SIAM J. SCI. COMPUT.
Vol. 19, No. 3, pp. 912–932, May 1998

009

DOMAIN DECOMPOSITION OPERATOR SPLITTINGS FOR THE SOLUTION OF PARABOLIC EQUATIONS*

T. P. MATHEW†, P. L. POLYAKOV†, G. RUSSO‡, AND J. WANG†

Abstract. We study domain decomposition counterparts of the classical alternating direction implicit (ADI) and fractional step (FS) methods for solving the large linear systems arising from the implicit time stepping of parabolic equations. In the classical ADI and FS methods for parabolic equations, the elliptic operator is split along coordinate axes; they yield tridiagonal linear systems whenever a uniform grid is used and when mixed derivative terms are not present in the differential equation. Unlike coordinate-axes-based splittings, we employ domain decomposition splittings based on a partition of unity. Such splittings are applicable to problems on nonuniform meshes and even when mixed derivative terms are present in the differential equation and they require the solution of one problem on each subdomain per time step, without iteration. However, the truncation error in our proposed method deteriorates with smaller overlap amongst the subdomains unless a smaller time step is chosen. Estimates are presented for the asymptotic truncation error, along with computational results comparing the standard Crank–Nicolson method with the proposed method.

Key words. domain decomposition, alternating direction implicit method, fractional step method, operator splitting, parabolic equation, partition of unity

AMS subject classifications. 65N20, 65F10

PII. S1064827595288206

1. Introduction.

The numerical solution of parabolic partial differential equations by *implicit* time stepping procedures requires the solution of large systems of linear equations. These linear systems need only be solved *approximately,* provided that the inexact solutions obtained by using approximate solvers preserve the stability and local truncation error of the original scheme. Though iterative methods with preconditioners are a popular way for solving such linear systems, see [5, 6, 3, 8, 15, 33, 4, 7], in this paper we consider only approximate solvers that do not involve iteration at each time step. Such approximate noniterative solution methods for parabolic equations include the alternating direction implicit (ADI) methods of Peaceman and Rachford [22] and Douglas and Gunn [12], the fractional step methods (FS) of Bagrinovskii and Godunov [1], Yanenko [34], and Strang [26], and also more recent one-iteration domain decomposition solvers of Kuznetsov [16, 17], Meurant [21], Dryja [13], Blum, Lisky, and Rannacher [2], Dawson, Du, and Dupont [9], Laevsky [19, 18] and Vabishchevich [28] and Vabishchevich and Matus [29], and Chen and Lazarov [20].

The method proposed here uses the same framework as the classical ADI and FS methods for solving parabolic equations of the form $u_t + Lu = f$ using an operator *splitting* $L = L_1 + \cdots + L_q$; however, the splittings chosen here are based on domain decomposition, unlike classical splittings along coordinate directions. Furthermore, they are applicable to problems with mixed derivative terms and on nonuniform grids. The basic idea is simple. Given a smooth partition of unity $\{\chi_k\}_{k=1,\ldots,q}$ subordinate to a decomposition of the domain, as described in section 4, an elliptic operator L,

*Received by the editors June 26, 1997; accepted for publication (in revised form) July 12, 1996. http://www.siam.org/journals/sisc/19-3/28820.html

†Department of Mathematics, University of Wyoming, Laramie, WY 82071-3036 (mathew@ledaig.uwyo.edu, polyakov@ledaig.uwyo.edu, junping@schwarz.uwyo.edu).

‡Dipartimento di Matematica, Universita dell'Aquila, Via Vetoio, Loc Coppito, 67010 L'Aquila, Italy (russo@smaq20.univaq.it).

such as the two-dimensional Laplacian, is split as a sum of several "simpler" operators:

$$Lu = -\Delta u = (L_1 + \cdots + L_q)u, \qquad \text{where } L_i u = -\nabla \cdot \chi_i \nabla u.$$

Given a splitting, an approximate solution of the parabolic equation is obtained by solving several "simpler" evolution equations of the form $u_t + L_k u = f_k$; see section 3. By comparison, classical splittings, see Richtmyer and Morton [23], of $L = -\Delta$ are of the form $L_1 = -\frac{\partial^2}{\partial x^2}$ and $L_2 = -\frac{\partial^2}{\partial y^2}$.

Our goal in this paper is to derive truncation error estimates of the proposed methods and to discuss their stability properties. The rest of the paper is outlined as follows. In section 2, we describe a parabolic equation and its discretization. In section 3, we describe the classical ADI and FS methods in matrix language. In section 4, we describe domain decomposition operator splittings using a partition of unity and derive asymptotic truncation error bounds and stability properties for the proposed method. A heuristic comparison is then made between the Crank–Nicolson method and the ADI method with domain decomposition splitting. Finally, in section 5, numerical results are presented.

2. A parabolic equation and its discretization. We consider the following parabolic equation for $u(x, y, t)$ on a domain $\Omega \subset R^2$:

$$\begin{cases} u_t &= \nabla \cdot a(x,y)\nabla u - c(x,y)u + f(x,y,t), & \text{in } \Omega \times [0,T], \\ u(x,y,0) &= u_0(x,y), & \text{in } \Omega, \\ u(x,y,t) &= 0, & \text{on } \partial\Omega \times [0,T], \end{cases}$$

where $a(x, y)$ is a 2×2 symmetric positive definite matrix function with $C^{2q-1}(\Omega)$ entries, $c(x, y) \geq 0$ is in $C^{2q}(\Omega)$, $f(x, y, t)$ is a $C^{2q}(\Omega \times [0, T])$ forcing term, and $u_0(x, y)$ is a $C^{2q}(\Omega)$ initial data. Here q is an integer to be specified later in section 4. With these smoothness assumptions, the exact solution $u(x, y, t)$ will be in $C^{2q}(\Omega \times [0, T])$. We will denote the elliptic operator by $Lu = -\nabla \cdot a(x, y)\nabla u + c(x, y)u$, and the parabolic equation by $u_t + Lu = f$.

2.1. Discrete problem. Let A_h denote the symmetric positive definite matrix obtained by second-order finite difference discretization of the elliptic operator L on a grid on Ω with mesh size h and grid points (x_i, y_j):

$$(A_h U)_{ij} = Lu(x_i, y_j) + O(h^2),$$

where $U_{ij} \approx u(x_i, y_j, t)$ denotes the discrete solution. To discretize in time, we apply an *implicit* scheme such as the backward Euler or Crank–Nicolson scheme, which results in a large sparse linear system to be solved.

For example, the backward Euler method yields

$$\frac{U^{n+1} - U^n}{\tau} = -A_h U^{n+1} + F^{n+1}, \quad \text{or} \quad (I + \tau A_h)\, U^{n+1} = U^n + \tau F^{n+1}, \tag{1}$$

where τ is the time step, $U^n = \{U(x_i, y_j, n\tau)\}$ denotes the vector of discrete unknowns at time $t^n = n\tau$, and $F^n = \{f(x_i, y_j, n\tau)\}$ is the discrete forcing term at time t^n. As is well known, see [23], the above scheme is unconditionally stable in the Euclidean norm $\|.\|$ and convergent with error: $\|u - U_{be}\| = O(\tau) + O\left(h^2\right)$. Thus, for transient problems $\tau = O(h^2)$ provides an optimal choice of time step τ for the backward Euler method.

Similarly, the Crank–Nicolson method yields

$$\frac{U^{n+1}-U^n}{\tau} = -A_h\left(\frac{U^{n+1}+U^n}{2}\right) + \left(\frac{F^{n+1}+F^n}{2}\right), \tag{2}$$

which requires the solution of the linear system

$$\left(I+\frac{\tau}{2}A_h\right)U^{n+1} = \left(I-\frac{\tau}{2}A_h\right)U^n + \tau\left(\frac{F^{n+1}+F^n}{2}\right). \tag{3}$$

The Crank–Nicolson method is also unconditionally stable, see [23], with error: $\|u - U_{cn}\| = O\left(\tau^2\right) + O\left(h^2\right)$ yielding second-order convergence when $\tau = O(h)$.

We note that both of the above discretizations can be written in the form

$$\left(I+\alpha\tau A_h\right)U^{n+1} + BU^n = g_n, \tag{4}$$

for suitable choices of constants α, matrices B, and forcing terms g_n. In particular, the coefficient matrices can be written in the form $I+\alpha\tau A_h$ for $\alpha = 1$ or $\alpha = \frac{1}{2}$, and they are symmetric positive definite. Furthermore, since it is known that the largest eigenvalue of A_h satisfies $\lambda_{max}(A_h) = O(h^{-2})$, it follows that the condition number of $I+\alpha\tau A_h$ is bounded by $1+c\tau h^{-2}$, for some positive constant c independent of τ and h. Thus, for $h^2 \ll \tau \ll 1$ systems (1) and (3) are ill conditioned, yet better conditioned than A_h. Consequently, if iterative methods are used, preconditioners such as multigrid [4, 15] or domain decomposition methods may be needed to reduce the number of iterations; see, for example, [5, 6, 3, 8, 33, 7].

However, as mentioned earlier, system (4) for U^{n+1} needs only to be solved approximately within the local truncation error. In this paper we only seek solvers that provide approximate solutions at the cost of *one iteration per time step*. Such approximate solvers will modify the scheme (4) and alter its stability and truncation error. In order to distinguish this with the modified schemes, we will henceforth refer to scheme (4) as the original scheme. In sections 3 and 4, we discuss *one-step* approximate solvers, and estimate their truncation errors using the framework of the classical ADI and FS methods. We also describe conditions under which the *stability* of the original scheme is preserved.

Given the original discretization (4), we define its local truncation error T_{orig} as follows:

$$T_{orig} = \left(I+\alpha\tau A_h\right)u^{n+1} + Bu^n - g_n, \tag{5}$$

where u is the exact solution of the parabolic equation restricted to the grid points. Discretization (4) is said to be *stable* in a norm $\|\cdot\|$ if the following holds for some numbers c_1, c_2:

$$\|U^{n+1}\| \le (1+c_1\tau)\|U^n\| + c_2\|g_n\|.$$

If c_1 and c_2 are independent of h and τ, then the discretization is said to be *unconditionally stable*, else *conditionally stable*.

