Instituto Nacional de Matemática Pura e Aplicada

Coarse grid correction operator splitting for parabolic partial differential equations

Carlos E. C. Borges

Adviser: Marcus Sarkis
Co-adviser: Christian E. Schaerer

Rio de Janeiro, November 14, 2007

Abstract

We develop a two-level splitting method for the numerical solution of parabolic partial differential equations (PDE). Instead of using the backward Euler method, we use an operator splitting scheme to advance the system of equations obtained from the spatial discretization of the PDE. The split operators are derived from a domain decomposition technique based on a partition of unity of the spatial domain. The analysis and numerical results show that the discretization error is of second order in space and of first order in time, however, it deteriorates when the number of subdomains increases and when the overlap between the subdomains decreases. To overcome this drawback we introduce a coarse grid correction scheme at each time step. The numerical results obtained with the coarse grid correction show that the discretization errors are smaller than the ones obtained from the one-level operator splitting method.

Resumo

Nós desenvolvemos um método de *splitting* de dois níveis para a solução numérica de equações diferenciais parciais parabólicas (EDP). Ao invés de usarmos o método de backward Euler, nós usamos um esquema de divisão de operador para avançar o sistema de equações obtido da discretização da EDP no espaço. Os operadores são obtidos por uma técnica de decomposição de domínio baseada numa partição da unidade do espaço. A análise e os resultados numéricos mostram que o erro de discretização é de primeira ordem em relação ao tempo e de segunda ordem em relação ao espaço, contudo, ele se deteriora com o aumento do número de subdomínios e a diminuição do tamanho da sobreposição entre esses subdomínios. Para resolver este problema, introduzimos um esquema de correção de malha grossa em cada passo de tempo. Os resultados numéricos obtidos pela correção da malha grossa mostram que os erros de discretização são menores que aqueles obtidos pelo método de splitting de operador de um nível.

Acknowledgments

My deepest gratitude goes to my co-advisor and friend, Professor Christian Schaerer, for all the time and energy he has devoted to me and my work. Without his guidance and help, this work would be much more difficult to be completed. During the past two and a half years, he has always guided and helped me working on this thesis. His contribution was fundamental.

My gratitude also goes to my advisor, Professor Marcus Sarkis, for his help and supervision during my research. Specially, for all the suggestions in the last months.

I also wish to thank Professor Dan Marchesin for his valuable suggestions.

In addition, I would like to thank to my friends at work, for all the support, and to all my friends in IMPA, for all the help in the past two and a half years.

I thank my parents, Nilson and Lucia, for the help and support during all these years of study, and Bianca for the patience, love and comprehension. This thesis is dedicated to them.

Notation

Ω	spatial domain of the problem contained in $\mathbb{R}^d$
$\bar{\Omega}$	the union of Ω and its boundary $\partial\Omega$
$u(x,t)$	exact solution of the parabolic partial differential equation (2.1)
$g(x)$	initial condition of problem (2.1)
$\mathcal{L}$	operator ∂_{xx}
P	operator $(\partial_t - \partial_{xx})$
$\hat{m}$	number of spatial subintervals
h	spatial mesh size
m	index of the spatial nodes
Ω_h	set of interior nodes in Ω
$\mathbf{g}_h$	pointwise evaluation of the initial condition $g(x)$ on Ω_h
A	spatial discretization of the operator $\mathcal{L}$ with mesh size h
$\hat{n}$	number of time subintervals
k	time discretization step
n	index of time step
ϕ_m^n	$\phi(mh, nk)$
$P_{h,k}$	discretization of P with the backward-time central-space scheme (BTCS)
$\mathbf{v}^n$	discrete solution of (2.1) using the BTCS scheme at time nk
I	identity matrix
O	matrix with all elements equal to zero
$\|\mathbf{y}\|_2$	Euclidean norm of the vector $\mathbf{y}$
$\|\mathbf{y}\|_h$	norm of the vector $\mathbf{y}$ as defined in (2.11)
$\|\|\|B\|\|\|_h$	spectral norm of the matrix B as defined in (2.12)
$\boldsymbol{\ell}_{be}^{n+1}(\phi)$	local truncation error for the smooth function ϕ of the BTCS scheme at time $(n+1)k$
$\hat{q}$	number of operators used in the splitting method
q	index of the operators used in the splitting method
$\mathbf{w}^n$	first-order splitting discrete solution of the problem (2.1) at time nk
2β	overlap between two neighboring subdomains composing an open covering of Ω
$\hat{s}$	number of overlapping subdomains $\{\Omega_s^*\}_{s=1,\ldots,\hat{s}}$
s	index of the partition of unity $\{\chi_s\}_{s=1,\ldots,\hat{s}}$ and the overlapping subdomains $\{\Omega_s^*\}_{s=1,\ldots,\hat{s}}$

$\{\chi_s\}_{s=1,\ldots,\hat{s}}$	partition of unity used to derive the overlapping subdomainsΩ_s^*
$\tilde{\chi}_q$	partition of unity used to derive operator $\mathcal{L}_q$
$\mathcal{L}_q$	operator $\partial_x(\tilde{\chi}_q \partial_x u)$
A_q	spatial discretization of the operator $\mathcal{L}_q$
$\boldsymbol{\ell}_{sp}^{n+1}(\phi)$	local truncation error for the smooth function ϕ of the first-order splitting method at time $(n+1)k$
H	coarse grid spatial size
Ω_H	$\{x \in \mathbb{R} : x = x_s = sH, s = 1, \ldots, \hat{s} - 1, H = 1/\hat{s}\}$
$\mathbf{v}_H$	a vector $\mathbf{v}$ in the coarse grid Ω_H
$\mathbf{v}_h$	a vector $\mathbf{v}$ in the fine grid Ω_h
$\mathbf{z}^n$	the two-level discrete solution of problem (2.1) at time nk

Index

Chapter 1

Introduction

Let us consider the parabolic equation for $u(x,t)$ on a domain $\Omega \subset \mathbb{R}^d$ [7]:

$$\begin{cases} \partial_t u(x,t) &= \nabla \cdot (D(x)\nabla u(x,t)) - c(x)u(x,t) + f(x,t) & \text{in } \Omega \times (0,T) \\ u(x,0) &= g(x) & \text{on } \Omega \times 0 \\ u(x,t) &= 0 & \text{in } \partial\Omega \times (0,T), \end{cases} \tag{1.1}$$

where $D(x)$ is a symmetric positive definite $d \times d$ matrix function with $C^{2\hat{q}-1}(\bar{\Omega})$ entries, $c(x) \geq 0$ is a scalar function in $C^{2\hat{q}}(\bar{\Omega})$, $f(x,t)$ is the forcing term in $C^{2\hat{q}}(\bar{\Omega} \times [0,T])$ and $g(x) \in C^{2\hat{q}}(\bar{\Omega})$ is the initial condition. It can be shown that the exact solution $u(x,t)$ of (1.1) is in $C^{2\hat{q}}(\bar{\Omega} \times [0,T])$. The regularity parameter $\hat{q}$ represents also the number of operators used in the operator splitting method presented in Chapter 3.

The numerical solution of the parabolic partial differential equation (1.1), using an implicit numerical scheme, requires the solution of a relatively large linear system of equations. This linear system can be solved inexactly provided that the inexact solutions preserve the stability and local truncation error of the original scheme [7]. We consider non-iterative inexact solvers at each time step. Such solvers include the alternating direction method (ADI) of Douglas and Gunn [3], and Peaceman and Rachford [8] (for two-dimensional problems), and the fractional step methods (FS) of Bagrinovskii and Godunov [1], Yanenko [13] and Strang [10]. In this thesis, we consider FS methods only.

The method proposed here uses the same framework as the classical FS method for solving parabolic equations of the form (1.1) [1, 13, 10, 4]. However, instead of using the approach of splitting the operator $\mathcal{L}$ in the coordinate axes, we use a splitting based on domain decomposition, as in [7]. The domain decomposition is obtained from a smooth partition of unity that

decomposes the domain in overlapping subdomains. As a result, the elliptic operator $\mathcal{L}$ can be split as the sum of simpler operators represented as:

$$\mathcal{L} = \mathcal{L}_1 + ... + \mathcal{L}_{\hat{q}}.$$

The advantage of splitting methods is that the operators $\mathcal{L}_i$ and associated equations

$$\partial_t u + \mathcal{L}_q u = f_q, \qquad \forall q = 1, \ldots, \hat{q}, \tag{1.2}$$

are cheaper to solve than the equation (1.1). By using a splitting based domain decomposition on the operator $\mathcal{L}$, we replace the problem (1.1) by solving in parallel equations of the form (1.2).

Although the splitting based on domain decomposition is easy to implement, the method shows a severe deterioration of the accuracy of the error when the number of subdomains increases since no global communication among the subdomains of Ω is inherited in the algorithm. Hence, the method does not resolve the low frequencies. To overcome the lack of global communication, we introduce a two-level approach by considering a coarse grid to resolve the low frequencies and improve the accuracy of the method.

In Chapter 2, we present the simplification of problem (1.1), which we use to demonstrate the theory. We introduce the discrete problem and the concepts of stability, local truncation error, consistency and global error.

In Chapter 3, we introduce the construction of the partition of unity used to derive the operator splitting based on domain decomposition. The first-order operator splitting is implemented with two operators (i.e., $\hat{q} = 2$). We analyze the error, consistency and stability of the splitting method, establishing an estimate of the error of the method in terms of the number of subdomains and the overlap between these subdomains. Our error estimate is more general than the estimate in [7] since it includes the dependence on the number of subdomains. Numerical results are presented to corroborate the results presented in the theory.

In Chapter 4, we introduce a coarse grid to accelerate the convergence of the splitting method. The numerical results show that the two-level schemes are more accurate than one-level operator splitting when the number of subdomains increases. The analysis of the coarse grid correction is not presented in this thesis. Conclusions are presented in Chapter 5.

Chapter 2

The parabolic PDE discretization

2.1 Model simplification

We are interested in developing a numerical method for the solution of (1.1). To simplify our work, we consider the one-dimensional case and use $\hat{q} = 2$. We take $D(x) = 1$ and $c(x) = 0$, obtaining the following equation:

$$\begin{cases} \partial_t u(x,t) &= \mathcal{L}u(x,t) + f(x,t) & \text{in } \Omega \times (0,T) \\ u(x,0) &= g(x) & \text{on } \Omega \times 0 \\ u(x,t) &= 0 & \text{on } \partial\Omega \times (0,T), \end{cases} \tag{2.1}$$

where $\mathcal{L}u = \partial_{xx}u$ and $\Omega = (0, L)$. The results in this thesis can be easily extended to the d-dimensional case of (1.1).

The problem (2.1) can be interpreted physically as the model for the time evolution of a temperature distribution over a one-dimensional bar of size L, where $g(x)$ is the temperature distribution at time $t = 0$ and $f(x,t)$ is the heat source. The values of the temperature at the boundary of Ω are set to zero, $\forall t \in [0,T]$. This problem is well-posed.

2.2 Problem discretization

First we use the following notation to define our problem:

$$Pu = (\partial_t - \mathcal{L})u = (\partial_t - \partial_{xx})u = f. \tag{2.2}$$

We subdivide $\Omega = (0, L)$ in $\hat{m}$-spatial subintervals of length $h = L/\hat{m}$, where the interior nodes are $x_m = mh$, for $m = 1, \ldots, \hat{m} - 1$. Ω_h denotes this set of the interior spatial nodes. We

also denote $x_0 = 0$ and $x_{\hat{m}} = L$ in order to impose the zero Dirichlet boundary condition. In time, we make a subdivision of the interval $[0, T]$ in $\hat{n}$ subintervals of size $k = T/\hat{n}$, obtaining time intervals $[t_n, t_{n+1}]$, where $t_n = nk$, $n = 0, 1, \ldots, \hat{n}$.

Applying the second-order central finite difference scheme on the elliptic operator $\mathcal{L}$, we obtain:

$$\mathcal{L}\phi(mh, (n+1)k) = \frac{\phi_{m+1}^{n+1} - 2\phi_m^{n+1} + \phi_{m-1}^{n+1}}{h^2} + \mathcal{O}(h^2), \tag{2.3}$$

where $\phi_m^{n+1} := \phi(mh, (n+1)k)$ and ϕ is a smooth function.

