A new class of parallel alternating-type iterative methods

David M. Young*, David R. Kincaid

Center for Numerical Analysis, The University of Texas at Austin, Austin, TX 78713-8510 USA

Received 15 June 1995; revised 27 December 1995

Abstract

This paper is concerned with parallel alternating-type iterative methods for solving large sparse linear systems of the form $Au = b$ arising in the numerical solution of partial differential equations by finite difference methods. Examples of alternating-type methods include the alternating direction implicit (ADI) method and the unsymmetric SOR (USSOR) method. Each iteration of an alternating-type method involves the use of two parameters, say $\rho$ and $\rho'$. We consider parallel alternating-type methods where, given an initial vector $u^{(0)}$, the positive integer $m$, and two sets of $m$ parameters $\{\rho_i\}$ and $\{\rho'_i\}$, one carries out $m^2$ single iterations in parallel, each involving one pair $(\rho_i, \rho'_i)$ of the parameters. It is shown that in some cases a linear combination $v^*$ of the vectors thus obtained is the same as the vector $v^{**}$ which would be obtained by a sequential process involving $m$ iterations based on the successive use of the parameter pairs $(\rho_1, \rho'_1), (\rho_2, \rho'_2), \ldots, (\rho_m, \rho'_m)$. Thus, the parallel procedure offers the potential of reducing the wall-clock time by a factor of $m$ as compared with the sequential procedure. Preliminary numerical results based on the use of a virtual parallel system of sequential computers confirm the expected reductions in the number of iterations.

Keywords: Interactive methods; Alternating-type; Parallel computing; Alternating direction; Implicit method; Unsymmetric successive overrelaxation method

AMS classification: 65F10

1. Introduction

In this paper, we are concerned with alternating-type iterative methods for solving large sparse linear systems of the form

$$Au = b,$$

where $A$ is a nonsingular matrix. To construct an alternating-type method for solving (1.1), we choose matrices $H, V,$ and $\Sigma$ such that

$$A = H + V$$

* Corresponding author. E-mail: young@cs.utexas.edu.

0377-0427/96/$15.00 © 1996 Published by Elsevier Science B.V. All rights reserved

PH S0377-0427(96)0030-1
and such that $\Sigma$ is a diagonal matrix with positive diagonal elements. We assume that $H + \rho \Sigma$ and $V + \rho \Sigma$ are nonsingular for any positive real number $\rho$ and that for any vector $w$, one can easily solve, for $v$, any linear system of the form $(H + \rho \Sigma)v = w$ or $(V + \rho \Sigma)v = w$.

To define an alternating-type iterative method, we choose positive numbers $\rho$ and $\rho'$ and, for a given $u^{(n)}$, we determine $u^{(n+1/2)}$ and $u^{(n+1)}$ by

\[
(H + \rho \Sigma)u^{(n+1/2)} = b - (V - \rho \Sigma)u^{(n)},
\]

\[
(V + \rho' \Sigma)u^{(n+1)} = b - (H - \rho' \Sigma)u^{(n+1/2)}.
\]

Thus,

\[
u^{(n+1)} = T_{\rho, \rho'}u^{(n)} + k_{\rho, \rho'},
\]

where

\[
T_{\rho, \rho'} = (V + \rho' \Sigma)^{-1}(H - \rho' \Sigma)(H + \rho \Sigma)^{-1}(V - \rho \Sigma)
\]

\[
= I - (\rho + \rho')(V + \rho' \Sigma)^{-1}\Sigma(H + \rho \Sigma)^{-1}(H + V)
\]

(1.5)

and

\[
k_{\rho, \rho'} = (\rho + \rho')(V + \rho' \Sigma)^{-1}\Sigma(H + \rho \Sigma)^{-1}b
\]

\[
= (I - T_{\rho, \rho'})A^{-1}b.
\]

(1.6)

Examples of alternating-type methods are the alternating-direction implicit method (ADI method) [5], the symmetric successive overrelaxation method (SSOR method) [6], and the unsymmetric successive overrelaxation method (USSOR method) [4, 10]. With the ADI method, $H$ and $V$ are either tridiagonal or are permutationally similar to tridiagonal matrices and $\Sigma = I$, the identity matrix. With the SSOR and USSOR methods, $H$ and $V$ are lower triangular and upper triangular matrices, respectively, and $\Sigma$ is a diagonal matrix with positive diagonal elements.

We are concerned with nonstationary alternating-type methods where the parameters $\rho$ and $\rho'$ vary from iteration to iteration. Thus, given $u^{(0)}$, $m$, and the parameters $\rho_1, \rho'_1, \rho_2, \rho'_2, \ldots, \rho_m, \rho'_m$, we can construct the vectors $u^{(1)}$, $u^{(2)}$, ..., $u^{(m)}$ by the sequential procedure

\[
u^{(1)} = T_{\rho_1, \rho'_1}u^{(0)} + k_{\rho_1, \rho'_1},
\]

\[
u^{(2)} = T_{\rho_2, \rho'_2}u^{(1)} + k_{\rho_2, \rho'_2},
\]

\[
\vdots
\]

\[
u^{(m)} = T_{\rho_m, \rho'_m}u^{(m-1)} + k_{\rho_m, \rho'_m}.
\]

(1.7)

We seek to determine the parameters $\{\rho_i\}$ and $\{\rho'_i\}$ so that $u^{(m)}$ will be as close to the true solution $u$ of (1.1) as possible. In practice, we seek to make the spectral radius $S(\prod_{i=1}^{m} T_{\rho_i, \rho'_i})$ of $\prod_{i=1}^{m} T_{\rho_i, \rho'_i}$ as small as possible.

