{\xi,\eta}-1 & \text{if } \Delta \leqq 0. \end{cases}$$

The above quantity is minimized for all ξ and η by choosing the following optimal relaxation parameter

$$\omega^* = \frac{2}{1+[1-\mu_{\max}^2]^{1/2}} \quad \text{where} \quad \mu_{\max} = \max_{1\leqq\xi,\eta\leqq N-1} |\mu_{\xi,\eta}|, \tag{3.8}$$

and the spectral radius of the corresponding SOR matrix is

$$\rho^* = \omega^* - 1.$$

3.3. Two-level SOR iteration. Suppose that one of the coefficients q_1, q_2, or q_3 is zero, or equivalently, that one of α_1, α_2, or α_3 is zero for all (ξ, η). Then, the four-color block SOR method described above reduces to an equivalent two-color SOR method, which corresponds to a degenerate case that will be discussed in § 4.

For the moment, consider the nondegenerate case where q_1, q_2, and q_3 are all strictly positive. In this case, the frequency-domain block SOR method given by (3.6) cannot be successfully transformed back to the space domain. We now describe a scheme, which we refer to as the two-level four-color SOR iteration for which a transformation back to the space domain is possible. To derive this method, we first rewrite (3.6) as

$$G_\omega(\xi, \eta) = (D(\xi, \eta) - \omega E(\xi, \eta))^{-1}\{(1-\omega)D(\xi, \eta) + \omega F(\xi, \eta)\},$$

and the corresponding space domain equation associated with the optimal block relaxation parameter ω^* becomes

$$\begin{aligned}
r_{j,k}^{n+1} - \alpha_3^S o_{j,k}^{n+1} &= (1-\omega^*)(r_{j,k}^n - \alpha_3^S o_{j,k}^n) + \omega^*(\alpha_1^S b_{j,k}^n + \alpha_2^S g_{j,k}^n), \\
-\alpha_3^S r_{j,k}^{n+1} + o_{j\,k}^{n+1} &= (1-\omega^*)(-\alpha_3^S r_{j,k}^n + o_{j,k}^n) + \omega^*(\alpha_2^S b_{j,k}^n + \alpha_1^S g_{j,k}^n), \\
b_{j,k}^{n+1} - \alpha_3^S g_{j,k}^{n+1} &= (1-\omega^*)(b_{j,k}^n - \alpha_3^S g_{j,k}^n) + \omega^*(\alpha_1^S r_{j,k}^{n+1} + \alpha_2^S o_{j,k}^{n+1}), \\
-\alpha_3^S b_{j,k}^{n+1} + g_{j,k}^{n+1} &= (1-\omega^*)(-\alpha_3^S b_{j,k}^n + g_{j,k}^n) + \omega^*(\alpha_2^S r_{j,k}^{n+1} + \alpha_1^S o_{j,k}^{n+1}),
\end{aligned} \tag{3.9}$$

where α_1^S, α_2^S, and α_3^S are space domain operators corresponding respectively to α_1, α_2, and α_3 respectively, i.e.,

$$\alpha_1^S = \frac{q_1}{q}(E_1 + E_1^{-1}), \qquad \alpha_2^S = \frac{q_2}{q}(E_2 + E_2^{-1}),$$

$$\alpha_3^S = \frac{q_3}{q}(E_1 + E_1^{-1})(E_2 + E_2^{-1}).$$

At the $(n+1)$th iteration, the values of the solution at the red and orange points and at the black and green points are coupled together as indicated by the left-hand side of (3.9).

The procedure defined by (3.9) is implicit in the sense that $r_{j,k}^{n+1}$ and $o_{j,k}^{n+1}$ are to be determined from the first pair of equations and $b_{j,k}^{n+1}$ and $g_{j,k}^{n+1}$ are to be determined from the second pair. We consider the following procedure.

Stage I. Solve the first pair of equations of (3.9) for $r_{j,k}^{n+1}$ and $o_{j,k}^{n+1}$ by using M steps of the SOR method using $\omega = \omega_p$.

Stage II. Solve the the second pair of equations of (3.9) for $b_{j,k}^{n+1}$ and $g_{j,k}^{n+1}$ by using M steps of the SOR method using $\omega = \omega_p$.

For convenience, we further decompose each stage into two steps. The formulas for Stage I are as follows.

Step 1. Compute $ff_{j,k}^{n+1} = (1-\omega_b)(u_{j,k}^n - \alpha_3^S u_{j,k}^n) + \omega_b(\alpha_1^S u_{j,k}^n + \alpha_2^S u_{j,k}^n - h^2 f_{j,k})$ at red and orange points.

Step 2. Perform iterations at red and orange points to compute $r_{j,k}^{n+1}$ and $o_{j,k}^{n+1}$ with $ff_{j,k}^{n+1}$ as driving function. Specifically, let $v_{j,k}^0 = u_{j,k}^n$, and for $m = 0, 1, 2, \cdots, M-1$ perform the following iterations:

at red points: $v_{j,k}^{m+1} = (1-\omega_p)v_{j,k}^m + \omega_p(\alpha_3^S v_{j,k}^m + ff_{j,k}^{n+1})$

at orange points: $v_{j,k}^{m+1} = (1-\omega_p)v_{j,k}^m + \omega_p(\alpha_3^S v_{j,k}^{m+1} + ff_{j,k}^{n+1})$

Finally, set $u_{j,k}^{n+1} = v_{j,k}^M$.

The formulas for Stage II are as follows.

Step 3. Compute $gg_{j,k}^{n+1} = (1-\omega_b)(u_{j,k}^n - \alpha_3^S u_{j,k}^n) + \omega_b(\alpha_1^S u_{j,k}^{n+1} + \alpha_2^S u_{j,k}^{n+1} - h^2 f_{j,k})$ at black and green points.

Step 4. Perform iterations at black and green points to compute $b_{j,k}^{n+1}$ and $g_{j,k}^{n+1}$ with $gg_{j,k}^{n+1}$ as driving function. Thus, let $v_{j,k}^0 = u_{j,k}^n$, and for $m = 0, 1, 2, \cdots, M-1$ perform the following iterations:

at black points: $v_{j,k}^{m+1} = (1-\omega_p)v_{j,k}^m + \omega_p(\alpha_3^S v_{j,k}^m + gg_{j,k}^{n+1})$

at green points: $v_{j,k}^{m+1} = (1-\omega_p)v_{j,k}^m + \omega_p(\alpha_3^S v_{j,k}^{m+1} + gg_{j,k}^{n+1})$.