Applying the first order backward difference scheme, we obtain:

$$\partial_t\phi(mh, (n+1)k) = \frac{\phi_m^{n+1} - \phi_m^n}{k} + \mathcal{O}(k). \tag{2.4}$$

Discretizing the operator P in time and in space using the backward-time central-space (BTCS) finite difference scheme [11] we obtain:

$$P_{k,h}\phi = \frac{\phi_m^{n+1} - \phi_m^n}{k} - \frac{\phi_{m+1}^{n+1} - 2\phi_m^{n+1} + \phi_{m-1}^{n+1}}{h^2}. \tag{2.5}$$

For the function $f(x, t)$ in (2.1), we use the following operator:

$$R_{k,h}f = f_m^{n+1}, \tag{2.6}$$

where $f_m^{n+1} := f(mh, (n+1)k)$.

So, applying BTCS in (2.1) gives us the finite difference scheme:

$$\begin{cases} (I - kA)\mathbf{v}^{n+1} = \mathbf{v}^n + k\mathbf{f}^{n+1}, \\ \mathbf{v}^0 = \mathbf{g}_h, \end{cases} \tag{2.7}$$

where A is a symmetric negative definite $(\hat{m}-1) \times (\hat{m}-1)$ matrix, $\mathbf{v}^n \in \mathbb{R}^{\hat{m}-1}$ is the BTCS discrete solution at time nk, $\mathbf{g}_h$ is the pointwise evaluation of the initial condition $g(x)$ on Ω_h and $\mathbf{f}^{n+1}$ is the pointwise evaluation of the forcing term $f(x, t)$ on Ω_h at time $(n+1)k$. For notation simplification, we make $B := I - kA$.

Remark 2.1. The matrix A is symmetric negative definite with eigenvalues λ_m given by [2]:

$$\lambda_m(A) = -\frac{4}{h^2}\sin^2\left(\frac{m\pi}{2\hat{m}}\right), \tag{2.8}$$

where $m = 1, ..., \hat{m} - 1$ and $h = L/\hat{m}$. The largest eigenvalue of A satisfies $\lambda_{max}(A) \approx -h^{-2}$ and the smaller one $\lambda_{min}(A) \approx -1$. It follows that the condition number of $(I - kA)$ is bounded by $1 + ckh^{-2}$, for some positive constant c independent of k and h.

2.3 Consistency, stability and convergence

The most important property that a numerical scheme must have is that its numerical solution must approximate the solution of the corresponding partial differential equation. The approximation must improve as the grid sizes, h and k, go to 0. We call such scheme convergent and it is defined as:

Definition 2.2. *A one-step difference scheme approximating a partial differential equation is a convergent scheme if for any solution to the partial differential equation, $u(x,t)$, and solutions to the difference scheme v_m^n, such that v_m^0 converges to $g(x)$ as mh converges to x, then v_m^n converges to $u(x,t)$ as (mh,nk) converges to (x,t) as h,k converge to 0.*

It is not easy in general to prove the convergence of a given scheme. So, we use the Lax-Richtmyer Equivalence Theorem to prove the convergence of the numerical scheme.

Theorem 2.3 (Lax-Richtmyer Equivalence Theorem)**.** *A consistent finite difference scheme accurate of order (p,q) for a partial differential equation for which the initial value problem is well posed is convergent if and only if it is stable.*

To use Theorem 2.3, we must define the concepts of consistency, order of accuracy and stability. We start with the definition of consistency:

Definition 2.4. *Given a partial differential equation, $Pu = (\partial_t - \partial_{xx})u = f$, and a finite difference scheme, $P_{k,h}v = R_{k,h}f$, we say that the finite difference scheme is consistent with the partial differential equation if for any smooth function $\phi(x,t)$*

$$P\phi - P_{k,h}\phi \;\rightarrow\; 0 \qquad \text{as } k,h \;\rightarrow\; 0$$

the convergence being pointwise convergence at each point (x,t).

For the scheme BTCS, by using (2.3)-(2.5), we have the following result:

$$P\phi - P_{k,h}\phi \;=\; \mathcal{O}(k) + \mathcal{O}(h^2). \tag{2.9}$$

Since the right hand side of (2.9) tends to 0 when $k,h \rightarrow 0$, we have that the scheme BTCS is consistent.

Difference schemes may differ considerably in how well their solutions approximate the solution of the differential equation. The "measure" of how well these schemes approximate the solution is called order of accuracy and it is defined as:

Definition 2.5. *A scheme $P_{k,h}v = R_{k,h}f$ that is consistent with the differential equation $Pu = f$ is accurate of order p in time and order q in space if for any smooth function $\phi(x,t)$*

$$P_{k,h}\phi - R_{k,h}P\phi \;=\; \mathcal{O}(k^p) + \mathcal{O}(h^q). \tag{2.10}$$

We say that this scheme is accurate of order (p,q).

From (2.3)-(2.6), the BTCS scheme is accurate of order (1,2).

Now, we define the norms used in this thesis. For a vector $\mathbf{y} \in \mathbb{R}^{\hat{m}-1}$, we have:

$$\|\mathbf{y}\|_h^2 \;=\; \sum_{m=1}^{\hat{m}-1} {y_m}^2 h. \tag{2.11}$$

For a matrix $B \in \mathbb{R}^{(\hat{m}-1)\times(\hat{m}-1)}$, we define the spectral norm as:

$$|||B|||_h \;=\; \sup_{\mathbf{y}} \frac{\|B\mathbf{y}\|_h}{\|\mathbf{y}\|_h}. \tag{2.12}$$

We use the norms defined in (2.11) in this thesis unless it is otherwise stated.

Next, we define the concept of local truncation error, which we will use in this thesis to obtain the order of accuracy of the scheme presented in Chapter 3. When we multiply the left member of (2.10) by the time step k, we can write the local truncation error of a scheme as:

Definition 2.6. *Given the discretization (2.7), we define its local truncation error $\boldsymbol{\ell}_{be}$ as:*

$$\boldsymbol{\ell}_{be}^{n+1}(\phi) = (I - kA)\boldsymbol{\phi}^{n+1} - \boldsymbol{\phi}^n - k(\boldsymbol{P\phi})^{n+1}$$

for any smooth function ϕ, where $\boldsymbol{\phi}^{n+1}$ and $(\boldsymbol{P\phi})^{n+1}$ correspond to the evaluation of ϕ and $P\phi$ in Ω_h at time $(n+1)k$ and $\boldsymbol{\ell}_{be}^{n+1}(\phi)$ is a $\mathbb{R}^{\hat{m}-1}$ vector. From (2.3)-(2.6), we have that each component of the vector $\boldsymbol{\ell}_{be}^{n+1}(\phi)$ is accurate of order $\mathcal{O}(k^2) + \mathcal{O}(kh^2)$. Taking the norm (2.11), we obtain for the BTCS scheme:

$$\|\boldsymbol{\ell}_{be}^{n+1}(\phi)\|_h \;=\; \mathcal{O}(k^2) + \mathcal{O}(kh^2). \tag{2.13}$$

We can easily see from Definition 2.5 and Definition 2.6 that the order of accuracy of a scheme is the order of the norm of the local truncation error divided by the time mesh size k.

In studying the convergence of the difference schemes, it is also important to define the concept of stability. We wish to include the zero boundary conditions [5], then we use the matrix method to examine the difference scheme (2.7).

In this work, we adopt the following definition of stability:

Definition 2.7. *We shall say that the scheme* (2.7) *is stable if and only if there is a constant* C *(independent of* k *and* h*) and some positive grid spaces* k_0 *and* h_0 *such that*

$$|||B|||_h \leq 1 + Ck, \tag{2.14}$$

for all $0 \leq k \leq k_0$ *and* $0 \leq h \leq h_0$*. If* $|||B|||_h$ *is independent of* h *and* k*, the stability condition can be replaced with*

$$|||B|||_h \leq 1.$$

Using the definition above, a sufficient condition for the stability of equation (2.7) is $|||B|||_h \leq 1$. The stability of the backward Euler method results from:

$$\begin{aligned} \frac{\|B\mathbf{y}\|_h^2}{\|\mathbf{y}\|_h^2} &= \frac{\|(I-kA)^{-1}\mathbf{y}\|_h^2}{\|\mathbf{y}\|_h^2} \\ &= \frac{\|\tilde{\mathbf{y}}\|_h^2}{\|(I-kA)\tilde{\mathbf{y}}\|_h^2} \\ &= \frac{\|\tilde{\mathbf{y}}\|_h^2}{\|\tilde{\mathbf{y}}\|_h^2 - 2k(A\tilde{\mathbf{y}},\tilde{\mathbf{y}}) + k^2\|A\tilde{\mathbf{y}}\|_h^2}, \end{aligned} \tag{2.15}$$

where we have used that $\tilde{\mathbf{y}} = (I - kA)^{-1}\mathbf{y}$. Since the matrix A, for the BTCS scheme, is symmetric negative definite, i.e., $(A\tilde{\mathbf{y}}, \tilde{\mathbf{y}}) < 0$, we obtain:

$$\frac{\|B\mathbf{y}\|_h^2}{\|\mathbf{y}\|_h^2} \leq 1, \tag{2.16}$$

and from (2.12) this implies that $|||B|||_h \leq 1$.

Following [9], the problem (2.1) is properly posed. We have obtained that the scheme (2.7) is accurate of order (1,2), stable and consistent. Therefore, it is convergent of order (1,2) by the Lax-Richtmyer's equivalence theorem.

Chapter 3

The operator splitting method

3.1 Introduction

Given a discrete problem of the form:

$$\begin{cases} (I - kA)\mathbf{v}^{n+1} &= \mathbf{v}^n + k\mathbf{f}^{n+1} \\ \mathbf{v}^0 &= \mathbf{g}_h, \end{cases} \tag{3.1}$$

where $n = 0, 1, \ldots, \hat{n} - 1$, the basic idea of the operator splitting method is to solve it approximately using an operator splitting of the form: $A = A_1 + A_2$. Therefore, the splitting method solves approximately the systems of the form (3.1) by solving simpler equations of the form $(I - kA_q)\mathbf{z}_q = \mathbf{z}_{q-1} + k\mathbf{f}_q^{n+1}$, $q = 1, 2$. Hence, A_q and $\mathbf{f}_q$ must be suitable chosen so that the latter systems are easier to solve than the original system.

Since A is a symmetric negative definite matrix, we consider A_q, $q = 1, 2$, to be symmetric negative semidefinite and to have a simpler structure than A, and such that:

$$A = A_1 + A_2, \qquad \text{where } A_q \leq 0.$$

We also chose $\mathbf{f}_q^{n+1}$, $q = 1, 2$, such that $\mathbf{f}^{n+1} = \mathbf{f}_1^{n+1} + \mathbf{f}_2^{n+1}$.

In this chapter, we introduce the operator splitting and its analysis. The splitting of the matrix A is performed using a domain decomposition technique. This technique subdivides the domain into several overlapping subdomains decoupling the solution associated to each subdomain except in the overlapping regions. We use a partition of unity to split the matrix A.

3.2 Operator splitting

These methods were originally conceived by Braginovskii and Godunov [1] and Yanenko [13], and further developed by Strang [10] and other authors [6, 4]. Given a matrix splitting $A = A_1 + A_2$, the operator splitting method approximates the matrix $(I - kA)$ by a product of several matrices of the form $(I - kA_q)$. The first-order operator splitting method (which is known as *sequential splitting*) replaces $(I - kA)$ by:

$$(I - kA) \;=\; (I - kA_1)(I - kA_2) + \mathcal{O}(k^2),$$

since A_1 and A_2 have derivatives and we have that

$$(I - kA_1)(I - kA_2) \;=\; I - k(A_1 + A_2) + k^2 A_1 A_2 \;=\; (I - kA) + \mathcal{O}(k^2) \tag{3.2}$$

is true when applied to a function with the double of derivatives than A_1 and A_2. We solve the following problem

$$(I - kA_1)\mathbf{z}^n \;=\; \mathbf{w}^n + k\mathbf{f}_1^{n+1} \qquad \text{and} \qquad (I - kA_2)\mathbf{w}^{n+1} \;=\; \mathbf{z}^n + k\mathbf{f}_2^{n+1}, \tag{3.3}$$

where $\mathbf{z}^n$ is an auxiliary variable and $\mathbf{f}^{n+1} = \mathbf{f}_1^{n+1} + \mathbf{f}_2^{n+1}$, for all $0 \leq n \leq \hat{n} - 1$.