3. ADI and FS methods. Given an evolution equation of the form $u_t + Lu = f$, the basic idea in the ADI and FS methods is to approximately solve it using an *operator splitting* $L = L_1 + \cdots + L_q$, where several simpler evolution equations of the form $w_t + L_i w = f_i$ are solved with suitably chosen f_i to provide an approximate solution $w \approx u$ to the original equation.

We will henceforth restrict our discussion to the matrix case, where a discretization in space of $u_t + Lu = f$ yields $U_t + A_h U = F_h$, and a subsequent discretization in time yields system (4). Corresponding to an operator splitting $L = L_1 + \cdots + L_q$, where $q > 1$, we assume a matrix *splitting* of A_h as a sum of symmetric positive semidefinite matrices A_i:

$$A_h = A_1 + \cdots + A_q, \text{ where } A_i \geq 0. \tag{6}$$

It will be assumed that the matrices A_i have a "simpler" structure than A_h, in the sense to be explained below.

In matrix language, given the splitting $\{A_i\}$, both the ADI and FS methods approximately solve systems of the form $(I + \alpha\tau A_h)w = z$ by solving several systems of the form $(I + \alpha\tau A_i)w_i = z_i$; see sections 3.1 and 3.2. Hence, the A_i must be suitably chosen so that the latter systems are easier to solve than the original system.

As an example of splittings, we briefly describe the classical splittings, see [23], of an elliptic operator along the x and y coordinate directions (when no mixed derivative terms are present). Domain decomposition splittings are described in section 4. For classical splittings, the two-dimensional Laplacian $L = -\Delta$ is split as a sum of $L_1 = -\frac{\partial^2}{\partial x^2}$ and $L_2 = -\frac{\partial^2}{\partial y^2}$. In matrix terms

$$A_h = -\Delta_h = -D^h_{xx} - D^h_{yy},$$

where Δ_h denotes the discrete Laplacian and D^h_{xx} and D^h_{yy} denote the discretization of $\frac{\partial^2}{\partial x^2}$ and $\frac{\partial^2}{\partial y^2}$, respectively. Both $A_1 = -D^h_{xx}$ and $A_2 = -D^h_{yy}$ will be tridiagonal matrices after a suitable ordering of the unknowns (if the grid is uniform) and hence $(I + \alpha\tau A_i)$ will also be tridiagonal and solvable in linear time on uniform grids!

Unfortunately, such splittings are not possible if mixed derivative terms are present and tridiagonality can be lost if the grid is nonuniform. By contrast, the domain decomposition splittings described in section 4 are applicable in more general cases, though the resulting linear systems are not tridiagonal.

3.1. ADI method. This method was originally proposed by Peaceman and Rachford [22] and Douglas [11, 10], and was further developed by several authors. We follow here the general formulation described by Douglas and Gunn [12], where additional references may be found. The reader is also referred to [30, 31, 32, 14, 24, 27]. In matrix language, given a splitting $A_h = A_1 + \cdots + A_q$ and an approximate solution $W^n \approx U^n$ to the discrete solution at time t^n, an approximate solution $W^{n+1} \approx U^{n+1}$ of the linear system

$$(I + \alpha\tau A_h)\, U^{n+1} = g^n - BU^n$$

is obtained by solving systems of the form $(I + \alpha\tau A_i)\, w_i = z_i$ as follows.

ADI Algorithm.

1. Given W^n, solve for W_1^{n+1}:

$$(I + \alpha\tau A_1)\, W_1^{n+1} + \sum_{j=2}^{q} \alpha\tau A_j W^n + BW^n = g_n.$$

2. For $i = 2, \ldots, q$, solve for W_i^{n+1}:

$$(I + \alpha\tau A_i)\, W_i^{n+1} - W_{i-1}^{n+1} - \alpha\tau A_i W^n = 0.$$

3. Define the ADI approximate solution at time $t^{n+1} = (n+1)\tau$ to be

$$W^{n+1} = W_q^{n+1}.$$

We note that in order to compute W^{n+1} in the ADI scheme, we need to solve a total of q linear systems, each of the form $I + \alpha\tau A_i$ of size $n \times n$ (the size of A_h). The following stability results for the ADI method can be found in [12].

THEOREM 3.1. *If the matrices A_i are symmetric positive semidefinite for $i = 1, \ldots, q$, if B is symmetric, and if A_i and B commute pairwise, then the ADI scheme is unconditionally stable. In the noncommuting case, if $q = 2$ and if the matrices A_i are symmetric positive semidefinite, and if $\|\cdot\|$ is any norm in which the original scheme* (4) *is stable, then the ADI scheme is stable in the norm $|||u||| \equiv \|(I+\alpha\tau A_2)u\|$. For $q \geq 3$, if the matrices A_i are positive semidefinite, then the ADI scheme can be shown to be conditionally stable.*

For $q \geq 3$ in the noncommuting case, examples are known of positive semidefinite splittings for which the ADI method can loose unconditional stability; see [12]. Nevertheless, the above stability results are a bit pessimistic and *instability is only rarely encountered in practice.*

Next, we describe the truncation error terms for the ADI method, derived in [12]. The magnitude of these truncation terms will be estimated in section 4.3 for domain decomposition splittings.

THEOREM 3.2. *Given the discrete problem*

$$(I + \alpha\tau A_h)\, U^{n+1} + BU^n = g_n,$$

with local truncation error T_{orig} (see (5)*), the ADI solution W^{n+1} solves the following perturbed problem:*

$$(I + \alpha\tau A_h)\, W^{n+1} + \sum_{m=2}^{q} \left(\alpha^m \tau^m \sum_{1 \leq \sigma_1 < \cdots < \sigma_m \leq q} A_{\sigma_1} \cdots A_{\sigma_m} \left(W^{n+1} - W^n \right) \right) + BW^n = g^n, \tag{7}$$

which introduces the following additional terms in the local truncation error:

$$T_{adi} = T_{orig} + \sum_{m=2}^{q} \left(\alpha^m \tau^m \sum_{1 \leq \sigma_1 < \cdots < \sigma_m \leq q} A_{\sigma_1} \cdots A_{\sigma_m} \left(u^{n+1} - u^n \right) \right), \tag{8}$$

where u is the exact solution of the parabolic equation restricted to the grid points.

Proof. The proof is given in [12]; however, for completeness we include it here. From step 2 of the ADI algorithm, we obtain that W_i^{n+1} satisfies

$$(I + \alpha\tau A_i)\, (W_i^{n+1} - W^n) = W_{i-1}^{n+1} - W^n \text{ for } i = 2, \ldots, q.$$

By recursively applying this, we deduce that

$$(I + \alpha\tau A_2) \cdots (I + \alpha\tau A_q)\, (W_q^{n+1} - W^n) = (W_1^{n+1} - W^n). \tag{9}$$

Now, from step 1 of the ADI algorithm, we deduce that

$$\begin{aligned} (I + \alpha\tau A_1)\, (W_1^{n+1} - W^n) &= g_n - BW^n - \textstyle\sum_{j=2}^{q} \alpha\tau A_j W^n - (I + \alpha\tau A_1)\, W^n \\ &= g_n - BW^n - (I + \alpha\tau A_h) W^n. \end{aligned} \tag{10}$$

Multiplying (9) by $(I + \alpha\tau A_1)$ and substituting (10) for the resulting right-hand side, we obtain

$$(I + \alpha\tau A_1)\cdots(I + \alpha\tau A_q)\,(W_q^{n+1} - W^n) = g_n - BW^n - (I + \alpha\tau A_h)W^n. \tag{11}$$

Noting that

$$(I + \alpha\tau A_1)\cdots(I + \alpha\tau A_q) = I + \alpha\tau A_h + \sum_{m=2}^{q} \alpha^m \tau^m \left(\sum_{1\le\sigma_1<\cdots<\sigma_m\le q} A_{\sigma_1}\cdots A_{\sigma_m} \right),$$

(7) follows. ∎

Explicit estimates of these truncation error terms for the case of domain decomposition splittings are given in section 4.3.

3.2. FS methods. These methods were originally developed by Bagrinovskii and Godunov [1] and Yanenko [34] and further developed by Strang [26] and other authors. Given a matrix splitting $A_h = A_1 + \cdots + A_q$, the FS methods approximate $(I + \alpha\tau A_h)$ by a product of several matrices of the form $(I + \alpha\tau A_i)$. For example, the first-order FS method uses the approximation

$$(I + \alpha\tau A_h) = (I + \alpha\tau A_1)\cdots(I + \alpha\tau A_q) + O(\tau^2), \tag{12}$$

which can be formally verified by multiplying all the terms, bearing in mind that the matrices A_i may not commute. Given (12), an approximate solution $W^{n+1} \approx U^{n+1}$ of

$$(I + \alpha\tau A_h)\,U^{n+1} + BU^n = g_n \tag{13}$$

can be obtained by solving

$$(I + \alpha\tau A_1)\cdots(I + \alpha\tau A_q)\,W^{n+1} = g_n - BW^n. \tag{14}$$

We summarize below the first-order *fractional step* algorithm to approximately solve $(I + \alpha\tau A_h)U^{n+1} = g_n - BU^n$ with $W_{fs}^{n+1} \approx U^{n+1}$.

First-Order FS Algorithm.

1. Let $z_1 = g_n - BW^n$ and for $i = 1, \ldots, q$ solve

$$(I + \alpha\tau A_i)w_i = z_i, \text{ and define } z_{i+1} = w_i.$$

2. Let $W_{fs}^{n+1} = w_q$.

Though (14) is a second-order approximation (locally) in τ of (13), it is only first-order accurate globally, due to accumulation of errors; see [23]. The following result concerns the stability of the first-order FS method.

