SIAM J. SCI. COMPUT.
Vol. 19, No. 2, pp. 341–363, March 1998

002

ADI METHODS FOR CUBIC SPLINE COLLOCATION DISCRETIZATIONS OF ELLIPTIC PDEs*

P. TSOMPANOPOULOU† AND E. VAVALIS†

Abstract. This paper presents the formulation, analysis, and implementation of alternating direction implicit (ADI) methods for solving the linear system of algebraic equations that arise from the discretization of multidimensional linear elliptic partial differential equations (PDEs). The theoretical analysis is carried out for a broad class of PDE problems. Numerical experiments confirm the theoretically determined characteristics of the ADI iterative schemes. The computational effectiveness of the proposed schemes is shown through a detailed theoretical complexity analysis confirmed with our experimental data.

Key words. alternating direction implicit methods, collocation methods, elliptic partial differential equations

AMS subject classifications. 65N35, 65N05, 65F10

PII. S1064827595281794

1. Introduction.

Collocation methods based on Hermite splines ([16], [26], [1], [19], [20]) or cubic splines ([28], [1], [18], [14]) have been proved lately to be very important and powerful discretization tools for the solution of elliptic partial differential equations (PDEs). It has been seen that both classes of methods can achieve optimal order of convergence and increased computational efficiency. A series of papers ([9], [10], [11], [2], [3]) have been devoted recently to the analysis, implementation, and performance evaluation of alternating direction implicit (ADI) methods applied to linear algebraic systems which arise from Hermite cubic collocation discretizations of elliptic PDEs in two and three dimensions. This study is the first attempt to propose, analyze, and implement ADI schemes for cubic spline collocation discretizations. Specifically, we formulate alternating direction implicit spline collocation (ADISC) methods based on cubic spline piecewise polynomials for approximating the solution u of the second-order elliptic PDE

$$Lu \equiv \sum_{i=1}^{k} \alpha_i \frac{\partial^2 u}{\partial x_i^2} + \gamma u = f \quad \text{in} \quad \Omega, \tag{1a}$$

subject to Dirichlet boundary conditions, where $\Omega \equiv \prod_{i=1}^{k} \otimes [a_i, b_i]$ is a rectangular domain in $\mathbb{R}^k$ and $\alpha_i (< 0)$, $\gamma (\geq 0)$, f, and g are functions of k variables. Although the formulation and the implementation of the proposed ADISC schemes are given for the above general PDE problem, we carry out most of the convergence analysis only for the three-dimensional Helmholtz problem, with Dirichlet boundary conditions and constant coefficients, that is,

$$Lu \equiv \sum_{i=1}^{3} \frac{\partial^2 u}{\partial x_i^2} + \gamma u = f \qquad \text{in} \quad \Omega, \tag{1b}$$

*Received by the editors February 17, 1995; accepted for publication (in revised form) May 14, 1996.

http://www.siam.org/journals/sisc/19-2/28179.html

†Department of Mathematics, University of Crete, 714 09 Heraklio, Greece, and Institute of Applied and Computational Mathematics, FORTH, 711 10 Heraklio, Greece (giwta@math.uch.gr, mav@math.uch.gr).

$$u = g \qquad \text{on} \quad \partial\Omega. \tag{2b}$$

The two-dimensional ADISC case can be treated similarly and it will not be presented here.

The rest of this paper is organized as follows. The derivation of the ADI schemes associated with the linear algebraic system of equations which arises from cubic spline collocation discretization schemes is given in section 2 where the convergence analysis and the complexity analysis of the proposed ADISC methods are also presented. The results of our extensive numerical experiments are given in section 3. Section 4 contains a summary of our results and some concluding remarks.

2. ADISC methods.

2.1. Cubic spline collocation methods. For a detailed formulation and analysis of the cubic spline collocation method in two or more dimensions the reader is referred to [18] and [29]. Next, we briefly describe how to derive the associated linear system of algebraic equations.

We start by discretizing uniformly each interval $[a_i, b_i], i = 1, \ldots, k$, to obtain the extended uniform partition of $[a_i, b_i]$

$$\Delta_i \equiv \left\{ \tau_\ell^i = a_i + \ell h_i \ ; \ \ell = -1, \ldots, N_i + 1 \text{ with } h_i = \frac{b_i - a_i}{N_i} \right\}.$$

Then $\Delta \equiv \prod_{i=1}^{k} \otimes \Delta_i$ is the induced uniform partition of the domain Ω. We denote by $S_{3,\Delta_i} \equiv \mathbf{P}_{3,\Delta_i} \cap C^2([a_i, b_i])$ the space of the one-dimensional splines defined by the partition Δ_i of $[a_i, b_i]$. The basis elements of the space of the k-dimensional splines $S_{3,\Delta}$ are obtained by taking the tensor product of the basis B_ℓ^i of the one-dimensional splines S_{3,Δ_i} (see [18]). The cubic spline collocation approximate $u_\Delta \in S_{3,\Delta}$ can then be represented as

$$u_\Delta(\mathbf{x}) = \sum_{\ell_1=-1}^{N_1+1} \sum_{\ell_2=-1}^{N_2+1} \cdots \sum_{\ell_k=-1}^{N_k+1} U_{\ell_1 \ell_2 \cdots \ell_k} B_{\ell_1}^1(x_1) B_{\ell_2}^2(x_2) \cdots B_{\ell_k}^k(x_k), \tag{3}$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_k)$ is a point in Δ and where $U_{\ell_1 \ell_2 \cdots \ell_k}$ (with $\ell_i = -1, \ldots, N_i + 1, i = 1, \ldots, k$) are the unknown spline collocation coefficients. In order to determine these $\prod_{i=1}^{k}(N_i + 3)$ unknowns we require u_Δ to satisfy the PDE (1) at all points in Δ and the boundary conditions (2b) at all boundary points $\Delta \cap \partial\Omega$. Using well-known results from the spline-interpolation theory ([22]) it can be easily seen that the solution u of the PDE problem satisfies the collocation equations within an error of order $O(h^2)$. In order to obtain an optimal ($O(h^4)$) spline approximation u_Δ of u we force it to satisfy the perturbed PDE $L'u = f$ where the operator L' is a perturbation of the operator L in (1) and can be obtained by replacing, for $i = 1, \ldots, k$,

$$\left.\frac{\partial^2 u}{\partial x_i^2}\right|_{\mathbf{x}} \quad \text{by} \quad \frac{1}{12}\left[\left.\frac{\partial^2 u}{\partial x_i^2}\right|_{\mathbf{x}^-} + 10 \left.\frac{\partial^2 u}{\partial x_i^2}\right|_{\mathbf{x}} + \left.\frac{\partial^2 u}{\partial x_i^2}\right|_{\mathbf{x}^+} \right],$$

where $\mathbf{x}^-, \mathbf{x}$, and $\mathbf{x}^+$ are grid points in Δ consecutive in the i-direction.

For both $O(h^2)$ and $O(h^4)$ collocation schemes for the PDE problem defined by (1a), the collocation equations obtained from the boundary conditions and from the differential equation at the end points of each line can be uncoupled. The reader is referred to [18] and [29] to check how such an uncoupling can be achieved. The

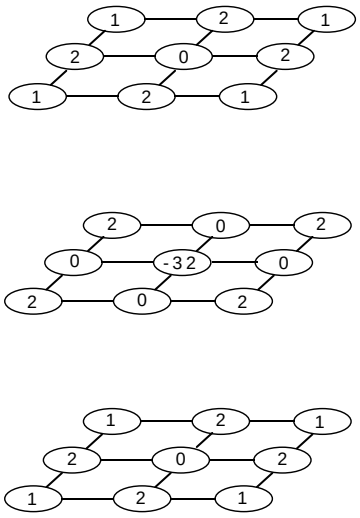

FIG. 1. *The three-dimensional coefficient stencil of the $O(h^2)$ collocation equations at points away from the boundary.*

collocation equations *away* ($2 \le \ell \le N_i - 2$) from the boundary can be presented, for the three-dimensional Poisson equation, in the form of stencils. The stencils associated with the $O(h^2)$ scheme and with the $O(h^4)$ scheme are given in Figures 1 and 2, respectively. The value of each entry in these stencils is the coefficient of the corresponding unknown. All entries in the $O(h^2)$ stencil have been multiplied by a factor of $-\frac{1}{12h^2}$, and all entries in the $O(h^4)$ stencil have been multiplied by a factor of $-\frac{1}{432h^2}$. For lines next to the boundary, the equations have similar form with appropriately modified right sides (see [14]).

From the above discussion, the assumed representation (3) of u_Δ, and the nature of the B-spline basis functions, we conclude ([18], [29]) that the *interior* collocation equations can be written in the form

$$\sum_{i=1}^{k} A_i \mathbf{U} = \mathbf{F}, \quad A_i \in \mathbb{R}^{\mathcal{K}\times\mathcal{K}}, \tag{4}$$

where $\mathcal{K} = \prod_{i=1}^{k}(N_i - 1)$ and where (for the Poisson equation)

$$A_i \equiv \left(\prod_{j=k}^{i+1} \otimes T_4^j\right) \otimes \frac{1}{6^{k-1}h_i^2}\mathcal{E}^i \otimes \left(\prod_{j=i-1}^{1} \otimes T_4^j\right), i = 1, \ldots, k, \tag{5}$$

$$\text{with } \mathcal{E}^i = \begin{cases} T_{-2}^i & \text{second-order scheme,} \\ \frac{1}{12}T_{10}^i T_{-2}^i & \text{fourth-order scheme,} \end{cases} \quad T_\alpha^i = \text{ tridiag}(1, \alpha, 1)$$

and $T_\alpha^i \in \mathbb{R}^{(N_i-1)\times(N_i-1)}$. The right-hand-side vector $\mathbf{F}$ holds the associated values of the right-hand-side function f of the PDE and contains the effect of the elimination of the "boundary" unknowns.