Finally, set $u_{j,k}^{n+1} = v_{j,k}^M$.

Each time we use (3.9), we are carrying out a first level *block* (or *outer*) SOR iteration. Each iteration within either Stage I or II is a second level *point* (or *inner*) SOR iteration. Usually the number M of inner iterations in Steps 2 and 4 is two (see § 5). For the outer iteration defined by the right-hand side of (3.9), we use the block relaxation parameter $\omega^* = \omega_b$, and we use the point relaxation parameter ω_p for the inner iterations.

A data-flow diagram that illustrates how grid points exchange values with their neighboring points at each step of one two-level iteration is shown in Fig. 2. For simplicity, only one inner iteration is illustrated in this data-flow diagram.

It is a well-known result that both the block and point SOR iterations applied to a symmetric positive definite matrix converge if and only if their relaxation parameters are between zero and two [11]. Hence, the convergence of the two-level SOR scheme can be achieved by first selecting

$$0 < \omega_p < 2 \quad M \text{ sufficiently large,} \tag{3.10a}$$

where M denotes the total number of point SOR iterations performed at the second level, so that the point SOR iteration converges inside each block SOR iteration. Under condition (3.10a), a two-level SOR iteration is not different from a single-level block SOR iteration. Therefore, by imposing the additional constraint,

$$0 < \omega_b < 2, \tag{3.10b}$$

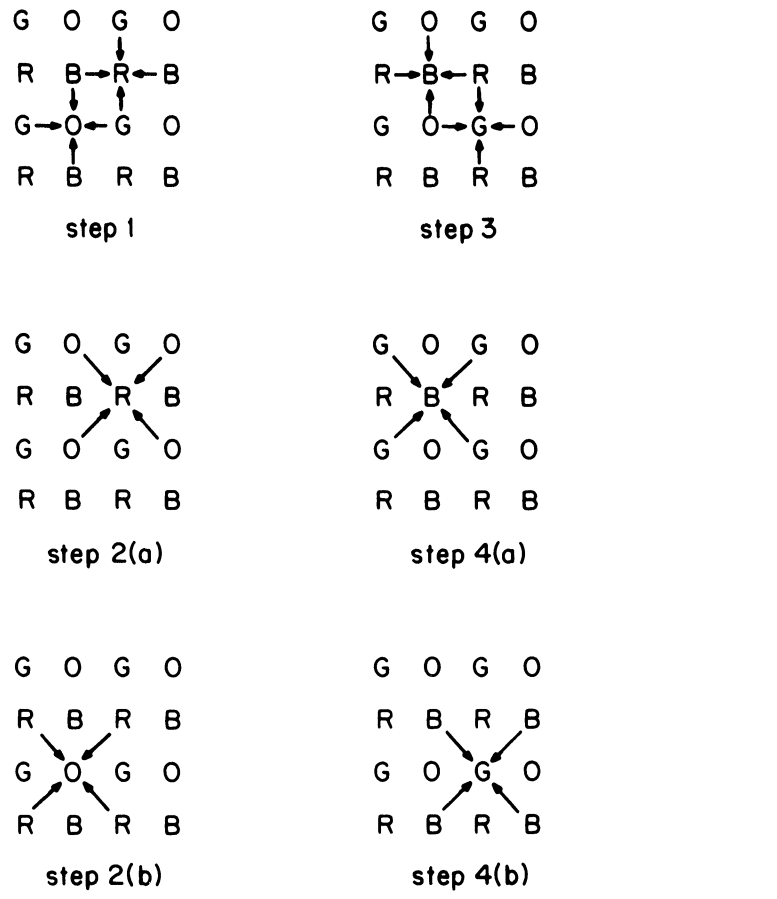

FIG. 2. *Data-flow diagram for a two-level four-color* SOR *method with computational order* {*red* → *orange* → *black* → *green*}. *Step* 1: *first half of a block* SOR *iteration. Step* 2(a) and (b): *one-point* SOR *iteration for red and orange points. Step* 3: *second half of a block* SOR *iteration. Step* 4(a) *and* (b): *one-point* SOR *iteration for black and green points.*

the two-level SOR method is guaranteed to converge. In § 5, we will discuss how to select the number M and optimal relaxation parameters ω_p^* and ω_b^* to maximize the convergence rate of the two-level SOR method.

3.4. Rederivation of Adams et al.'s nine-point SOR results. It is possible to derive the SOR results of Adams, LeVeque, and Young [2] directly from the frequency domain matrix equation (3.3). To do so, rewrite the coefficient matrix $A(\xi, \eta)$ as

$$A(\xi, \eta) = \tilde{D}(\xi, \eta) - \tilde{E}(\xi, \eta) - \tilde{F}(\xi, \eta)$$

where

$$\tilde{D}(\xi, \eta) = I = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad \tilde{E}(\xi, \eta) = \begin{bmatrix} 0 & \alpha_3 & \alpha_1 & \alpha_2 \\ 0 & 0 & \alpha_2 & \alpha_1 \\ 0 & 0 & 0 & \alpha_3 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

and $\tilde{F}(\xi, \eta) = \tilde{E}^T(\xi, \eta)$. In the frequency domain, we can then consider an SOR iteration of the form

$$\tilde{G}_\omega(\xi, \eta) = (I - \omega\tilde{F}(\xi, \eta))^{-1}\{(1-\omega)I + \omega\tilde{E}(\xi, \eta)\}. \tag{3.11}$$

In the space domain, (3.11) corresponds to Adams et al.'s SOR method with $R/O/B/G$ ordering, which can be written as

$$\begin{aligned}
r_{j,k}^{n+1} &= (1-\omega)r_{j,k}^n + \omega(\alpha_1^S b_{j,k}^n + \alpha_2^S g_{j,k}^n + \alpha_3^S o_{j,k}^n),\\
o_{j,k}^{n+1} &= (1-\omega)o_{j,k}^n + \omega(\alpha_1^S g_{j,k}^n + \alpha_2^S b_{j,k}^n + \alpha_3^S r_{j,k}^{n+1}),\\
b_{j,k}^{n+1} &= (1-\omega)b_{j,k}^n + \omega(\alpha_1^S r_{j,k}^{n+1} + \alpha_2^S o_{j,k}^{n+1} + \alpha_3^S g_{j,k}^n),\\
g_{j,k}^{n+1} &= (1-\omega)g_{j,k}^n + \omega(\alpha_1^S o_{j,k}^{n+1} + \alpha_2^S r_{j,k}^{n+1} + \alpha_3^S b_{j,k}^{n+1}).
\end{aligned} \tag{3.12}$$