Hence, using elementary algebra, the first-order operator splitting solution associated to the BTCS scheme is given by:

$$\begin{cases} (I - kA_1)(I - kA_2)\mathbf{w}^{n+1} \;=\; \mathbf{w}^n + k\mathbf{f}^{n+1} + k^2 A_1 \mathbf{f}_2^{n+1} \\ \mathbf{w}^0 \;=\; \mathbf{g}_h. \end{cases} \tag{3.4}$$

We prove later in this chapter that the solution $\mathbf{w}$ above is an approximation to $\mathbf{u}$ that is first order in time and second order in space. We will see in Remark 3.4 that A_q is symmetric negative semidefinite, therefore, the stability of the scheme (3.4) is proven in the next theorem.

Theorem 3.1. *For symmetric negative semidefinite A_q, the scheme* (3.4) *is stable.*

Proof: It is sufficient to verify, see inequality (2.16), that:

$$|||(I - kA_1)^{-1}(I - kA_2)^{-1}|||_h \;\leq\; 1. \tag{3.5}$$

Since we have $|||(I - kA_q)^{-1}|||_h \leq 1$, we obtain:

$$|||(I - kA_1)^{-1}(I - kA_2)^{-1}|||_h \;\leq\; |||(I - kA_1)^{-1}|||_h |||(I - kA_2)^{-1}|||_h \;\leq\; 1.$$

The proof is complete.□

In the next section, we construct a $C^3(\Omega)$ partition of unity and show how to obtain the operators A_q from it.

3.3 Domain-decomposition of the operators

Let $\{\Omega_s\}_{s=1,...,\hat{s}}$ (where, for simplicity, $\hat{s} = 2^{\hat{i}}$ and $\hat{i}$ is a positive integer) denotes a non overlapping collection of subintervals of Ω. Each subinterval has length $H = L/\hat{s}$, and $\overline{\Omega} = \bigcup_{s=1}^{\hat{s}} \overline{\Omega}_s$ and $\Omega_{s_1} \bigcap \Omega_{s_2} = \emptyset$ for $s_1 \neq s_2$.

We extend the subdomains Ω_s by a length β in each side to obtain Ω_s^*. Hence, $\Omega_s \subset \Omega_s^*$ and two neighboring subdomains have an overlap of length 2β. Notice that the nodes $X_s = sH$, for $s = 1, ..., \hat{s} - 1$, determine a grid of $(0, L)$. $\{\Omega_s^*\}_{s=1,...,\hat{s}}$ is an overlapping covering of Ω.

A $C^3(\Omega)$ partition of unity subordinated to this covering is a family of $\hat{s}$ nonnegative $C^3(\Omega)$ functions $\{\chi_s(x)\}_{s=1,...,\hat{s}}$ whose sum is equal to one and such that $\chi_s(x)$ vanishes outside Ω_s^*. We have:

$$\begin{array}{lcl} \text{a) } 0 \leq \chi_s(x) & \leq & 1 \\ \text{b) } \sum_{s=1}^{\hat{s}} \chi_s(x) & = & 1 \\ \text{c) } \text{supp}(\chi_s(x)) & = & \overline{\Omega}_s^*. \end{array} \tag{3.6}$$

We take the intervals Ω_s^* as an open covering for $(0, L)$ such that:

$$\begin{array}{lcl} \Omega_1 & = & (0, H) \\ \Omega_1^* & = & (0, H + \beta) \\ \Omega_s & = & (H(s-1), Hs) \\ \Omega_s^* & = & (H(s-1) - \beta, Hs + \beta) \\ \Omega_{\hat{s}} & = & (L - H, L) \\ \Omega_{\hat{s}}^* & = & (L - H - \beta, L), \end{array} \tag{3.7}$$

where $2 \leq s \leq \hat{s} - 1$.

Once the overlapping subdomains $\{\Omega_s^*\}_{s=1,...,\hat{s}}$ are formed, a $C^3(\Omega)$ partition of unity $\{\chi_s\}_{s=1,...\hat{s}}$

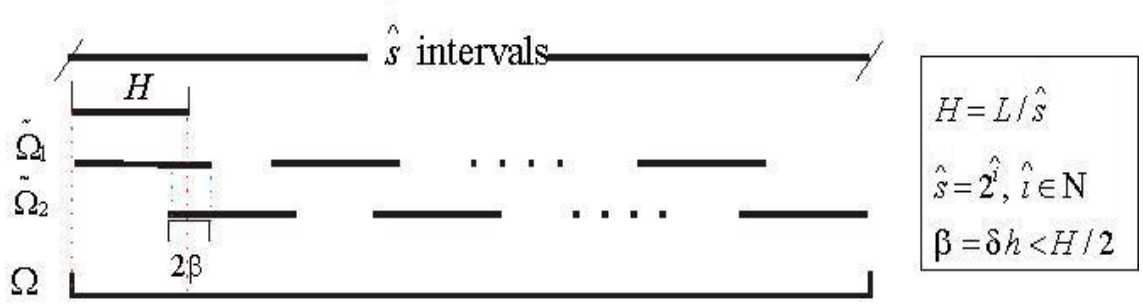

Figure 3.1: Open covering of Ω.

can be constructed as follows:

$$\chi_1(x) = \begin{cases} 1 & \text{if } x \in (0, H-\beta] \\ h((x-H+\beta)/(2\beta)) & \text{if } x \in (H-\beta, H+\beta) \\ 0 & \text{otherwise} \end{cases}$$

$$\chi_s(x) = \begin{cases} 1-h((x-H(s-1)+\beta)/(2\beta)) & \text{if } x \in (H(s-1)-\beta, H(s-1)+\beta) \\ 1 & \text{if } x \in [H(s-1)+\beta, Hs-\beta] \\ h((x-Hs+\beta)/(2\beta)) & \text{if } x \in (Hs-\beta, Hs+\beta) \\ 0 & \text{otherwise} \end{cases}$$

$$\chi_{\hat{s}}(x) = \begin{cases} 1-h((x-L+H+\beta)/(2\beta)) & \text{if } x \in (L-H-\beta, L-H+\beta) \\ 1 & \text{if } x \in [L-H+\beta, L) \\ 0 & \text{otherwise,} \end{cases}$$

where $2 \le s \le \hat{s}-1$. If we use a $C^3(\Omega)$ partition of unity, it suffices to use $h(x) = 20x^7 - 70x^6 + 84x^5 - 35x^4 + 1$, because $h(0) = 1$, $h(1) = 0$ and $h^{(i)}(0) = h^{(i)}(1) = 0$ for $i = 1, \ldots, 3$. It is easy to check that the functions $\chi_s(x)$ are in $C^3(\Omega)$ for this $h(x)$.

Now, we consider a splitting $A = A_1 + A_2$ based on domain-decomposition of the discretization of the operator $\mathcal{L} = \frac{\partial^2}{\partial x^2}$. We are interested in providing an estimate on how the truncation error depends on the overlap β between the subdomains. For related algorithms, see [7] and references therein.

We obtain the partition of unity $\{\tilde{\chi}_q\}_{q=1,2}$ from $\{\chi_s\}_{s=1,\ldots,\hat{s}}$ by constructing a new covering $\{\tilde{\Omega}_q\}_{q=1,2}$ for $(0, L)$, derived from $\{\Omega_s^*\}_{s=1,\ldots,\hat{s}}$, and defined as $\tilde{\Omega}_1 = \bigcup_{s=1}^{\hat{s}/2} \Omega_{2s-1}^*$ and $\tilde{\Omega}_2 = \bigcup_{s=1}^{\hat{s}/2} \Omega_{2s}^*$. We assume that $\beta < H/2$ (see Figure 3.1). Summarizing, we split the domain into two subdomains which can be painted with two different colors. We make $\tilde{\chi}_1 = \sum_{s=1}^{\hat{s}/2} \chi_{2s-1}$ and $\tilde{\chi}_2 = \sum_{s=1}^{\hat{s}/2} \chi_{2s}$. We have that $\tilde{\chi}_1$ and $\tilde{\chi}_2$ are $C^3(\Omega)$ as they inherit these properties from the functions χ_s.

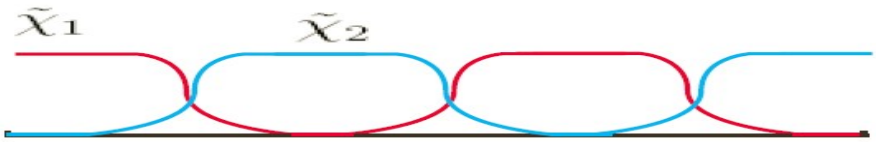

Figure 3.2: Partition of unity $\{\tilde{\chi}_s\}_{s=1,2}$.

We generate the matrices A_q using the $C^3(\Omega)$ partition of unity $\{\tilde{\chi}_q\}_{q=1,2}$ for two open subdomains $\{\tilde{\Omega}_q\}_{q=1,2}$ which cover $(0, L)$. We split the operator $\mathcal{L} = \frac{\partial^2}{\partial x^2}$ into the following sum:

$$\begin{aligned} \mathcal{L}u &= \partial_x(\tilde{\chi}_1\partial_x u) + \partial_x(\tilde{\chi}_2\partial_x u) \\ &= (\mathcal{L}_1 + \mathcal{L}_2)u, \end{aligned}$$

where $\mathcal{L}_q = \partial_x(\tilde{\chi}_q\partial_x)$, for $q = 1, 2$.

The matrices A_q are defined by:

$$(A_q\mathbf{u}^{n+1})_m = \frac{\tilde{\chi}_{q,m+\frac{1}{2}}u^{n+1}_{m+1} - (\tilde{\chi}_{q,m+\frac{1}{2}} + \tilde{\chi}_{q,m-\frac{1}{2}})u^{n+1}_m + \tilde{\chi}_{q,m-\frac{1}{2}}u^{n+1}_{m-1}}{h^2}, \tag{3.8}$$

where $\tilde{\chi}_{q,m+\frac{1}{2}} := \tilde{\chi}_q(h(m + 1/2))$.

We implement the partition of unity in the function f, obtaining the functions $f_q = \tilde{\chi}_q f$, for $q = 1, 2$.

After splitting the operator A and $\mathbf{f}$ and using $\mathbf{f}_2 = \tilde{\chi_2}\mathbf{f}$ in (3.4), we obtain the following problem:

$$\begin{cases} (I - kA_1)(I - kA_2)\mathbf{w}^{n+1} &= \mathbf{w}^n + k\mathbf{f}^{n+1} + k^2A_1(\tilde{\chi}_2\mathbf{f})^{n+1} \\ \mathbf{w}^0 &= \mathbf{g}_h. \end{cases} \tag{3.9}$$

This scheme allows the full parallelization in space of the problem. We solve in each $\tilde{\Omega}_q$ the continuous problem $\partial_t u = \mathcal{L}_q u + kf_q$, where $\mathcal{L}_q = \partial_x(\tilde{\chi}_q\partial_x)$ is the elliptic operator with Dirichlet boundary conditions in $\partial\tilde{\Omega}_q$, and $f_q = \chi_q f$. From the discrete point of view, the Dirichlet condition is given by $\mathbf{w}^n$ and $\mathbf{z}^n$; see (3.3).

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

$$\|\phi\|_{C^j(\Omega)} \equiv \sup_{\alpha \leq j} \max_{\overline{\Omega}} |\frac{\partial^\alpha \phi}{\partial x^\alpha}|,$$

and we have the following result about a partition of unity; the main ideas of the proof can be found in [7].