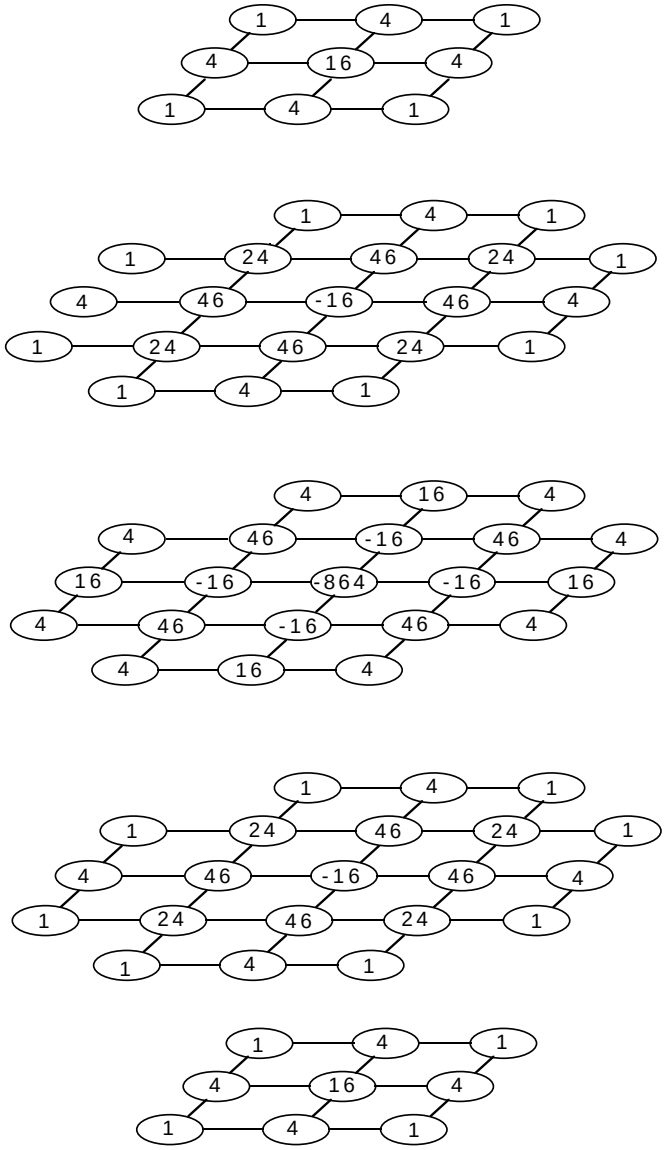

FIG. 2. *The three-dimensional coefficient stencil of the $O(h^4)$ collocation equations at points away from the boundary.*

2.2. Derivation. The collocation linear system of coefficients (4) can be solved by several direct and iterative solvers (see [18], and [17] for the two-dimensional case, and [14] for the k-dimensional one). Our first ADISC method[1] associated with the linear system (4), obtained by generalizing the approach found in [7], can be described by the following recurrence relation:

[1]For a detailed introduction to the theory of tensor products (Kronecker products) of matrices, the first usage of this theory for the analysis of PDEs, and the efficient computer manipulation of tensor products, the reader is referred to [15], [23], and [4], respectively.

Given a guess $\mathbf{U}^{(0)}$, iterate for $s = 0, 1, \ldots$

$$(A_1 + r_{s+1}D)\,\mathbf{U}^{(s+1/k)} = \left[(A_1 + r_{s+1}D) - \omega \sum_{i=1}^{k} A_i\right] \mathbf{U}^{(s)} + \omega \mathbf{F}, \tag{6}$$

$$(A_j + r_{s+1}D)\,\mathbf{U}^{(s+j/k)} = A_j \mathbf{U}^{(s+(j-1)/k)} + r_{s+1} D \mathbf{U}^{(s)},\ j = 2, \ldots, k, \tag{7}$$

where $D \equiv \frac{1}{6^k} \prod_{j=k}^{1} \otimes T_4^j$, $r_{s+1}, s = 0, 1, \ldots$, are positive parameters to accelerate the convergence and ω is a relaxation parameter. It should be pointed out that for $\omega = 1$ our scheme reduces to the so-called Douglas–Rachford scheme [8] and for $\omega = 2$ it reduces to the Douglas scheme [7]. The above ADISC scheme can be written in the following matrix form:

$$\mathbf{U}^{(s+1)} = M_{r_{s+1},\omega} \mathbf{U}^{(s)} + r_{s+1}^{k-1} \omega \prod_{i=k}^{1} \left(\frac{1}{r_{s+1}} D^{-1} A_i + I\right)^{-1} \mathbf{F}, \tag{8}$$

where the iteration matrix $M_{r_{s+1},\omega}$ is given by

$$M_{r_{s+1},\omega} = I - \omega \frac{1}{r_{s+1}} \prod_{i=k}^{1} \left(\frac{1}{r_{s+1}} D^{-1} A_i + I\right)^{-1} \left(\sum_{i=1}^{k} D^{-1} A_i\right). \tag{9}$$

In an effort to increase the per iteration efficiency of the above-described ADISC scheme we express all the tridiagonal matrices T_α^i involved in terms of the associated identity matrix I and the tridiagonal matrix T_{-2}^i. Substituting these expressions in (4), the $O(h^2)$ interior collocation equations are now given (for the Poisson equation) in the form

$$\left[\sum_{i=1}^{k} X_i + \sum_{i=2}^{k} \sum_{j=1}^{\binom{k}{i}} H_{ij}\right] \mathbf{U} = \mathbf{F} \tag{10}$$

with

$$X_i = -\frac{1}{h^2} \left(\prod_{j=k}^{i+1} \otimes I\right) \otimes T_{-2}^i \otimes \left(\prod_{j=i-1}^{1} \otimes I\right) \quad \text{and} \quad H_{ij} = -\frac{i}{6^{i-1} h^2} C_{ij},$$

where C_{ij} is a tensor product of k matrices, where i of them are T_{-2}^i and the remaining $k - i$ of them are identity matrices. The above form of the linear system naturally leads us to our second ADISC scheme given by the following recurrence relation:

$$(r_{s+1}I + X_1)\,\mathbf{U}^{(s+1/k)} = (r_{s+1}I + X_1)\,\mathbf{U}^{(s)} - \omega \left(\sum_{i=1}^{k} X_i + \sum_{i=2}^{k} \sum_{j=1}^{\binom{k}{i}} H_{ij}\right) \mathbf{U}^{(s)} + \omega \mathbf{F}, \tag{11}$$

$$(r_{s+1}I + X_i)\,\mathbf{U}^{(s+i/k)} = r_{s+1} \mathbf{U}^{(s+(i-1)/k)} + X_i \mathbf{U}^{(s)}\ ,\ i = 2, \ldots, k, \tag{12}$$

which in matrix form is given by

$$\mathbf{U}^{(s+1)} = M_{r_{s+1},\omega} \mathbf{U}^{(s)} + r_{s+1}^{k-1} \omega \left(\prod_{i=k}^{1} (r_{s+1}I + X_i)^{-1}\right) \mathbf{F} \tag{13}$$

and where the associated iteration matrix is

$$M_{r_{s+1},\omega} = I - \omega \left(\prod_{i=k}^{1} \left(I + \frac{1}{r_{s+1}} X_i \right)^{-1} \right) \left(\sum_{i=1}^{k} \frac{1}{r_{s+1}} X_i + \sum_{i=2}^{k} \sum_{j=1}^{\binom{k}{i}} \frac{1}{r_{s+1}} H_{ij} \right). \tag{14}$$

The new scheme is closer to the ADI scheme associated with the five-point star discretization method as far as matrix inversions are concerned [23]. It has been observed ([24]) that ADI formulations like (11)–(12) may lead to an increased accuracy. The analysis of this phenomenon is beyond the scope of this study and it will not be presented here. Therefore, in this study we will only consider the $O(h^2)$ version of (11)–(12). In what follows we will call the ADI method defined by the relations (6)–(7) ADISC1 and call the one defined by the relations (11)–(12) ADISC2.

2.3. Complexity analysis. For our analysis and implementation we naturally represent the individual matrix factors of a tensor product as separate matrices stored in Fortran fashion as two-dimensional arrays while it is computationally convenient to represent the vectors as k-dimensional arrays. We have the following definition for such an array and its transpose.

DEFINITION 2.1. *Let* $\mathbf{x}$ *be a vector of order* $\prod_{i=1}^{k} N_i$*, where* $N_i \in \mathbb{N}, i = 1, \ldots, k$*. We define the* $N_1 N_2 \cdots N_k$ *k-multidimensional matrix representation* $\mathbf{X} = \{\mathbf{X}_{i_1,i_2,\ldots,i_k}\}$ *of* $\mathbf{x}$ *by*

$$\mathbf{X}_{i_1,i_2,\ldots,i_k} = \mathbf{x}_{i_1+N_1(i_2-1+N_2(i_3-1+\cdots+N_{k-1}(i_k-1)\cdots))},$$

where $i_j = 1, \ldots, N_j$*,* $j = 1, \ldots, k$*.*

DEFINITION 2.2. *Let* $\mathbf{X}$ *be a k-multidimensional matrix of order* $\prod_{i=1}^{k} N_i$*. Then* $\mathbf{X}^T$ *is a k-multidimensional matrix of order* $\prod_{i=2}^{k} N_i \times N_1$ *such that*

$$\mathbf{X}^T_{l_2,l_3,\ldots,l_k,l_1} = \mathbf{X}_{l_1,l_2,\ldots,l_k},$$

where $l_i = 1, \ldots, N_i$*,* $i = 1, \ldots, k$*.*

For the implementation of the ADI schemes we need efficient procedures for computing matrix-times-vector operations of the tensor product form $(\prod_{i=1}^{k} \otimes A_i)\mathbf{x}$ and for solving systems of linear equations given in the following tensor product form:

$$\left(\prod_{i=k}^{1} \otimes A_i \right) \mathbf{X} = \mathbf{B}. \tag{15}$$

These procedures should use only the factors A_i and avoid explicitly forming the matrix $A = \prod_{i=1}^{k} \otimes A_i$. Using the above definitions, the following theorem (its proof, given in [29], is a simple generalization of the analysis found in [4] for the $k = 2$ case) gives us such efficient procedures which only involve the matrices A_i, $\mathbf{X}$, and $\mathbf{B}$.