As an alternative to the (sequential) nonstationary method defined by (1.7), we will consider the parallel alternating-type method defined for any set of $m^2$ coefficients $\{\alpha_{i,j}\}$ for $i, j = 1, 2, \ldots, m$, by

\[
\tilde{u}^{(m)} = \left(1 - \sum_{i,j=1}^{m} \alpha_{i,j}\right)u^{(0)} + \sum_{i,j=1}^{m} \alpha_{i,j}u^{(1)}_{\rho_i, \rho'_j},
\]

(1.8)
where

\[
[u^{(1)}]_{p_i,p_j} = T_{p_i,p_j}u^{(0)} + k_{p_i,p_j}, \quad i,j = 1,2,...,m.
\]  

We note that the parallel procedure defined by (1.8) and (1.9) represents an iterative method which is consistent in the sense defined in [11], since if \(u^{(0)} = \bar{u}\), where \(\bar{u} = A^{-1}b\) is the true solution of (1.1), then \(\bar{u}^{(m)} = \bar{u}\).

We plan to attempt to identify classes of problems where the parallel procedure is equivalent, or nearly equivalent, to a related sequential procedure. The objective is to reduce the wall-clock time. We are not attempting to obtain speedup, in the usual sense, or scalability. As far as speedup is concerned, we could hope to achieve a factor of \(m\) at best with the use of \(m^2\) processors; hence, the speedup ratio (whose theoretical optimum value is 1.0) would be only \(m^{-1}\) at best.

In Section 2, we show that in the case where \(H, V, \) and \(\Sigma\) are mutually commutative, that is, where

\[
HV = VH, \quad H\Sigma = \Sigma H, \quad V\Sigma = \Sigma V
\]  

then, for suitable \(\{\alpha_{i,j}\}\), we have \(\bar{u}^{(m)} = u^{(m)}\) and the parallel procedure agrees with the sequential procedure. Explicit formulas for the \(\{\alpha_{i,j}\}\) are given in terms of the \(\{\rho_i\}\) and \(\{\rho'_i\}\). A generalized partial fraction representation of a rational function of two real variables is used.

In Section 3, we consider the ADI method. First, we review known results on the choice of the \(\{\rho_i\}\) and \(\{\rho'_i\}\) for the case where \(H\) and \(V\) commute and are symmetric and positive definite (SPD) and where the eigenvalues of \(H\) and \(V\) lie in known intervals on the real line. Good values of the \(\{\rho_i\}\) and \(\{\rho'_i\}\) are given along with the corresponding convergence factors for model problems involving Poisson’s equation on the rectangle. The results of some preliminary numerical experiments based on the use of the parallel and the sequential procedures for model problems are given in Section 4. Good agreement was obtained in a number of cases between the actual and the expected reductions in the number of iterations required for convergence for the parallel procedures as compared with the sequential procedures.

In Section 5, we consider the solution of discrete periodic problems in two dimensions based on the use of the implicit SSOR method and the implicit USSOR method. These methods are alternating-type methods and are slightly modified versions of the SSOR and the USSOR methods, respectively, see [2] and [12]. We show that for any given sequential procedure based on the implicit SSOR method, one can construct an equivalent parallel procedure based on the the implicit USSOR method.

2. Sequential nonstationary and parallel alternating-type methods

In this section, we seek to derive the coefficients \(\{\alpha_{i,j}\}\) of (1.8) so that the parallel alternating-type method (1.8)-(1.9) will be equivalent to the sequential nonstationary method defined by (1.7).

First, for the alternating-type method (1.3), we have

\[
u^{(n+1)} - \bar{u} = T_{p,\rho'}(u^{(n)} - \bar{u}),
\]  