Theorem 3.2. *Given a suitable collection of overlapping subdomains* $\{\Omega_s^*\}_{s=1,\ldots,\hat{s}}$ *with overlap* 2β*, there exists a smooth partition of unity* $\{\tilde{\chi}_s\}_{j=1,\ldots,\hat{s}}$ *satisfying*

$$\|\tilde{\chi}_s\|_{C^j(\Omega)} \leq c(2\beta)^{-j}, \tag{3.10}$$

where $c > 0$ *is a constant independent of* β*.*

Remark 3.3. We can use domain decomposition to split the operator in a d-dimensional geometry. For example, in two dimensions, two possible open coverings of a domain $\Omega \in \mathbb{R}^2$ can be seen in Figure 3.3. For the covering in the left of Figure 3.3, we can construct a partition of unity in a similar way viewed in Section 3.3, with a higher degree of differentiability. For the covering in the right of Figure 3.3, we split the operator $\mathcal{L}$ in four operators $\mathcal{L}_q, q = 1, \ldots, 4$, each operator for one of the sets composing the open covering. For more than two dimensions, the splitting of the operators is similar. We do not analyze the splitting of the operator $\mathcal{L}$ for more than one dimension in this thesis.

Remark 3.4. The matrices A_q are symmetric by construction; see (3.8). They are symmetric negative semidefinite because, in each line of A_q, the diagonal element is negative and its absolute value is greater or equal to the sum of the absolute values of the off-diagonal elements in the line. The negative semidefiniteness follows from the Gershgorin's Theorem.

3.4 Error analysis of the splitting scheme

As we mentioned before, given a partition of unity $\{\tilde{\chi}_q\}_{q=1,2}$, we take the operator $\mathcal{L} = \partial_{xx} = \partial_x\partial_x$ and the function $f(x,t)$, and obtain $\mathcal{L}_q = \partial_x(\tilde{\chi}_q\partial_x)$ and $f_q(x,t) = \tilde{\chi}_q f(x,t)$. We have that $\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2$, and by construction, $\mathcal{L}_q$ are self-adjoint operators [7]. We take the matrices A_q as the discretization of the operators $\mathcal{L}_q$, hence, we have $A = A_1 + A_2$. The $\mathbb{R}^{\hat{m}-1}$ vectors $\mathbf{f}_q^{n+1}, q = 1, 2$, are the spatial evaluation of $\tilde{\chi}_q f$ in Ω_h at time $(n+1)k$.

We now provide an estimate for the order of accuracy of the first-order splitting method using the concept of local truncation error for the scheme. Analogously to the BTCS scheme, we define the local truncation error for this method as:

Definition 3.5. *The local truncation error* $\boldsymbol{\ell}_{sp}^{n+1}(\phi)$ *for the first-order approximate splitting is:*

$$\boldsymbol{\ell}_{sp}^{n+1}(\phi) = (I - kA)\boldsymbol{\phi}^{n+1} - \boldsymbol{\phi}^n - k(\boldsymbol{P\phi})^{n+1} + k^2 A_1 A_2 \boldsymbol{\phi}^{n+1} - k^2 A_1(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{P\phi})^{n+1}. \tag{3.11}$$

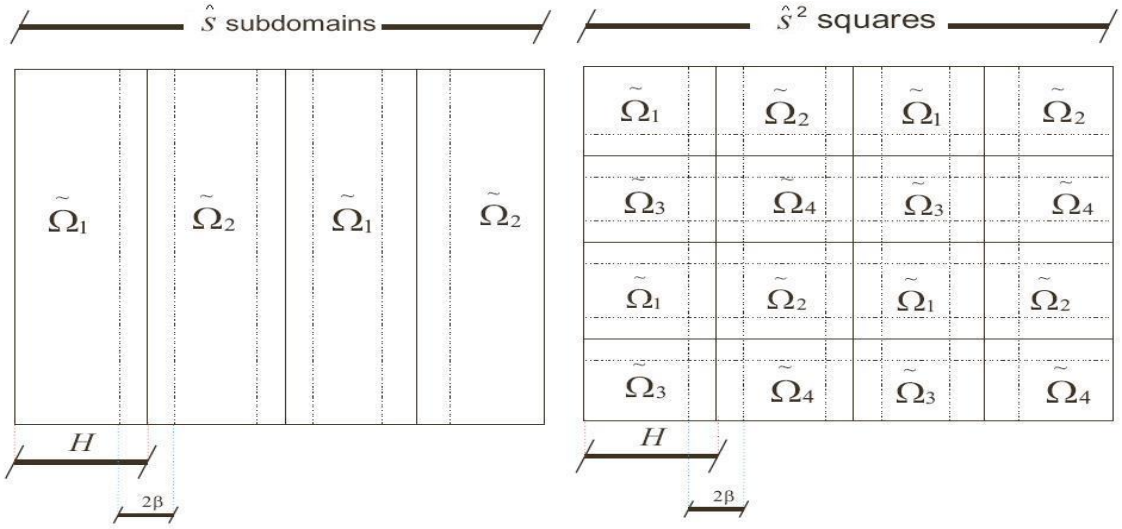

Figure 3.3: Two examples of coverings for two-dimensional domains. We use $\hat{s} = 4$ in this examples.

for any smooth function ϕ. *The* $\mathbb{R}^{\hat{m}-1}$ *vectors* $\boldsymbol{\phi}^{n+1}$, $(\boldsymbol{P\phi})^{n+1}$ *and* $(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{P\phi})^{n+1}$ *correspond to the evaluation of* ϕ, $(P\phi)$ *and* $(\tilde{\chi}_2 P\phi)$ *in* Ω_h *at time* $(n+1)k$. *Using the Definition 2.6, we obtain the following relation between the local truncation error for the splitting method and for the BTCS scheme:*

$$\boldsymbol{\ell}_{sp}^{n+1}(\phi) = \boldsymbol{\ell}_{be}^{n+1}(\phi) + k^2 A_1 A_2 \boldsymbol{\phi}^{n+1} - k^2 A_1 (\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{P\phi})^{n+1}. \tag{3.12}$$

From (3.12), the splitting method has the same order of accuracy of BTCS plus the terms $k^2 A_1 A_2 \boldsymbol{\phi}^{n+1}$ and $-k^2 A_1 (\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{P\phi})^{n+1}$. Since the BTCS scheme is accurate of order (1,2), the splitting scheme is accurate of order (1,2) plus the contribution of the terms $k^2 A_1 A_2 \boldsymbol{\phi}^{n+1}$ and $-k^2 A_1 (\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{P\phi})^{n+1}$.

We now investigate the influence of the term $k^2 A_1 A_2 \boldsymbol{\phi}^{n+1}$ for the first-order splitting scheme. To simplify the analysis we introduce the following notation:

$$\begin{aligned}
\mathcal{I}_h^x f(x) &= h^{-1} \int_x^{x+h} f(\eta) d\eta, \\
\mathcal{I}_h^{xx} f(x) &= h^{-2} \int_{x-h}^{x} \int_{\xi}^{\xi+h} f(\eta) d\eta d\xi, \\
\mathcal{I}_{h,1/2}^x f(x) &= h^{-1} \int_{x-h/2}^{x+h/2} f(\eta) d\eta, \\
\partial^\alpha \phi(x) &= \frac{\partial^\alpha \phi(x)}{\partial x^\alpha}.
\end{aligned}$$

Theorem 3.6. *If $\phi \in C_0^4(\Omega)$, the discretization $(A_1A_2\boldsymbol{\phi})_m$ satisfies*

$$|(A_1A_2\boldsymbol{\phi})_m| \leq c_1\beta^{-3}\|\phi\|_{C^4(\Omega)},$$

where $(A_1A_2\boldsymbol{\phi})_m$ is the m-component of the vector $(A_1A_2\boldsymbol{\phi})$ and $\boldsymbol{\phi}$ is the spatial interpolation of the function ϕ.

Proof: First observe that the finite difference approximation for $(A_2\boldsymbol{\phi})_m$ is:

$$(A_2\boldsymbol{\phi})_m = \frac{\tilde{\chi}_2(x_m+\frac{h}{2})\left(\frac{\boldsymbol{\phi}_{m+1}-\boldsymbol{\phi}_m}{h}\right) - \tilde{\chi}_2(x_m-\frac{h}{2})\left(\frac{\boldsymbol{\phi}_m-\boldsymbol{\phi}_{m-1}}{h}\right)}{h}.$$

Applying repeatedly the fundamental theorem of calculus and using the multi-index notation, we have:

$$(A_2\boldsymbol{\phi})_m = \left(\mathcal{I}_{h,1/2}^{x_m}\partial_x\tilde{\chi}_2\right)\left(\mathcal{I}_h^{x_m}\partial_x\phi\right) - \tilde{\chi}_2\left(x_m-\frac{h}{2}\right)\left(\mathcal{I}_h^{x_mx_m}\partial_{xx}\phi\right). \tag{3.13}$$

We have an analogous expression for $(A_1\boldsymbol{\phi})_m$. Using $A_2\boldsymbol{\phi}$ instead of $\boldsymbol{\phi}$ in the expression for $(A_1\boldsymbol{\phi})_m$ gives:

$$(A_1A_2\boldsymbol{\phi})_m = \left(\mathcal{I}_{h,1/2}^{x_m}\partial_x\tilde{\chi}_1\right)\left(\mathcal{I}_h^{x_m}\partial_x(A_2\boldsymbol{\phi})_m\right) - \tilde{\chi}_1\left(x_m-\frac{h}{2}\right)\left(\mathcal{I}_h^{x_mx_m}\partial_{xx}(A_2\boldsymbol{\phi})_m\right).$$

The resulting expression can be simplified by the following observation. The partial derivative operator ∂^α "commutes" with all the integral operators $\mathcal{I}_h^{x_m}$, $\mathcal{I}_h^{x_mx_m}$, etc. The commutativity of ∂_x and $\mathcal{I}_h^{x_m}$ can be seem by the following example:

$$\partial_x\mathcal{I}_h^{x_m}F(x) = \partial_x\left(h^{-1}\int_{x_m}^{x_m+h}F(\xi)d\xi\right) = \left(h^{-1}\int_{x_m}^{x_m+h}\partial_\xi F(\xi)d\xi\right) = \mathcal{I}_h^{x_m}\partial_x F(x).$$

Analogously, we have for $\mathcal{I}_h^{x_mx_m}$ and $\mathcal{I}_{h,1/2}^{x_m}$:

$$\begin{aligned}\partial_x\mathcal{I}_h^{x_mx_m}F(x) &= \mathcal{I}_h^{x_m}\partial_x F(x)\\ \partial_x\mathcal{I}_{h,1/2}^{x_m}F(x) &= \mathcal{I}_{h,1/2}^{x_m}\partial_x F(x).\end{aligned}$$

Using the fact that ∂_x commutes with $\mathcal{I}_h^x, \mathcal{I}_h^{xx}$ and $\mathcal{I}_{h,1/2}^x$ and performing elementary calculations we obtain:

$$\begin{aligned}(A_1A_2\boldsymbol{\phi})_m = &\left(\mathcal{I}_{h,1/2}^{x_m}\partial_x\tilde{\chi}_1\right)\left(\mathcal{I}_h^{x_m}\left(\mathcal{I}_{h,1/2}^{x_m}\partial_{xx}\tilde{\chi}_2\right)\left(\mathcal{I}_h^{x_m}\partial_x\phi\right) + \left(\mathcal{I}_{h,1/2}^{x_m}\partial_x\tilde{\chi}_2\right)\left(\mathcal{I}_h^{x_m}\partial_{xx}\phi\right)\right.\\
&-\partial_x\tilde{\chi}_2\left(x_m-h/2\right)\left(\mathcal{I}_h^{x_mx_m}\partial_{xx}\phi\right) - \tilde{\chi}_2\left(x_m-h/2\right)\left(\mathcal{I}_h^{x_mx_m}\partial_{xxx}\phi\right))+\\
&-\tilde{\chi}_1\left(x_m-h/2\right)\mathcal{I}_h^{x_mx_m}\left(\left(\mathcal{I}_{h,1/2}^{x_m}\partial_{xxx}\tilde{\chi}_2\right)\left(\mathcal{I}_h^{x_m}\partial_x\phi\right) + \left(\mathcal{I}_{h,1/2}^{x_m}\partial_{xx}\tilde{\chi}_2\right)\left(\mathcal{I}_h^{x_m}\partial_{xx}\phi\right)+\right.\\
&+\left(\mathcal{I}_{h,1/2}^{x_m}\partial_{xx}\tilde{\chi}_2\right)\left(\mathcal{I}_h^{x_m}\partial_{xx}\phi\right) + \left(\mathcal{I}_{h,1/2}^{x_m}\partial_x\tilde{\chi}_2\right)\left(\mathcal{I}_h^{x_m}\partial_{xxx}\phi\right)+\\
&-\partial_{xx}\tilde{\chi}_2\left(x_m-h/2\right)\left(\mathcal{I}_h^{x_mx_m}\partial_{xx}\phi\right) - \partial_x\tilde{\chi}_2\left(x_m-h/2\right)\left(\mathcal{I}_h^{x_mx_m}\partial_{xxx}\phi\right)+\\
&\left.-\partial_x\tilde{\chi}_2\left(x_m-h/2\right)\left(\mathcal{I}_h^{x_mx_m}\partial_{xxx}\phi\right) - \tilde{\chi}_2\left(x_m-h/2\right)\left(\mathcal{I}_h^{x_mx_m}\partial_{xxxx}\phi\right)\right).\end{aligned} \tag{3.14}$$

Now, we can obtain an estimate for $|(A_1A_2\boldsymbol{\phi})_m|$ in terms of the value of the derivatives of ϕ, $\tilde{\chi}_1$ and $\tilde{\chi}_2$.