THEOREM 2.3. *Let* A_i *be matrices of order* $N_i \times N_i, i = 1, \ldots, k$*, let* $\mathbf{x}$ *be a vector of order* $\prod_{i=1}^{k} N_i$*, let* $\mathbf{X}$ *be its k-multidimensional matrix representation of order* $N_1 \cdots N_{k-1} N_k$*, let* $\mathbf{B}$ *be k-multidimensional matrices of order* $N_1 \cdots N_{k-1} N_k$*, and consider the linear system* $(\prod_{i=k}^{1} \otimes A_i)\mathbf{X} = \mathbf{B}$*. Then we have that*

$$\left(\prod_{i=k}^{1} \otimes A_i \right) \mathbf{x} = \left(A_k \left(A_{k-1} \left(\cdots A_2 \left(A_1 \mathbf{X} \right)^T \cdots \right)^T \right)^T \right)^T, \tag{16}$$

$$\left(\left(\prod_{i=k}^{1}\otimes A_i\right)\mathbf{X}\right)^T = \left(A_1\otimes\left(\prod_{i=k}^{2}\otimes A_i\right)\right)\mathbf{X}^T, \tag{17}$$

and that if A_i^{-1} *exists for* $i = 1,\ldots,k$ *and if*

$$A_1\mathbf{Y}_1 = \mathbf{B}, A_2\mathbf{Y}_2 = \mathbf{Y}_1^T,\ldots,A_i\mathbf{Y}_i = \mathbf{Y}_{i-1}^T,\ldots,A_k\mathbf{Y}_k = \mathbf{Y}_{k-1}^T,$$

then $\mathbf{X} = \mathbf{Y}_k^T$.

It is worthwhile to point out that while the first and third parts of this theorem give us efficient ways for computing the matrix–vector product and for solving linear systems of equations of the form (15), respectively, the second one gives us the ability to switch the order of the terms in a tensor product. Instead of solving linear systems of the form

$$\left(\left(\prod_{i=k}^{m+1}\otimes I\right)\otimes A_m\otimes\left(\prod_{i=m-1}^{1}\otimes I\right)\right)\mathbf{X} = \mathbf{B}\ , \tag{18}$$

this allows us to use existing numerical software to compute the solution of the equivalent, in view of the above theorem, linear system

$$\left(\left(\prod_{i=m-1}^{1}\otimes I\right)\otimes\left(\prod_{i=k}^{m+1}\otimes I\right)\otimes A_m\right)\mathbf{X}^{\overbrace{T\cdots T}^{m-1\ \ times}} = \mathbf{B}^{\overbrace{T\cdots T}^{m-1\ \ times}}\ . \tag{19}$$

In order to compare the efficiency of our ADI algorithms we estimate, in the following lemma (its proof based on the above theorem is simple and can be found in [29]), the computer work by computing the so-called operation counts in the traditional manner by counting only floating-point operations.

LEMMA 2.4. *Let* A_i, $i = 1,\ldots,k$, *be matrices of order* $N_i\times N_i$ *and let* $\mathbf{X}$ *and* $\mathbf{B}$ *be k-multidimensional matrices of order* $\mathcal{K}\equiv N_1N_2\cdots N_k$. *Then the computer work required to*

- *compute* $(A_k\otimes A_{k-1}\otimes\cdots\otimes A_1)\mathbf{B}$ *is* $\mathcal{K}\sum_{i=1}^{k}(2N_i-1)$ *and reduces to* $4\mathcal{K}\sum_{i=1}^{k}M_i$ *when the* A_i*'s are banded matrices with respective bandwidths* M_i*;*
- *solve the linear system* (15) *using Gauss elimination with partial pivoting is* $2(\mathcal{K}\sum_{i=1}^{k}N_i+\sum_{i=1}^{k}N_i^3/3)$ *and reduces to* $2(3\mathcal{K}\sum_{i=1}^{k}M_i+\sum_{i=1}^{k}M_i^2N_i)$ *when the* A_i*'s are banded matrices with respective bandwidths* M_i.

To clearly observe the significant difference in the efficiency we mark that for matrices $A_i,\mathbf{X},\mathbf{B}$ given as above, where now $N_i = N$ and $M_i = M$ for $i = 1,\ldots,k$, the work to solve the linear system (15) reduces from $2((M^2+3M)N^k)$ when the tensor product $\prod_{i=k}^{1}\otimes A_i$ is expanded to $2k(3MN^k+M^2N)$ when the above theorem is used.

The procedures described above are not only time efficient but also memory efficient. Using simple calculations we can obtain Table 1 which shows the amount of memory required to store the data by using the data structures described at the beginning of this section (in the second row) and by storing $\prod_{i=k}^{1}\otimes A_i$ expanded (in the third row). As we see, the usage of tensor products can significantly reduce the computer memory required to store $\prod_{i=k}^{1}\otimes A_i$. This is particularly important for solving three- (or more) dimensional problems for which memory can easily be exhausted even on modern computers.

TABLE 1
Memory requirements for storing $\prod_{i=1}^{k} \otimes A_i$.

Storage format	Memory	
	Full	Band
$(A_k \otimes \cdots \otimes A_1)$	$\sum_{i=1}^{k} N_i^2$	$2\sum_{i=1}^{k} M_i N_i$
expand $\prod_{i=k}^{1} \otimes A_i$	$\prod_{i=1}^{k} N_i^2$	$2M_1N_1\prod_{i=2}^{k} N_i^2$

TABLE 2
*Work to sweep the first direction of ADISC*1.

Procedure	Work	
	+,−	*
$W_1 := D + r_{s+1}A_1$	$3(N_1 - 1)$	$3(N_1 - 1)$
$U_i^{(s)} := A_i U^{(s)}$	$2k\mathcal{K}$	$2k\mathcal{K}$
$Y := \sum_{i=1}^{k} U_i^{(s)}$	$(2k^2 + k - 1)\mathcal{K}$	$2k^2\mathcal{K}$
$\mathrm{R} := W_1U^{(s)} - \omega Y + \omega F$	$2(k + 1)\mathcal{K}$	$2(k + 1)\mathcal{K}$
Solve $W_1U^{(s+1/k)} = \mathrm{R}$	$3k\mathcal{K} + \sum_{i=1}^{k} N_i$	$3k\mathcal{K} + \sum_{i=1}^{k} N_i$
Total	$(2k^2 + 6k + 1)\mathcal{K}+$ $3(N_1 - 1) + \sum_{i=1}^{k} N_i$	$(2k^2 + 5k + 2)\mathcal{K}+$ $3(N_1 - 1) + \sum_{i=1}^{k} N_i$

TABLE 3
Work to sweep the last $k - 1$ *directions of ADISC*1.

Procedure	Work	
	+,−	*
$W_i := D + r_{s+1}A_i$	$3(N_i - 1)$	$3(N_i - 1)$
$Y_1 := DU^{(s+(i-1)/k)}$	$2k\mathcal{K}$	$2k\mathcal{K}$
$Y_2 := A_iU^{(s)}$	$2k\mathcal{K}$	$2k\mathcal{K}$
$Y := Y_1 + r_{s+1}Y_2$	$\mathcal{K}$	$\mathcal{K}$
Solve $W_iU^{(s+i/k)} = \mathrm{Y}$	$3k\mathcal{K} + \sum_{j=1}^{k} N_j$	$3k\mathcal{K} + \sum_{j=1}^{k} N_j$
Total	$(7k + 1)\mathcal{K}+$ $3(N_i - 1) + \sum_{j=1}^{k} N_j$	$(7k + 1)\mathcal{K}+$ $3(N_i - 1) + \sum_{j=1}^{k} N_j$

Based on the above complexity results it is easy to calculate the per iteration work involved in the two $O(h^2)$ ADISC schemes derived in the previous section. This way, a detailed arithmetic count for the sweep of our schemes in the first direction (relations (6) and (11)) is presented in Tables 2 and 4, respectively, while the arithmetic count in the other directions (associated with relations (7) and (12)) are presented in Tables 3 and 5. We can calculate the total number of additions (O_j^A) and multiplications (O_j^M) for our ADISCj $j = 1, 2$ scheme by multiplying the last rows in Tables 3 and 5, by $k - 1$ and adding them to the last rows in Tables 2 and 4, respectively.

To compare the per iteration complexity of our two schemes we need to compare the overall total work required. Assuming that the time to perform (by the CPU) a multiplication is twice the time to perform an addition, we calculate the quantities $O_j = O_j^A + 2O_j^M$, $j = 1, 2$, which correspond to the CPU time needed for the ADISC1 and ADISC2 schemes to perform an iteration, respectively. In Figure 3 we plot the quantity $100\frac{O_1 - O_2}{O_2}$ $(= 100\frac{27k^2 - 2^k(3k+1) - 25k - 4}{2^k(3k+1) + 23k + 2})$ in the y-axis versus the dimension k of the PDE problem. As depicted the ADISC2 iterative method is, per iteration, more efficient than the ADISC1 one when Ω is a k-dimensional domain for $k = 2, 3, 4$, and 5 achieving its maximum relative efficiency, of approximately 55%, for three-dimensional

TABLE 4
Work to sweep the first direction of ADISC2.

Procedure	Work	
	+,–	*
$Z_1 := r_{s+1}I + X_1$	$3(N_1 - 1)$	$3(N_1 - 1)$
$U_i^{(s)} := X_i U^{(s)}$	$2\mathcal{K}$	$2\mathcal{K}$
$V_{ij}^{(s)} := H_j^i U(s)$	$2i\mathcal{K}$	$2i\mathcal{K}$
$\sum_{i=1}^{k} U_i^{(s)} + \sum_{i=2}^{k} \sum_{j=1}^{\binom{k}{i}} V_{ij}^{(s)} = V$	$(2^k(k+1) + k - 1)\mathcal{K}$	$(2^k + 2)k\mathcal{K}$
$\mathrm{R} := Z_1 U^{(s)} - \omega V + \omega F$	$4\mathcal{K}$	$4\mathcal{K}$
Solve $Z_1 U^{(s+1/k)} = \mathrm{R}$	$3\mathcal{K} + N_1$	$3\mathcal{K} + N_1$
Total	$(2^k(k+1) + k + 6)\mathcal{K}$ $+4N_1 - 3$	$((2^k + 2)k + 7)\mathcal{K}$ $+4N_1 - 3$

TABLE 5
Work to sweep the last $k-1$ directions of ADISC2.