where
where
\[ \bar{u} = A^{-1}b \]  
(2.2)
is the true solution of (1.1). This follows from (1.3) since
\[ (H + \rho \Sigma)\bar{u} = b - (V - \rho \Sigma)\bar{u}, \]
\[ (V + \rho' \Sigma)\bar{u} = b - (H - \rho' \Sigma)\bar{u}. \]  
(2.3)
It then follows that, for \( u^{(m)} \) defined by the nonstationary method (1.7), we have
\[ u^{(m)} - \bar{u} = \left( \prod_{s=1}^{m} T_{\rho_s \rho'_s} \right) (u^{(0)} - \bar{u}) \]  
(2.4)
and, hence,
\[ u^{(m)} = \left( \prod_{s=1}^{m} T_{\rho_s \rho'_s} \right) u^{(0)} + \left( I - \prod_{s=1}^{m} T_{\rho_s \rho'_s} \right) \bar{u}. \]  
(2.5)
We now seek to determine a parallel representation of \( \prod_{s=1}^{m} T_{\rho_s \rho'_s} \). By (1.4), we have
\[ \prod_{s=1}^{m} T_{\rho_s \rho'_s} = \prod_{s=1}^{m} (V + \rho'_s \Sigma)^{-1}(H - \rho'_s \Sigma)(H + \rho \Sigma)^{-1}(V - \rho \Sigma) \]
\[ = \prod_{s=1}^{m} (\tilde{V} + \rho'_s \Sigma)(\tilde{H} - \rho'_s \Sigma)(\tilde{H} + \rho \Sigma)^{-1}(\tilde{V} - \rho \Sigma) \]  
(2.6)
where
\[ \tilde{H} = \Sigma^{-1} H, \quad \tilde{V} = \Sigma^{-1} V. \]  
(2.7)
If we assume that \( H, V, \) and \( \Sigma \) are mutually commutative, then \( \tilde{H} \) and \( \tilde{V} \) commute. We are thus led to consider the following rational function of the real variables \( x \) and \( y \):
\[ \prod_{s=1}^{m} t_{\rho_s \rho'_s} = \prod_{s=1}^{m} \frac{(x - \rho'_s)(y - \rho_s)}{(x + \rho_s)(y + \rho'_s)}, \]  
(2.8)
where
\[ t_{\rho_s \rho'_s} = \frac{(x - \rho'_s)(y - \rho_s)}{(x + \rho_s)(y + \rho'_s)} = 1 - \frac{(\rho_s + \rho'_s)(x + y)}{(x + \rho_s)(y + \rho'_s)}. \]  
(2.9)

**Theorem 2.1.** Let \( m \) be a positive integer and let \( \rho_1, \rho_2, \ldots, \rho_m \) and \( \rho'_1, \rho'_2, \ldots, \rho'_m \) be two sets of distinct positive numbers. Then for all \( x \) and \( y \), we have
\[ \prod_{s=1}^{m} t_{\rho_s \rho'_s} = 1 + \sum_{i,j=1}^{m} \alpha_{i,j} \left( t_{\rho_s \rho'_s} - 1 \right), \]  
(2.10)
where

\[
\alpha_{i,j} = \frac{\prod_{s \neq j}^{m} (-\rho_i - \rho_s) \prod_{s \neq i}^{m} (-\rho_j - \rho_s)}{\prod_{s \neq i}^{m} (-\rho_i + \rho_s) \prod_{s \neq j}^{m} (-\rho_j + \rho_s)}. \tag{2.11}
\]

We will give a proof for the case \(m = 2\). (A proof for the general case is given in \([3]\).) If \(m = 2\), the relations (2.10) and (2.11) become

\[
t_{p_1, \rho'_1} t_{p_2, \rho'_2} = 1 + \alpha_{1,1} \left( t_{p_1, \rho'_1} - 1 \right) + \alpha_{1,2} \left( t_{p_1, \rho'_1} - 1 \right) + \alpha_{2,1} \left( t_{p_2, \rho'_2} - 1 \right) + \alpha_{2,2} \left( t_{p_2, \rho'_2} - 1 \right), \tag{2.12}
\]

where

\[
\alpha_{1,1} = \alpha_{2,2} = \frac{(\rho_1 + \rho'_1)(\rho'_1 + \rho_2)}{(\rho_1 - \rho_2)(\rho'_1 - \rho'_2)},
\]

\[
\alpha_{1,2} = \alpha_{2,1} = - \frac{(\rho_1 + \rho'_1)(\rho_2 + \rho'_2)}{(\rho_1 - \rho_2)(\rho'_1 - \rho'_2)}. \tag{2.13}
\]

We remark that the results (2.12) and (2.13) were previously given in \([12]\). However, the method used in \([12]\) does not appear to generalize easily to the case \(m > 2\). On the other hand, the proof given in \([3]\) is based on a straightforward inductive argument.

Note that from (2.9), we have

\[
1 - t_{p_1, \rho'_1} = \frac{(\rho_1 + \rho'_1)(x + y)}{(x + \rho_1)(y + \rho'_1)},
\]

\[
1 - t_{p_1, \rho'_2} = \frac{(\rho_1 + \rho'_2)(x + y)}{(x + \rho_1)(y + \rho'_2)},
\]

\[
1 - t_{p_2, \rho'_1} = \frac{(\rho_2 + \rho'_1)(x + y)}{(x + \rho_2)(y + \rho'_1)},
\]

\[
1 - t_{p_2, \rho'_2} = \frac{(\rho_2 + \rho'_2)(x + y)}{(x + \rho_2)(y + \rho'_2)}. \tag{2.14}
\]