From the mean value property of integrals, we have that $|\mathcal{I}_h^x\phi(x)| \leq \max_{\xi\in[x_m,x_m+1]}|\phi(\xi)|$. Using that property and taking the absolute value of the expression (3.14), we can write:

$$|(A_1A_2\boldsymbol{\phi})_m| \leq \sum_{\substack{(\alpha_1,\alpha_2,\alpha_3)\\ |\alpha_1|+|\alpha_2|+|\alpha_3|=4\\ |\alpha_1|\leq 1}} c_{\alpha_1,\alpha_2,\alpha_3}|\partial^{\alpha_1}\tilde{\chi}_1|_{C^0(\Omega)}|\partial^{\alpha_2}\tilde{\chi}_2|_{C^0(\Omega)}|\partial^{\alpha_3}\phi|_{C^0(\Omega)},$$

where $c_{\alpha_1,\alpha_2,\alpha_{\hat{q}+1}} = 0$ or 1. Using the definition of the Hölder norm we obtain:

$$|(A_1A_2\boldsymbol{\phi})_m| \leq \sum_{\substack{(\alpha_1,\alpha_2,\alpha_3)\\ |\alpha_1|+|\alpha_2|+|\alpha_3|=4\\ |\alpha_1|\leq 1}} c_{\alpha_1,\alpha_2,\alpha_3}\|\tilde{\chi}_1\|_{C^{\alpha_1}(\Omega)}\|\tilde{\chi}_2\|_{C^{\alpha_2}(\Omega)}\|\phi\|_{C^{\alpha_3}(\Omega)}.$$

Applying the estimation obtained in Theorem 3.2, we have:

$$|(A_1A_2\boldsymbol{\phi})_m| \leq c_1\beta^{-3}\|\phi\|_{C^4(\Omega)},$$

which proves our assertion.□

Theorem 3.7. *If $\phi \in C_0^4(\Omega)$ and $\partial_t\phi \in C_0^2(\Omega)$, the discretization $(A_1\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{P\phi})_m$ satisfies*

$$|(A_1(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{P\phi}))_m| \leq \beta^{-2}(c_2\|\phi\|_{C^4(\Omega)} + c_3\|\partial_t\phi\|_{C^2(\Omega)}),$$

where $(A_1(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{P\phi}))_m$ is the m-component of the vector $(A_1(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{P\phi}))$, $\boldsymbol{\phi}$ is the spatial interpolation of the function ϕ, $\phi_t := \partial_t\phi$ and $\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{P\phi}$ is the spatial interpolation of $\tilde{\chi}_2P\phi$.

Proof: To simplify our notation we make $\psi := P\phi = \partial_t\phi - \partial_{xx}\phi$, consequently we have $\boldsymbol{\psi} = \boldsymbol{P\phi}$. First observe that the finite difference approximation of $(A_1(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{\psi}))_m$ is:

$$(A_1(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{\psi}))_m = \frac{\tilde{\chi}_1(x_m+\frac{h}{2})\left(\frac{(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{\psi})_{m+1}-(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{\psi})_m}{h}\right) - \tilde{\chi}_1(x_m-\frac{h}{2})\left(\frac{(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{\psi})_m-(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{\psi})_{m-1}}{h}\right)}{h}.$$

Applying repeatedly the fundamental theorem of calculus and using the multi-index notation, we have:

$$\begin{aligned}(A_1(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{\psi}))_m &= \left(\mathcal{I}_{h,1/2}^{x_m}\partial_x\tilde{\chi}_1\right)\left(\mathcal{I}_h^{x_m}\partial_x(\tilde{\chi}_2\psi)\right) - \tilde{\chi}_1\left(x_m-\frac{h}{2}\right)\left(\mathcal{I}_h^{x_mx_m}\partial_{xx}(\tilde{\chi}_2\psi)\right),\\ &= \left(\mathcal{I}_{h,1/2}^{x_m}\partial_x\tilde{\chi}_1\right)\left(\mathcal{I}_h^{x_m}(\partial_x\tilde{\chi}_2+\partial_x\psi)\right) - \tilde{\chi}_1\left(x_m-\frac{h}{2}\right)\mathcal{I}_h^{x_mx_m}(\psi\partial_{xx}\tilde{\chi}_2 +\\ &\quad +\partial_x\tilde{\chi}_2\partial_x\psi + \partial_x\psi\partial_x\tilde{\chi}_2 + \tilde{\chi}_2\partial_{xx}\psi). \qquad (3.15)\end{aligned}$$

Using the fact that ∂_x commutes with $\mathcal{I}_h^x, \mathcal{I}_h^{xx}$ and $\mathcal{I}_{h,1/2}^x$ and the mean value property for integrals and taking modulus of expression (3.15), we can obtain an estimate for $|(A_1\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{\psi})_m|$ in terms of the value of the derivatives of ψ, $\tilde{\chi}_1$ and $\tilde{\chi}_2$, given by:

$$|(A_1(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{\psi}))_m| \leq \sum_{\substack{(\alpha_1,\alpha_2,\alpha_3)\\ |\alpha_1|+|\alpha_2|+|\alpha_3|=2\\ |\alpha_1|\leq 1}} c_{\alpha_1,\alpha_2,\alpha_3}|\partial^{\alpha_1}\tilde{\chi}_1|_{C^0(\Omega)}|\partial^{\alpha_2}\tilde{\chi}_2|_{C^0(\Omega)}|\partial^{\alpha_3}\phi|_{C^0(\Omega)},$$

where $c_{\alpha_1,\alpha_2,\alpha_{\hat{q}+1}} = 0$ or 1. Using the definition of the Hölder norm we obtain:

$$|(A_1(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{\psi}))_m| \leq \sum_{\substack{(\alpha_1,\alpha_2,\alpha_3)\\ |\alpha_1|+|\alpha_2|+|\alpha_3|=2\\ |\alpha_1|\leq 1}} c_{\alpha_1,\alpha_2,\alpha_3}\|\tilde{\chi}_1\|_{C^{\alpha_1}(\Omega)}\|\tilde{\chi}_2\|_{C^{\alpha_2}(\Omega)}\|\psi\|_{C^{\alpha_3}(\Omega)}.$$

Using the estimation derived in Theorem 3.2 we obtain:

$$|(A_1(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{\psi}))_m| \leq c\beta^{-2}\|\psi\|_{C^2(\Omega)}. \tag{3.16}$$

We know that $\psi = P\phi = \partial_t\phi - \partial_{xx}\phi$, so we have that

$$\begin{aligned}\|\psi\|_{C^2(\Omega)} &\leq c_2\|\partial_t\phi\|_{C^2(\Omega)} + c_3\|\partial_{xx}\phi\|_{C^2(\Omega)}\\ &\leq c_2\|\partial_t\phi\|_{C^2(\Omega)} + c_3\|\phi\|_{C^4(\Omega)},\end{aligned} \tag{3.17}$$

where c_2 and c_3 are constants.

Substituting (3.17) in (3.16) we obtain:

$$|(A_1(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{\psi}))_m| \leq \beta^{-2}(c_2\|\partial_t\phi\|_{C^2(\Omega)} + c_3\|\phi\|_{C^4(\Omega)}), \tag{3.18}$$

which proves our assertion.□

Now, we can estimate the contribution of the terms $k^2A_1A_2\boldsymbol{\phi}$ and $k^2A_1(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{P\phi})$ to the local truncation error.

Theorem 3.8. *The local truncation error $\boldsymbol{\ell}_{sp}^{n+1}(\phi)$ of the splitting method with domain decomposition satisfies*

$$\begin{aligned}\|\boldsymbol{\ell}_{sp}^{n+1}(\phi)\|_h \leq\ & \|\boldsymbol{\ell}_{be}^{n+1}(\phi)\|_h + c_1k^2\left(\frac{H}{\beta}\right)^{5/2}\left(\frac{1}{H}\right)^3\|\phi\|_{C^4(\Omega)} +\\ &+k^2\left(\frac{H}{\beta}\right)^{3/2}\left(\frac{1}{H}\right)^2(c_2\|\partial_t\phi\|_{C^2(\Omega)} + c_3\|\phi\|_{C^4(\Omega)})\end{aligned} \tag{3.19}$$

for any smooth function ϕ, where $\boldsymbol{\ell}_{be}^{n+1}(\phi)$ is the local truncation error of the backward Euler method at time $(n+1)k$, c_1, c_2 and c_3 are constants. The order of accuracy of the method is

$$\mathcal{O}(k)+\mathcal{O}(h^2)+\mathcal{O}\left(k\left(\frac{H}{\beta}\right)^{5/2}\left(\frac{1}{H}\right)^3\right)+\mathcal{O}\left(k\left(\frac{H}{\beta}\right)^{3/2}\left(\frac{1}{H}\right)^2\right). \tag{3.20}$$

Proof: We have from (3.11) that the local truncation error is:

$$\boldsymbol{\ell}_{sp}^{n+1}(\phi) \;=\; \boldsymbol{\ell}_{be}^{n+1}(\phi)+k^2A_1A_2\boldsymbol{\phi}^{n+1}-k^2A_1(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{P}\boldsymbol{\phi})^{n+1}, \tag{3.21}$$

where $\boldsymbol{\phi}^{n+1}$ corresponds to the evaluation of ϕ in Ω_h at time $(n+1)k$. The matrix A_1A_2 is of the form:

$$\begin{pmatrix} O & & & & & & & & \\ & K & & & & & & & \\ & & O & & & & & & \\ & & & K^T & & & & & \\ & & & & O & & & & \\ & & & & & \ddots & & & \\ & & & & & & O & & \\ & & & & & & & K & \\ & & & & & & & & O \end{pmatrix} \tag{3.22}$$

where O is a matrix with all elements equal to zero. If we take the overlap $2\beta=\delta h<H$, where δ is a positive even integer, K is a $(\delta+3)\times(\delta+3)$ matrix. If we take $\hat{s}=1/H$ as the number of subdomains that subdivides $(0,L)$, then we have $\hat{s}-1$ matrices of the form K and K^T. The matrix K comes from the product of $\tilde{\chi}_1$ in the right boundary of Ω_1^* and $\tilde{\chi}_2$ in the left boundary of Ω_2^*, while its transpose comes from the product of $\tilde{\chi}_2$ in the left boundary of Ω_2^* and $\tilde{\chi}_1$ in the right boundary of Ω_1^*. We have a total of $(\delta+3)((1/H)-1)$ terms different from zero in the vector $(A_1A_2\boldsymbol{\phi})$. In a similar way, we can see that the vector $(A_1(\boldsymbol{\chi}_{\mathbf{2}}\boldsymbol{P}\boldsymbol{\phi}))$ has a total of $\mathcal{O}(\delta/H)$ components differents from zero. Taking the Euclidean norm of (3.21), which we denote by $\|\cdot\|_2$, and using the estimates from Theorems 3.6 and 3.7, we obtain:

$$\begin{aligned}
\|\boldsymbol{\ell}_{sp}^{n+1}(\phi)\|_2 &\leq \|\boldsymbol{\ell}_{be}^{n+1}(\phi)\|_2+k^2\|A_1A_2\boldsymbol{\phi}\|_2+k^2\|A_1(\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{P}\boldsymbol{\phi})^{n+1}\| \\
&\leq \|\boldsymbol{\ell}_{be}^{n+1}(\phi)\|_2+c_1\sqrt{(\delta+3)((1/H)-1)}k^2(\beta)^{-3}\|\phi\|_{C^4(\Omega)}+ \\
&\quad +k^2\sqrt{\delta/H}\beta^{-2}(c_2\|\partial_t\phi\|_{C^2(\Omega)}+c_3\|\phi\|_{C^4(\Omega)}) \\
&\leq \|\boldsymbol{\ell}_{be}^{n+1}(\phi)\|_2+c_1\sqrt{\delta/H}k^2(\beta)^{-3}\|\phi\|_{C^4(\Omega)}+ \\
&\quad +\sqrt{\delta/H}k^2(\beta)^{-2}(c_2\|\partial_t\phi\|_{C^2(\Omega)}+c_3\|\phi\|_{C^4(\Omega)}).
\end{aligned}$$