Procedure	Work	
	+,–	*
$Z_i := r_{s+1}I + X_i$	$3(N_i - 1)$	$3(N_i - 1)$
$Y_1 := X_i U^{(s)}$	$2\mathcal{K}$	$2\mathcal{K}$
$Y := r_{s+1}U^{(s+(i-1)/k)} + Y_1$	$\mathcal{K}$	$\mathcal{K}$
Solve $Z_i U^{(s+i/k)} = Y$	$3\mathcal{K} + N_i$	$3\mathcal{K} + N_i$
Total	$6\mathcal{K} + 4N_i - 3$	$6\mathcal{K} + 4N_i - 3$

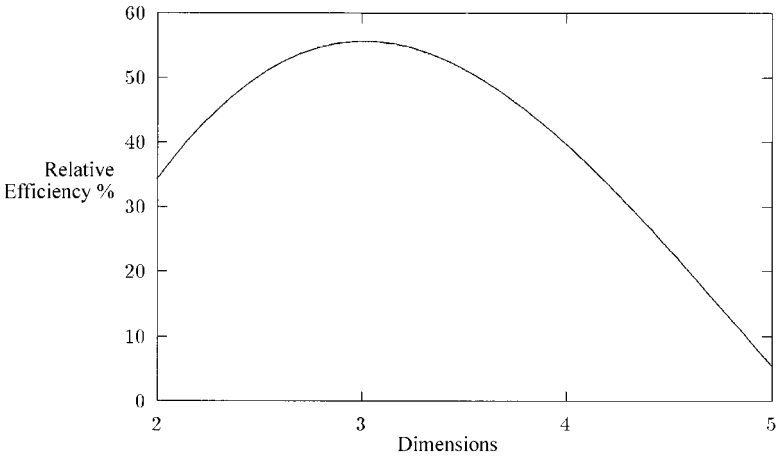

FIG. 3. *Efficiency improvement (percent).*

problems. In more than five dimensions ADISC1 seems to win, increasing its relative efficiency as k increases.

2.4. Convergence analysis. In this section we present our theoretical results concerning the convergence analysis of the proposed ADISC methods. Although the analysis can be carried out for the generalized Helmholtz PDE problem (i.e., $a_i = 1, i = 1, \ldots, k$) (1b)–(2b), for simplicity in the presentation, we consider the Poisson PDE equation (i.e., $\gamma = 0$). Recall that our two ADI methods for solving the interior collocation equation can be written in the form

$$\mathbf{U}^{(s+1)} = M_{r_{s+1},\omega}\mathbf{U}^{(s)} + \mathbf{G}, \tag{20}$$

where the iteration matrix $M_{r_{s+1},\omega}$ is given by relations (9) and (14) for the ADISC1 and ADISC2 methods, respectively. We start our analysis by giving below analytic expressions of the eigenvalues of these iteration matrices.

LEMMA 2.5. *The eigenvalues* $\underline{\nu}$ *of the iteration matrix* $M_{r_{s+1},\omega}$ *defined by the relation* (9) *for the ADISC*1 *scheme and by* (14) *for the ADISC*2 *scheme are given by*

$$\underline{\nu} = \underline{\nu}^{(\underline{\ell})} = 1 - \omega \left[\frac{\sum_{i=1}^{k} \frac{\alpha_i^{(\ell_i)}}{r_{s+1}} + C}{\prod_{i=1}^{k} \left(\frac{\alpha_i^{(\ell_i)}}{r_{s+1}} + 1 \right)} \right] \tag{21}$$

with $\underline{\ell} = (\ell_1, \ldots, \ell_k)$, $\ell_i = 1, \ldots, N_i - 1$, *where*

$$\alpha_i^{(\ell_i)} = \begin{cases} \frac{-6N_i^2 \lambda_i^{(\ell_i)}}{6+\lambda_i^{(\ell_i)}} & \text{for ADISC1} \quad O(h^2), \\ \frac{-N_i^2 \lambda_i^{(\ell_i)} \left(\lambda_i^{(\ell_i)}+12\right)}{2\left(\lambda_i^{(\ell_i)}+6\right)} & \text{for ADISC1} \quad O(h^4), \\ -N_i^2 \lambda_i^{(\ell_i)} & \text{for ADISC2} \quad O(h^2), \end{cases} \tag{22}$$

$$\lambda_i^{(\ell_i)} = -4 \sin^2 \left(\frac{\ell_i \pi}{2N_i} \right), \tag{23}$$

and $C = 0$ *for the ADISC*1 *scheme and*

$$C = \sum_{i=2}^{k} \frac{i}{r_{s+1}} \sum_{j=1}^{\binom{k}{i}} \frac{N_j^2}{6^{i-1}} \prod_{m=1}^{k} p_m^{(ij)} \lambda_m^{(\ell_m)}$$

*for the ADISC*2 *scheme with*

$$p_m^{(ij)} = \begin{cases} 1 & \text{if there is } T_{-2} \text{ at the position } m \text{ of the matrix } H_{ij}, \\ \frac{1}{\lambda_m^{(\ell_m)}} & \text{if there is } I \text{ at the position } m \text{ of the matrix } H_{ij}. \end{cases}$$

Proof. It is known ([14]) that the eigenvalues $\lambda_i^{(\ell_i)}$ of the matrix $T_{-2}^i \in \mathbb{R}^{N_i \times N_i}$ are given by the relation (23). It is easy to see that all the matrices $T_a^i \in \mathbb{R}^{N \times N}$, $T_a^i \equiv \text{tridiag}(1, a, 1)$, $a \in \mathbb{R}$, have a common set of linearly independent eigenvectors and that the eigenvalues of the matrix T_a^i are given by $(a+2) + \lambda_i^{(\ell_i)}$. Relations (21) and (22) can be now easily obtained by applying Lemma 3.5 in [13] to the iteration matrices given by (9) and (14). ∎

Denote now the error at the sth iteration by $\mathbf{E}^{(s)} = \mathbf{U} - \mathbf{U}^{(s)}$, where $\mathbf{U}$ and $\mathbf{U}^{(s)}$ are the coefficients of u_Δ and $u_\Delta^{(s)}$ in (3), respectively. Then from (20) we have that

$$\mathbf{E}^{(s+1)} = M_{r_{s+1},\omega} \mathbf{E}^{(s)}.$$

We can expand the error $\mathbf{E}^{(s)}$ in terms of the eigenvectors $\mathbf{p}_{i_j}, j = 1, \ldots, k$, of $D^{-1}A_i$ or of X_i (see Lemma 3.5 in [13]) and use the above to get

$$\mathbf{E}^{(s+1)} = \sum_{i_1=1}^{N_1-1} \cdots \sum_{i_k=1}^{N_k-1} \nu_{i_1,\ldots,i_k}(r_{s+1}) E_{i_1,\ldots,i_k} \mathbf{p}_{i_1} \otimes \cdots \otimes \mathbf{p}_{i_k},$$

where $\nu_{i_1,\ldots,i_k}(r_{s+1})$ are the eigenvalues of the iteration matrix given by Lemma 2.5. Thus we have

$$\mathbf{E}^{(s)} = \sum_{i_1=1}^{N_1-1} \cdots \sum_{i_k=1}^{N_k-1} \left[\prod_{j=0}^{s} \nu_{i_1,\ldots,i_k}(r_j) \right] E_{i_1,\ldots,i_k} \mathbf{p}_{i_1} \otimes \cdots \otimes \mathbf{p}_{i_k}. \tag{24}$$

For the ADISC1 scheme (the ADISC2 scheme can be treated similarly) we use relation (24), Lemma 3.5 in [13], and the fact that the function

$$g(x_1, \ldots, x_k) := \frac{\sum_{i=1}^{k} x_i}{\prod_{i=1}^{k}(x_i + 1)} - 1$$

is always negative to obtain our first convergence result.

THEOREM 2.6. *For any given set of positive acceleration parameters $r_{s+1}, s = 0, 1, \ldots$, and $0 < \omega \leq 2$, the proposed iterative methods ADISC1 and ADISC2 converge from any initial guess.*

For the rest of this section we restrict ourselves in three dimensions since most of our results cannot be easily extended to more dimensions. Without loss of generality, and for simplicity in the presentation only, we will assume that we have a uniform discretization grid of equal spacing in all dimensions (i.e., $N_i = N, i = 1, 2, 3$).

It is worthwhile to note that the ADISC1 iterative method can be exact (except for round-off) in a number of iterations equal to the number of unknowns. This can be easily seen by observing that $\nu_{i_1,i_2,i_3}(r_s)$ is a fraction whose numerator is a cubic polynomial in r_s which has a real root $r_{i_1 i_2 i_3}$ for which the denominator does not vanish. Therefore, $\mathbf{E}^{(s)}$ can be made zero in $(N-1)^3$ iterations by setting $r_s = r_{i_1 i_2 i_3}$, $s = 1, \ldots, (N-1)^3$, and $i_j = 1, \ldots, (N-1)$, $j = 1, 2, 3$.

If the number of iterations s required is known in advance, one can determine the optimum values for the sequence of iteration parameters r_s by solving a minimax problem. This minimax problem becomes a whole sequence of such problems since in practice we very rarely know s in advance. Our objective here is to choose a sequence of "good" acceleration parameters r_s that will reduce the number of iterations required to produce a satisfactory approximation to the solution. We will do that, following the methodology found in [7], for the Douglas scheme, i.e., we set $\omega = 2$.