Theorem 3.3. *If A_i are symmetric and positive semidefinite, and if $\|B\| \le 1$ in the Euclidean norm $\|.\|$, then the first-order FS method is stable with*

$$\|W^{n+1}\| \le \|W^n\| + c\|g_n\|$$

for some $c > 0$ and independent of τ and h.

Proof. Since $W^{n+1} = (I+\alpha\tau A_q)^{-1}\cdots(I+\alpha\tau A_1)^{-1}\,(g_n - BW^n)$ and since $\|B\| \le 1$ in the spectral norm, we only need verify that

$$\|\,(I + \alpha\tau A_q)^{-1}\cdots(I + \alpha\tau A_1)^{-1}\,\| \le 1$$

in the spectral norm. But this follows easily: since each of the terms $(I + \alpha\tau A_i)$ are symmetric positive definite with eigenvalues greater than one, the spectral norms of their inverses are each bounded by one. ∎

The following result concerns the local truncation error of the first-order FS method.

THEOREM 3.4. *The first-order FS approximate solution W^{n+1} of*

$$(I + \alpha\tau A_h)U^{n+1} = g_n - BU^n$$

solves

$$\begin{aligned}(I + \alpha\tau A_h)W^{n+1} + \textstyle\sum_{m=2}^{q} \left(\alpha^m\tau^m \sum_{1\le\sigma_1<\cdots<\sigma_m\le q} A_{\sigma_1}\cdots A_{\sigma_m}\right) W^{n+1} \\ = g_n - BW^n.\end{aligned} \tag{15}$$

The local truncation error T_{fs} of the first-order FS method has the following terms in addition to the terms in the local truncation error T_{orig} when exact solvers are used:

$$T_{fs} = T_{orig} + \sum_{m=2}^{q} \left(\alpha^m\tau^m \sum_{1\le\sigma_1<\cdots<\sigma_m\le q} A_{\sigma_1}\cdots A_{\sigma_m}\right) u^{n+1}. \tag{16}$$

Here u^{n+1} is the exact solution of the parabolic equation at time $(n+1)\tau$ restricted to the grid points.

Proof. Since the product

$$(I + \alpha\tau A_1)\cdots(I + \alpha\tau A_q) = I + \alpha\tau A_h + \sum_{m=2}^{q} \left(\alpha^m\tau^m \sum_{1\le\sigma_1<\cdots<\sigma_m\le q} A_{\sigma_1}\cdots A_{\sigma_m}\right),$$

the proof immediately follows by replacing $(I + \alpha\tau A_h)$ by $(I + \alpha\tau A_1)\cdots(I + \alpha\tau A_q)$ in the original discretization. ∎

For a given splitting $\{A_i\}$, the above local truncation error terms are $O(\tau^2)$ (though the constant in the leading-order term τ^2 can vary for different choices of splittings). However, the global error will be $O(\tau)$, see [23], due to accumulation of errors, and consequently, the above first-order FS method is suitable only for globally first-order schemes such as backward Euler but not for globally second-order discretizations such as Crank–Nicolson.

For globally second-order accurate schemes such as Crank–Nicolson, second-order fractional step approximations can be obtained by using *Strang splitting* [26]. We outline the main idea for a splitting involving two matrices: $A_h = A_1 + A_2$. The aim is to approximate $e^{-\alpha\tau A_h}$ by a locally third-order accurate approximation as follows:

$$e^{-\tau A_h} = e^{-\frac{\tau}{2}A_1}e^{-\tau A_2}e^{-\frac{\tau}{2}A_1} + O(\tau^3). \tag{17}$$

Then, as in the Crank–Nicolson method, a locally third-order approximation in τ of the terms $e^{-\alpha\tau A_i}$ can be used to replace each of the exponentials

$$e^{-\alpha\tau A_i} = \left(I + \frac{\alpha\tau}{2}A_i\right)^{-1}\left(I - \frac{\alpha\tau}{2}A_i\right) + O(\tau^3).$$

Applying this to each of the terms in (17) yields the implementation of *Strang splitting*. We state without proof the following.

THEOREM 3.5. *Strang splitting provides unconditionally stable approximations when A_i are symmetric positive semidefinite: $A_i \geq 0$. For any fixed choice of positive semidefinite matrices A_1 and A_2, the truncation error for Strang splitting is third-order accurate locally in τ, and globally second-order accurate in τ.*

Since *Strang splitting* provides a globally second-order accurate approximation in time, it can be applied to inexactly solve the Crank–Nicolson scheme. However, it is more expensive and complicated to generalize when there are more than two matrices in the splitting, and it requires more linear systems to be solved than the first-order FS method. In the applications we consider, the ADI methods which were described earlier are preferable and generate smaller truncation errors, though they are not guaranteed to be unconditionally stable without additional assumptions.

4. Domain-decomposition-based operator splittings. In this section, we describe domain-decomposition-based splittings

$$A_h = A_1 + \cdots + A_q$$

of the discretization A_h of L such that $(I + \alpha\tau A_i)w_i = z_i$ can be solved at the cost of solving a problem on a subdomain. For such splittings, we provide in section 4.3 an explicit estimate on how the truncation error depends on the overlap β amongst the subdomains. For related algorithms, we refer the reader to Dryja [13], Laevsky [19, 18], Vabishchevich [28] and Vabishchevich and Matus [29], Kuznetsov [17, 16], Blum, Lisky, and Rannacher [2], Meurant [21], Dawson, Du, and Dupont [9], and Chen and Lazarov [20].

As noted in the introduction, given a smooth partition of unity $\{\chi_k(x,y)\}_{k=1,\ldots,q}$ subordinate to a collection of open subdomains $\{\Omega_k^*\}_{k=1,\ldots,q}$ which cover Ω, we split an elliptic operator L as a sum of several "simpler" operators as follows:

$$\begin{aligned} Lu &= -\nabla \cdot a(x,y)\nabla u + c(x,y)u \\ &= \textstyle\sum_{k=1}^{q} -\nabla \cdot (\chi_k(x,y)a(x,y)\nabla u) + \chi_k(x,y)c(x,y)u \\ &= L_1 u + \cdots + L_q u, \end{aligned} \tag{18}$$

where $L_k u = -\nabla \cdot (\chi_k(x,y)a(x,y)\nabla u) + \chi_k(x,y)c(x,y)u$. In what follows, we will denote the coefficients of L_k by $a_k(x,y) = \chi_k(x,y)a(x,y)$ and $c_k(x,y) = \chi_k(x,y)c(x,y)$. For numerical purposes, we could replace a smooth partition of unity by one which is *piecewise smooth*. In such a case, it may be necessary to define the operators L_k weakly using a variational approach. The matrices A_k are obtained by discretization of L_k.

4.1. Partition of unity. Given an overlapping collection of *open* subregions $\{\Omega_k^*\}_{k=1,\ldots,q}$ whose union contains Ω, a partition of unity subordinate to this covering is a family of q nonnegative, $C^\infty(\Omega)$ smooth functions $\{\chi_k(x,y)\}_{k=1,\ldots,q}$ whose sum equals one and such that $\chi_k(x,y)$ vanishes outside $\overline{\Omega_k^*}$:

$$0 \leq \chi_k(x,y) \leq 1, \quad \chi_1(x,y) + \cdots + \chi_q(x,y) = 1, \quad \operatorname{supp}(\chi_k(x,y)) \subset \overline{\Omega}_k^*.$$

For numerical purposes, we will consider $\chi_k(x,y)$ which are not necessarily $C^\infty(\Omega)$ but are continuous and piecewise smooth. We will refer to such a partition of unity as *piecewise smooth*. For more information on partitions of unity, see Sternberg [25].

Construction of piecewise smooth partitions of unity. Given a collection of *overlapping* subdomains $\{\Omega_k^*\}_{k=1,\ldots,q}$ which form a covering of Ω, we wish to describe how a *piecewise smooth* partition of unity can be constructed. However, first we describe how to construct an overlapping covering $\{\Omega_k^*\}_{k=1,\ldots,q}$.

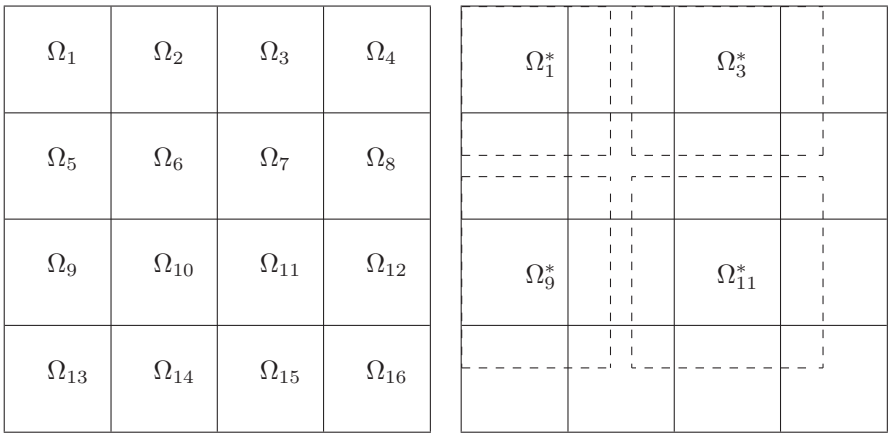

FIG. 1. 16 *nonoverlapping subdomains* Ω_i*; enlarged subdomains* Ω_k^* *of color* 1.

1. Let $\{\Omega_k\}_{k=1,\ldots,q}$ denote a *nonoverlapping* collection of subregions of Ω, each of width H, with $\overline{\Omega} = \cup_{i=1}^q \overline{\Omega}_i$ and $\Omega_i \cap \Omega_j = \emptyset$ for $i \neq j$. For example, the subdomains Ω_k can be chosen to be the coarse elements in a coarse grid triangulation of Ω with mesh size H; see Figure 1.
2. Next, enlarge each subdomain Ω_k to Ω_k^* to contain all points in Ω within a distance β of Ω_k, where $\beta > 0$; see Figure 1.