Multiplying the whole expression by $\sqrt{h}$ and using that $\sqrt{h}\|.\|_2 \equiv \|.\|_h$, we have:

$$\begin{aligned}\|\boldsymbol{\ell}_{sp}^{n+1}(\phi)\|_h &\leq \|\boldsymbol{\ell}_{be}^{n+1}(\phi)\|_h + c\sqrt{\beta/H}k^2(\beta)^{-3}\|\phi\|_{C^4(\Omega)} + \\ &\quad +\sqrt{\beta/H}k^2(\beta)^{-2}(c_2\|\partial_t\phi\|_{C^2(\Omega)} + c_3\|\phi\|_{C^4(\Omega)}) \\ &\leq \|\boldsymbol{\ell}_{be}^{n+1}(\phi)\|_h + c_1k^2\left(\frac{1}{H}\right)^3\left(\frac{H}{\beta}\right)^{5/2}\|\phi\|_{C^4(\Omega)} + \\ &\quad +k^2\left(\frac{1}{H}\right)^2\left(\frac{H}{\beta}\right)^{3/2}(c_2\|\partial_t\phi\|_{C^2(\Omega)} + c_3\|\phi\|_{C^4(\Omega)}). \end{aligned} \tag{3.23}$$

The order of accuracy of the method is the order of the local truncation error divided by k. So, using the Lax-Richtmyer Theorem, the splitting method converges and is accurate of order

$$\mathcal{O}(k) + \mathcal{O}(h^2) + \mathcal{O}\left(k\left(\frac{H}{\beta}\right)^{5/2}\left(\frac{1}{H}\right)^3\right) + \mathcal{O}\left(k\left(\frac{H}{\beta}\right)^{3/2}\left(\frac{1}{H}\right)^2\right). \tag{3.24}$$

□

Remark 3.9. We can implement the splitting method using $f_1 = f$ and $f_2 = 0$. With this choice of splitting for the function f we have to solve the problem:

$$\begin{cases} (I - kA_1)\mathbf{z}^n &= \mathbf{w}^n + k\mathbf{f}^{n+1} \\ (I - kA_2)\mathbf{w}^{n+1} &= \mathbf{z}^n \\ \mathbf{w}^0 &= \mathbf{g}_h, \end{cases} \tag{3.25}$$

which, after elementary calculations, becomes:

$$\begin{cases} (I - kA_1)(I - kA_2)\mathbf{w}^{n+1} &= \mathbf{w}^n + k\mathbf{f}^{n+1} \\ \mathbf{w}^0 &= \mathbf{g}_h, \end{cases} \tag{3.26}$$

The analysis of (3.26) is similar to the analysis of (3.4). This method is also stable and is convergent of order:

$$\mathcal{O}(k) + \mathcal{O}(h^2) + \mathcal{O}\left(k\left(\frac{H}{\beta}\right)^{5/2}\left(\frac{1}{H}\right)^3\right), \tag{3.27}$$

because there is no influence of the term $k^2A_1\tilde{\boldsymbol{\chi}}_{\mathbf{2}}\boldsymbol{P\phi}$ in the order of accuracy of the method

The choice (3.25) of the splitting of the function $f(x,t)$ also permits some parallelization of the scheme, but not as much as for the scheme (3.9). In the choice (3.25), the boundary conditions for the operator $\mathcal{L}_q$ will be the same as for the problem (2.1), however, values of $\mathbf{z}^n$ need to be computed outside of $\tilde{\Omega}_1$.

3.5 Numerical results

We solve the following problem in the domain $(0,1)\times(0,1)$:

$$\begin{cases} \partial_t u &= \partial_{xx} u + f(x,t) & \text{in } (0,1)\times(0,1) \\ u(x,0) &= x(1-x)\exp(x) & \text{on } (0,1)\times\{0\} \\ u(x,t) &= 0 & \text{on } \{0,1\}\times(0,1), \end{cases} \tag{3.28}$$

where $f(x,t) = x\exp(x)[(x-1)\pi\sin(\pi t) + (3+x)\cos(\pi t)]$. The function $f(x,t)$ and $u(x,0)$ are chosen such that the analytical solution of problem (3.28) is $u(x,t) = x(1-x)\exp(x)\cos\pi t$ in $(0,1)\times(0,1)$. The BTCS scheme and the Sequential Splitting solutions are denoted respectively by BT and SS. We calculate the error of the methods at time $T=1$ for all experiments. The error is $\|\mathbf{u}^{\hat{n}} - \mathbf{w}^{\hat{n}}\|_h$, where $\mathbf{u}_m^{\hat{n}} = u(mh, \hat{n}k)$, $\mathbf{w}^{\hat{n}}$ is the numerical solution given by the splitting method and $\|.\|_h = \sqrt{h}\|.\|_2$.

Experiment 1 - In Tables 3.1-3.2, we keep constant the spatial discretization mesh size $h = 1/1024$ and the time discretization step $k = 1/2048$. In each row of these tables, we keep H/β constant, while in each column we keep constant the number of subdomains $\hat{s} = 1/H$. In Table 3.1, we solve (3.9), while in Table 3.2, we solve (3.26). We examine the effect of H/β on the error. The values presented in Table 3.1 are multiplied by 10^3. The error for BT is 0.0680×10^{-3} when $k = 1/2048$ and $h = 1/1024$.

Table 3.1: We divide $(0,1)$ in $\hat{s}$ subdomains of length $H = 1/\hat{s}$ and overlap β and keep the mesh parameters $k = 1/2048$ and $h = 1/1024$. We use the SS scheme (3.9). The values for the error at $T = 1$ are multiplied by 10^3.

H	1/2	1/4	1/8	1/16	1/32
$H/\beta = 4$	0.3859	2.9652	17.8233	74.9567	238.3453
$H/\beta = 8$	0.7271	5.7262	31.9605	125.7292	334.9536
$H/\beta = 16$	1.3536	10.5073	55.3489	198.9055	412.0838
$H/\beta = 32$	2.4300	18.5890	92.8414	286.1308	451.2088

We see that the errors for the solution of (3.9) and (3.26) are very similar. So the implementation of full parallelization in space does not cause a deterioration on the error.

We verify that the error deteriorates when we increase the number of subdomains $\hat{s}$, and keep the ratio H/β constant. This behavior was predicted in Theorem 3.8. The error deteriorates

Table 3.2: We divide $(0,1)$ in $\hat{s}$ subdomains of length $H = 1/\hat{s}$ and overlap β and keep the mesh parameters $k = 1/2048$ and $h = 1/1024$. We use the SS scheme (3.26). The values for the error at $T = 1$ are multiplied by 10^3.

H	1/2	1/4	1/8	1/16	1/32
$H/\beta = 4$	0.2701	3.3550	18.2621	75.3914	238.6964
$H/\beta = 8$	0.5476	6.1325	32.4113	126.1702	335.3240
$H/\beta = 16$	1.1562	10.9227	55.8070	199.3570	412.7376
$H/\beta = 32$	2.2283	19.0108	93.3087	286.7788	451.9897

when we increase the quotient H/β, and keep the number of subdomains $\hat{s}$ constant. This means that when we increase H/β and keep $\hat{s}$ constant, then β decreases, i.e., the overlap between the subdomains decreases. When we decrease the overlap, we decrease the communication between the neighbors subdomains, and this justifies the error deterioration. Summarizing, the decrease of the ratio H/β deteriorates the error for SS.

Experiment 2 - In this experiment, we obtain four tables using the SS scheme (3.9). For each table, we keep H/β constant. In each row, we vary the number of subdomains, and keep constant the time and spatial discretization mesh sizes such that $k = h$. In each column, we vary the time and spatial discretizations mesh sizes such that $k = h$ and keep constant the number of subdomains. We examine the effect of H/β in the error when we vary the mesh discretization sizes. The values for the error at time $T = 1$ presented are multiplied by 10^3.

Table 3.3: We divide $(0,1)$ in $\hat{s}$ subdomains of length $H = 1/\hat{s}$ and overlap β and make $H/\beta = 32$. In the column labeled BT, the error for the BTCS scheme is presented. The other columns have the error for the SS scheme for different values of H. The values for the error are multiplied by 10^3.

H	1/2	1/4	1/8	1/16	BT
$h = k = 1/512$	8.1564	61.6603	247.8635	432.1676	0.2718
$h = k = 1/1024$	4.5175	34.7734	165.2721	388.1498	0.1359
$h = k = 1/2048$	2.4357	18.9688	96.9683	305.8543	0.0679

Table 3.4: We divide $(0,1)$ in $\hat{s}$ subdomains of length $H = 1/\hat{s}$ and overlap β and make $H/\beta = 16$. The values for the error are multiplied by 10^3.

H	1/2	1/4	1/8	1/16
$h = k = 1/512$	4.7515	36.0040	168.6354	392.0056
$h = k = 1/1024$	2.5566	19.7149	100.6397	310.0240
$h = k = 1/2048$	1.3526	10.5565	56.4390	206.3469

Table 3.5: We divide $(0,1)$ in $\hat{s}$ subdomains of length $H = 1/\hat{s}$ and overlap β and make $H/\beta = 8$. The values are multiplied by 10^3.

H	1/2	1/4	1/8	1/16
$h = k = 1/512$	2.6738	20.3071	105.0133	314.5113
$h = k = 1/1024$	1.4042	10.8733	59.1623	213.0031
$h = k = 1/2048$	0.7261	5.7347	32.1099	128.0164

In the Tables (3.3)-(3.6), as predicted in our analysis, we verify that the error is almost linearly dependent with respect to the size of the time discretization step. This linear relation is less clear for higher values of $\hat{s}$, since the term $\mathcal{O}\left(k\left(\frac{1}{H}\right)^3\left(\frac{H}{\beta}\right)^{5/2}\right)$ dominates the error with respect to the BTCS scheme.

We also can see from these tables, by keeping $\hat{s}$, k and h constant, that the same conclusion as above follows, that is, the error decreases almost linearly when we decrease the ratio H/β.

However, from these results, we also see that our estimate (3.24) of the error is not sharp. It is more complete and accurate than the estimate of the error obtained in [7].

Remark 3.10. A second-order splitting method (Strang-Marchuk) was also implemented during our research, however the results for this method were worse than for SS and are not presented in this thesis.

Table 3.6: We divide $(0,1)$ in $\hat{s}$ subdomains of length $H = 1/\hat{s}$ and overlap β and make $H/\beta = 4$. The values are multiplied by 10^3.