If $\tilde{\lambda}_{\xi,\eta}$ is an eigenvalue of $\tilde{G}_\omega(\xi, \eta)$ we have

$$|\tilde{G}_\omega(\xi, \eta) - \tilde{\lambda}_{\xi,\eta} I| = 0, \tag{3.13}$$

which is a quartic equation of the variable $\tilde{\lambda}_{\xi,\eta}$. In [2], Adams et al. derived a quartic equation in terms of the variable $\gamma = (\tilde{\lambda}_{\xi,\eta})^{1/2}$ and showed that if

$$\gamma^4 + c_1\gamma^3 + c_2\gamma^2 + c_3\gamma + c_4 = 0, \tag{3.14a}$$

where c_i, $1 \leqq i \leqq 4$ are functions of ω, ξ, and η is the quartic equation for the frequency $(\xi\pi, \eta\pi)$, then

$$\gamma^4 - c_1\gamma^3 + c_2\gamma^2 - c_3\gamma + c_4 = 0 \tag{3.14b}$$

is the quartic equation for the frequency $(\xi\pi, (N-\eta)\pi)$. It turns out that the equation (3.13) obtained by our approach is equal to

$$(\gamma^4 + c_1\gamma^3 + c_2\gamma^2 + c_3\gamma + c_4)(\gamma^4 - c_1\gamma^3 + c_2\gamma^2 - c_3\gamma^2 - c_3\gamma + c_4) = 0.$$

In other words, (3.13) and (3.14) contain the same amount of information. From (3.13) or (3.14), the optimal relaxation parameter ω^* has to be selected so that the maximum value of $\rho[\tilde{G}_\omega(\xi, \eta)]$ is minimized over all ξ and η. For the details of this procedure, we refer to [2].

The major advantage of deriving SOR methods directly from the frequency domain coefficient matrix $A(\xi, \eta)$ is that this procedure does not require the knowledge of the eigenvectors of the SOR iteration matrices such as $G_\omega(\xi, \eta)$ in (3.6) and $\tilde{G}_\omega(\xi, \eta)$ in (3.11) for the determination of the optimal relaxation parameters and the corresponding spectral radii. We only have to know the eigenvectors associated with the scalar operators α_1^S, α_2^S, and α_3^S that describe the coupling between grid points of different colors. Consequently, the derivation is usually simpler. In addition, if the frequency domain coefficient matrix is block consistently ordered, the standard SOR theory can be applied separately at the block and point levels as shown above, and the determination of the optimal block and point relaxation parameters becomes straightforward.

However, our Fourier/Fourier approach has several limitations. Sometimes, eigenvectors provide valuable information for understanding the convergence property of an SOR iteration scheme. For example, for the SOR method (3.12), it was found that the eigenvector associated with the spectral radius is highly oscillatory. Therefore, the observed convergence rate for a test problem with a smooth initial error is faster than the predicted convergence rate [2]. Since the eigenvectors for the SOR iteration matrix cannot be found by our approach, this phenomenon cannot be appropriately explained. In addition, our approach does not apply to the SOR method with natural ordering.

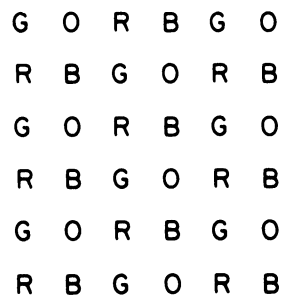

FIG. 3. *Another four-color partitioning scheme.*

Even for different coloring schemes such as the one shown in Figure 3, for which we have

$$\alpha_1^S = \frac{1}{q}[q_1 E_1 + q_3(E_1^{-1}E_2 + E_1^{-1}E_2^{-1})], \qquad \alpha_2^S = \frac{q_2}{q}(E_2 + E_2^{-1}),$$

$$\alpha_3^S = \frac{1}{q}[q_1 E_1^{-1} + q_3(E_1 E_2 + E_1 E_2^{-1})],$$

and where (3.1) can be rewritten in terms of this choice of α_1^S, α_2^S, and α_3^S, it is not clear whether a frequency domain equation exists corresponding to (3.3). The difficulty arises because $\sin(\xi\pi jh)\sin(\eta\pi kh)$ is no longer an eigenvector of the operators α_1^S and α_3^S. In fact, the results of our paper rely exclusively on the fact that α_1^S, α_2^S, and α_3^S admit $\sin(\xi\pi jh)\sin(\eta\pi kh)$ as a common set of eigenvectors, and this requirement is probably the most serious limitation of our approach. Note, however, that the coloring scheme of Fig. 3 is asymmetric and is therefore less natural than the one considered in this paper.

4. Degenerate case: Five-point stencils. In this section, we show that the traditional single-level two-color SOR method for a five-point stencil is in fact a degenerate case of the general two-level four-color SOR method. The following discussion also gives us more insight into the two-level SOR algorithm.

4.1. Standard five-point stencil. The standard five-point stencil discretization of the Poisson equation is

$$\frac{1}{h^2}(u_{j+1,k} + u_{j-1,k} + u_{j,k+1} + u_{j,k-1} - 4u_{j,k}) = f_{j,k},$$

which is a special case of (3.1) with

$$q_1 = 1, \quad q_2 = 1, \quad q_3 = 0, \quad q = 4.$$

Hence, we have

$$\alpha_1 = \frac{\cos(\xi\pi h)}{2}, \quad \alpha_2 = \frac{\cos(\eta\pi h)}{2}, \quad \alpha_3 = 0,$$

and

$$\alpha_1^S = \frac{E_1 + E_1^{-1}}{4}, \quad \alpha_2^S = \frac{E_2 + E_2^{-1}}{4}, \quad \alpha_3^S = 0.$$

For this case, we know that $\omega_p^* = 0$ from (2.8). It is easy to check that the second-level point SOR iteration becomes trivial and that only the first-level block SOR iteration is necessary, which is identical to the traditional red/black SOR method with the following optimal relaxation parameter

$$\omega^* = \omega_b^* = \frac{2}{1+[1-\cos^2(\pi h)]^{1/2}} \approx 2 - 2\pi h. \tag{4.1}$$