Therefore,

\[
t_{p_1, \rho'_1} t_{p_2, \rho'_2} = \left[ 1 - (1 - t_{p_1, \rho'_1}) \right] \left[ 1 - (1 - t_{p_2, \rho'_2}) \right]
= 1 - \left\{ (1 - t_{p_1, \rho'_1}) + (1 - t_{p_2, \rho'_2}) - (1 - t_{p_1, \rho'_1})(1 - t_{p_2, \rho'_2}) \right\} \tag{2.15}
\]

and

\[
t_{p_1, \rho'_1} t_{p_2, \rho'_2} - 1 = \left( t_{p_1, \rho'_1} - 1 \right) + \left( t_{p_2, \rho'_2} - 1 \right) + \left( t_{p_1, \rho'_1} - 1 \right) \left( t_{p_2, \rho'_2} - 1 \right)
= \left( t_{p_1, \rho'_1} - 1 \right) + \left( t_{p_2, \rho'_2} - 1 \right) + (\rho_1 + \rho'_1)(\rho_2 + \rho'_2)(x + y)\Delta_2. \tag{2.16}
\]
and, where

\[
\Delta_2 = \frac{(x + y)}{(x + \rho_1)(y + \rho'_1)(x + \rho_2)(y + \rho'_2)}
\]

\[
= \left[ \frac{x}{(x + \rho_1)(x + \rho_2)} \right] \left[ \frac{1}{(y + \rho'_1)(y + \rho'_2)} \right] + \left[ \frac{1}{(x + \rho_1)(x + \rho_2)} \right] \left[ \frac{y}{(y + \rho'_1)(y + \rho'_2)} \right].
\]

(2.17)

Since,

\[
\frac{x}{(x + \rho_1)(x + \rho_2)} = \left( \frac{-\rho_1}{\rho_2 - \rho_1} \right) \left( \frac{1}{x + \rho_1} \right) + \left( \frac{\rho_2}{\rho_2 - \rho_1} \right) \left( \frac{1}{x + \rho_2} \right),
\]

\[
\frac{1}{(y + \rho'_1)(y + \rho'_2)} = \left( \frac{1}{\rho'_2 - \rho'_1} \right) \left( \frac{1}{y + \rho'_1} \right) + \left( \frac{-1}{\rho'_2 - \rho'_1} \right) \left( \frac{1}{y + \rho'_2} \right),
\]

\[
\frac{1}{(x + \rho_1)(x + \rho_2)} = \left( \frac{1}{\rho_2 - \rho_1} \right) \left( \frac{1}{x + \rho_1} \right) + \left( \frac{-1}{\rho_2 - \rho_1} \right) \left( \frac{1}{x + \rho_2} \right),
\]

\[
\frac{y}{(y + \rho'_1)(y + \rho'_2)} = \left( \frac{-\rho'_1}{\rho'_2 - \rho'_1} \right) \left( \frac{1}{y + \rho'_1} \right) + \left( \frac{\rho'_2}{\rho'_2 - \rho'_1} \right) \left( \frac{1}{y + \rho'_2} \right),
\]

(2.18)

it follows that

\[
\Delta_2 = \left[ \frac{-\rho_1 - \rho'_1}{(\rho_2 - \rho_1)(\rho'_2 - \rho'_1)} \right] \left[ \frac{1}{(x + \rho_1)(y + \rho'_1)} \right] + \left[ \frac{\rho_1 + \rho'_2}{(\rho_2 - \rho_1)(\rho'_2 - \rho'_1)} \right] \left[ \frac{1}{(x + \rho_1)(x + \rho_2)} \right] + \left[ \frac{\rho_2 + \rho'_1}{(\rho_2 - \rho_1)(\rho'_2 - \rho'_1)} \right] \left[ \frac{1}{(x + \rho_2)(y + \rho'_1)} \right] + \left[ \frac{-\rho_2 - \rho'_2}{(\rho_2 - \rho_1)(\rho'_2 - \rho'_1)} \right] \left[ \frac{1}{(x + \rho_2)(y + \rho'_2)} \right].
\]

(2.19)

Therefore, by (2.14), we have

\[
(x + y)\Delta_2 = \left( \frac{1}{(\rho_2 - \rho_1)(\rho'_2 - \rho'_1)} \right) (t_{\rho_1, \rho'_1} - 1)
\]
\[ + \left( \frac{-1}{(\rho_2 - \rho_1)(\rho'_2 - \rho'_1)} \right) (t_{p_i, p'_i} - 1) \]
\[ + \left( \frac{-1}{(\rho_2 - \rho_1)(\rho'_2 - \rho'_1)} \right) (t_{p_i, p'_i} - 1) \]
\[ + \left( \frac{1}{(\rho_2 - \rho_1)(\rho'_2 - \rho'_1)} \right) (t_{p_i, p'_i} - 1) \]  

(2.20)

Evidently, (2.12) follows from (2.16), (2.17), and (2.20).

A proof of Theorem 2.1 for the general case can be carried out using mathematical induction. We remark that first one should show that (2.10) holds for some \( \{\alpha_{i,j}\} \). Then, given \( i \) and \( j \), one can determine \( \alpha_{i,j} \) by multiplying both sides of (2.10) by \((t_{p_i, p'_i})^{-1}\) and letting \( x = -\rho_i \) and \( y = -\rho'_j \).