Then $\{\Omega_k^*\}$ will form an overlapping covering of Ω: $\Omega = \cup_{k=1}^q \Omega_k^*$. We note here that the subdomains $\{\Omega_k^*\}$ can often be grouped into a small number of *colors* so that any two subdomains of the same color are disjoint, see Figure 1, for a coloring of 16 overlapping subdomains into *four colors* ($q = 4$, only color 1 is indicated).

Once $\{\Omega_k^*\}_{k=1,\cdots,q}$ are formed, a *piecewise smooth* partition of unity $\{\chi_k\}_{k=1,\ldots,q}$ can be constructed as follows:

1. Inside Ω_k^*, let $\omega_k(x,y)$ denote the distance of (x,y) to the boundary $\partial\Omega_k^*$:

$$\omega_k(x,y) = \begin{cases} \text{dist}((x,y), \partial\Omega_k^*), & (x,y) \in \Omega_k^*, \\ 0, & (x,y) \notin \Omega_k^*. \end{cases}$$

   Note that $0 \leq \omega_k(x,y)$ is a continuous function with support in Ω_k^*.
2. Then define $\chi_k(x,y)$ by normalizing the $\omega_k(x,y)$ in order that their sum equals one:

$$\chi_k(x,y) \equiv \frac{\omega_k(x,y)}{\sum_{j=1}^q \omega_j(x,y)} \qquad \text{for } k = 1, \ldots, q.$$

The above constructed $\{\chi_k(x,y)\}$ will be continuous and piecewise smooth (with discontinuous derivatives across $\cup_k \partial\Omega_k^*$). To obtain $C^\infty(\Omega)$ smooth partitions of unity, we refer the reader to [25].

Before we proceed further, we define the Hölder norms for integer $k \geq 0$ as follows:

$$\|u\|_{C^k(\Omega)} \equiv \sup_{|\alpha| \leq k} \max_{\overline{\Omega}} |\partial^\alpha u|,$$

where we have used the multi-index notation

$$\partial^\alpha = \frac{\partial^{|\alpha|}}{\partial x^{\alpha_1} \partial y^{\alpha_2}}, \qquad \alpha = (\alpha_1, \alpha_2).$$

For *smooth* partitions of unity with overlap β, the following result holds.

PROPOSITION 4.1. *Given a suitable collection of overlapping subregions, $\{\Omega_k^*\}_{k=1,\ldots,q}$ with overlap β, there exists a smooth partition of unity $\{\chi_k(x,y)\}_{k=1,\ldots,q}$ satisfying*

$$\|\chi_k\|_{C^m(\Omega)} \le c_m \beta^{-m} \tag{19}$$

in the Hölder $C^m(\Omega)$ norm, where $c_m > 0$ is a constant independent of k and β but dependent on the geometry of the region.

Proof. We sketch the main ideas.

1. First, map Ω to an enlarged region $\tilde{\Omega}$ using the change of variables $(\tilde{x}, \tilde{y}) = (x/\beta, y/\beta)$. This mapping enlarges Ω, and consequently all its subregions $\{\Omega_k^*\}$, by a factor of β^{-1}. Denote by $\{\tilde{\Omega}_k^*\}_{k=1,\ldots,q}$ the corresponding covering of $\tilde{\Omega}$. These subregions will have an overlap parameter of size $O(1)$.
2. Next, construct a $C^\infty(\tilde{\Omega})$ partition of unity $\{\tilde{\chi}_k\}_{k=1,\ldots,q}$ on $\tilde{\Omega}$, subordinate to the covering $\{\tilde{\Omega}_k^*\}_{k=1,\ldots,q}$; see for instance Sternberg [25]. This partition of unity on $\tilde{\Omega}$ defines a corresponding $C^\infty(\Omega)$ partition of unity on Ω as follows:

$$\chi_k(x,y) \equiv \tilde{\chi}_k(\tilde{x}, \tilde{y}) \qquad \text{for } k = 1, \ldots, q.$$

3. By repeated applications of the chain rule, it follows that

$$\frac{\partial^{|\alpha|} \chi_k}{\partial x^{\alpha_1} \partial y^{\alpha_2}} = \beta^{-|\alpha|} \frac{\partial^{|\alpha|} \tilde{\chi}_k}{\partial \tilde{x}^{\alpha_1} \partial \tilde{y}^{\alpha_2}}.$$

Using the definition of the Hölder norm it follows that

$$\|\chi_k\|_{C^m(\Omega)} \le \beta^{-m} \|\tilde{\chi}_k\|_{C^m(\tilde{\Omega})} = c_m \beta^{-m},$$

where c_m is defined by $c_m \equiv \|\tilde{\chi}_k\|_{C^m(\tilde{\Omega})}$.

We omit further details. ∎

For numerical implementation, smooth partitions of unity can be replaced by piecewise smooth ones based on the distance functions $w_k(x,y)$. See section 5 for an example of an alternative partition of unity for rectangular subdomains.

4.2. Domain decomposed matrix splittings. As mentioned before, given a partition of unity $\{\chi_k\}_{k=1,\ldots,q}$ as in section 4.1, a domain decomposition *splitting* of the operator $Lu = -\nabla \cdot a(x,y) \nabla u - c(x,y)u$ is obtained as $L_k u = -\nabla \cdot (a_k(x,y) \nabla u) + c_k(x,y)u(x,y)$. By construction, $\sum_{k=1}^q L_k = L$, and each L_k are self-adjoint. A_k are then defined as suitable discretizations of the differential operators L_k, and so $A_h = \sum_{k=1}^q A_k$. We note that if $c(x) \ge 0$, then each L_k and A_k will be positive semidefinite. The A_k constructed above can then be used in the ADI and FS methods described in section 3, where the solution of $(I + \alpha\tau A_k)w = z$ corresponds to solving a problem on subregion Ω_k^*. Additionally, by *coloring*, problems on disjoint subregions of the same color can be solved in parallel.

4.3. Truncation errors of the ADI and FS methods for domain decomposition splittings. As described in sections 3.1 and 3.2, when the ADI and FS methods are used to approximately solve a discretization

$$(I + \alpha\tau A_h)\, U^{n+1} = g_n - BU^n, \tag{20}$$

the resulting approximate solutions W_{adi}^{n+1} and W_{fs}^{n+1} solve the *perturbed* problems (7) and (15), respectively. These perturbed problems correspond to new discretizations of the original parabolic equation $u_t + Lu = f$ with additional truncation terms of the form $\tau^m A_{\sigma_1} \cdots A_{\sigma_m}(W^{n+1} - W^n)$ and $\tau^m A_{\sigma_1} \cdots A_{\sigma_m} W^{n+1}$ for the ADI and FS methods, respectively. Our goal in this section is to estimate the magnitude of these additional terms for the case of domain decomposition splittings.

As is well known, see for instance Richtmyer and Morton [23], once the local truncation errors T_{adi} and T_{fs} are estimated, the global errors $E_{adi}(t) = \|u(t) - W_{adi}(t)\|$ and $E_{fs}(t) = \|u(t) - W_{fs}(t)\|$ will follow by Lax's convergence theorem, if the schemes are stable. We note, however, that the global error will be larger than the local truncation errors by a factor of τ^{-1} due to the accumulation of truncation errors.

Since the local truncation errors T_{orig} for the case of backward Euler or Crank–Nicolson with exact solvers are known, our estimates will focus on the new truncation terms $A_{\sigma_1} \cdots A_{\sigma_m} W$ and $A_{\sigma_1} \cdots A_{\sigma_m}(W^{n+1} - W^n)$. For simplicity, we will assume that $c(x, y) = 0$ and that $a(x, y)$ is a smooth scalar function. The analysis for the cases when $c(x, y) \geq 0$ and for smooth matrix functions $a(x, y)$ is similar. Our analysis will require the following smoothness assumptions for the functions $a(x, y)$, $W(x, y, t)$, and $\{\chi_k(x, y)\}$; $a(x, y)$ should be in $C^{2q-1}(\Omega)$, $W(x, y, .)$ should be in $C_0^{2q}(\Omega)$, and $\{\chi_k(x, y)\}$ should be in $C^{2q-1}(\Omega)$. With some abuse of notation, W will also refer to $W = \{W(x_i, y_j, .)\}$, the vector of nodal values of $W(x, y, .)$ on the grid points of Ω.

Our estimates will be based on two simple observations, the details of which will be provided subsequently.

1. The finite difference approximations $A_{\sigma_1} \cdots A_{\sigma_m} W$ (which occurs in T_{fs}) and $A_{\sigma_1} \cdots A_{\sigma_m}(W^{n+1} - W^n)$ (which occurs in T_{adi}) can be shown to be "similar" to their continuous counterparts $L_{\sigma_1} \cdots L_{\sigma_m} W$ and $L_{\sigma_1} \cdots L_{\sigma_m}(W^{n+1} - W^n)$, respectively. This result is obtained by repeated applications of the integral form of the mean value theorem to the difference quotients.
2. Using step 1, the expressions $A_{\sigma_1} \cdots A_{\sigma_m} W$ and $A_{\sigma_1} \cdots A_{\sigma_m}(W^{n+1} - W^n)$ can be estimated in terms of the maximum values of the derivatives of W and the derivatives of the coefficients $\{a_k(x, y)\}$. The derivatives of $\{a_k(x, y)\}$ can be estimated in terms of the derivatives of $a(x, y)$ and of the partition of unity functions $\{\chi_k(x, y)\}$. Using the bounds (19) for the partition of unity, we can then obtain that

$$\left|(A_{\sigma_1} \cdots A_{\sigma_m} W)_{ij}\right| \leq C(a)\beta^{-2m+1}\|W\|_{C^{2m}(\Omega)},$$

where $C(a)$ is a positive constant that depends on the coefficients $a(x, y)$ and its derivatives. Similar estimates will be valid for $A_{\sigma_1} \cdots A_{\sigma_m}\left(W^{n+1} - W^n\right)$ but with an additional factor of τ due to the difference in time $W^{n+1} - W^n$.