4.2. Rotated five-point stencil. Another five-point stencil discretization of the Poisson equation is [5]

$$\frac{1}{2h^2}(u_{j+1,k+1} + u_{j+1,k-1} + u_{j-1,k+1} + u_{j-1,k-1} - 4u_{j,k}) = f_{j,k},$$

which is also a special case of (3.1) with

$$q_1 = 0, \quad q_2 = 0, \quad q_3 = \tfrac{1}{2}, \quad q = 2.$$

Consequently, we find that

$$\alpha_1 = 0, \quad \alpha_2 = 0, \quad \alpha_3 = \cos(\xi\pi h)\cos(\eta\pi h),$$

and

$$\alpha_1^S = 0, \quad \alpha_2^S = 0, \quad \alpha_3^S = \frac{(E_1 + E_1^{-1})(E_2 + E_2^{-1})}{4}.$$

It turns out that $\omega_b^* = 1$, and in this case the first-level block SOR iteration becomes trivial. Only the second-level point SOR iteration is necessary, which can be written as

$$u_{j,k}^{n+1} = (1-\omega^*)u_{j,k}^n + \omega^*(\alpha_3^S u_{j,k}^n - \tfrac{1}{2}h^2 f_{j,k}) \qquad (j, k) \text{ red or black},$$

$$u_{j,k}^{n+1} = (1-\omega^*)u_{j,k}^n + \omega^*(\alpha_3^S u_{j,k}^{n+1} - \tfrac{1}{2}h^2 f_{j,k}) \qquad (j, k) \text{ orange or green},$$

where

$$\omega^* = \omega_p^* = \frac{2}{1+[1-\cos^4(\pi h)]^{1/2}} \approx 2 - 2\sqrt{2}\pi h. \tag{4.2}$$

By comparing (4.1) and (4.2), we find that the only difference between the standard and rotated five-point stencil discretizations is that the mesh size is h in the first case and $\sqrt{2}h$ in the second case. Therefore, the optimal relaxation parameter ω^* and spectral radius $\rho^* = \omega^* - 1$ have to be adjusted accordingly. Note, however, that the above observation depends on the isotropy of the Poisson equation, since the standard and rotated five-point stencils give rise to different discretizations in the anisotropic case.

5. Convergence rate analysis. In this section, we show how to select the optimal relaxation parameters ω_b^* and ω_p^* for the two-level four-color SOR method described in § 3, and we analyze the convergence rate of the resulting method when it is applied to (3.1) with nondegenerate coefficients, i.e., for

$$q_1 > 0, \quad q_2 > 0, \quad q_3 > 0.$$

5.1. Determination of optimal two-level relaxation parameters. First, let us concentrate on the second-level point iteration. In order to determine the optimal relaxation parameter, we need to find the spectral radius of the point Jacobi iteration, given by

$$\mu_{p,\max} = \max_{1 \le \xi, \eta \le N-1} |\alpha_3| = \frac{4q_3 \cos^2(\pi h)}{q} \approx \frac{4q_3}{q}(1 - \pi^2 h^2),$$

where the maximum value of $|\alpha_3|$ occurs for $(\xi, \eta) = (1, 1)$ and $(N-1, N-1)$. Since the spectral radius of the point Jacobi iteration is bounded by the constant $4q_3/q$, which is less than one, even a simple point Jacobi relaxation converges reasonably fast. Nevertheless, this can be further improved by a point SOR iteration using the following optimal relaxation parameter:

$$\omega_p^* = 2\Big/1+\left[1-\frac{(4q_3)^2\cos^4(\pi h)}{q^2}\right]^{1/2} \approx 1+\frac{1}{4}\left(\frac{4q_3}{q}\right)^2, \tag{5.1}$$

with spectral radius

$$\rho_p^* = \omega_p^* - 1 \approx \frac{1}{4}\left(\frac{4q_3}{q}\right)^2. \tag{5.2}$$

For a typical example, we have $q_3 = 1$ and $q = 20$ (see § 6) so that $\rho_p^* \approx 0.01$. Since the error can be damped approximately at the rate 10^{-2M}, where M is the number of second-level iterations, only two- or three-point SOR iterations inside each block SOR iteration are necessary. The fact that the second-level point SOR iteration requires only a constant number M of steps to converge, where M is usually two or three, plays a crucial role in our analysis of the convergence rate of the two-level SOR method. By using this observation, it will be shown below that the convergence rate of the two-level SOR scheme is similar to that of the standard SOR method for a five-point stencil, or of the nine-point SOR scheme discussed in [2].

Next, we examine the first-level block iteration. The spectral radius of the block Jacobi iteration matrix (3.5) is given by

$$\mu_{b,\max} = \max_{1\le\xi,\eta\le N-1}\left|\frac{\alpha_1+\alpha_2}{1-\alpha_3}\right| = \frac{2(q_1+q_2)\cos(\pi h)}{q-4q_3\cos^2(\pi h)},$$

which occurs at $(\xi, \eta) = (1, 1)$, $(1, N-1)$, $(N-1, 1)$, and $(N-1, N-1)$. By using the fact that $q = 2q_1 + 2q_2 + 4q_3$, we can simplify $\mu_{b,\max}$ as

$$\mu_{b,\max} = \frac{(q-4q_3)\cos(\pi h)}{q-4q_3\cos^2(\pi h)} \approx 1-\left(\frac{1}{2}+\frac{4q_3}{q-4q_3}\right)\pi^2h^2.$$

Hence, the optimal relaxation parameter for the block SOR iteration is

$$\omega_b^* = \frac{2}{1+(1-\mu_{b,\max}^2)^{1/2}} \approx 2-2\left(1+\frac{8q_3}{q-4q_3}\right)^{1/2}\pi h, \tag{5.3}$$

and the spectral radius is

$$\rho_b^* = \omega_b^* - 1 \approx 1-2\left(1+\frac{8q_3}{q-4q_3}\right)^{1/2}\pi h. \tag{5.4}$$

Therefore, if $q_3 = 1$ and $q = 20$, then $\rho_b^* = 1-\sqrt{6}\pi h$.

