

ALTERNATING DIRECTION IMPLICIT FINITE DIFFERENCE METHODS
FOR THE HEAT EQUATION ON GENERAL DOMAINS
IN TWO AND THREE DIMENSIONS

by
Steven Wray

A thesis submitted to the Faculty and Board of Trustees of the Colorado School of Mines in partial fulfillment of the requirements for the degree of Master of Science in Applied Mathematics and Statistics - Computational and Applied Mathematics Specialty.

Golden, Colorado

Date: _____

Signed: _____

Steven Wray

Signed: _____

Dr. Bernard Bialecki

Thesis Advisor

Golden, Colorado

Date: _____

Signed: _____

Dr. Willy Hereman

Professor and Head

Department of Applied Mathematics and Statistics

ABSTRACT

Alternating direction implicit methods are a class of finite difference methods for solving parabolic PDEs in two and three dimensions. The convergence properties of these methods on rectangular domains are well-understood. We wish to extend this approach to solve the heat equation on arbitrary domains. We begin by dropping a perturbation term for the boundary conditions of the Peaceman-Rachford method in the Dirichlet problem on a two-dimensional box. We show theoretically that this modified method converges with order two under the discrete maximum norm. This is confirmed by numerical tests that show the modified method converges with order two under both the discrete maximum norm and the discrete L^2 norm. In three dimensions, similar modifications allow us to extend the Douglas method. On an arbitrary domain, the extended method converges with order two under the discrete L^2 norm but with order one under the discrete maximum norm.

TABLE OF CONTENTS

ABSTRACT	iii
LIST OF FIGURES.	vi
LIST OF TABLES	vii
LIST OF EQUATIONS	xi
1 INTRODUCTION	1
2 ADI METHODS ON A TWO-DIMENSIONAL BOX	3
2.1 The Crank-Nicolson method in 2D	4
2.2 The Peaceman-Rachford method	6
2.3 Matrix-vector equations for the Peaceman-Rachford method	9
2.4 Numerical results for the Peaceman-Rachford method	12
2.5 The Peaceman-Rachford method without perturbation terms on the boundary	16
2.6 Convergence analysis for the Peaceman-Rachford method without perturba- tion terms on the boundary	17
2.7 The Dyakonov method	26
2.8 The Dyakonov method without perturbation terms on the boundary	28
3 ADI METHODS ON A GENERAL 2D REGION.	32
3.1 The discrete problem on a general 2D region	33
3.2 Difference quotients on irregular grids	39
3.3 The Peaceman-Rachford method on a general region	41
3.4 Matrix-vector equations for the Peaceman-Rachford method on a general region	42

3.5	Numerical results for the Peaceman-Rachford method on a general 2D region	48
4	ADI METHODS ON A THREE-DIMENSIONAL BOX	60
4.1	The Douglas method	61
4.2	Numerical results for the Douglas method	63
4.3	Douglas method without perturbation terms on the boundary	63
4.4	The Douglas method with partial perturbation on the boundary	67
5	ADI METHODS ON A GENERAL 3D REGION.	69
5.1	The discrete problem on a general 3D region	70
5.2	The Douglas method for a general 3D region	77
5.3	Numerical results for the Douglas method on a general 3D region	78
	APPENDIX: ELECTRONIC FILES.	92
	REFERENCES CITED	95

LIST OF FIGURES

2.1	Ten point stencil for one timestep of the 2D Crank-Nicolson method. Known values are shaded, and unknown values are white.	5
2.2	Stencil for two partial steps of the Peaceman-Rachford method. Each partial step uses six points.	7
2.3	Stencil for boundary point on the partial timestep of the Peaceman-Rachford method. The calculation uses six known boundary values from neighboring integer timesteps.	9
3.1	A large array of regularly-spaced gridpoints that cover the irregular domain Ω shown in gray	34
3.2	Ω_N , the gridpoints that lie in the interior of the original domain. We call these points interior points.	34
3.3	A nonempty column showing interior points and boundary points	36
3.4	The diamond-shaped domain Ω_D	57
3.5	Ω_L , a domain with discontinuous boundary functions	57
5.1	Ω_O , an octahedron	84
5.2	Ω_L , a domain with discontinuous boundary functions	87