Before we proceed, we introduce some notation. In addition to multi-index derivatives, $\partial^\alpha = \frac{\partial^{|\alpha|}}{\partial x^{\alpha_1} \partial y^{\alpha_2}}$, we will use multi-index integrals

$$\mathcal{I}_h^x F(x_i, y_j) = \frac{1}{h} \int_{x_i}^{x_i+h} F(\xi, y_j) d\xi.$$

Iterated integrals are defined as follows:

$$\mathcal{I}_h^{xx} F(x_i, y_j) = \frac{1}{h^2} \int_{x_i - h}^{x_i} \int_{\xi}^{\xi + h} F(\eta, y_j) d\eta d\xi.$$

We also define $\mathcal{I}^x_{h,\frac{1}{2}}$ as

$$\mathcal{I}^x_{h,\frac{1}{2}} F(x_i, y_j) = \frac{1}{h} \int_{x_i - \frac{h}{2}}^{x_i + \frac{h}{2}} F(\xi, y_j) d\xi.$$

Similar integrals are defined in y.

We now outline the details of step 1. By repeatedly applying the integral form of the mean value theorem (or the fundamental theorem of calculus), one can express all the difference quotients in $A_{\sigma_1} \cdots A_{\sigma_m} W$ in terms of integrals of derivatives of W and integrals of the derivatives of the coefficients $a_{\sigma_k}(x, y)$. We have the following result.

LEMMA 4.2. *Let $W \in C_0^{2m}(\Omega)$ and let the expansion of $L_{\sigma_1} \cdots L_{\sigma_m} W$ be of the form*

$$L_{\sigma_1} \cdots L_{\sigma_m} W = \sum_{\substack{(\alpha_1, \ldots, \alpha_{m+1}) \\ |\alpha_1| \le 1 \\ |\alpha_1| + \cdots + |\alpha_{m+1}| = 2m}} C_{\alpha_1, \ldots, \alpha_{m+1}} \left(\partial^{\alpha_1} a_{\sigma_1}\right) \cdots \left(\partial^{\alpha_m} a_{\sigma_m}\right) \left(\partial^{\alpha_{m+1}} W\right),$$

where each $C_{\alpha_1, \ldots, \alpha_{m+1}} = 0$ or 1. Then, for $W(x, y)$ the product $A_{\sigma_1} \cdots A_{\sigma_m} W$ satisfies

$$\begin{aligned} &(A_{\sigma_1} \cdots A_{\sigma_m} W)_{ij} \\ &= \sum_{\substack{(\alpha_1, \ldots, \alpha_{m+1}) \\ |\alpha_1| \le 1 \\ |\alpha_1| + \cdots + |\alpha_{m+1}| = 2m}} C_{\alpha_1, \ldots, \alpha_{m+1}} \mathcal{I}_h^* \cdots \mathcal{I}_h^* \left(\mathcal{I}_h^* \partial^{\alpha_1} a_{\sigma_1}\right) \cdots \left(\mathcal{I}_h^* \partial^{\alpha_m} a_{\sigma_m}\right) \left(\mathcal{I}_h^* \partial^{\alpha_{m+1}} W\right), \end{aligned}$$

where each of the $\mathcal{I}_h^*$ terms (with at most $2m$ such terms) results from a suitable choice of single or multiple integrals $\mathcal{I}_h$.

Proof. The proof will follow by induction on m. For $m = 1$, the standard five-point finite difference approximation $(A_k W)_{i,j}$ of $L_k W(x_i, y_j)$

$$\begin{aligned} (A_k W)_{i,j} \quad = \quad & - \frac{a_k(x_i + \frac{h}{2}, y_j)\left(\frac{W_{i+1,j} - W_{i,j}}{h}\right) - a_k(x_i - \frac{h}{2}, y_j)\left(\frac{W_{i,j} - W_{i-1,j}}{h}\right)}{h} \\ & - \frac{a_k(x_i, y_j + \frac{h}{2})\left(\frac{W_{i,j+1} - W_{i,j}}{h}\right) - a_k(x_i, y_j - \frac{h}{2})\left(\frac{W_{i,j} - W_{i,j-1}}{h}\right)}{h}, \end{aligned}$$

satisfies

$$\begin{aligned} (A_k W)_{i,j} \quad = \quad & - \left(\tfrac{1}{h} \int_{x_i - \frac{h}{2}}^{x_i + \frac{h}{2}} \tfrac{\partial a_k}{\partial x}(\xi, y_j) d\xi\right) \left(\tfrac{1}{h} \int_{x_i}^{x_i + h} \tfrac{\partial W}{\partial x}(\xi, y_j) d\xi\right) \\ & - a_k(x_i - \tfrac{h}{2}, y_j) \left(\tfrac{1}{h^2} \int_{x_i - h}^{x_i} \int_{\xi}^{\xi + h} \tfrac{\partial^2 W}{\partial x^2}(\eta, y_j) d\eta d\xi\right) \\ & - \left(\tfrac{1}{h} \int_{y_j - \frac{h}{2}}^{y_j + \frac{h}{2}} \tfrac{\partial a_k}{\partial y}(x_i, \xi) d\xi\right) \left(\tfrac{1}{h} \int_{y_j}^{y_j + h} \tfrac{\partial W}{\partial y}(x_i, \xi) d\xi\right) \\ & - a_k(x_i, y_j - \tfrac{h}{2}) \left(\tfrac{1}{h^2} \int_{y_j - h}^{y_j} \int_{\xi}^{\xi + h} \tfrac{\partial^2 W}{\partial y^2}(x_i, \eta) d\eta d\xi\right). \end{aligned} \tag{21}$$

This follows trivially by repeatedly applying the fundamental theorem of calculus to the difference quotients in the finite difference discretization. For instance, $\frac{1}{h}(W_{i+1,j}$

$-W_{i,j}) = \frac{1}{h}\int_{x_i}^{x_{i+1}} \frac{\partial W}{\partial x}(\xi, y_j)d\xi$, and similarly $\frac{1}{h}(W_{i,j+1} - W_{i,j}) = \frac{1}{h}\int_{y_j}^{y_{j+1}} \frac{\partial W}{\partial y}(x_i, \xi)d\xi$. Next, we subtract and add the terms $a_k(x_i - \frac{h}{2}, y_j)\frac{1}{h^2}\int_{x_i}^{x_{i+1}} \frac{\partial W}{\partial x}(\xi, y_j)d\xi$ and $a_k(x_i, y_j + \frac{h}{2})\frac{1}{h^2}\int_{y_j}^{y_{j+1}} \frac{\partial W}{\partial y}(x_i, \xi)d\xi$ to the resulting terms in the discretization and apply the fundamental theorem of calculus again. Result (21) follows.

We can restate equation (21) using the multi-index derivative and integral notation as

$$
\begin{aligned}
(A_k W)_{i,j} \quad = \quad & -\left(\mathcal{I}^x_{h,\frac{1}{2}}\partial_x a_k\right)\left(\mathcal{I}^x_h \partial_x W\right) - a_k(x_i - \tfrac{h}{2}, y_j)\left(\mathcal{I}^{xx}_h \partial^2_x W\right) \\
& -\left(\mathcal{I}^y_{h,\frac{1}{2}}\partial_y a_k\right)\left(\mathcal{I}^y_h \partial_y W\right) - a_k(x_i, y_j - \tfrac{h}{2})\left(\mathcal{I}^{yy}_h \partial^2_y W\right).
\end{aligned} \tag{22}
$$

But this is just an expression for $L_k W$ with multi-index integrals applied to its terms. Consequently, our proof for the case $m = 1$ is complete.

For $m \geq 2$, we use the induction hypothesis and assume that the result holds true for $m - 1$. We will then show that the result holds for m. To do this, define $V = A_{\sigma_2} \cdots A_{\sigma_m} W$. Then, by the induction hypothesis

$$
V = \sum_{\substack{(\alpha_2,\ldots,\alpha_{m+1}) \\ |\alpha_2|\leq 1 \\ |\alpha_1|+\cdots+|\alpha_{m+1}|=2m-2}} C_{\alpha_2,\ldots,\alpha_{m+1}} \mathcal{I}_h \cdots \mathcal{I}_h \left(\mathcal{I}_h \partial^{\alpha_2} a_{\sigma_2}\right) \cdots \left(\mathcal{I}_h \partial^{\alpha_m} a_{\sigma_m}\right)\left(\mathcal{I}_h \partial^{\alpha_{m+1}} W\right).
$$

Now, $A_{\sigma_1} V$ can be calculated by using (22) which involves first- and second-order partial derivatives of $V(x, y)$ locally.

The resulting expression can be simplified by the following observation. The partial derivative operators ∂^α "commute" with all the integral operators $\mathcal{I}^x_h$, $\mathcal{I}^y_h$, etc. The commutativity of ∂_x and $\mathcal{I}^y_h$ follows from the differentiation rule for integrals depending on a parameter. The commutativity of ∂_x and $\mathcal{I}^x_h$ can be seen by the following example:

$$
\partial_x \mathcal{I}^x_h F(x, y_j) = \partial_x \left(\frac{1}{h}\int_x^{x+h} F(\xi, y_j)d\xi\right) = \frac{1}{h}\int_x^{x+h} \partial_\xi F(\xi, y_j)d\xi = \mathcal{I}^x_h \partial_x F(x, y_j).
$$

When higher-order partial derivatives and iterated integrals are involved, they can be handled by repeated applications of the above. Consequently, all the partial derivatives can be moved inside the integrals.

When all the partial derivatives are moved inside the integrals, we obtain an expression similar to $L_{\sigma_1} \cdots L_{\sigma_m} W$ but containing integrals of the terms. This gives the desired result for m. ∎

This completes the sketch of the proof of step 1. We obtain step 2 immediately from Lemma 4.2 by applying the mean value property of integrals.