Since for a fixed point, the two-level SOR method divides neighboring points into two groups and operates on one group at the block iteration level and on the other group at the point iteration level, and since each block SOR iteration at the first level requires M point SOR iterations at the second level, it is convenient to define the *effective* number of iterations for one two-level SOR iteration as

$$n_{\text{eff}} \equiv \frac{w_pM + w_b}{w_b + w_p}, \tag{5.5}$$

where w_b and w_p represent the amount of work required per block and per point iteration respectively. The number n_{eff} measures approximately the computational burden of one full two-level SOR iteration in terms of equivalent nine-point Jacobi iterations.

If the point SOR iteration converges in M iterations, the convergence rate of the two-level SOR method is then only determined by that of the block SOR iteration. Therefore, we can define the *effective* spectral radius of the two-level SOR iteration as

$$\rho_{\text{eff}}^* \equiv (\rho_b^*)^{1/n_{\text{eff}}} = (\rho_b^*)^{(w_b+w_p)/(w_pM+w_b)}, \tag{5.6}$$

which is used to measure the average smoothing rate per effective iteration of the two-level SOR scheme.

For the above example, since the amount of computational work for each block and point SOR iteration is the same, we have $w_p = w_b$, so that

$$n_{\text{eff}} = \frac{M+1}{2}, \qquad \rho_{\text{eff}}^* \approx 1 - \frac{2}{M+1}\sqrt{6}\pi h.$$

When $M = 2$, we find therefore that

$$n_{\text{eff}} = \frac{3}{2} = 1.5, \qquad \rho_{\text{eff}}^* \approx 1 - 1.63\pi h. \tag{5.7}$$

The above effective spectral radius ρ_{eff}^* should be compared with the spectral radius $\rho_9^* \approx 1 - 1.79\pi h$ obtained for the nine-point SOR method discussed in [2]. In the next section, we will present a two-level SOR method with a different computationl ordering for which the effective spectral radius is $\rho_{\text{eff}}^* \approx 1 - 2.26\pi h$.

We see from the above comparison that the two-level SOR method and the nine-point SOR procedure of [2] have very similar convergence rates. The main difference is of course that the method of [2] is a single-level method that uses only one relaxation parameter ω^*. In addition, its convergence rate analysis requires the study of the solution of a quartic equation, and does not yield closed-form relations between ρ^*, ω^* and the spectral radius μ of the nine-point Jacobi iteration matrix. By comparison, the approach we used above to study the convergence of the two-level SOR method relies on the standard SOR theory, and provides closed-form relations between ρ_p^*, ω_p^*, and $\mu_{p,\max}$, and between ρ_b^*, ω_b^*, and $\mu_{b,\max}$.

Finally, note that the amount of work required by each effective iteration for the nine-point stencil case is about twice as large as for a standard five-point SOR iteration. Thus, to compare the convergence rate of the two-level SOR method with that of the standard five-point SOR scheme, we must compare ρ_{eff}^* with the spectral radius $(\rho_5^*)^2 \approx 1 - 4\pi h$ corresponding to two five-point SOR iterations. This comparison seems to indicate that the five-point SOR iteration converges faster than the two-level SOR method, or the nine-point SOR method discussed in [2]. However, the nine-point stencil discretization is more accurate than the corresponding five-point stencil discretization. Thus, for the same accuracy, we can select h larger for the nine-point stencil discretization so that in actuality the two-level or single-level nine-point SOR methods may converge faster than the standard five-point SOR method.

5.2. Computational order. In the above discussion, we have used a particular computational order, i.e., {red → orange → black → green}. Now, let us consider other computational orderings. Although there exist $4! = 24$ different ways to permute the computational order for these four colors, they only result in three different two-level SOR iteration schemes. By interchanging the relative positions of α_1, α_2, and α_3 in

the matrix $A(\xi, \eta)$, we can obtain only six different matrices, each of which corresponds to four different computational orderings. Furthermore, we can divide these six matrices into three classes:

$$\text{Class 1:} \quad \begin{bmatrix} 1 & -\alpha_1 & -\alpha_2 & -\alpha_3 \\ -\alpha_1 & 1 & -\alpha_3 & -\alpha_2 \\ -\alpha_2 & -\alpha_3 & 1 & -\alpha_1 \\ -\alpha_3 & -\alpha_2 & -\alpha_1 & 1 \end{bmatrix} \text{ and } \begin{bmatrix} 1 & -\alpha_1 & -\alpha_3 & -\alpha_2 \\ -\alpha_1 & 1 & -\alpha_2 & -\alpha_3 \\ -\alpha_3 & -\alpha_2 & 1 & -\alpha_1 \\ -\alpha_2 & -\alpha_3 & -\alpha_1 & 1 \end{bmatrix}.$$

$$\text{Class 2:} \quad \begin{bmatrix} 1 & -\alpha_2 & -\alpha_1 & -\alpha_3 \\ -\alpha_2 & 1 & -\alpha_3 & -\alpha_1 \\ -\alpha_1 & -\alpha_3 & 1 & -\alpha_2 \\ -\alpha_3 & -\alpha_1 & -\alpha_2 & 1 \end{bmatrix} \text{ and } \begin{bmatrix} 1 & -\alpha_2 & -\alpha_3 & -\alpha_1 \\ -\alpha_2 & 1 & -\alpha_1 & -\alpha_3 \\ -\alpha_3 & -\alpha_1 & 1 & -\alpha_2 \\ -\alpha_1 & -\alpha_3 & -\alpha_2 & 1 \end{bmatrix}.$$

$$\text{Class 3:} \quad \begin{bmatrix} 1 & -\alpha_3 & -\alpha_1 & -\alpha_2 \\ -\alpha_3 & 1 & -\alpha_2 & -\alpha_1 \\ -\alpha_1 & -\alpha_2 & 1 & -\alpha_3 \\ -\alpha_2 & -\alpha_1 & -\alpha_3 & 1 \end{bmatrix} \text{ and } \begin{bmatrix} 1 & -\alpha_3 & -\alpha_2 & -\alpha_1 \\ -\alpha_3 & 1 & -\alpha_1 & -\alpha_2 \\ -\alpha_2 & -\alpha_1 & 1 & -\alpha_3 \\ -\alpha_1 & -\alpha_2 & -\alpha_3 & 1 \end{bmatrix}.$$

It is easy to see that the same two-level SOR method applies to matrices within the same class. Although the discussion in § 5.1 applies only to Class 3 matrices, we can use a similar approach to obtain optimal block and point relaxation parameters and spectral radii for a two-level SOR method for matrices of Classes 1 and 2. For Class 1 matrices, we find

$$\omega_p^* \approx 1+\frac{1}{4}\left(\frac{2q_1}{q}\right)^2, \qquad \rho_p^* \approx \frac{1}{4}\left(\frac{2q_1}{q}\right)^2, \tag{5.8}$$

$$\omega_b^* \approx 2-2\left(\frac{q+4q_3}{q-2q_1}\right)^{1/2}\pi h, \qquad \rho_b^* \approx 1-2\left(\frac{q+4q_3}{q-2q_1}\right)^{1/2}\pi h \tag{5.9}$$

and for Class 2 matrices, we need only to replace q_1 by q_2 in the above expressions.