Since, \( H, V, \) and \( \Sigma \) are mutually commutative, it follows from Theorem 2.1 that

\[ \prod_{s=1}^{m} T_{\rho_s, \rho'_s} - I = \sum_{i,j=1} \alpha_{i,j} \left( T_{\rho_i, \rho'_i} - I \right), \]

(2.21)

where the \( \{\alpha_{i,j}\} \) are given by (2.11).

**Theorem 2.2.** If \( H, V, \) and \( \Sigma \) are mutually commutative and if \( u^{(m)} \) and \( \tilde{u}^{(m)} \) are determined by (1.7) and (1.8), respectively, where the \( \{\alpha_{i,j}\} \) are given by (2.11), then

\[ \tilde{u}^{(m)} = u^{(m)}. \]  

(2.22)

To prove Theorem 2.2, first we note that by (1.9) and (1.8) we have

\[ \tilde{u}^{(m)} = \left( 1 - \sum_{i,j=0}^{m} \alpha_{i,j} \right) u^{(0)} + \sum_{i,j=1}^{m} \alpha_{i,j} \left[ u^{(1)} \right]_{p_i, p'_j} \]
\[ = \left( 1 - \sum_{i,j=0}^{m} \alpha_{i,j} \right) u^{(0)} + \sum_{i,j=1}^{m} \alpha_{i,j} \left\{ T_{p_i, p'_j}u^{(0)} + k_{p_i, p'_j} \right\}, \]

(2.23)

where, by (1.6),

\[ k_{p_i, p'_j} = (I - T_{p_i, p'_j}) \tilde{u}. \]

(2.24)

By (2.5), (2.21), (1.5), and (1.9), we have

\[ u^{(m)} = \left( \prod_{s=1}^{m} T_{\rho_s, \rho'_s} \right) u^{(0)} + \left( I - \prod_{s=1}^{m} T_{\rho_s, \rho'_s} \right) \tilde{u} \]
\[ = \left( I + \sum_{i,j=1}^{m} \alpha_{i,j} \left( T_{p_i, p'_i} - I \right) \right) u^{(0)} + \sum_{i,j=1}^{m} \alpha_{i,j} \left( I - T_{p_i, p'_i} \right) \tilde{u} \].
$$\begin{align*}
&(1 - \sum_{i,j=0}^{m} \alpha_{i,j}) \mathbf{u}(0) + \sum_{i,j=1}^{m} \alpha_{i,j} \left( T_{\rho,\rho'} \mathbf{u}(0) + k_{\rho,\rho'} \right) \\
&= \left( 1 - \sum_{i,j=0}^{m} \alpha_{i,j} \right) \mathbf{u}(0) + \sum_{i,j=1}^{m} \alpha_{i,j} \left[ \mathbf{u}^{(1)} \right]_{\rho,\rho'}.
\end{align*}$$

Thus, (2.22) follows.

3. ADI method

We now assume that $A = H + V$, where $H$ and $V$ are symmetric and positive-definite (SPD) matrices which are either tridiagonal or permutationally similar to tridiagonal matrices. We will also assume that $H$ and $V$ commute, that is,

$$HV = VH.$$  

We assume that the eigenvalues $\mu$ of $H$ and $\nu$ of $V$ lie in the ranges

$$0 < \alpha \leq \mu \leq \beta,$$

$$0 < \alpha \leq \nu \leq \beta.$$  

Given $m$, we seek to determine parameters $\rho = (\rho_1, \rho_2, \ldots, \rho_m)$ and $\rho' = (\rho'_1, \rho'_2, \ldots, \rho'_m)$ to minimize

$$S(\rho, \rho') = \max_{0 < \alpha \leq \nu < \beta, \mu < \nu} \prod_{s=1}^{m} \left( \frac{\mu - \rho'_s (v - \rho_s)}{(\mu + \rho_s) (v + \rho'_s)} \right)^2.$$  

The problem of determining the $\{ \rho_i \}$ and the $\{ \rho'_i \}$ to minimize $S(\rho, \rho')$ can be solved in terms of elliptic functions; see [7, 8]. However, for simplicity, we will consider the use of good parameters which, though not optimum, yield nearly optimum values of $S(\rho, \rho')$. Thus for the case where $a = \alpha$ and $b = \beta$, we consider the parameters

$$\rho_i = \rho'_i = b \left( \frac{a}{b} \right)^{(2i-1)/(2m)}, \quad i = 1, 2, \ldots, m.$$  

The corresponding value of $S(\rho, \rho')$ satisfies

$$S(\rho, \rho') \leq \left( \frac{1 - (a/b)^{1/(2m)}}{1 + (a/b)^{1/(2m)}} \right)^2.$$  

The optimum value of $m$, in the sense of minimizing $S^{1/m}$, is the smallest $m$ satisfying

$$\left( \sqrt{2} - 1 \right)^{2m} \leq \frac{a}{b}.$$  

We remark that if one uses the truly optimum parameters for each \( m \), then the average convergence factor decreases monotonically as \( m \) increases.