LEMMA 4.3. *The discretization* $(A_{\sigma_1} \cdots A_{\sigma_m} W)_{ij}$ *satisfies*

$$
|(A_{\sigma_1} \cdots A_{\sigma_m} W)_{ij}| \leq \sum_{\substack{(\alpha_1,\ldots,\alpha_{m+1}) \\ |\alpha_1|\leq 1 \\ |\alpha_1|+\cdots+|\alpha_{m+1}|=2m}} C_{\alpha_1,\ldots,\alpha_{m+1}} \|a_{\sigma_1}\|_{C^{\alpha_1}} \cdots \|a_{\sigma_m}\|_{C^{\alpha_m}} \|W\|_{C^{\alpha_{m+1}}},
$$

where $C_{\alpha_1,\ldots,\alpha_{m+1}} = 0$ *or* 1.

Proof. The mean value property of integrals yields

$$
\left|\frac{1}{h}\int_{x_i}^{x_i+h} F(\xi, y_j)d\xi\right| \leq \max_{\xi\in[x_i, x_i+h]} |F(\xi, y_j)|.
$$

Applying this to all the integrals $\mathcal{I}_h$ in Lemma 4.2, we obtain that

$$|(A_{\sigma_1}\cdots A_{\sigma_m}W)_{ij}|$$

$$\leq \sum_{\substack{(\alpha_1,\ldots,\alpha_{m+1})\\ |\alpha_1|\leq 1\\ |\alpha_1|+\cdots+|\alpha_{m+1}|=2m}} C_{\alpha_1,\ldots,\alpha_{m+1}}\|\partial^{\alpha_1}a_{\sigma_1}\|_{C^0(\Omega)}\cdots\|\partial^{\alpha_m}a_{\sigma_m}\|_{C^0(\Omega)}\|\partial^{\alpha_{m+1}}W_{\sigma_m}\|_{C^0(\Omega)}$$

$$\leq \sum_{\substack{(\alpha_1,\ldots,\alpha_{m+1})\\ |\alpha_1|\leq 1\\ |\alpha_1|+\cdots+|\alpha_{m+1}|=2m}} C_{\alpha_1,\ldots,\alpha_{m+1}}\|a_{\sigma_1}\|_{C^{|\alpha_1|}(\Omega)}\cdots\|a_{\sigma_m}\|_{C^{|\alpha_m|}(\Omega)}\|W_{\sigma_m}\|_{C^{|\alpha_{m+1}|}(\Omega)}.$$

This gives us the desired result. □

Using Leibnitz's rule for partial differentiation of $a_k(x,y)=\chi_k(x,y)a_k(x,y)$ and applying bounds (19) for the partition of unity, we deduce that

$$\begin{aligned}\|a_k(x,y)\|_{C^p(\Omega)} &\leq C(a)\|a\|_{C^p(\Omega)}\|\chi_k\|_{C^p(\Omega)}\\ &\leq C(a)\beta^{-p}.\end{aligned}$$

Here $C(a)$ is a constant that depends only on the coefficients $a(x,y)$ and some of its derivatives.

By substituting these results in Lemma 4.3, we obtain the following.

COROLLARY 4.4. *If $W\in C_0^{2m}(\Omega)$, the discretization $(A_{\sigma_1}\cdots A_{\sigma_m}W)_{ij}$ satisfies*

$$|(A_{\sigma_1}\cdots A_{\sigma_m}W)_{ij}|\leq C(a)\beta^{-2m+1}\|W\|_{C^{2m}(\Omega)}.$$

Similarly, the discretization $\left(A_{\sigma_1}\cdots A_{\sigma_m}(W^{n+1}-W^n)\right)_{ij}$ satisfies

$$|\left(A_{\sigma_1}\cdots A_{\sigma_m}(W^{n+1}-W^n)\right)_{ij}|\leq C(a)\tau\beta^{-2m+1}\sup_{[0,t_f]}\|\partial_t W\|_{C^{2m}(\Omega)}.$$

This completes the details of step 2.

We are finally able to estimate the local truncation errors and the global errors of the ADI method with domain decomposition splitting.

THEOREM 4.5. *Let the exact solution $u(x,y,t)$ of the parabolic equation be in $C_0^{2q}(\Omega)$ for each $t\in[0,t_f]$ and in $C^1([0,t_f])$ for each (x,y). Then the local truncation error T_{adi} of the ADI method with domain decomposition splitting satisfies*

$$\|T_{adi}\|_{L^2(\Omega)}\leq\|T_{orig}\|_{L^2(\Omega)}+C(a)\tau^3\left(\frac{1}{\beta^3}+\frac{\tau}{\beta^5}+\cdots+\frac{\tau^{q-2}}{\beta^{2q-1}}\right)\sup_{[0,t_f]}\|\partial_t u\|_{C^{2q}(\Omega)},$$

where T_{orig} is the local truncation error of the original scheme (4) when exact solvers are used. Whenever the ADI scheme is stable, the error $E_{adi}(t)=u(t)-W_{adi}(t)$ satisfies

(23)
$$\|E_{adi}(t)\|_{L^2(\Omega)}\leq\|E_{orig}(t)\|_{L^2(\Omega)}+C(a)\tau^2\left(\frac{1}{\beta^3}+\frac{\tau}{\beta^5}+\cdots+\frac{\tau^{q-2}}{\beta^{2q-1}}\right)\sup_{[0,t_f]}\|\partial_t u\|_{C^{2q}(\Omega)}$$

for $t\in[0,t_f]$.

Proof. By (8), the local truncation error satisfies

$$T_{adi} = T_{orig} + \sum_{m=2}^{q} \left(\alpha^m \tau^m \sum_{1 \le \sigma_1 < \cdots < \sigma_m \le q} A_{\sigma_1} \cdots A_{\sigma_m} \left(u^{n+1} - u^n \right) \right).$$

Considering the Euclidean norm $\| \cdot \|$ of the above expression, and estimating the additional terms using Corollary 4.4, we obtain

$$\begin{aligned} \|T_{adi}\| &\le \|T_{orig}\| + \sqrt{n} C(a) \textstyle\sum_{m=2}^{q} \tau^{m+1} \beta^{-2m+1} \sup_{[0,t_f]} \|\partial_t u\|_{C^{2m}(\Omega)} \\ &\le \|T_{orig}\| + \sqrt{n} C(a) \tau^3 \left(\tfrac{1}{\beta^3} + \tfrac{\tau}{\beta^5} + \cdots + \tfrac{\tau^{q-2}}{\beta^{2q-1}} \right) \sup_{[0,t_f]} \|\partial_t u\|_{C^{2q}(\Omega)}, \end{aligned}$$

where n is the number of grid points in Ω. Multiplying the entire expression by h and using that $h\| \cdot \|$ is equivalent to $\| \cdot \|_{L^2(\Omega)}$, we can replace $h\|T_{adi}\|$ and $h\|T_{orig}\|$ by $\|T_{adi}\|_{L^2(\Omega)}$ and $\|T_{orig}\|_{L^2(\Omega)}$, respectively. The desired estimate for the local truncation error then follows by noting that $\sqrt{n}h$ is proportional to the square root of the area of Ω, which can be absorbed in $C(a)$.

The error follows from the local truncation error by an application of the Lax convergence theorem. ∎

Remark 1. Each of the terms $(\partial^{\alpha_1} a_{\sigma_1}) \cdots (\partial^{\alpha_m} a_{\sigma_m}) (\partial^{\alpha_{m+1}} W)$ will be zero in most of Ω, more specifically, outside $\cap_{k=1}^{m} \Omega^*_{\sigma_k}$, and our proof has not taken this into account. This is due to the compact support of the coefficients $a_{\sigma_k}(x, y)$ in $\Omega^*_{\sigma_k}$. However, we expect that an estimate taking this into account will be much more complicated. Consequently, our theoretical estimate for the truncation errors are pessimistic when compared with actual numerical results from section 5.

Remark 2. For a fixed choice of subdomains $\{\Omega^*_k\}$ and partition of unity $\{\chi_k\}$, the overlap β is fixed. Consequently, the above truncation error bounds for the ADI method with domain decomposition splitting is asymptotically second order in time, i.e., as $\tau \to 0$

$$\|u(t) - W_{adi}(t)\|_{L^2(\Omega)} \le C(a) \left(h^2 + \tau^2 \left(1 + \frac{1}{\beta^3} + \cdots + \frac{\tau^{q-2}}{\beta^{2q-1}} \right) \right) \sup_{[0,t_f]} \|\partial_t u\|_{C^{2q}(\Omega)}.$$

Remark 3. As the number of subdomains increase, and their diameters H and overlap β decrease (i.e., $\beta, H \to 0$), the accuracy of the proposed ADI method deteriorates. However, since all the terms involving β are multiplied by powers of τ, we can choose a smaller time step τ_{adi} for the ADI method so that the resulting global error is $O(h^2)$, the same as for the Crank–Nicolson method with exact solvers. A simple calculation yields that substituting

$$\tau_{adi} \sim h \beta^{\frac{2q-1}{q}}, \text{ where } \beta < 1,$$

produces a global error $E_{adi}(t)$ satisfying

$$\begin{aligned} &\|E_{adi}(t)\|_{L^2(\Omega)} \\ &\sim \left(h^2 + h^2 \beta^{\frac{4q-2}{q}} + h^2 \beta^{\frac{4q-2}{q} - 3} + \cdots + h^q \beta^{\frac{2q^2-q}{q} - 2q + 1} \right) \sup_{[0,t_f]} \|\partial_t u\|_{C^{2q}(\Omega)}, \\ &\sim h^2 \sup_{[0,t_f]} \|\partial_t u\|_{C^{2q}(\Omega)}. \end{aligned}$$

However, since $\tau_{adi} < h$, the ADI method will require more time steps than the Crank–Nicolson method. For a heuristic comparison of the complexity of the ADI method with the standard Crank–Nicolson method, see section 4.4.

The truncation errors of the first-order FS method can be analyzed similarly. They are larger than the corresponding errors for the ADI method by a factor of τ^{-1}.