We start by letting for $j = 1, 2, 3$

$$\zeta_\ell = \frac{12}{r_\ell h^2},\ \ell = 1, 2, \ldots, \quad \text{and} \quad \xi_{i_j} = \frac{\sin^2 \frac{i_j \pi}{2N}}{3 - 2\sin^2 \frac{i_j \pi}{2N}},\ \ i_j = 1, \ldots, N-1\ . \tag{25}$$

Then we have that

$$\nu_{i_1,i_2,i_3}(r_\ell) = 1 - 2\frac{\zeta_\ell \xi_{i_1} + \zeta_\ell \xi_{i_2} + \zeta_\ell \xi_{i_3}}{(1 + \zeta_\ell \xi_{i_1})(1 + \zeta_\ell \xi_{i_2})(1 + \zeta_\ell \xi_{i_3})}. \tag{26}$$

We would like to find a sequence $\{\zeta^{(\ell)}\}$, $\ell = 1, \ldots, P$, such that $\mu \leq \zeta^{(\ell)} \xi_{i_j} \leq \nu$ for every ℓ for at least one j. μ and ν are parameters that will be determined later on. Since $\xi_1 = \frac{\sin^2 \frac{\pi}{2N}}{3 - 2\sin^2 \frac{\pi}{2N}}$ and $\xi_{N-1} \approx 1$ we define $\xi^{(1)} = \xi_1$ and the sequence $\{\zeta^{(\ell)}, \xi^{(\ell)}\}$ such that

$$\zeta^{(\ell)} \xi^{(\ell)} = \mu \ \text{ and } \ \zeta^{(\ell)} \xi^{(\ell+1)} = \nu$$

from which we have for $\ell = 1, \ldots, P-1$ that

$$\zeta^{(\ell)} = \mu \left(\frac{\mu}{\nu}\right)^{\ell-1} \frac{3 - 2\sin^2 \frac{\pi}{2N}}{\sin^2 \frac{\pi}{2N}} \tag{27}$$

and

$$\xi^{(\ell)} = \left(\frac{\nu}{\mu}\right)^{\ell-1} \frac{\sin^2 \frac{\pi}{2N}}{3 - 2\sin^2 \frac{\pi}{2N}}, \tag{28}$$

for which $\mu \leq \zeta^{(\ell)}\xi \leq \nu$ for every ξ such that $\xi^{(\ell)} \leq \xi \leq \xi^{(\ell+1)}$. We stop generating terms when we cross one, i.e., when $\xi^{(P+1)} \approx 1 \approx \xi_{N-1}$, and thus we have that

$$P = \log^{-1}\left(\frac{\nu}{\mu}\right) \log\left(\frac{3 - 2\sin^2 \frac{\pi}{2N}}{\sin^2 \frac{\pi}{2N}}\right). \tag{29}$$

Using the expressions for ζ_ℓ in (25) and (27) we obtain the following expression for the acceleration parameters:

$$r_s = \frac{12N^2}{\mu}\left(\frac{\nu}{\mu}\right)^{s-1} \frac{\sin^2 \frac{\pi}{2N}}{3 - 2\sin^2 \frac{\pi}{2N}}, \qquad s = 1, \ldots, P. \tag{30}$$

We proceed by stating two lemmas found in [7].

LEMMA 2.7. *Let*

$$\rho \equiv \rho(a,b,c) = 1 - \frac{2(a+b+c)}{(1+a)(1+b)(1+c)} \tag{31}$$

and

$$\begin{aligned}\hat{\rho}(\mu,\nu) \equiv \max\{|\rho(a,b,c)| : [\mu \leq a \leq \nu; 0 \leq b,c \leq \nu] \ \text{ or} \\ [\mu \leq b \leq \nu; 0 \leq a,c \leq \nu] \ \text{ or } \ [\mu \leq c \leq \nu; 0 \leq a,b \leq \nu]\}.\end{aligned}$$

Then if $\mu < 1 < \nu$

$$\hat{\rho}(\mu,\nu) = \max\left[1 - \frac{6\nu}{(1+\nu)^3}, 1 - \frac{2\mu}{1+\mu}\right]. \tag{32}$$

LEMMA 2.8. *Let* $\nu \geq 1$ *and*

$$\mu = \frac{3\nu(1+\nu)^{-3}}{1 - 3\nu(1+\nu)^{-3}} = \frac{3\nu}{1 - 3\nu^2 + \nu^3}; \tag{33}$$

then

$$\hat{\rho}(\mu,\nu) = 1 - \frac{6\nu}{(1+\nu)^3} = 1 - \frac{2\mu}{1+\mu}. \tag{34}$$

If we denote by R_s the operator which maps $\mathbf{E}^{(0)}$ to $\mathbf{E}^{(s+1)}$, then we easily see from (24) that its L_2-norm is

$$||R_s|| = \max_{i_1,\ldots,i_k}\left[\prod_{j=0}^{s} \nu_{i_1,\ldots,i_k}(r_j)\right]. \tag{35}$$

From Lemmas 2.7 and 2.8 we have that if we iterate P times using the parameters given in (30) we have

$$||R_P|| \leq \hat{\rho}(\mu, \nu),$$

and if we iterate mP times using the parameters in (30) cyclically (i.e., $r_{mP+\ell} = r_\ell$, $m = 1, 2, \ldots$) we have that

$$||R_{mP}|| \leq \hat{\rho}^m(\mu, \nu).$$

If we now want to have $||R_{mP}|| \approx \epsilon$ we get

$$m \approx \frac{\log \epsilon}{\log \hat{\rho}(\mu, \nu)}$$

and therefore

$$mP \approx \frac{-\log \epsilon \log\left(\frac{3-2\sin^2\frac{\pi}{2N}}{\sin^2\frac{\pi}{2N}}\right)}{\log\left(\hat{\rho}(\mu,\nu)\right)^{-1} \log\left(\frac{\nu}{\mu}\right)}. \tag{36}$$

Assuming (for our convenience) that μ satisfies (33) and evaluating the denominator of relation (36) for $\nu = 1(.01)2$ we find that the number of iterations mP is minimized for $\nu = 1.78$ and $\mu = 0.33$. We can summarize the above discussion (the $O(h^4)$ case can be treated in a similar way) in the following theorem.

THEOREM 2.9. *If the acceleration parameters r_s are selected as*

$$r_s = \begin{cases} \frac{12N^2}{\mu}\left(\frac{\nu}{\mu}\right)^{s-1} \frac{\sin^2\frac{\pi}{2N}}{3-2\sin^2\frac{\pi}{2N}} & \text{in the } O(h^2) \text{ case,} \\ \frac{8N^2}{\mu}\left(\frac{\nu}{\mu}\right)^{s-1} \frac{\sin^2\frac{\pi}{2N}\left(3-\sin^2\frac{\pi}{2N}\right)}{6-4\sin^2\frac{\pi}{2N}} & \text{in the } O(h^4) \text{ case,} \end{cases} \qquad s = 1, \ldots, P, \tag{37}$$

and used cyclically, i.e., $r_{iP+s} = r_s$, $i = 1, 2, \ldots$, $s = 0, \ldots, P-1$, the ADISC1 iterative methods with $\omega = 2$ will reduce the initial error $\mathbf{E}^{(0)}$ by a preassigned factor of ϵ in mP iterations where $m \approx \frac{\log \epsilon}{\log .5}$

$$P \approx \begin{cases} 0.59 \log\left(\frac{3-2\sin^2\frac{\pi}{2N}}{\sin^2\frac{\pi}{2N}}\right) & \text{in the } O(h^2) \text{ case,} \\ 0.59 \log\left(\frac{6-4\sin^2\frac{\pi}{2N}}{\sin^2\frac{\pi}{2N}\left(3-\sin^2\frac{\pi}{2N}\right)}\right) & \text{in the } O(h^4) \text{ case.} \end{cases} \tag{38}$$

In order to further increase the efficiency of our ADISC1 scheme we would like to find a value of ω for which either the asymptotic rate of convergence is a maximum or the number of calculations needed to reduce the norm of the matrix operator that maps the error vector $\mathbf{E}^{(0)}$ to the error vector $\mathbf{E}^{(s)}$, $||R_s||$, below a preassigned tolerance is a minimum. We will follow the latter approach and the techniques found in [12]. Recall that the eigenvalues of the ADISC1 iteration matrix are given by

$$\rho = \rho(a, b, c) \equiv \underline{\nu} = 1 - \omega \frac{a+b+c}{(1+a)(1+b)(1+c)}, \tag{39}$$

where $a = \zeta_\ell \xi_{i_1}$, $b = \zeta_\ell \xi_{i_2}$, and $c = \zeta_\ell \xi_{i_3}$ with ζ_ℓ and ξ_{i_j} given by (25). We decide to use as acceleration parameters r_s the ones obtained above for $\omega = 2$. Thus we replace, for $\ell = 1, \ldots, n_0$, $(n_0 = P)$ the ζ_ℓ by $\zeta^{(\ell)}$ that are given in (27). So if we iterate n_0

times with any triple (a, b, c) such that $a \geq b \geq c$, then there exists $n^* \in \{1, \ldots, n_0\}$ for which one of the following is satisfied:

$$\mu \leq a_{n^*} \leq \nu, \quad t\mu \leq b_{n^*}, c_{n^*} \leq \nu, \tag{40}$$

$$\mu \leq b_{n^*} \leq \nu, \quad t\mu \leq a_{n^*}, c_{n^*} \leq \nu, \tag{41}$$

$$\mu \leq c_{n^*} \leq \nu, \quad t\mu \leq a_{n^*}, b_{n^*} \leq \nu \tag{42}$$

with

$$t = \frac{6\tan^2 \frac{\pi}{2N} - 4\sin^2 \frac{\pi}{2N}}{6 - 4\sin^2 \frac{\pi}{2N}}. \tag{43}$$

For $n \neq n^*$ the considered triple does not satisfy any of (40), (41), and (42) anymore. Instead, we have

$$t\mu \leq a_n, b_n, c_n \leq t^{-1}\nu. \tag{44}$$

By simply taking derivatives and following an elementary but tedious analysis we can prove the following lemma.

LEMMA 2.10. *Let t be given by relation* (43), $f = f(a, b, c) = \frac{a+b+c}{(1+a)(1+b)(1+c)}$,

$$f_{max} = \max\{f(a, b, c) \text{ for } a, b, c \text{ that satisfy } (44)\},$$

$$f^*_{max} = \max\{f(a, b, c) \text{ for } a, b, c \text{ that satisfy one of } (40),\ (41),\ \text{and } (42)\},$$

and

$$f^*_{min} = \min\{f(a, b, c) \text{ for } a, b, c \text{ that satisfy one of } (40),\ (41),\ \text{and } (42)\}.$$

Then we have

$$f_{max} = \frac{2t\mu + t^{-1}\nu}{(1 + t\mu)^2(1 + t^{-1}\nu)}, \tag{45}$$

$$f^*_{max} = \frac{2t\mu + \nu}{(1 + t\mu)^2(1 + \nu)}, \tag{46}$$

and

$$f^*_{min} = \min\left\{\frac{\mu + 2t\mu}{(1 + t\mu)^2(1 + \mu)}, \frac{3\nu}{(1 + \nu)^3}\right\}. \tag{47}$$

We are now in a position to determine the optimum value of ω as function of N, μ, and ν as follows. From (39) we observe that the possible range of ω is limited by the fact that we must have $|\rho| < 1$. So for triples a, b, c that satisfy the inequality (44) we have $-1 < 1 - \omega f < 1$ from which we have that

$$0 < \omega \leq \frac{2}{f_{max}}. \tag{48}$$

With ω in the above range we can easily verify that $|\rho| < 1$ for any triple (a, b, c) that satisfies one of the inequalities (40), (41), and (42). For convenience we take

$$\frac{\mu + 2t\mu}{(1+t\mu)^2(1+\mu)} = \frac{3\nu}{(1+\nu)^3}. \tag{49}$$

We can now easily see that for optimality we must have $1 - \omega f^*_{min} = -(1 - \omega f^*_{max})$, that is,

$$\omega = \frac{2}{f^*_{min} + f^*_{max}}. \tag{50}$$

To conclude, we consider the following two cases for ω_{opt}:

- If $\frac{2}{f^*_{min}+f^*_{max}} \leq \frac{2}{f_{max}}$ and if we choose ω_{opt} as in relation (50) which also satisfies (48), then $|\rho| \leq 1 - \omega_{opt} f^*_{min} \equiv \rho(\mu, \nu)$.
- If $\frac{2}{f^*_{min}+f^*_{max}} > \frac{2}{f_{max}}$ and for ω_{opt} in the range defined by (48) we have that $1 - \omega_{opt} f^*_{min} < -(1 - \omega_{opt} f^*_{max})$, thus we have again $|\rho| \leq \rho(\mu, \nu)$.