For the more general case, where the ranges of the eigenvalues of \( H \) and of \( V \) may be different, we consider the following good parameters [11]:

\[
\rho_i = \frac{-p + q\hat{\rho}_i}{1 - s\hat{\rho}_i}, \quad \rho_i' = \frac{p + q\hat{\rho}_i'}{1 + s\hat{\rho}_i'}, \quad i = 1, 2, \ldots, m, \tag{3.7}
\]

where

\[
\hat{\rho}_i = \hat{\rho}_i' = e^{(2i-1)/(2m)}, \quad i = 1, 2, \ldots, m, \tag{3.8}
\]

and

\[
c = \left[ 1 + \theta + \sqrt{\theta(2 + \theta)} \right]^{-1},
\]

\[
p = \frac{1}{2}[(b - \beta) + (b + \beta)s],
\]

\[
q = \frac{1}{2}[(b + \beta) + (b - \beta)s], \tag{3.9}
\]

\[
s = \frac{(\beta - \alpha) - (b - a)}{(b + \beta) - (a + \alpha)c}.
\]

The corresponding value of \( S(p, p') \) satisfies

\[
S(p, p') \leq \left( \frac{1 - e^{1/(2m)}}{1 + e^{1/(2m)}} \right)^2. \tag{3.10}
\]

The optimum value of \( m \) in the sense of minimizing \( S^{1/m} \) is the smallest \( m \) satisfying

\[
\left( \sqrt{2} - 1 \right)^{2m} \leq c. \tag{3.11}
\]

Let us now consider a boundary-value problem involving the elliptic equation

\[
\frac{\partial}{\partial x}(A(x, y)u_x) + \frac{\partial}{\partial y}(C(x, y)u_y) = G(x, y) \tag{3.12}
\]

in the rectangle \( 0 \leq x \leq L_x, \ 0 \leq y \leq L_y \) with values of \( u(x, y) \) prescribed on the boundary. It is assumed that \( A(x, y) \) and \( C(x, y) \) are positive in the rectangle. By standard finite difference methods we can obtain a linear system of the form

\[
(H + V)u = b, \tag{3.13}
\]

where the matrices \( H \) and \( V \) are SPD and correspond to the operators \( (\partial/\partial x)(A(x, y)u_x) \) and \( (\partial/\partial y)(C(x, y)u_y) \), respectively. Moreover, if \( A(x, y) \) and \( C(x, y) \) are separable, that is, if we have \( A(x, y) = E(x)F(y) \) and \( C(x, y) = E^*(x)F^*(y) \), then the matrices \( H \) and \( V \) commute [1].
Table 1
Model problem $h = \frac{1}{40}$

<table>
<thead>
<tr>
<th>$m$</th>
<th>$m^2$</th>
<th>No. iterations sequential</th>
<th>No. iterations parallel</th>
<th>Ratio seq./par.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>72</td>
<td>72</td>
<td>1.0</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>28</td>
<td>14</td>
<td>2.0</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td>23</td>
<td>8</td>
<td>2.9</td>
</tr>
<tr>
<td>4</td>
<td>16</td>
<td>20</td>
<td>5</td>
<td>4.0</td>
</tr>
<tr>
<td>5</td>
<td>25</td>
<td>20</td>
<td>4</td>
<td>5.0</td>
</tr>
<tr>
<td>6</td>
<td>36</td>
<td>18</td>
<td>3</td>
<td>6.0</td>
</tr>
<tr>
<td>7</td>
<td>49</td>
<td>21</td>
<td>3</td>
<td>7.0</td>
</tr>
<tr>
<td>8</td>
<td>64</td>
<td>22</td>
<td>3</td>
<td>7.3</td>
</tr>
<tr>
<td>9</td>
<td>81</td>
<td>18</td>
<td>2</td>
<td>9.0</td>
</tr>
<tr>
<td>10</td>
<td>100</td>
<td>20</td>
<td>2</td>
<td>10.0</td>
</tr>
<tr>
<td>11</td>
<td>121</td>
<td>21</td>
<td>2</td>
<td>10.5</td>
</tr>
<tr>
<td>12</td>
<td>144</td>
<td>23</td>
<td>2</td>
<td>11.5</td>
</tr>
<tr>
<td>13</td>
<td>169</td>
<td>24</td>
<td>2</td>
<td>12.0</td>
</tr>
<tr>
<td>14</td>
<td>196</td>
<td>25</td>
<td>2</td>
<td>12.5</td>
</tr>
<tr>
<td>15</td>
<td>225</td>
<td>24</td>
<td>2</td>
<td>12.0</td>
</tr>
<tr>
<td>16</td>
<td>256</td>
<td>16</td>
<td>1</td>
<td>16.0</td>
</tr>
<tr>
<td>17</td>
<td>289</td>
<td>17</td>
<td>1</td>
<td>17.0</td>
</tr>
<tr>
<td>18</td>
<td>324</td>
<td>18</td>
<td>1</td>
<td>18.0</td>
</tr>
<tr>
<td>19</td>
<td>360</td>
<td>19</td>
<td>1</td>
<td>19.0</td>
</tr>
<tr>
<td>20</td>
<td>400</td>
<td>20</td>
<td>1</td>
<td>20.0</td>
</tr>
</tbody>
</table>