THEOREM 4.6. *The local truncation error T_{fs} of the FS method with domain decomposition splitting satisfies*

$$\|T_{fs}\|_{L^2(\Omega)} \leq \|T_{orig}\|_{L^2(\Omega)} + C(a)\tau^2 \left(\frac{1}{\beta^3} + \frac{\tau}{\beta^5} + \cdots + \frac{\tau^{q-2}}{\beta^{2q-1}} \right) \sup_{[0,t_f]} \|\partial_t u\|_{C^{2q}(\Omega)},$$

where u is the exact solution of the partial differential equation. The global error satisfies

$$\begin{aligned} &\|E_{fs}(t)\|_{L^2(\Omega)} \equiv \|u(t) - W_{fs}(t)\|_{L^2(\Omega)} \\ &\leq \|E_{orig}(t)\|_{L^2(\Omega)} + C(a)\tau \left(\tfrac{1}{\beta^3} + \tfrac{\tau}{\beta^5} + \cdots + \tfrac{\tau^{q-2}}{\beta^{2q-1}} \right) \sup_{[0,t_f]} \|\partial_t u\|_{C^{2q}(\Omega)}. \end{aligned}$$

Proof. The proof follows by an application of Lemma 4.3 to the truncation error of the first-order FS method. ∎

4.4. A comparison of the Crank–Nicolson method with the proposed ADI method. In this section, we heuristically compare the *work complexity* (total floating point operations) of the Crank–Nicolson method using exact solvers with that of the ADI method using domain decomposition splitting. Our comparison will be based on calculating the cost of each method for computing the solution up to time t_f and to the *same specified accuracy.* In order to obtain an explicit comparison, we make the following assumptions:

1. An algebraic direct solver is used to solve the linear systems occurring in both the Crank–Nicolson method and in the ADI method with domain decomposition splitting. The cost of the direct solver to solve a linear system in n unknowns is assumed to be $C_\alpha n^\alpha$, where $1 \leq \alpha \leq 3$.
2. The number of subdomains in the ADI method is chosen to be N^d, for some positive integer N, where d is the space dimension ($d = 2$ or $d = 3$). The diameter H and overlap β of each subdomain Ω_k^* is chosen to be proportional to
$$H \sim N^{-1} \qquad \text{and} \qquad \beta \sim N^{-1}.$$
3. The number of unknowns in each subdomain Ω_k^* is assumed to be $\left(n/N^d\right) M$, where $M > 1$ is some magnification factor that enlarges the number of unknowns in each subdomain.
4. The time step τ_{cn} of the Crank–Nicolson method is chosen to be $\tau_{cn} = h$. In order to ensure that both the Crank–Nicolson method and the ADI method have similar errors, see Remark 3 from section 4.3, the time step τ_{adi} of the ADI method is chosen to be
$$\tau_{adi} \sim h\beta^{\frac{2q-1}{q}}.$$

For the above choices, both methods have global errors satisfying

$$\|E_{cn}\| \sim \|E_{adi}\| \sim h^2.$$

Finally, since we chose $\beta \sim N^{-1}$ in assumption 2, the ADI time step becomes

$$\tau_{adi} \sim hN^{-\frac{2q-1}{q}}.$$

For both methods, the cost of computing the solution up to time t_f satisfies

$$\text{Cost} = (\text{Number of time steps}) \times (\text{Cost per time step}).$$

For the Crank–Nicolson method we obtain

$$\text{Cost of CN} = \left(\frac{t_f}{h}\right)(C_\alpha n^\alpha),$$

where n is the number of space unknowns. For the ADI method with domain decomposition splitting, we obtain

$$\begin{aligned}
\text{Cost of ADI} &= \left(\tfrac{t_f}{\tau_{adi}}\right)\left(N^d C_\alpha \left(\tfrac{Mn}{N^d}\right)^\alpha\right), \\
&\sim \left(\frac{t_f}{hN^{-\frac{2q-1}{q}}}\right)\left(N^d C_\alpha \left(\tfrac{Mn}{N^d}\right)^\alpha\right), \\
&\sim \left(\tfrac{t_f C_\alpha n^\alpha}{h}\right)\left(M^\alpha N^{\frac{2q-1}{q}-\alpha d+d}\right), \\
&\sim (\text{Cost of CN})\left(M^\alpha N^{\frac{2q-1}{q}-\alpha d+d}\right),
\end{aligned}$$

where $M > 1$ is the constant magnification factor and q is the number of colors in the domain decomposition splitting; see section 4.1. Asymptotically, when $1 \ll N^d < n$, we obtain that

$$\left(M^\alpha N^{\frac{2q-1}{q}-\alpha d+d}\right) < 1, \text{ if } (1-\alpha d) + \frac{2q-1}{q} < 0,$$

or equivalently if $\alpha > 1 + \frac{2q-1}{qd}$.

Thus, in two dimensions ($d = 2$), the ADI method will cost asymptotically less than Crank–Nicolson if the direct algebraic solver costs $C_\alpha n^\alpha$ with $\alpha > 2 - \frac{1}{2q}$. In three dimensions, the ADI method will be preferable if $\alpha > \frac{5}{3} - \frac{1}{3q}$.

Remark 4. Numerical tests in section 5 indicate that a more reasonable time step for the ADI method is

$$\tau_{adi} \sim h\beta,$$

which leads to similar global errors for the ADI and Crank–Nicolson methods. For such choices of time steps, the ADI method will be competitive with the Crank–Nicolson method if the complexity of the direct solvers is $C_\alpha n^\alpha$ where

$$\alpha > \begin{cases} \frac{3}{2} & d = 2, \\ \frac{4}{3} & d = 3, \end{cases}$$

where $\Omega \subset R^d$.

5. Numerical results. We report here on the results of numerical tests conducted using the generalized ADI scheme and the first-order FS method described in section 3, using the *domain decomposition splittings* of section 4. Our goal in these

TABLE 1
Two nonoverlapping subdomains: Errors at time $t_f = 1$; $\tau_{cn} = \tau_{adi} = h$.

h^{-1}	17	33	49	65
CN	0.0039	0.0010	4.5×10^{-4}	2.5×10^{-4}
ADI	0.0092	0.0024	0.0011	6.2×10^{-4}
FS	0.39	0.19	0.12	0.090

tests was to compare the accuracies at a fixed time t_f, say $t_f = 1$, of the Crank–Nicolson solution when *exact* solvers are used with that of the ADI and first-order FS solutions. Accordingly, we tabulate the global errors for the exact solver-based Crank–Nicolson method and the ADI and first-order FS methods.

To enable computing the global error, we use the heat equation with a known exact solution $u(x, y, t) = e^t \sin(\pi x) \sin(\pi y)$ on the unit square $\Omega = [0, 1]^2$:

$$(24)\quad \begin{cases} u_t &= \Delta u + (1 + 2\pi^2) e^t \sin(\pi x) \sin(\pi y) & \text{on } \Omega \times [0, T], \\ u(x, y, t = 0) &= \sin(\pi x) \sin(\pi y) & \text{on } \Omega, \\ u(x, y, t) &= 0 & \text{on } \partial\Omega \times [0, T]. \end{cases}$$

The standard five-point finite difference scheme was used to discretize in space on a uniform grid of mesh size h. The Crank–Nicolson method was used to discretize in time with time steps τ_{cn} as indicated in the tables. In the tables, we tabulate the scaled Euclidean norm (scaled according to area) of the global error at time $t_f = 1$ for the Crank-Nicolson solution using exact solvers denoted by CN, and Crank–Nicolson with inexact ADI-based solvers denoted by ADI, and Crank–Nicolson with inexact first-order fractional step solvers denoted by FS.

Partitions of unity: The subdomains Ω_k^* chosen in our experiments were rectangular. For the case of overlapping subregions, we constructed *piecewise smooth partitions of unity* as follows. For a model rectangle $\Omega_k^* = [0, a] \times [0, b]$ we define $\omega_k(x, y) = \sin(\pi x/a) \sin(\pi y/b)$ inside Ω_k^* and zero outside it. Then, $\{\chi_k(x, y)\}$ is derived from $\{\omega_k(x, y)\}$ as described in section 4.1.

Table 1 contains the results for the limiting case of *zero overlap*, i.e., $\beta = 0$. This case does not strictly fit in the theoretical framework developed in the paper. We consider *two nonoverlapping* subdomains $\Omega_1^* = \Omega_1 = [0, \frac{1}{2}] \times [0, 1]$ and $\Omega_2^* = \Omega_2 = [\frac{1}{2}, 1] \times [0, 1]$, and we define $\chi_i(x, y) = \chi_{\Omega_i}(x, y)$, for $i = 1, 2$, i.e., the characteristic or indicator functions of the subdomains. In this case, the operators L_k can only be defined weakly, in a variational sense. However, the finite difference approximations were well defined without additional modification since the grid we chose was aligned with the interface. Though the overlap $\beta = 0$, the numerical results indicate that the domain-decomposition-based ADI solution has, approximately, only thrice as large errors compared to the solution of Crank–Nicolson with exact solvers. The FS solution, however, had about 100 times larger error than the CN solution.

Table 2 contains the results for *two overlapping* subdomains: $\Omega_1^* = [0, \frac{3}{4}] \times [0, 1]$ and $\Omega_2^* = [\frac{1}{4}, 1] \times [0, 1]$. In this case the overlap $\beta = 1/4$, and we note that the domain-decomposition-based ADI solution has error very close to the CN solution (using exact solvers). The FS solution had approximately 50 times larger error than this CN solution.