H	$1/2$	$1/4$	$1/8$	$1/16$
$h = k = 1/512$	1.4746	10.9644	62.0467	220.9803
$h = k = 1/1024$	0.7580	5.7481	33.6596	133.8819
$h = k = 1/2048$	0.3853	2.9676	17.8515	75.2981

Chapter 4

Two-grid scheme for the solution of the parabolic PDE's

4.1 Introduction

In the preceding chapters, we have studied numerical methods to approximate the solution of a parabolic PDE on a spatial grid with mesh size h and time step size k. These methods consist of solving, in each time step, equations of the type:

$$B\mathbf{z} = \mathbf{b}, \tag{4.1}$$

where B, $\mathbf{z}$ and $\mathbf{b}$ depend on the scheme used for the discretization of the problem. For instance, in BTCS we have $B := B_{be} = (I - kA)$, $\mathbf{z} = \mathbf{v}^{n+1}$ and $\mathbf{b} = \mathbf{v}^n + k\mathbf{f}^{n+1}$, which give us:

$$(I - kA)\mathbf{v}^{n+1} \;\; = \;\; \mathbf{v}^n + k\mathbf{f}^{n+1}, \tag{4.2}$$

while in the splitting method $B := B_{sp} = (I - kA_1)(I - kA_2)$, $\mathbf{z} = \mathbf{w}^{n+1}$ and $\mathbf{b} = \mathbf{w}^n + k\mathbf{f}^{n+1} - k^2 A_1 \mathbf{f}_2^{n+1}$, and we have:

$$(I - kA_1)(I - kA_2)\mathbf{w}^{n+1} \;\; = \;\; \mathbf{w}^n + k\mathbf{f}^{n+1} - k^2 A_1 \mathbf{f}_2^{n+1}. \tag{4.3}$$

When we use an operator splitting based on a spatial partition of unity of the domain, we have concluded in the preceding chapter that the accuracy of the discrete solution deteriorates as the number of subdomains increases. To deal with this latter drawback, in this chapter, we implement a coarse grid correction at each time step.

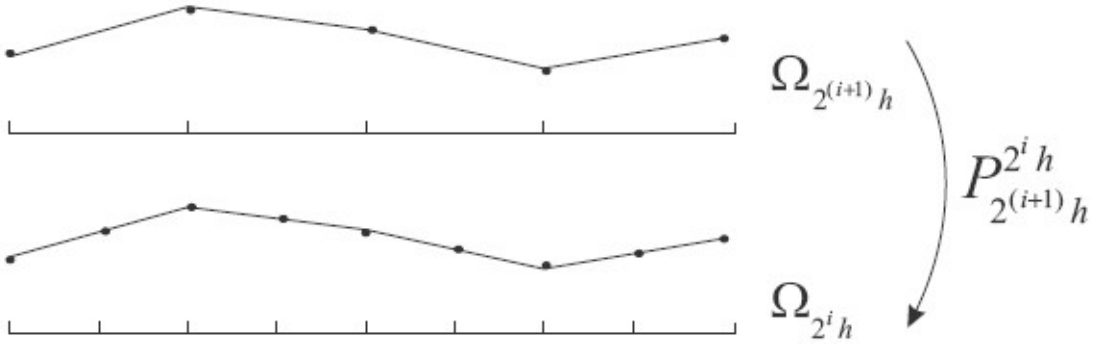

Figure 4.1: Interpolation of a vector on coarse grid $\Omega_{2^{i+1}h}$ to fine grid $\Omega_{2^i h}$.

4.2 Intergrid operators

Let $\Omega = I \subset \mathbb{R}$ be a finite interval with $I = (0, L)$. Let the fine grid on Ω be given by $\Omega_h = \{x \in \mathbb{R} : x = x_m = mh, m = 1, \ldots, \hat{m} - 1, h = L/\hat{m}\}$ and the coarse grid given by $\Omega_H = \{x \in \mathbb{R} : x = x_s = sH, s = 1, \ldots, \hat{s} - 1\}$, where $\hat{s} = 1/H = 2^{-\hat{i}}\hat{m}$, and $\hat{i}$ is a positive integer measuring the coarsening of h, i.e., $i = 0$ refers to h while $i = \hat{i}$ refers to H. On Ω_h, we define the grids $\Omega_{2^i h}$ by $\Omega_{2^i h} = \{x \in \mathbb{R} : x = x_j = j2^i h, j = 1, \ldots, \hat{m}/2^i - 1, h = L/\hat{m}\}$. To transfer data from the coarse grid $\Omega_{2^{i+1}h}$ to the fine grid $\Omega_{2^i h}$, we define the prolongation operator $P^{2^i h}_{2^{i+1}h} \in \mathbb{R}^{(\hat{m}/2^i - 1)\times(\hat{m}/2^{i+1} - 1)}$, in stencil notation, as:

$$P^{2^i h}_{2^{i+1}h} = \frac{1}{2}\begin{bmatrix} 1 & 2 & 1 \end{bmatrix}.$$

Composing the intergrid data transfers for all grids from Ω_h to Ω_H, we obtain the prolongation operator P^h_H from Ω_H to Ω_h as:

$$P^h_H = \prod_{i=0}^{\hat{i}-1} P^{2^i h}_{2^{i+1}h}.$$

The next step consists in defining the restriction operator R^H_h. This operator restricts values from Ω_h to Ω_H. We will proceed as above and define the operator $R^{2^{i+1}h}_{2^i h} \in \mathbb{R}^{(\hat{m}/2^{i+1} - 1)\times(\hat{m}/2^i - 1)}$. There are different forms to construct this operator. One is by direct injection, where the values at the coarse grid vector are equal to the values at the fine grid. Another choice is a full-weighting restriction, where the elements of the coarse grid vector are given by averages of the values of the neighbor points on the fine grid. We choose a full-weighting restriction proportional to the transpose of the prolongation operator. We have:

$$R^{2^{i+1}h}_{2^i h} = \frac{1}{2}\left(P^{2^i h}_{2^{i+1}h}\right)^T.$$

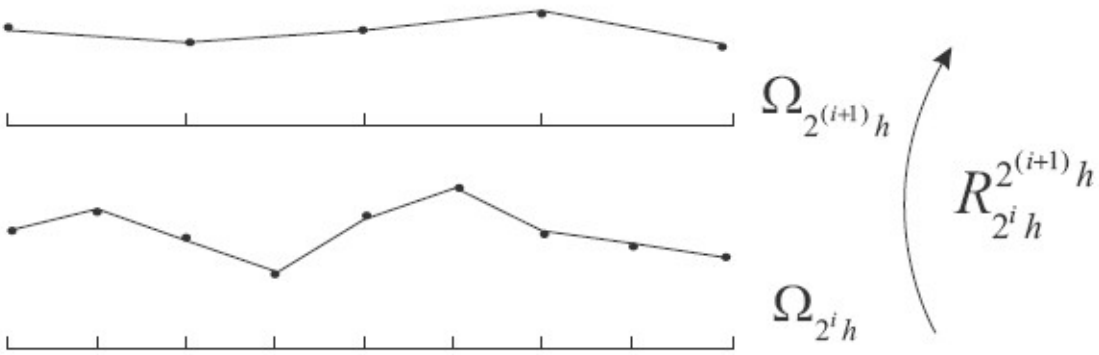

Figure 4.2: Restriction by full weighting of a fine-grid vector to the coarse grid.

So, similarly to the prolongation operator, the restriction operator R_h^H that transfer values from Ω_h to Ω_H is given by:

$$R_h^H = \frac{1}{2^{\hat{i}-1}} \left(P_H^h\right)^T. \tag{4.4}$$

4.3 Coarse grid correction

To define the discrete solution $\mathbf{z}_h^{n+1}$ of (4.3) in Ω_h with coarse grid correction, we first solve at each time step the splitting scheme (4.3). We have:

$$(I - kA_1)(I - kA_2)\tilde{\mathbf{z}}_h^{n+1} = \mathbf{z}_h^n + k\mathbf{f}^{n+1} - k^2 A_1 \mathbf{f}_2^{n+1}. \tag{4.5}$$

From the intermediate solution $\tilde{\mathbf{z}}_h^{n+1}$ we obtain in Ω_h the residue:

$$\mathbf{r}_h^{n+1} = \mathbf{z}_h^n + k\mathbf{f}^{n+1} - B_h \tilde{\mathbf{z}}_h^{n+1}, \tag{4.6}$$

where $B_h = B_{be} = (I - kA)$. The residue $\mathbf{r}_h^{n+1}$ measures how far from the BTCS discrete solution is our approximation $\tilde{\mathbf{z}}_h^{n+1}$ given by sequential splitting at each time step. We now restrict the residue to the grid Ω_H, obtaining:

$$\mathbf{r}_H^{n+1} = R_h^H \mathbf{r}_h^{n+1}. \tag{4.7}$$

We next solve a coarse problem with right-hand side $\mathbf{r}_H^{n+1}$ to update the solution $\tilde{\mathbf{z}}_h^{n+1}$. Note that the solution error for (4.3) at time step $(n+1)$ in the fine mesh is given by

$$\mathbf{e}_h^{n+1} = \mathbf{v}_h^{n+1} - \tilde{\mathbf{z}}_h^{n+1}, \tag{4.8}$$

where $\mathbf{v}_h^{n+1}$ is the BTCS solution of (4.3) at time $(n+1)k$ in the fine mesh.

If we multiply (4.8) by B_h we obtain the residue equation:

$$B_h\mathbf{e}_h^{n+1} = B_h\mathbf{v}_h^{n+1} - B_h\tilde{\mathbf{z}}_h^{n+1} = \mathbf{v}_h^n + k\mathbf{f}^{n+1} - B_h\tilde{\mathbf{z}}_h^{n+1} = \mathbf{r}_h^{n+1}. \tag{4.9}$$

Instead of solving the equation (4.9) in the grid Ω_h, we approximate $\mathbf{e}_h^{n+1}$ by the solution in the grid Ω_H.

To this objective, we need to approximate the operator B_h in Ω_H, which we call B_H. There are two ways to do this. One is called the discretization coarse grid approximation(DCA), and the other is called the Galerkin coarse grid approximation(GCA). The DCA consists in discretizing the Equation (2.1) in the coarse grid, while the GCA is obtained by defining B_H as:

$$B_H \;=\; R_h^H B_h P_H^h. \tag{4.10}$$

We can find a comparison of these two discretizations in [12]. We will use GCA in our problem because of its purely algebraic nature, and therefore, we do not need to discretize the differential equation again in Ω_H.

We obtain the error in the coarse grid by solving the equation $B_H\mathbf{e}_H^{n+1} = \mathbf{r}_H^{n+1}$ in the grid Ω_H, where B_H is defined in (4.10) and $\mathbf{r}_H^{n+1}$ in (4.7).

We prolongate $\mathbf{e}_H^{n+1}$ on the fine grid using the prolongation operator:

$$\mathbf{e}_h^{n+1} \;=\; P_H^h\mathbf{e}_H^{n+1}. \tag{4.11}$$

Finally, we update the solution for time step $(n+1)$ by the two-grid scheme:

$$\bar{\mathbf{z}}_h^{n+1} \;=\; \tilde{\mathbf{z}}_h^{n+1} + \mathbf{e}_h^{n+1}. \tag{4.12}$$

Summarizing, the splitting scheme using coarse grid correction for the solution of problem (1.1) is given as:

Two-Level (2L)

For each time step $n = 1, \ldots, \hat{n}$

- Solve $(I - kA_1)(I - kA_2)\tilde{\mathbf{z}}_h^{n+1} = \mathbf{z}_h^n + k\mathbf{f}^{n+1} - k^2A_1\mathbf{f}_2^{n+1}$ in Ω_h;
- Obtain the residue $\mathbf{r}_h^{n+1} = \mathbf{z}_h^n + k\mathbf{f}^{n+1} - (I - kA)\tilde{\mathbf{z}}_h^{n+1}$;
- Restrict the residue to the coarse grid $\mathbf{r}_H^{n+1} = R_h^H\mathbf{r}_h^{n+1}$;

- Solve the equation $(R_h^H(I-kA)P_H^h)\mathbf{e}_H^{n+1} = \mathbf{r}_H^{n+1}$;
- Prolong the error to the fine grid $\mathbf{e}_h^{n+1} = P_H^h\mathbf{e}_H^{n+1}$;
- Update the approximate solution $\bar{\mathbf{z}}_h^{n+1} = \tilde{\mathbf{z}}_h^{n+1} + \mathbf{e}_h^{n+1}$;
- Make $\mathbf{z}_h^{n+1} = \bar{\mathbf{z}}_h^{n+1}$.