For a rectangle of length $L_x$ and height $L_y$, we have

\[
\begin{align*}
a &= 4 \sin^2 \frac{\pi h}{2L_x}, \\
b &= 4 \cos^2 \frac{\pi h}{2L_x}, \\
\alpha &= 4 \sin^2 \frac{\pi h}{2L_y}, \\
\beta &= 4 \cos^2 \frac{\pi h}{2L_y}.
\end{align*}
\] (3.14)

4. Preliminary numerical results

We considered a model problem involving the differential equation

\[-u_{xx} - u_{yy} = -1\] (4.1)
on the unit square $0 \leq x \leq 1, 0 \leq y \leq 1$ with $u = 0$ on the boundary. The grid size $h = \frac{1}{40}$ was used. The numbers of iterations needed for convergence are given in Table 1. These results are for the parallel ADI method using parallel execution on a system of parallel virtual machines (PVM on a collection of Sun-workstations). The good ADI parameters which were used correspond to cases where the eigenvalues of $H$ and $V$ lie in the same range.

For the parallel implementation if we let $m = 16$ then we get convergence in one cycle. This involves carrying out 256 single iterations to obtain a factor of improvement of 16, in terms of
numbers of iterations, as compared with the sequential procedure with \( m = 16 \). If we let \( m = 9 \), then we get convergence in two cycles. This involves carrying out 81 iterations to obtain a factor of improvement of 9.

The accuracy obtained by the use of the parallel procedure is quite remarkable in view of the fact that the \( \{ z_{i,j} \} \) are very large in absolute value and frequently change sign.

5. Nonstationary implicit SSOR and parallel implicit USSOR methods

In this section, we consider the nonstationary SSOR and the parallel USSOR methods as applied to a class of discrete periodic problems in two dimensions involving Poisson's equation. For such problems, the implicit SSOR and the implicit USSOR methods are alternating-type methods of the form (1.3) such that the commutativity condition (1.10) holds. Therefore, the parallel implicit USSOR method generates the same iteration vectors as the nonstationary implicit SSOR method.

Let us consider a discrete periodic problem involving the Poisson equation

\[
-u_{xx} - u_{yy} = f(x, y),
\]

with periods of length one in each coordinate direction and with grid size \( h = 1/M \), for some integer \( M \) (see [2, 13]). We assume that the function \( f(x, y) \) is periodic with period one in both \( x \) and \( y \) and that the sum of the values of \( f(x, y) \), taken over all of the grid points \( (x, y) \) such that \( x, y = 0, \pm h, \pm 2h, \ldots \) and \( 0 \leq x < 1, 0 \leq y < 1 \) vanishes. We require that \( u(x, y) \) be periodic in \( x \) and \( y \) with period one on the grid and that \( u(x, y) \) satisfies the difference equation

\[
4u(x, y) - u(x + h, y) - u(x - h, y)
- u(x, y + h) - u(x, y - h) = h^2 f(x, y)
\]

at each grid point.

It can be shown that the discrete periodic problem thus defined can be reduced to the problem of solving a linear system of the form (1.1) with \( M^2 \) unknowns (see [2, 9, 13]). The unknowns \( u_i \) correspond to values of \( u(x, y) \) at the \( M^2 \) grid points \( (x, y) \) such that \( 0 \leq x < 1 \) and \( 0 \leq y < 1 \). Moreover, the matrix \( A \) is singular and the null space of \( A \) is of dimension one and is spanned by the vector \( v \), where

\[
v = (1, 1, \ldots, 1)^T.
\]

We seek to obtain a least-squares solution of (1.1). To do this, first we *purify* the right-hand side \( b \) by replacing \( b \) with \( b^* \), where

\[
b^* = b - \frac{(b, v)}{(v, v)} v.
\]

Evidently, the modified system

\[
Av = b^*
\]

is solvable, though in general not uniquely solvable. We seek the solution \( u^* \) for which \((u^*, u^*)^{1/2}\) is minimum.
To define the implicit USSOR method for solving (1.1), we first represent the matrix $A$ in the form

$$A = D - C_L - C_U,$$

where

$$D = 4I.$$  \hfill (5.7)

For the ordinary, nonimplicit, USSOR method $C_L$ and $C_U$ are strictly lower triangular and strictly upper triangular matrices, respectively. For the implicit USSOR method, following Chan and Elman [2], we choose $C_L$ and $C_U$ such that $C_L$ and $C_U$ are not, in general, lower triangular and upper triangular, respectively, but such that $C_L$ and $C_U$ commute. The implicit USSOR method with relaxation factors $\omega$ and $\omega'$, where $0 < \omega, \omega' < 2$, is defined by

$$
\left( \frac{1}{\omega} D - C_L \right) u^{(n+1/2)} = b + \left[ \left( \frac{\omega - 1}{\omega} \right) + C_U \right] u^{(n)},
$$

$$
\left( \frac{1}{\omega'} D - C_U \right) u^{(n+1)} = b + \left[ \left( \frac{\omega' - 1}{\omega'} \right) + C_L \right] u^{(n+1/2)}.
$$

where $b^*$ is given by (5.4). Thus, the implicit USSOR method is an alternating-type method of the form (1.3) with

$$H = \frac{1}{2} D - C_L,$$

$$V = \frac{1}{2} D - C_U,$$

$$\Sigma = D,$$

$$\rho = \frac{2 - \omega}{2\omega},$$

$$\rho' = \frac{2 - \omega'}{2\omega'}.$$  \hfill (5.9)