LIST OF TABLES

2.1	Results of numerical testing for the Peaceman-Rachford method using exact solution $u_0(x, y, t) = x(1 - x)y(1 - y)e^{x+y+t}$	15
2.2	Results of numerical testing for the Peaceman-Rachford method using exact solution $u_1(x, y, t) = e^{x+y+t}$	15
2.3	Results of numerical testing for the Peaceman-Rachford method using exact solution $u_2(x, y, t) = e^{x+2y+3t}$	15
2.4	Results of numerical testing for the Peaceman-Rachford method using exact solution $u_3(x, y, t) = e^{xyt}$	15
2.5	Results of numerical testing for the Peaceman-Rachford method using exact solution $u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t)$	18
2.6	Results of numerical testing for the Peaceman-Rachford method without perturbation terms on the boundary using exact solution $u_1(x, y, t) = e^{x+y+t}$	18
2.7	Results of numerical testing for the Peaceman-Rachford method without perturbation terms on the boundary using exact solution $u_2(x, y, t) = e^{x+2y+3t}$	18
2.8	Results of numerical testing for the Peaceman-Rachford method without perturbation terms on the boundary using exact solution $u_3(x, y, t) = e^{xyt}$	18
2.9	Results of numerical testing for the Peaceman-Rachford method without perturbation terms on the boundary using exact solution $u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t)$	29
2.10	Results of numerical testing for the Dyakonov method with exact solution $u_2(x, y, t) = e^{x+2y+3t}$	29
2.11	Results of numerical testing for the Dyakonov method with exact solution $u_3(x, y, t) = e^{x+2y+3t}$	29
2.12	Results of numerical testing for the Dyakonov method with exact solution $u_4(x, y, y) = 10 \cos(16x^2 + 4y^2 + t)$	29
2.13	Results of numerical testing for the Dyakonov method without perturbation terms on the boundary using exact solution $u_2(x, y, t) = e^{x+2y+3t}$	31

2.14	Results of numerical testing for the Dyakonov method without perturbation terms on the boundary using exact solution $u_3(x, y, y) = e^{xyt}$	31
2.15	Results of numerical testing for the Dyakonov method without perturbation terms on the boundary using exact solution $u_4(x, y, y) = 10 \cos(16x^2 + 4y^2 + t)$	31
3.1	Results of numerical testing for the general Peaceman-Rachford method on the square domain Ω_S using exact solution $u_0(x, y, t) = x(1-x)y(1-y)e^{x+y+t}$.	51
3.2	Results of numerical testing for the general Peaceman-Rachford method on the square domain Ω_S using exact solution $u_1(x, y, t) = e^{x+y+t}$	51
3.3	Results of numerical testing for the general Peaceman-Rachford method on the square domain Ω_S using exact solution $u_2(x, y, t) = e^{x+2y+3t}$	51
3.4	Results of numerical testing for the general Peaceman-Rachford method on the square domain Ω_S using exact solution $u_3(x, y, t) = e^{xyt}$	51
3.5	Results of numerical testing for the general Peaceman-Rachford method on the square domain Ω_S using exact solution $u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t)$	52
3.6	Results of numerical testing for the general Peaceman-Rachford method on the circular domain Ω_C using exact solution $u_2(x, y, t) = e^{x+2y+3t}$	52
3.7	Results of numerical testing for the general Peaceman-Rachford method on the circular domain Ω_C using exact solution $u_3(x, y, t) = e^{xyt}$	52
3.8	Results of numerical testing for the general Peaceman-Rachford method on the circular domain Ω_C using exact solution $u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t)$	52
3.9	Results of numerical testing for the general Peaceman-Rachford method on the elliptic domain Ω_E using exact solution $u_2(x, y, t) = e^{3x+2y+t}$	54
3.10	Results of numerical testing for the general Peaceman-Rachford method on the elliptic domain Ω_E using exact solution $u_3(x, y, t) = e^{xyt}$	54
3.11	Results of numerical testing for the general Peaceman-Rachford method on the elliptic domain Ω_E using exact solution $u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t)$	54
3.12	Results of numerical testing for the general Peaceman-Rachford method on the diamond-shaped domain Ω_D using exact solution $u_2(x, y, t) = e^{3x+2y+t}$	56

3.13	Results of numerical testing for the general Peaceman-Rachford method on the diamond-shaped domain Ω_D using exact solution $u_3(x, y, t) = e^{xyt}$	56
3.14	Results of numerical testing for the general Peaceman-Rachford method on the diamond-shaped domain Ω_D using exact solution $u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t)$	56
3.15	Results of numerical testing for the general Peaceman-Rachford method on the L-shaped domain Ω_L using exact solution $u_2(x, y, t) = e^{3x+2y+t}$	59
3.16	Results of numerical testing for the general Peaceman-Rachford method on the L-shaped domain Ω_L using exact solution $u_3(x, y, t) = e^{xyt}$	59
3.17	Results of numerical testing for the general Peaceman-Rachford method on the L-shaped domain Ω_L using exact solution $u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t)$	59
4.1	Results of numerical testing for the Douglas method using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$	64
4.2	Results of numerical testing for the Douglas method using exact solution $u_2(x, y, z, t) = e^{xyz t}$	64
4.3	Results of numerical testing for the Douglas method using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$	64
4.4	Results of numerical testing for the Douglas method without perturbation on the boundary terms using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$	66
4.5	Results of numerical testing for the Douglas method without perturbation terms on the boundary using exact solution $u_2(x, y, z, t) = e^{xyz t}$	66
4.6	Results of numerical testing for the Douglas method without perturbation terms on the boundary using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$	66
4.7	Results of numerical testing for the Douglas method with partial perturbation on the boundary terms using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$	68
4.8	Results of numerical testing for the Douglas method with partial perturbation terms on the boundary using exact solution $u_2(x, y, z, t) = e^{xyz t}$	68
4.9	Results of numerical testing for the Douglas method with partial perturbation terms on the boundary using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$	68