So by choosing

$$\omega_{opt} = \min\left\{\frac{2}{f^*_{min} + f^*_{max}}, \frac{2}{f_{max}}\right\} \tag{51}$$

we have $|\rho| < 1 - \omega_{opt} f^*_{min}$.

Going back to (49), we can see that function $y(\nu) = \frac{3\nu}{(1+\nu)^3}$ decreases for $\nu \geq 1$ and so we have $y(\nu) \leq y(1) = 3/8$ for $\nu \geq 1$. Thus to find the optimum pair of (μ, ν), which minimizes the total number of iterations, we search among the pairs (μ, ν) that satisfy (47), $0 < \mu \leq 1$, and

$$0 < \frac{\mu + 2t\mu}{(1+t\mu)^2(1+\mu)} \leq \frac{3}{8},$$

and maximize the function $\log \rho(\mu, \nu)^{-1} \log(\mu/\nu)$. Notice that since $0 < f^*_{min} < 3/8$ and we want $0 < 1 - \omega f^*_{min} \leq 1$, the desired inequality is equivalent to $0 < 1 - \frac{3}{8}\omega \leq 1$ or $0 \leq \omega < \frac{8}{3}$.

2.5. More general PDE operators. The problem of selecting the acceleration parameters r_s of the proposed ADI iterative schemes for more general than Helmholtz PDE operators is obviously very important. The assumption of commutativity, while necessary for our analysis presented in the previous sections, is rarely satisfied in practice. Fortunately, as observed by the authors and others the acceleration parameters based on the commutative assumption have worked well in a variety of problems. In particular Young and Ehrlich in [31] indicated how successful such ADI methods could be, even when it is apparent that the commutativity theory on which the selection of parameters was based do not hold, while Pearcy in [25] has shown that we can always get convergence for any given problem if we choose the cycle length m sufficiently large, have a mild restriction on the size of iterative parameters, and use them in certain order. Furthermore, some preliminary studies have shown to us that the noncommutative case associated with a general self-adjoint PDE operator can be treated for our ADISC scheme in a way similar to the one found in [30] for the five-point star ADI scheme, but this is beyond the scope of the present study.

In this section we address the problem of theoretically selecting the acceleration parameters r_s of the ADISC1 iterative scheme for more general than Helmholtz operators and specifically the following PDE operator:

$$\sum_{i=1}^{k} -\alpha_i \frac{\partial^2 u}{\partial x_i^2} = f \quad \text{in} \quad \Omega, \tag{52}$$

where $\alpha_i (> 0)$ are functions of x_i and f is a function of k variables. Following the procedure presented in the previous section (see also [29]), one can easily see that the matrices $\mathcal{E}_i$ involved in the interior collocation equations are given for the PDE operator (52) by

$$\mathcal{E}_i = \mathcal{T}^T \mathcal{A}_i \mathcal{T} \quad \text{with} \quad \mathcal{A}_i = \text{diag}(\alpha_{ij}), \tag{53}$$

where $\alpha_{ij} = \alpha_i((x_{j-1} + x_j)/2), j = 1, \ldots, N_i + 1$, and

$$\mathcal{T} = \begin{pmatrix} 1 & & & & \\ -1 & 1 & & & \\ & -1 & 1 & & \\ & & & \ddots & \\ & & & -1 & 1 \\ & & & & -1 \end{pmatrix}_{(N_i+1)\times(N_i)}.$$

The ADISC1 scheme which is defined by the recurrence relations (6)–(7) can be expressed in the equivalent form

$$\begin{gathered} \left(\tfrac{1}{r_{s+1}} B_1 + I\right) \mathbf{W}^{(s+1/2)} = \left(\tfrac{1}{r_{s+1}} B_1 + I\right) \mathbf{W}^{(s)} - \omega \textstyle\sum_{j=1}^{k} \tfrac{1}{r_{s+1}} B_j \mathbf{W}^{(s)} + \omega D^{1/2} \mathbf{F}, \\ \left(\tfrac{1}{r_{s+1}} B_i + I\right) \mathbf{W}^{(s+1)} = \tfrac{1}{r_{s+1}} B_i \mathbf{W}^{(s)} + \mathbf{W}^{(s+1/2)} \text{ for } i = 2, \ldots, k, \end{gathered} \tag{54}$$

where $B_i = D^{-1/2} \mathcal{A}_i D^{1/2}, i = 1, \ldots, k$, and where $\mathbf{W}^{(s)} = D^{1/2} \mathbf{U}^{(s)}, s = 1, 2, \ldots$. The iteration matrix associated with the above scheme is given in the form

$$M_{r_{s+1},\omega} = I - \omega \frac{1}{r_{s+1}} \prod_{i=k}^{1} \left(\frac{1}{r_{s+1}} B_i + I\right)^{-1} \left(\sum_{i=1}^{k} B_i\right). \tag{55}$$

To derive the spectral radius of the above ADISC1 iteration matrix we first obtain expressions for the eigenvalues of the matrices B_i. For this we denote with λ^i an eigenvalue of the matrix B_i and with $\phi \equiv \phi^k \otimes \cdots \otimes \phi^1$ the associated eigenvector; we observe that

$$\lambda^i = (\lambda^i \phi, \phi) = \left(B_i \prod_{j=k}^{1} \otimes \phi^j, \prod_{j=k}^{1} \otimes \phi^j\right). \tag{56}$$

Therefore we have that

$$\lambda^i = \prod_{j=k}^{i+1} ||\phi^j||^2 \times \left(\mathcal{A}\hat{\phi}^i, \hat{\phi}^i\right) \times \prod_{j=i-1}^{1} ||\phi^j||^2 \quad \text{with} \quad \hat{\phi}^i = \frac{1}{h} \mathcal{T} T_4^{-1/2} \phi^i$$

from which we easily get

$$\lambda^i \leq \max_{0\leq x\leq 1} a_i(x)||\hat{\phi}^i||^2 \prod_{j=k,j\neq i}^{1} ||\phi^j||^2. \tag{57}$$

Also we see that

$$||\hat{\phi}^i||^2 = \left(T_{-4}\phi, -\tfrac{1}{h^2}T_{-2}T_{-4}\phi\right) \leq \tfrac{1}{h^2}\rho\left(-T_{-2}\right)\left(T_4^{-1}\phi^i, \phi^i\right) \leq \rho\left(-T_{-2}\right)\rho\left(T_4^{-1}\right)||\phi^i||^2.$$

If we assume, without loss of generality, that $||\phi^j|| = 1$, $j = 1, \ldots, k$, and use the analytic expressions for the eigenvalues of the tridiagonal matrices involved we obtain the right-hand-side of the inequality below:

$$\frac{1}{h^2}\min_{0\leq x\leq 1}\{a_i(x)\}\frac{2\sin^2\frac{\pi}{2N_i}}{3-2\sin^2\frac{\pi}{2N_i}} \leq \lambda^i \leq \frac{1}{h^2}\max_{0\leq x\leq 1}\{a_i(x)\}\frac{2\cos^2\frac{\pi}{2N_i}}{3-2\cos^2\frac{\pi}{2N_i}},\ i = 1, \ldots, k, \tag{58}$$

while its left side can be found in a similar way. Therefore, although analytic expressions for the eigenvalues of the matrices $B_i, i = 1, \ldots, k$, cannot be obtained, we have successfully computed sharp bounds for the spectral radii of the matrices B_i's. From this point on, the analysis is almost identical with that in the previous section but rather tedious and complicated and will not be presented here. The outcome of this analysis can be summarized in the following theorem.

THEOREM 2.11. *A "good" choice for the acceleration parameters r_s of the $O(h^2)$ ADISC1 scheme associated with the PDE operator* (52) *is the following:*

$$r_s = \frac{2N^2}{\mu}\left(\frac{\nu}{\mu}\right)^{s-1}\min_{0\leq x\leq 1, 1\leq i\leq k}\{a_i(x)\}\frac{\sin^2\frac{\pi}{2N}}{3-2\sin^2\frac{\pi}{2N}}, \qquad s = 1, \ldots, P \tag{59}$$

and used cyclically, i.e., $r_{iP+s} = r_s$, $i = 1, 2, \ldots$, $s = 0, \ldots, P-1$, the ADISC1 iterative methods with $\omega = 2$ will reduce the initial error $\mathbf{E}^{(0)}$ by a preassigned factor of ϵ in mP iterations where $m \approx \frac{\log\epsilon}{\log .5}$ and

$$P \approx 0.59\log\left(\frac{3-2\sin^2\frac{\pi}{2N}}{\sin^2\frac{\pi}{2N}}\frac{\max_{0\leq x\leq 1, 1\leq i\leq k}\{a_i(x)\}}{\min_{0\leq x\leq 1, 1\leq i\leq k}\{a_i(x)\}}\right). \tag{60}$$

3. Numerical experiments. In this section we present the results of our numerical experiments obtained for the $O(h^2)$ and the $O(h^4)$ ADISC1 iterative methods. Our computer implementation of these ADISC1 methods uses software components available to us through BLAS ([21], [6]) and LINPACK ([5]) and is based on algorithms for the basic tensor product operations proposed in [4]. Our codes perform very few floating-point operations outside BLAS and LINPACK. For the discretization of the PDE domain Ω we have used a uniform, equal in all dimensions, discretization with grid spacing h. As an initial guess we have used the zero function. We stop the iterations of our ADISC schemes when we obtain three significant digits correct in the max-norm of the relative error or when the max-norm of the difference of two successive iterations ($||\mathbf{U}^{(s+1)} - \mathbf{U}^{(s)}||_\infty$) is less than 10^{-7}. All experiments presented in this section have been performed in double precision Fortran on a CONVEX C-3420 configured with 128Mb RAM and all CPU times reported are in seconds.