The data-flow diagram for the computational order {red → black → green → orange}, which corresponds to a two-level SOR method applied to Class 1 matrices, is shown in Fig. 4. Let us analyze the convergence rate for this two-level SOR iteration. From Fig. 4, it is easy to see that $w_p = \frac{1}{3}w_b$. Therefore, from (5.5) and (5.6), we have

$$n_{\text{eff}} = \frac{3+M}{4}, \qquad \rho_{\text{eff}}^* = (\rho_b^*)^{4/(3+M)}.$$

Consider now the typical example where $q_1 = q_2 = 4$ and $q_3 = 1$. By using (5.8) and (5.9), we find that the spectral radius of the point SOR iteration becomes larger, but the spectral radius of the block SOR iteration becomes smaller, i.e.,

$$\rho_p^* \approx 4 \times 10^{-2}, \qquad \rho_b^* \approx 1-\sqrt{8}\,\pi h.$$

Therefore, the effective spectral radius can be expressed as

$$\rho_{\text{eff}}^* \approx 1-\frac{4}{M+3}\sqrt{8}\,\pi h.$$

This gives

$$\rho_{\text{eff}}^* \approx 1-2.26\pi h \quad \text{if } M=2, \qquad \rho_{\text{eff}}^* \approx 1-1.89\pi h \quad \text{if } M=3. \tag{5.10}$$

By comparing (5.7) and (5.10), we observe that the performance of a two-level SOR iteration applied to Class 1 or Class 2 matrices is in fact better for this specific example.

6. Numerical examples. We consider the system of equations obtained from a nine-point stencil discretization of the Poisson equation, i.e.,

$$\frac{1}{6h^2}\{4(u_{j+1,k}+u_{j-1,k})+4(u_{j,k+1}+u_{j,k-1}) + (u_{j+1,k+1}+u_{j+1,k-1}+u_{j-1,k+1}+u_{j-1,k-1})-20u_{j,k}\}=f_{j,k}, \quad j, k=1, 2, \cdots, N-1, \tag{6.1}$$

with zero boundary conditions and $h=1/N=1/20$. In this case, $q_1=q_2=4$, and $q_3=1$. Since in this example the performance of the two-level SOR method for matrices $A(\xi, \eta)$ of Classes 1 and 2 is the same, we compare only the following two computational orders:

order (a): {red → orange → black → green},

order (b): {red → black → green → orange}.

The computational orders (a) and (b) are obtained by applying the two-level SOR iteration to matrices $A(\xi, \eta)$ belonging, respectively, to Classes 3 and 1. Their spectral radii and optimal relaxation parameters for the block SOR and point SOR iterations are summarized in Table 1.

TABLE 1

order	ω_b^*	ρ_b^*	ω_p^*	ρ_p^*
(a)	1.679931	0.679931	1.009702	0.009702
(b)	1.640105	0.640105	1.042400	0.042400

We use the following two test problems:

Example 1. The driving function is $e^{5x}[2x(x-1)+y(y-1)(25x^2-5x-8)]$ and the true solution is $e^{5x}x(x-1)y(y-1)$. In this case, the solution is a smooth function with a wideband two-dimensional Fourier spectrum concentrated in the region where ξ and η are small.

Example 2. The driving function is $-74\pi^2 \sin(5\pi x)\sin(7\pi y)$ and the true solution is $\sin(5\pi x)\sin(7\pi y)$. This corresponds to the case when the solution is a rapidly oscillatory function containing a single Fourier component at $(\xi, \eta)=(5, 7)$.

The computed results are shown in Figs. 5 and 6, where we plot the maximum error at each iteration as a function of the number of block SOR iterations. Each curve is parametrized by the number M of point SOR iterations we used. It is almost impossible to distinguish the curves with $M=2, 3, 4$ for computational order (a) in both examples. Hence, it is reasonable to choose $M=2$ in this case. When the computational order (b) is applied to the first example, where the solution contains low frequency components, the curve for $M=3$ is slightly better than for $M=2$. Nevertheless, the difference is very small. For the second example, the curves with $M=2, 3, 4$ are in fact not distinguishable. Thus, for computational order (b), it is still preferable to choose $M=2$, since fewer computations are required.

To demonstrate the convergence rate of the two-level SOR method, we choose another test problem with zero driving function and boundary conditions. This is in fact a homogeneous Laplace equation and its solution is zero. Two initial guesses are considered: (1) a smooth function, which is chosen to be $x(x-1)y(y-1)$; and (2) a

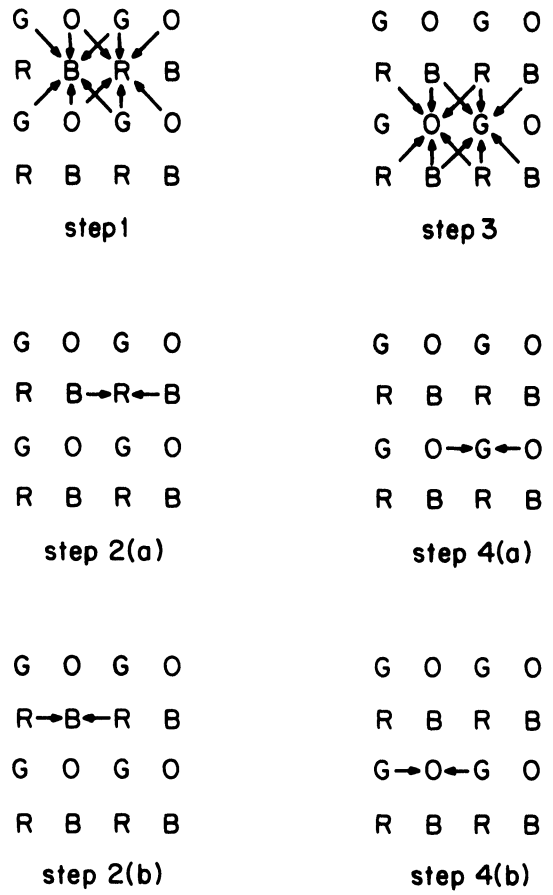

FIG. 4. *Data-flow diagram for a two-level four-color* SOR *method with computational order* {*red* → *black* → *green* → *orange*}. *Step* 1: *first half of a block* SOR *iteration. Step* 2(a) and (b): *one-point* SOR *iteration for red and black points. Step* 3: *second half of a block* SOR *iteration. Step* 4(a) *and* (b): *one-point* SOR *iteration for orange and green points.*