Table 3 contains the results for the case of *16 overlapping* subdomains, as shown in Figure 1. A 4×4 rectangular partition was first formed and each subrectangle was enlarged to include overlap of $\beta = 1/12$ (1/3 of the subdomain width). The subdomains were grouped into $q = 4$ colors, and one of the four colored subregions is shown in Figure 1. In solving $(I + \alpha\tau A_k)w = z$, parallelization is possible with

TABLE 2
Two overlapping subdomains: Errors at time $t_f = 1$; $\tau_{cn} = \tau_{adi} = h$.

h^{-1}	17	33	49	65
CN	0.0039	0.0010	4.5×10^{-4}	2.5×10^{-4}
ADI	0.0042	0.0011	4.9×10^{-4}	2.8×10^{-4}
FS	0.15	0.0050	0.032	0.024

TABLE 3
16 *overlapping subdomains: $H^{-1} = 4$; $\tau_{cn} = \tau_{adi} = h$.*

h^{-1}	17	33	49	65
CN	0.0039	0.0010	4.5×10^{-4}	2.5×10^{-4}
ADI	0.021	0.0058	0.0030	0.0018
FS	0.31	0.14	0.09	0.063

TABLE 4
Many overlapping subdomains: fixed $h^{-1} = 64$ and $\tau_{cn} = h$. Varying τ_{adi}, H, and β.

h^{-1}	64	64	64
τ_{adi}	$h/4$	$h/8$	$h/16$
$H^{-1} \times H^{-1}$	4×4	8×8	16×16
ADI	2.9×10^{-4}	3.16×10^{-4}	3.03×10^{-4}
CN	2.64×10^{-4}	2.64×10^{-4}	2.64×10^{-4}

different solvers on disjoint subregions of the kth color. The results indicate that the domain-decomposition-based ADI solution has approximately seven times larger error than the CN solution with exact solvers. This was expected due to the larger β^{-1} term in the truncation error formula of (23). The FS solution had approximately 26 times larger error than this CN solution.

Table 4 tabulates the global error for a fixed grid of size $h^{-1} = 64$ on which 16, 64, and 256 subdomains were chosen. According to the asymptotic error bounds of equation (23), the global error should remain approximately $O(h^2)$ if $\tau_{adi} = h\beta^{\frac{2q-1}{q}}$. We chose $\tau_{adi} = h\beta$, which is more optimistic than the theoretical bounds (23). Thus, as the number of subdomains increased, the cost of calculating the solution also increased proportionally to the inverse of the subdomain width. The results indicate that the error in the ADI solution remained approximately constant and close to the error for the CN solution with exact solvers.

5.1. Discussion. The numerical results indicate that in the case of two subdomains with reasonable overlap, the global error of the ADI method with domain decomposition splitting is similar to errors for the Crank–Nicolson method with exact solvers. However, for the case of many subdomains, the terms β^{-1} in the error (23) causes a deterioration in the global error. If a smaller time step $\tau_{adi} = h\beta$ is used in the ADI method, then the resulting global error of the ADI solution remains close to the error for CN with exact solvers and $\tau_{cn} = h$.

Our heuristic study of the complexity of these methods in section 4.4 leads us to note that if algebraic direct solvers of complexity $C_\alpha n^\alpha$ are used to solve all linear systems, then the ADI method with domain decomposition splitting can be competitive with the Crank–Nicolson method provided $\alpha > 1 + \frac{2q-1}{qd}$, where d is the space dimension.

Finally, we note that though the ADI method can become unstable, when $q \geq 3$ colors are used, such instability was not observed for the range of mesh, subdomain, and time step sizes used in our numerical experiments.

Acknowledgments. The authors thank Professors Max Dryja and Olof Widlund and the referees for helpful comments. They also express their gratitude to Professor Patrick Le Tallec for a helpful suggestion regarding the ADI method during a presentation of this paper at the Seventh International Conference on Domain Decomposition Methods.

REFERENCES

[1] K. A. Bagrinovskii and S. K. Godunov, *Difference schemes for multidimensional problems*, Dokl. Acad. Nauk USSR, 115 (1957), p. 431.

[2] H. Blum, S. Lisky, and R. Rannacher, *A domain splitting algorithm for parabolic problems*, Computing: Archiv fur Informatik und Numerik, 49 (1992), pp. 11–23.

[3] F. A. Bornemann, *An adaptive multilevel approach to parabolic equations:* III. $2D$ *error estimation and multilevel preconditioning*, Impact Comput. Sci. Engrg., 4 (1992), pp. 1–45.

[4] A. Brandt, *Guide to Multigrid Development*, Lecture Notes in Math. 960, W. Hackbusch and U. Trottenberg, eds., Springer-Verlag, New York, 1981.

[5] X. C. Cai, *Additive Schwarz algorithms for parabolic convection-diffusion equations*, Numer. Math., 60 (1991), pp. 41–61.

[6] X. C. Cai, *Multiplicative Schwarz methods for parabolic problems*, SIAM J. Sci. Comput., 15 (1994), pp. 587–603.

[7] T. Chan and T. Mathew, *Domain decomposition algorithms*, Acta Numerica, 1994, pp. 61–143.

[8] T. F. Chan and J. Zou, *Optimal Domain Decomposition Algorithms for Non-symmetric Problems on Unstructured Meshes*, Tech. report, Department of Mathematics, University of California, Los Angeles, CA, 1994.

[9] C. Dawson, Q. Du, and T. F. Dupont, *A finite difference domain decomposition algorithm for numerical solution of the heat equation*, Math. Comp., 57 (1991), pp. 63–71.

[10] J. Douglas, Jr., *On the numerical integration of* $u_{xx} + u_{yy} = u_t$ *by implicit methods*, J. Soc. Ind. Appl. Math., 3 (1955), pp. 42–65.

[11] J. Douglas, Jr., *Alternating direction methods for three space variables*, Numer. Math., 4 (1961), pp. 41–63.

[12] J. Douglas Jr. and J. E. Gunn, *A general formulation of alternating direction method: Part* I. *Parabolic and hyperbolic problems*, Numer. Math., 6 (1964), pp. 428–453.

[13] M. Dryja, *Substructuring methods for parabolic problems*, in Fourth Internat. Symposium on Domain Decomposition Methods for Partial Differential Equations, R. Glowinski, Y. A. Kuznetsov, G. A. Meurant, J. Périaux, and O. Widlund, eds., SIAM, Philadelphia, PA, 1991, pp. 264–271.

[14] R. Glowinski and P. L. Tallec, *Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics*, SIAM Studies in Applied Mathematics 9, SIAM, Philadelphia, PA, 1989.

[15] W. Hackbusch, *Multi-grid Methods and Applications*, Springer Ser. Comput. Math. 4, Springer Verlag, Berlin, Heidelberg, 1985.

[16] Y. A. Kuznetsov, *New algorithms for approximate realization of implicit difference schemes*, Soviet J. Numer. Anal. Math. Modelling, 3 (1988), pp. 99–114.

[17] Y. A. Kuznetsov, *Overlapping domain decomposition methods for FE-problems with elliptic singular perturbed operators*, in Fourth Internat. Symposium on Domain Decomposition Methods for Partial Differential Equations, R. Glowinski, Y. A. Kuznetsov, G. A. Meurant, J. Périaux, and O. Widlund, eds., SIAM, Philadelphia, PA, 1991, pp. 223–241.

[18] Y. M. Laevsky, *Direct Domain Decomposition Method for Solving Parabolic Equations*, Tech. report preprint 940, Computing Center, Siberian Branch Academy of Sciences, Novosibirsk, Russia, 1992 (in Russian).

[19] Y. M. Laevsky, *On the domain decomposition method for parabolic problems*, Bulletin of the Novosibirsk Computing Center, 1 (1993), pp. 41–62.

[20] H. Chen and R. D. Lazarov, *Domain splitting algorithm for mixed finite element approximations to parabolic problems*, East-West J. Numer. Math., 4 (1996), pp. 121–135.

[21] G. A. Meurant, *Numerical experiments with a domain decomposition method for parabolic problems on parallel computers*, in Fourth Internat. Symposium on Domain Decomposition Methods for Partial Differential Equations, R. Glowinski, Y. A. Kuznetsov, G. A. Meurant, J. Périaux, and O. Widlund, eds., SIAM, Philadelphia, PA, 1991, pp. 394–408.
[22] D. W. Peaceman and H. H. Rachford Jr., *The numerical solution of parabolic and elliptic differential equations*, J. Soc. Ind. Appl. Math., 3 (1955), pp. 28–41.
[23] R. D. Richtmyer and K. W. Morton, *Difference Methods for Initial-value Problems*, Wiley-Interscience, New York, 1967.
[24] G. Starke, *Alternating direction preconditioning for nonsymmetric systems of linear equations*, SIAM J. Sci. Comput., 15 (1994), pp. 369–384.
[25] S. Sternberg, *Lectures on Differential Geometry*, Chelsea, New York, 1964.
[26] G. Strang, *On the construction and comparison of difference schemes*, SIAM J. Numer. Anal., 5 (1968), pp. 506–517.
[27] X. C. Tai, *Domain decomposition for linear and non-linear elliptic problems via function or space decomposition*, in Domain Decomposition Methods in Scientific and Engineering Computing, D. Keyes and J. Xu, eds., AMS, Providence, RI, 1994, pp. 355–360.
[28] P. N. Vabishchevich, *Parallel domain decomposition algorithms for time-dependent problems of mathematical physics*, in Advances in Numerical Methods and Applications, $O(h^3)$, I. T. Dimov, B. Sendov, and P. S. Vassilevski, eds., World Scientific, Singapore, 1994, pp. 293–299.
[29] P. N. Vabishchevich and P. Matus, *Difference schemes on grids locally refined in space as well as in time*, in Advances in Numerical Methods and Applications, $O(h^3)$, I. T. Dimov, B. Sendov, and P. S. Vassilevski, eds., World Scientific, Singapore, 1994, pp. 146–153.
[30] O. B. Widlund, *On the rate of convergence of an alternating direction implicit method in a noncommutative case*, Math. Comp., 20 (1966), pp. 500–515.
[31] O. B. Widlund, *On difference methods for parabolic equations and alternating-direction-implicit methods for elliptic equations*, IBM J. Research and Development, 11 (1967), pp. 239–243.
[32] O. B. Widlund, *On the effects of scaling of the Peaceman-Rachford method*, Math. Comp, 25 (1971), pp. 33–41.
[33] J. Xu, *Iterative methods by space decomposition and subspace correction*, SIAM Rev., 34 (1992), pp. 581–613.
[34] N. N. Yanenko, *On weak approximation of systems of differential equations*, Sibirsk. Mat. Zh., 5 (1964), p. 1430.