For our experimental study we have considered three PDE model problems defined by the following PDE equations:

PDE 1: The Poisson equation

$$u_{xx} + u_{yy} + u_{zz} = f.$$

PDE 2: The generalized Helmholtz equation

$$u_{xx} + u_{yy} + u_{zz} - \gamma u = f$$

with $\gamma(x, y, z) = 100 + \cos(2\pi x) + \sin(3\pi y) + \cos(\pi z)$.

PDE 3: The general elliptic equation

$$(1 + x^2)u_{xx} + e^{y-1}u_{yy} + (3 + \sin^2(\pi z))u_{zz} + \gamma u = f$$

with $\gamma(x, y, z) = e^{2x}\cos(3\pi x) + y^3 - 2y + \sin(\pi z)\cos(2\pi z)$ on the unit cube with homogeneous Dirichlet boundary conditions. The right-hand-side function f was selected so that the true solution of PDE 1 and PDE 3 is

$$u(x, y, z) = 10e^{x+y+z}(x^2 - x)(y^2 - y)(z^2 - z)$$

and the true solution of PDE 2 is

$$u(x, y, z) = -0.31c(x)s(x)(y^2 - y)c(y)s(z)\left(\frac{1}{1 + (4\,(r(x) + r(y) + r(z)))^4} - 0.5\right),$$

where $c(t) := 5.4 - \cos(4\pi t)$, $s(t) := \sin(\pi t)$, and $r(t) = (t - .5)^2$. As we easily see all solutions are analytic with the one associated with the model problem PDE 2 (a problem from stratospheric physics [27]) being oscillatory. All the experimental data presented in this section were obtained using as acceleration parameters r_s and as cycle length m the ones determined by using Theorem 2.9 for PDE 1 and PDE 2 and by using Theorem 2.11 for PDE 3.

In Table 6 we present our results for the $O(h^2)$ scheme. Specifically we present the following for different numbers of discretization points N ($N = 5(5)30$) and for the three PDE model problems:

The achieved accuracy. By *error* in the third column of Table 6 we denote the maximum norm of the error at the discretization points, i.e., $error = ||u - u_\Delta||_\infty$.

The order of convergence. We compute an estimation of the order of the discretization from the expression

$$-\log\left(\frac{||(u - u_{\Delta_1})||_\infty}{||(u - u_{\Delta_2})||_\infty}\right) \Big/ \log\left(\frac{h_1}{h_2}\right),$$

where u_{Δ_i} represents the collocation spline approximate obtained using a uniform grid step h_i in all directions. As we easily see, the order of convergence is $O(h^2)$ for all three model problems as was theoretically expected.

Optimum ω. The optimum value of the relaxation parameter ω was obtained experimentally by systematically searching the value of ω in $[0, 4]$ which corresponds to the minimum number of iterations required by the ADISC1 method to satisfy the stopping criteria. These experimental estimations of ω_{opt} agree within a reasonable accuracy with the theoretical ones obtained (when applicable) by relation (51). It should be pointed out that the optimal point of ω for the model problem PDE 2

TABLE 6

Error, order of convergence, ω_{opt}, time, and number of iterations for the $O(h^2)$ scheme applied to the three PDE model problems.

	N	Error	Order	ω_{opt}	Time	Number of iterations		
						$\omega = 1$	$\omega = 2$	$\omega = \omega_{opt}$
PDE 1	5	1.01e-3		2.3	.04	21	7	5
	10	3.20e-4	1.88	2.3	.28	37	13	7
	15	1.55e-4	1.93	2.3	.93	44	13	9
	20	8.99e-5	2.00	2.3	2.20	57	18	13
	25	5.90e-5	1.97	2.3	4.24	62	21	14
	30	4.18e-5	1.96	2.3	7.30	75	18	13
PDE 2	5	7.87e-2		2.1 – 2.3	.04	17	6	6
	10	3.29e-2	1.43	2.1 – 2.6	.28	20	10	8
	15	1.52e-2	2.05	2.2 – 2.6	.93	22	8	6
	20	9.43e-3	1.76	2.2 – 2.6	2.19	42	10	10
	25	6.07e-3	2.06	2.1 – 2.4	4.20	34	13	10
	30	4.27e-3	2.00	2.2 – 2.6	7.19	37	14	10
PDE 3	5	1.01e-3		2.1 – 2.2	.06	15	3	4
	10	3.31e-4	1.84	2.2	.29	23	10	7
	15	1.53e-4	2.05	2.1	.97	28	7	7
	20	9.16e-5	1.89	2.1	2.23	45	13	10
	25	6.00e-5	1.98	2.1	5.20	57	17	14
	30	4.21e-5	2.01	2.1	7.40	49	14	13

becomes an interval of length up to .5. During this systematic search for ω_{opt} we were able to confirm the theoretically obtained interval of convergence for ω which agrees with remarkable accuracy with the experimental one.

Time. In order to check the per iteration efficiency of our implementations we give in the sixth column the average CPU time required to perform one iteration. A least squares logarithmic fit of these data shows that the per iteration total time T is given by $T = .0003N^{2.93}$. This experimental estimation confirms our theoretical one obtained in the complexity analysis in section 2.3 (see Tables 2, 3, 4, and 5).

Number of iterations. In the last three columns of Table 6 we present the number of iterations required by the Douglas–Rachford ($\omega = 1$), the Douglas ($\omega = 2$), and the optimum ADISC1 iteration methods to satisfy the stopping criteria. As can be observed there is a significant increase in the rate of convergence as we move from the Douglas–Rachford scheme to the Douglas scheme and to the optimum scheme for all the PDE model problems. We can also observe another nice feature of, at least, our optimal scheme in the fact that the associated number of iterations required for convergence for a specific discretization step remains almost constant for all model problems.

The same observations as above can be made for the data in Table 7 where we present, similar to the $O(h^2)$ case, our results obtained by the $O(h^4)$ ADISC1 scheme applied to the model problem PDE 1. In addition we can point out that the number of iterations required does not increase significantly as we move from the $O(h^2)$ to the $O(h^4)$ scheme.

In Figures 4 and 5 we present a detailed history of the convergence of the $O(h^2)$ and the $O(h^4)$ ADISC1 methods applied to the model problem PDE 1 with all stopping criteria removed. We can split both figures in two branches, the upper one corresponding to the Douglas–Rachford scheme and the lower one corresponding to the optimum scheme. Each of these branches can be further split into two groups of lines corresponding to *error* being $||u - u_\Delta||_\infty$ (lines "———" and "– . — ." on the upper branch) or being $||\mathbf{U}^{(s+1)} - \mathbf{U}^{(s)}||_\infty$ (lines "- - - - " and "– ... —" on the upper

TABLE 7
Error, order of convergence, ω_{opt}, and number of iterations for the $O(h^4)$ scheme applied to model problem PDE 1.

				Number of iterations		
N	Error	Order	ω_{opt}	$\omega = 1$	$\omega = 2$	$\omega = \omega_{opt}$
5	3.01e-4		2.3	31	11	9
10	3.31e-5	3.91	2.3	51	16	10
15	7.90e-6	3.82	2.3	60	19	12
20	3.16e-6	3.66	2.3	61	19	13
25	1.51e-6	3.36	2.3	73	21	13
30	8.95e-7	3.19	2.3	77	25	17

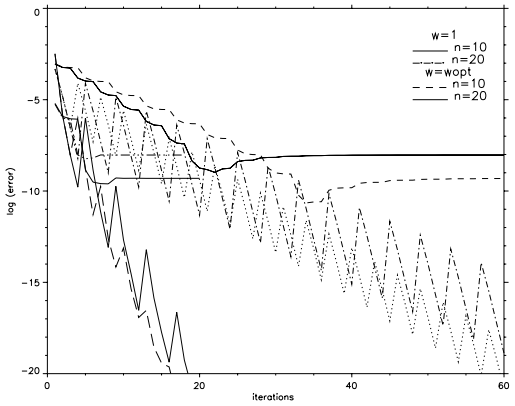

FIG. 4. *History of convergence for $O(h^2)$ and PDE 1.*

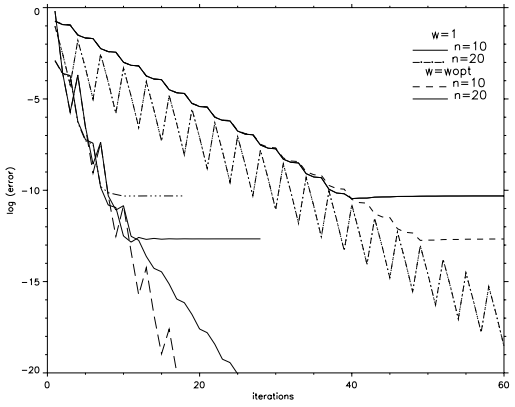

FIG. 5. *History of convergence for $O(h^4)$ and PDE 1.*

branch). We easily see the effect of using the acceleration parameters cyclically which is apparent for the $||\mathbf{U}^{(s+1)} - \mathbf{U}^{(s)}||_\infty$ case and almost invisible in the $||u - u_\Delta||_\infty$ case.