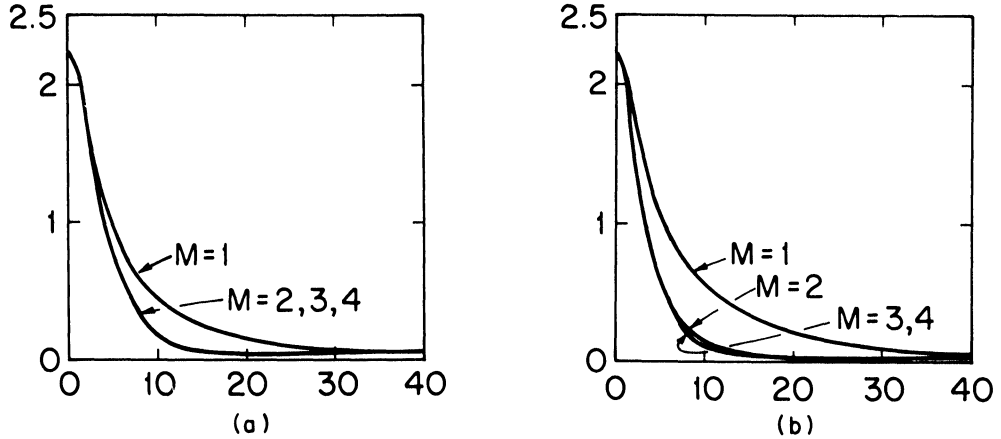

FIG. 5. *Computer simulation results for Example* 1 *with computational orders* (a) {*red* → *orange* → *black* → *green*}; *and* (b) {*red* → *black* → *green* → *orange*}. *The x-axis is the number of first-level block iterations and the y-axis is the maximum error at each iteration.*

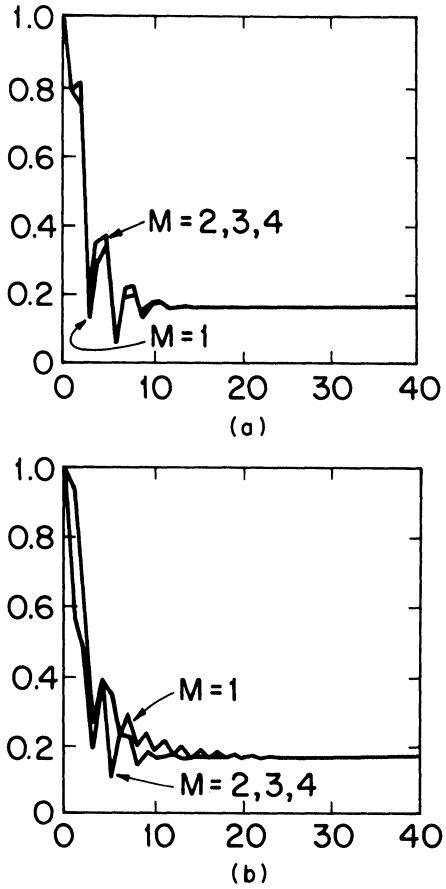

FIG. 6. *Computer simulation results for Example* 2 *with computational orders* (a) {*red* → *orange* → *black* → *green*}; *and* (b) {*red* → *black* → *green* → *orange*}. *The x-axis is the number of first-level block iterations and the y-axis is the maximum error at each iteration.*

random two-dimensional sequence. In Fig. 7, we plot the two-norm of the error versus the effective number (n_{eff}) of iterations for the above two computational orders and $M = 2$. The results show that the two-level SOR method with computational order (b) is better than that with order (a) and that the convergence rate of the two-level SOR method is not sensitive to the smoothness of the initial errors. Since the problem with initial guess $x(x-1)y(y-1)$ was also used to demonstrate the convergence rate of Adams et al.'s SOR method in [2], we are able to compare the convergence rates of our method with theirs for this test problem. It turns out that these two methods have very similar convergence rates.

7. Conclusions and generalizations. In this paper, we have transformed the system of equations for a discretized elliptic PDE from the space domain to the frequency domain in order to interpret the SOR method from a new viewpoint. This new formulation has helped us to design a two-level SOR method with optimal block and point relaxation parameters. The resulting two-level four-color SOR method for the nine-point stencil discretization of the Poisson equation was shown to be efficient with spectral radius $1 - C\pi h$, and numerical examples confirm our analysis.

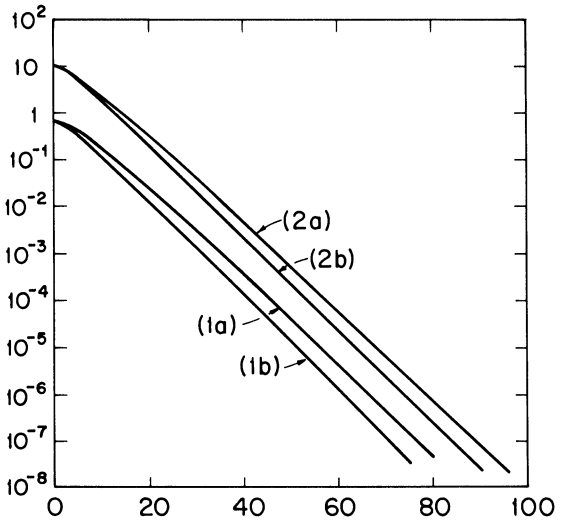

FIG. 7. *Convergence history (two-norm of the error versus the number of effective iterations) for computational orders* (a) $\{red \to orange \to black \to green\}$; *and* (b) $\{red \to black \to green \to orange\}$ *with* $M = 2$. *The driving function is zero and the initial values are* (1) $x(x-1)y(y-1)$; *and* (2) *a random sequence.*

The constant C of Adams et al.'s SOR method with various orderings and the line SOR method was compared in [2]. The results for the nine-point stencil discretization can be summarized as follows. The constant C ranges from 1.6 to 2.45 for Adams et al.'s method, $C = 1.63$ or 2.26 for the two-level method, and $C = 2.82$ for the line SOR method. In practice, when the initial error is smooth, the convergence rate of the two-level SOR method is similar to that of Adams et al.'s method. By comparing the constant C, we see that the line SOR method is slightly faster than both the two-level and Adams et al.'s methods. However, it should be emphasized that the line SOR method is less parallelizable since it needs a sequential direct method to solve tridiagonal matrix equations that describe the coupling between points of each line. Thus, from a parallel processing point of view, the two-level and Adams et al.'s SOR methods are more attractive.