4.4 The symmetrized two-level method

We can improve the two-level method by applying another fine grid correction. After we obtain the approximate solution (4.12), we calculate the residue in the fine grid due to $\bar{\mathbf{z}}_h^{n+1}$ and obtain:

$$\tilde{\mathbf{r}}_h^{n+1} \;=\; \mathbf{z}_h^n + k\mathbf{f}^{n+1} - B_h\bar{\mathbf{z}}_h^{n+1}.$$

From the residue $\tilde{\mathbf{r}}_h^{n+1}$, we use the first order operator splitting scheme to obtain:

$$(I-kA_1)(I-kA_2)\tilde{\mathbf{e}}_h^{n+1} \;=\; \tilde{\mathbf{r}}_h^{n+1}. \tag{4.13}$$

Finally, we use the error (4.13) to obtain the approximate solution:

$$\mathbf{z}_h^{n+1} \;=\; \bar{\mathbf{z}}_h^{n+1} + \tilde{\mathbf{e}}_h^{n+1}.$$

We call this scheme the symmetrized two-level method and it is summarized as:
Symmetrized Two-level (S2L)
For each time step $n = 1, \ldots, \hat{n}$

- Solve $(I-kA_1)(I-kA_2)\tilde{\mathbf{z}}_h^{n+1} = \mathbf{z}_h^n + k\mathbf{f}^{n+1} - k^2A_1\mathbf{f}_2^{n+1}$ in Ω_h;
- Obtain the residue $\mathbf{r}_h^{n+1} = \mathbf{z}_h^n + k\mathbf{f}^{n+1} - (I-kA)\tilde{\mathbf{z}}_h^{n+1}$;
- Restrict the residue to the coarse grid $\mathbf{r}_H^{n+1} = R_h^H\mathbf{r}_h^{n+1}$;
- Solve the equation $(R_h^H(I-kA)P_H^h)\mathbf{e}_H^{n+1} = \mathbf{r}_H^{n+1}$;
- Prolong the error to the fine grid $\mathbf{e}_h^{n+1} = P_H^h\mathbf{e}_H^{n+1}$;
- Update the approximate solution $\bar{\mathbf{z}}_h^{n+1} = \tilde{\mathbf{z}}_h^{n+1} + \mathbf{e}_h^{n+1}$;
- Obtain the new residue $\tilde{\mathbf{r}}_h^{n+1} = \mathbf{z}_h^n + k\mathbf{f}^{n+1} - (I-kA)\bar{\mathbf{z}}_h^{n+1}$;

- Solve in the fine grid $(I - kA_1)(I - kA_2)\tilde{\mathbf{e}}_h^{n+1} = \tilde{\mathbf{r}}_h^{n+1}$;
- Update the solution $\mathbf{z}_h^{n+1} = \bar{\mathbf{z}}_h^{n+1} + \tilde{\mathbf{e}}_h^{n+1}$;

4.5 Numerical results

We solve problem (3.28), repeating the Experiments 1 and 2 from Chapter 3 for both methods. We calculate the error for the two-level sequential splitting(2L) and the symmetrized two-level sequential splitting(S2L) at time $T = 1$ using the norm $\|.\|_h$.

Experiment 1 - In Table 4.1, we keep constant the spatial discretization mesh size $h = 1/1024$ and the time discretization step $k = 1/2048$. In each line of this table, we keep constant the relative overlap H/β, while for each column of this table we make constant the number of subdomains $\hat{s}$ and consequently the size of the subdomains is $H = 1/\hat{s}$. We examine the effect of H/β in the global error. The values for the error presented in Table 4.1 are multiplied by 10^3. The results for 2L and S2L are displayed in this order separated by the symbol "\".

Table 4.1: We divide $(0, 1)$ in $\hat{s}$ subdomains of length $H = 1/\hat{s}$, with overlap β, keeping the mesh parameters $k = 1/2048$ and $h = 1/1024$. The values for the error are multiplied by 10^3. The results for 2L and S2L are displayed in this order separated by the symbol "\", i.e., 2L\S2L

H	1/2	1/4	1/8	1/16	1/32
$H/\beta = 4$	0.6433\0.0671	2.2298\0.0719	3.6847\1.2905	4.2604\6.0258	3.6352\6.3063
$H/\beta = 8$	1.2250\0.0693	4.4287\0.4872	7.2203\5.2070	8.4364\19.8011	6.8811\20.4576
$H/\beta = 16$	2.1914\0.1436	7.9984\1.8833	12.6752\14.4814	13.6547\43.3528	9.8914\39.7007
$H/\beta = 32$	3.7703\0.4558	13.8837\5.2082	20.9029\31.5017	19.6685\78.2567	12.7409\53.2530

As expected, in general, the results for 2L and S2L are much better than for SS for large number of subdomains; see Tables 3.1 and 4.1. If we keep the ratio H/β constant and vary the parameter $\hat{s}$, we can see that for the 2L method, the error remains roughly on the same size.

Experiment 2 - In this experiment, we obtain four tables. For each table, we keep H/β constant and vary the number of subdomains and the mesh size h and keep $k = h$. We examine the effect of H/β in the error when we vary the discretization mesh sizes. The values presented are multiplied by 10^3. The results for 2L and S2L are displayed in this order separated by the symbol "\".

Table 4.2: We divide $(0,1)$ in $\hat{s}$ subdomains of length $H = 1/\hat{s}$, with overlap β, and make $H/\beta = 32$. The values for the error are multiplied by 10^3.

H	1/2	1/4	1/8	1/16
$h = k = 1/512$	12.2936\2.4031	44.1046\22.0934	49.6599\78.6257	33.4156\62.3839
$h = k = 1/1024$	6.8947\1.0848	25.4412\11.4008	34.6324\57.4985	27.0295\82.2786
$h = k = 1/2048$	3.7810\0.4573	14.1638\5.3332	21.8186\33.9411	21.1568\85.5807

Table 4.3: We divide $(0,1)$ in $\hat{s}$ subdomains of length $H = 1/\hat{s}$, with overlap β, and make $H/\beta = 16$. The values for the error are multiplied by 10^3.

H	1/2	1/4	1/8	1/16
$h = k = 1/512$	7.3447\0.9359	25.9395\9.9907	32.0812\43.1580	23.9805\46.5872
$h = k = 1/1024$	4.0410\0.3802	14.6136\4.5243	21.1500\28.1137	19.0555\52.6627
$h = k = 1/2048$	2.1919\0.1436	8.0384\1.8945	12.9505\14.8266	14.1972\46.6185

From Tables 4.2-4.5, we conclude that the error decreases almost linearly with k for the 2L and S2L methods. This is expected since SS is a first order method. This linear dependence on k is more noticeable for small values of $\hat{s}$. For $\hat{s} = 16$, it is difficult to predict the behavior of the S2L method. A possible reason is because we apply the SS method twice in the fine grid and we know that SS method does not work well for high $\hat{s}$.

If we look the tables, and keeping fixed $\hat{s}$, k and h, we verify that the 2L and S2L methods, the error decreases almost linearly when we increase H/β.

We obtain more accurate results for the 2L and S2L methods than for the SS method; see Tables 3.3-3.6 and 4.2-4.5. This is expected due to the communications provided by the coarse grid correction.

Table 4.4: We divide $(0,1)$ in $\hat{s}$ subdomains of length $H = 1/\hat{s}$, with overlap β, and make $H/\beta = 8$. The values for the error are multiplied by 10^3.

H	1/2	1/4	1/8	1/16
$h = k = 1/512$	4.2852\0.3673	14.6586\3.5636	19.2037\19.9828	46.5872\25.0959
$h = k = 1/1024$	2.3123\0.1544	8.1377\1.4045	12.2111\11.1564	12.0869\26.1968
$h = k = 1/2048$	1.2249\0.0692	4.4376\0.4884	7.2705\5.2386	8.6670\20.2609

Table 4.5: We divide $(0,1)$ in $\hat{s}$ subdomains of length $H = 1/\hat{s}$, with overlap β, and make $H/\beta = 4$. The values are multiplied by 10^3.

H	1/2	1/4	1/8	1/16
$h = k = 1/512$	2.3731\0.2624	7.6628\0.9054	10.0427\6.6444	8.3484\9.3710
$h = k = 1/1024$	1.2465\0.1322	4.1867\0.2733	6.2642\3.2539	6.2640\8.5327
$h = k = 1/2048$	0.6433\0.0670	2.2328\0.0719	3.6990\1.2939	4.3153\6.0643

Chapter 5

Conclusions

We develop a method to parallelize the solution of equation (2.1) (and more generally the equation (1.1)). The numerical results from Chapter 3 are in accordance with our analysis. The results for the sequential splitting (SS) method has the same "order" of accuracy with respect to k as the BTCS scheme when the number of subdomains is small. However, when we increase too much the number of subdomains, it is more difficult to predict the behavior of the order of accuracy with respect to k. Another fact to be mentioned is that the error decreases when we increase the relative overlap between the subdomains, and keeping the number of subdomains constant. This happens because when we increase the overlap, we increase the low frequency communications between neighboring subdomains. We also introduced a new splitting scheme which is better for parallel computing. A more complete error analysis than in [7] was developed.

To overcome the drawback caused by the deterioration of the error when we increase the number of subdomains, we implemented the coarse grid correction in Chapter 4 to obtain the 2L and S2L methods. Numerical results show that when we increase the number of subdomains, the 2L and S2L methods are more accurate than the SS method.

We mention some possible extensions and future work. We should analyze the 2L and S2L methods, taking into consideration the numerical results obtained in Chapter 4. These methods use the simplicity of SS method and the correction of the coarse grid scheme. It is important to analyze the influence on the global error with respect to the number of subdomains, the size of the overlapping between the intervals, and the ratio H/β. Other kinds of splitting of the domain should be tested.

Bibliography

[1] K. A. Bagrinovskiĭ and S. K. Godunov. Difference schemes for multidimensional problems. *Dokl. Akad. Nauk SSSR (N.S.)*, 115:431–433, 1957.

[2] William L. Briggs, Van Emden Henson, and Steve F. McCormick. *A multigrid tutorial*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 2000.

[3] Jim Douglas, Jr. and James E. Gunn. A general formulation of alternating direction methods. I. Parabolic and hyperbolic problems. *Numer. Math.*, 6:428–453, 1964.

[4] J. Geiser and I. Faragó. Iterative operator splitting methods for linear problems. *Preprint Weierstrass Inst. Appl. Anal. and Stoch.*, 2005.

[5] Leon Lapidus and George F. Pinder. *Numerical solution of partial differential equations in science and engineering*. A Wiley-Interscience Publication. John Wiley & Sons Inc., New York, 1999.

[6] G. I. Marchuk. Splitting and alternating direction methods. In *Handbook of numerical analysis, Vol. I*, Handb. Numer. Anal., I, pages 197–462. North-Holland, Amsterdam, 1990.

[7] T. P. Mathew, P. L. Polyakov, G. Russo, and J. Wang. Domain decomposition operator splittings for the solution of parabolic equations. *SIAM J. Sci. Comput.*, 19(3):912–932 (electronic), 1998.

[8] D. W. Peaceman and H. H. Rachford, Jr. The numerical solution of parabolic and elliptic differential equations. *J. Soc. Indust. Appl. Math.*, 3:28–41, 1955.

[9] Robert D. Richtmyer and K. W. Morton. *Difference methods for initial-value problems*. Robert E. Krieger Publishing Co. Inc., Malabar, FL, second edition, 1994.

[10] Gilbert Strang. On the construction and comparison of difference schemes. *SIAM J. Numer. Anal.*, 5:506–517, 1968.

[11] John C. Strikwerda. *Finite difference schemes and partial differential equations.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 2004.

[12] Pieter Wesseling. *An introduction to multigrid methods.* Pure and Applied Mathematics (New York). John Wiley & Sons Ltd., Chichester, 1992.

[13] N. N. Yanenko. On weak approximation of systems of differential equations. *Sibirsk. Mat. Zh.*, 5(6):1431–1434, 1964.