In order to rank the proposed ADISC1 iterative method we compare it with the two three-dimensional PDE solving methods available in ELLPACK ([27]), namely, the standard seven-point star finite difference method which is an $O(h^2)$ method and an $O(h^4)$ 27-point difference method called HODIE. The result of the seven-point star linear system (which involves N^3 equations and unknowns) was solved using

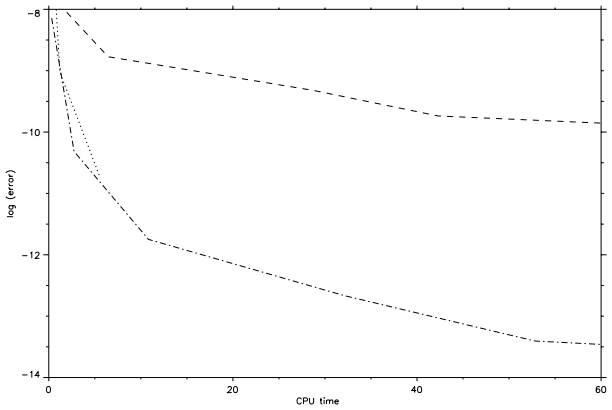

FIG. 6. *Efficiency of $O(h^2)$ ADISC1 (_ _) , $O(h^4)$ ADISC1 (_ . _ .), and $O(h^4)$ HODIE (- - -) methods.*

the ITPACK's SOR iterative method. The HODIE module discretizes the PDE using high-order finite difference stencils and solves the resulting linear system using the Fourier (FFT) method. The HODIE-FFT requires $2N^3 + 5N^2$ places memory workspace and its CPU execution time is proportional to $2N^2 \ln(N-1)$. In Figure 6 we plot the logarithm of the error $||u - u_\Delta||_\infty$ versus the required CPU time to achieve it. The data for the seven-point star method do not appear in the graph since its efficiency is too low and the associated line is further above the line -8 parallel to the x-axis. We mention here that for $N = 40$ the seven-point star generates a linear system of 54872 equations solved by the SOR method which converged in 124 iterations, took 69 seconds of CPU time, and achieved an error whose maximum norm was approximately 2.6×10^{-3}. To compare, the HODIE-FFT for $N = 20$ involves 5824 equations, took 1.21 seconds CPU time, and reduced the maximum norm of the error to 1.2×10^{-4}. Although Figure 6 compares the implementation and not the actual methods themselves, we can claim that the proposed methods easily outperform the standard seven-point star method and perform equally well with the high-order HODIE method. Here we should point out that we were unable to obtain more points to extend the HODIE line in the graph further due to memory limitations. Furthermore, the applicability of the HODIE method is restricted to generalized Helmholtz problems only and cannot be applied to general self-adjoint PDEs.

4. Synopsis and conclusions. In this study we have formulated, analyzed, and implemented efficient ADI iterative methods for the solution of the linear algebraic systems which arise from the discretization of self-adjoint elliptic PDE problems in k dimensions using $O(h^2)$ and $O(h^4)$ cubic spline collocation.

Two ADISC schemes have been proposed, namely, ADISC1 and ADISC2. A detailed per iteration complexity analysis has been presented showing us that the ADISC2 scheme is much more efficient (within an iteration) for $k = 2, 3, 4$ dimensions while ADISC1 takes over for $k > 5$. More specifically we have shown that the total number of operations required to perform one iteration step in the $O(h^2)$ ADISC1 and ADISC2 schemes are $(18k^2 - k - 1)N^k + O(N)$ and $(2^k(k+2) + 15k + 1)N^k + O(N)$, respectively, where N represents the number of discretization points in one direction.

Most of our convergence analysis of the proposed schemes has been carried out for the Poisson PDE and can be easily extended to the generalized Helmholtz PDE. Specifically we first prove the convergence of the $O(h^2)$ and $O(h^4)$ ADISC1 and the

$O(h^2)$ ADISC2 schemes in k dimensions for any set of positive acceleration parameters and for $0 < \omega < 2$. The rest of our analysis is restricted to our ADISC1 schemes and to three dimensions since most of our results cannot be easily extended to more dimensions. We obtain values for the acceleration parameters r_s which are not optimal but reasonably good. Furthermore, we estimate the number of iterations required by our schemes to reduce the initial error by a preassigned factor ϵ. By estimating bounds on the one-dimensional collocation matrices involved we were able to estimate the r_s's and the required number of iterations for more general than Helmholtz PDE operators. We also obtain analytic expressions for the optimal values and give intervals of convergence of the relaxation parameter ω.

We have implemented our ADISC1 schemes using software components that take full advantage of the tensor product formulation of our iterative methods. Our extensive numerical results confirm the increased efficiency of the methods predicted by our complexity analysis which is verified by our timing results. Furthermore, a careful experimental comparison has been carried out which shows that our ADI schemes outperform well-known methods. We have used our implementation to solve three PDE model problems, namely, a Poisson, a Helmholtz, and a general PDE on a unit cube. All experimental data obtained exhibit good agreement with our theoretical results.

REFERENCES

[1] D. N. Arnold and J. Saraven, *On the asymptotic convergence of spline-collocation methods for partial differential equations*, SIAM J. Numer. Anal., 21 (1984), pp. 459–472.

[2] B. Bialecki, *An alternating direction implicit method for orthogonal spline collocation linear systems*, Numer. Math., 59 (1991), pp. 413–429.

[3] K. Cooper and P. Prenter, *Alternating direction collocation method for separable elliptic partial differential equations*, SIAM J. Numer. Anal., 28 (1991), pp. 711–727.

[4] C. de Boor, *Efficient computer manipulation of tensor products*, ACM Trans. Math. Software, 5 (1979), pp. 173–182.

[5] J. Dongarra, J. Bunch, C. Moler, and G. Stewart, *LINPACK User's Guide*, SIAM, Philadelphia, PA, 1979.

[6] J. Dongarra, J. Du Croz, I. Duff, and S. Hammarling, *A set of level 3 basic linear algebra subprograms*, ACM Trans. Math. Software, 16 (1990), pp. 1–17.

[7] J. Douglas, *Alternating direction methods for three space variables*, Numer. Math., 4 (1962), pp. 41–53.

[8] J. Douglas and H. Rachford, *On the numerical solution of heat conduction problems in two or three space variables*, Trans. Amer. Math. Soc., 82 (1956), pp. 421–438.

[9] W. R. Dyksen, *A tensor product generalized ADI method for elliptic problems on cylindrical domains with holes*, J. Comput. Appl. Math., 16 (1986), pp. 43–58.

[10] W. R. Dyksen, *Tensor product generalized ADI methods for elliptic problems*, SIAM J. Numer. Anal., 24 (1987), pp. 59–76.

[11] W. R. Dyksen, *A tensor product generalized ADI method for the method of planes*, Numer. Methods Partial Differential Equations, 4 (1988), pp. 283–300.

[12] A. Hadjidimos, *On a generalized alternating direction implicit method for solving Laplace's equation*, IMA J. Numer. Anal., 10 (1968), pp. 324–328.

[13] A. Hadjidimos, E. Houstis, J. Rice, and E. Vavalis, *On the Iterative Solution of Line Spline Collocation Schemes for Elliptic PDEs*, Tech. report CSD-TR-020, Purdue University, West Lafayette, IN, 1991.

[14] A. Hadjidimos, E. Houstis, J. Rice, and E. Vavalis, *Iterative line cubic spline collocation methods for elliptic partial differential equations in several dimensions*, SIAM J. Sci. Statist. Comput., 14 (1993), pp. 715–734.

[15] P. Halmos, *Finite Dimensional Vector Spaces*, Van Nostrand, New York, 1958.

[16] E. Houstis, *Collocation methods for linear elliptic problems*, BIT, 18 (1978), pp. 301–310.

[17] E. Houstis, J. Rice, and E. Vavalis, *Spline collocation methods for elliptic partial differential equations*, in Advances in Computer Methods for Partial Differential Equations, R. Vichnevetsky and R. Stepleman, eds., Vol. V, IMACS Rutgers University, New Brunswick, NJ, 1984, pp. 191–194.

[18] E. Houstis, E. Vavalis, and J. Rice, *Convergence of $O(h^4)$ cubic spline collocation methods for elliptic partial differential equations*, SIAM J. Numer. Anal., 25 (1988), pp. 54–74.

[19] E. N. Houstis, W. F. Mitchell, and J. R. Rice, *GENCOL: Collocation on general domains with bicubic Hermite polynomials*, ACM Trans. Math. Software, 11 (1985), pp. 413–415.

[20] E. N. Houstis, W. F. Mitchell, and J. R. Rice, *INTCOL and HERMCOL: Collocation on rectangular domains with bicubic Hermite polynomials*, ACM Trans. Math. Software, 11 (1985), pp. 416–418.

[21] C. L. Lawson, R. J. Hanson, D. R. Kincaid, and F. T. Krogh, *Basic linear algebraic subprogram for Fortran usage*, ACM Trans. Math. Software, 5 (1979), pp. 324–325.

[22] B. K. Lucas, *Error bounds for interpolating cubic splines under various end conditions*, SIAM J. Numer. Anal., 11 (1974), pp. 569–584.

[23] R. Lynch, J. Rice, and D. Thomas, *Tensor product analysis of alternating direction implicit methods*, J. Soc. Industrial Appl. Math., 13 (1965), pp. 995–1006.

[24] A. Mitchell and G. Fairweather, *Improved forms of the alternating direction methods of Douglas, Peaceman, and Rachford for solving parabolic and elliptic equations*, Numer. Math., 6 (1964), pp. 285–292.

[25] C. Pearcy, *On the convergence of alternating direction procedures*, Numer. Math., 4 (1962), pp. 172–176.

[26] P. Percell and M. F. Wheeler, *A C^1 finite element collocation method for elliptic equations*, SIAM J. Numer. Anal., 17 (1980), pp. 605–622.

[27] J. Rice and R. Boisvert, *Solving Elliptic Problems Using ELLPACK*, Springer-Verlag, New York, 1985.

[28] S. Rubin and R. Graves, *Viscous flow solution with a cubic spline approximation*, Comput. & Fluids, 3 (1974), pp. 1–36.

[29] P. Tsompanopoulou and E. Vavalis, *Alternating Direction Implicit-Spline Collocation Methods for Elliptic Partial Differential Equations*, Tech. report 95-3, Department of Mathematics, University of Crete, Heraklion, Greece, 1995.

[30] O. D. Widlund, *On the rate of convergence of an alternating direction implicit method in a noncommutative case*, Math. Comp., 20 (1966), pp. 500–515.

[31] D. Young and L. Ehrlich, *Some numerical studies of iterative methods for solving elliptic difference equations*, in Boundary Problems in Differential Equations, R. Langer, ed., University of Wisconsin Press, Madison, WI, 1960, pp. 143–162.