The two-level SOR iteration method presented here can be generalized easily to higher-dimensional problems. A three-level eight-color SOR scheme can be described as follows. Consider a nondegenerate 27-point discretization of the three-dimensional Poisson equation. Suppose that each grid point is indexed by (j, k, l). We can label these points with eight colors depending on whether j, k, and l are even or odd. Following a procedure similar to the one used in § 3, we transform the discretized system from the space domain to the frequency domain so that in the frequency domain we obtain a discretization matrix which is block diagonal with 8×8 block matrices along the diagonal. Each of these blocks describes the coupling of the Fourier components of the eight colors at a fixed frequency. Since the discretization scheme is nondegenerate, each 8×8 matrix block is full. In order to apply the SOR method for each of these 8×8 matrices, we can block partition them into 4×4 submatrices. This results in a first-level block SOR iteration. However, the first-level block SOR iteration requires inverting 4×4 full matrices, which can be accomplished by performing several second-level block SOR and third-level point SOR iterations. Note that both the second-level block SOR and third-level point SOR iterations require a constant number

of steps to converge. The total number of iterations required by the above three-level SOR method, which is $O(1/h)$, is therefore determined primarily by the convergence rate of the first-level block SOR iteration.

There are many different possible computational orders for the above three-level SOR procedure. A typical one can be chosen as follows. At the first-level, we can distinguish two big blocks depending on whether $(j+k+l)$ is even or odd. At the second-level, within each big block, points are further divided into two smaller blocks according to whether $(j+k)$ is even or odd. Finally, at the third-level, each color can be separated from each other.

It is straightforward to generalize the above procedure to obtain an n-dimensional n-level 2^n-color SOR method. Here, we have considered the case where $n=2$.

Another generalization of interest would be to extend the two-level SOR iteration procedure described in this paper to PDEs with space-varying coefficients. It is natural in this context to combine the two-level SOR method discussed here with the local relaxation procedure developed in [4], [6], and [9]. The main idea of the local relaxation method can be roughly stated as follows. Each local finite difference equation is viewed as if it were homogeneous over the entire problem domain so that at each point a local relaxation parameter is determined on the basis of the local coefficients of the PDE and of the boundary conditions for the whole domain. Hence, a two-level local relaxation method would use the local coefficients and boundary conditions to choose optimal local block and point relaxation parameters at each grid point, so that different grid points would therefore have different block and point relaxation parameters.

Note that the Fourier/Fourier approach described in this paper depends heavily on the specific coloring and partitioning scheme we used. The relation existing between the single-level rowwise and multicolor SOR methods for the five-point stencil and the nine-point stencil cases can be explained by introducing a tilted grid [10], [2]. There does not seem to be an easy way to apply the tilted grid concept to obtain a two-level rowwise SOR method.

Acknowledgments. The authors thank Professor Lloyd N. Trefethen for his comments at various stages of this work. The authors are also grateful to the referees for several valuable suggestions, and in particular for pointing out the rederivation of Adams et al.'s SOR method presented in § 3.4.

REFERENCES

[1] L. ADAMS AND H. F. JORDAN, *Is* SOR *color-blind*? SIAM J. Sci. Statist. Comput., 7 (1986), pp. 490–506.

[2] L. ADAMS, R. J. LEVEQUE, AND D. M. YOUNG, *Analysis of the* SOR *iteration for the nine-point Laplacian*, SIAM J. Numer. Anal., 25 (1988), pp. 1156–1180.

[3] L. ADAMS AND J. M. ORTEGA, *A multi-color* SOR *method for parallel computation*, ICASE Report, 82-9, ICASE-NASA Langley Research Center, Hampton, VA, April 1982.

[4] E. F. BOTTA AND A. E. P. VELDMAN, *On local relaxation methods and their application to convection-diffusion equations*, J. Comput. Phys., 48 (1981), pp. 127–149.

[5] G. DAHLQUIST, A. BJORCK, AND N. ANDERSON, *Numerical Methods*, Prentice-Hall, Englewood Cliffs, NJ, 1974.

[6] L. W. ERHLICH, *The Ad-Hoc* SOR *Method: A Local Relaxation Scheme*, in Elliptic Problem Solvers II, Academic Press, New York, 1984, pp. 257–269.

[7] S. P. FRANKEL, *Convergence rates of iterative treatments of partial differential equations*, Math. Tables Aids Comput., 4 (1950), pp. 65–75.

[8] P. R. GARABEDIAN, *Estimation of the relaxation factor for small mesh size*, Math. Tables Aids Comput., 10 (1956), pp. 183–185.

[9] C.-C. J. KUO, B. C. LEVY, AND B. R. MUSICUS, *A local relaxation method for solving elliptic* PDEs *on mesh-connected arrays*, SIAM J. Sci. Statist. Comput., 8 (1987), pp. 530–573.

[10] R. J. LeVeque and L. N. Trefethen, *Fourier analysis of the* SOR *iteration*, Numerical Analysis Report 86-6, Department of Mathematics, Massachusetts Institute of Technology, Cambridge, MA; ICASE Report 86-93 ICASE-NASA Langley Research Center, Hampton, VA, September 1986.
[11] R. S. Varga, *Matrix Iterative Analysis*, Prentice-Hall, Englewood Cliffs, NJ, 1962.
[12] D. M. Young, *Iterative methods for solving partial differential equations of elliptic type*, Ph.D. thesis, Harvard University, Cambridge, MA, 1950.
[13] D. M. Young, *Iterative Solution of Large Linear Systems*, Academic Press, New York, 1971.