To iterate with the implicit USSOR method, one should choose an arbitrary vector $u^{(0)}$ and then purify $u^{(0)}$ to obtain $u^{*0}$. It can be shown that, in the absence of rounding error, all subsequent iterations based on the implicit USSOR method will produce purified vectors. However, because of rounding errors, it is advisable to purify the iteration vectors, $u^{(n)}$, from time to time.

It can be shown that the matrices $H, V$ and $\Sigma$ are mutually commutative. (Details will be given in a later paper along with a discussion of various procedures for implementing the implicit USSOR method.) Consequently, there is a parallel implicit SSOR method which is based on the use of the implicit USSOR method.

We remark that Chan and Elman [2] considered several implicit iterative methods including the implicit Gauss–Seidel, SOR, and SSOR methods. They did not recommend the use of these methods for actual computation. Their objective in studying the methods was to determine their convergence factors and to use these to obtain estimates of the convergence factors of the corresponding nonimplicit methods for certain problems with Dirichlet boundary conditions.
In determining the convergence factors for several implicit methods Chan and Elman used the fact that each of the matrices involved has a complete set of eigenvectors, which are given by

\[ \psi^{(s,t)}(x, y) = e^{2\pi i s} e^{2\pi i t y}, \quad s, t = 0, 1, 2, \ldots . \]  

(5.10)

Since for discrete periodic problems, the eigenvectors of the implicit SSOR method do not depend on \( \omega \) or \( \omega' \), we can obtain formulas for the eigenvalues of the nonstationary implicit SSOR method where \( \omega \) and \( \omega' \) vary from iteration to iteration. This is in contrast to the situation for discrete Dirichlet problems where the eigenvectors of the matrices corresponding to the SSOR method with different \( \omega \) and \( \omega' \) may vary from iteration to iteration. Thus, in general, we cannot obtain analytic expressions for the eigenvalues of the product iteration matrix for discrete Dirichlet problems.

We remark that given an elliptic problem involving Poisson’s equation on a rectangular region with Dirichlet boundary conditions, one can construct a discrete periodic problem whose periods are twice the lengths of the corresponding sides of the rectangle (see [9, 13]).

We plan to apply the procedure described in Section 2 to the nonstationary implicit SSOR method. The first step is to choose \( m \) and a set of \( m \) values of \( \omega \). For the ADI method it is an advantage to use different values of the parameters \( \rho \) and \( \rho' \). Thus, one can obtain a smaller average convergence factor using several different values of \( \rho \) and \( \rho' \) than if one uses a single pair. On the other hand, for the implicit SSOR method the average convergence factor is minimized, or nearly minimized, by using a fixed value of \( \omega \). Consequently, when one needs to choose several different values of \( \omega \) in order to obtain parallelism, one can only hope not to increase the average convergence factor too much. In trying to choose \( m \) different values of \( \omega \), if one chooses values too different from the optimum one might slow the convergence. On the other hand, by choosing values of \( \omega \) too close to the optimum value, one can expect to obtain very large values of the \( \{x_{i,j}\} \) and this could lead to numerical instability.

Fortunately, in contrast to the situation for the SOR method for problems with Dirichlet boundary conditions where the convergence factor is very sensitive to \( \omega \) for \( \omega \) near the optimum value, the convergence factor for the SSOR method for discrete problems and for implicit SSOR method for discrete problems is a relatively slowly varying function of \( \omega \) for \( \omega \) near its optimum value. Therefore, we expect to be able to choose sets of values that will be close enough to the optimum value to give near-optimum convergence and yet will be far enough away from each other so that the \( \{x_{i,j}\} \) will not be excessively large.

6. Summary, conclusions, and future work

In this paper, we have described some novel parallel alternating-type iterative methods that are designed to converge more rapidly, in terms of number of iterations, than certain nonstationary alternating-type iterative methods that are usually carried out sequentially.

Preliminary numerical experiments show that the parallel procedures perform very well, in terms of number of iterations, as compared to the corresponding sequential procedures.

We plan to test and compare the parallel and the sequential procedures on a massively parallel computer for a variety of problems involving cases where commutativity properties hold as well as for cases for which they do not.
Acknowledgements

We wish to thank Wan Chen for obtaining the numerical results. Also, we wish to thank Karen Baker for her help. In addition, the comments and suggestions of the referee were most helpful.

This work was done at The University of Texas at Austin and was supported, in part, by the Department of Energy under grant DE-FG03-93ER25183, by Cray Research, Inc., under grant LTR DTD, by the State of Texas Higher Education Coordinating Board, under grant TARP-212 of the Texas Advanced Research Program, and by the National Science Foundation under grant CCR-9504954.

References