5.1	Results of numerical testing for the general Douglas method on the unit cube Ω_C using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$	81
5.2	Results of numerical testing for the general Douglas method on the unit cube Ω_C using exact solution $u_2(x, y, z, t) = e^{xyzt}$	81
5.3	Results of numerical testing for the general Douglas method on the unit cube Ω_C using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$	81
5.4	Results of numerical testing for the general Douglas method on the unit sphere Ω_S using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$	83
5.5	Results of numerical testing for the general Douglas method on the unit sphere Ω_S using exact solution $u_2(x, y, z, t) = e^{xyzt}$	83
5.6	Results of numerical testing for the general Douglas method on the unit sphere Ω_S using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$	83
5.7	Results of numerical testing for the general Douglas method on the ellipsoid Ω_E using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$	83
5.8	Results of numerical testing for the general Douglas method on the ellipsoid Ω_E using exact solution $u_2(x, y, z, t) = e^{xyzt}$	88
5.9	Results of numerical testing for the general Douglas method on the ellipsoid Ω_E using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$	88
5.10	Results of numerical testing for the general Douglas method on the octahedron Ω_O using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$	88
5.11	Results of numerical testing for the general Douglas method on the octahedron Ω_O using exact solution $u_2(x, y, z, t) = e^{xyzt}$	88
5.12	Results of numerical testing for the general Douglas method on the octahedron Ω_O using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$	91
5.13	Results of numerical testing for the general Douglas method on the L-shaped domain Ω_L using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$	91
5.14	Results of numerical testing for the general Douglas method on the L-shaped domain Ω_L using exact solution $u_2(x, y, z, t) = e^{xyzt}$	91
5.15	Results of numerical testing for the general Douglas method on the L-shaped domain Ω_L using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$	91

LIST OF EQUATIONS

1.1	The heat equation	2
2.1	The heat equation in two dimensions	3
2.2	The two-dimensional Crank-Nicolson method	4
2.3	The 2D Crank-Nicolson method with known terms collected on the right-hand side	4
2.4	Initial condition for the 2D Crank-Nicolson method	5
2.5	Boundary condition for the 2D Crank-Nicolson method	5
2.6	Modified 2D Crank-Nicolson method written in factored form	6
2.7	The first partial timestep of the Peaceman-Rachford method	6
2.8	The second partial timestep of the Peaceman-Rachford method	6
2.9	Step 1 in the Bialecki & Fernandes implementation of the Peaceman-Rachford method	7
2.10	Step 2 in the Bialecki & Fernandes implementation of the Peaceman-Rachford method	7
2.11	Step 2 in the Bialecki & Fernandes implementation of the Peaceman-Rachford method	7
2.12	Initial condition for the Peaceman-Rachford method	8
2.13	Boundary values for the partial timestep of the Peaceman-Rachford method	9
2.14	Matrix-vector form for step 1 in the Bialecki & Fernandes implementation of the Peaceman-Rachford method	10
2.15	Matrix-vector form for step 2 in the Bialecki & Fernandes implementation of the Peaceman-Rachford method	11
2.16	Matrix-vector form for step 3 in the Bialecki & Fernandes implementation of the Peaceman-Rachford method	12

2.17	Definition of the discrete maximum norm for a 2D box	13
2.18	Definition of the discrete L^2 norm for a 2D box	13
2.19	Definition of the numerical order of convergence determined from a sequence of tests	14
2.20	Boundary values calculated without perturbation terms for the partial timestep of the Peaceman-Rachford method	16
2.21	Comparison function for analysis of the error of the partial timestep for the Peaceman-Rachford method without perturbation terms on the boundary . . .	17
2.22	The error of the approximations	17
2.23	Boundary conditions for the error of the approximation	19
2.24	Order of convergence for the error of the approximation on the partial timestep	19
2.25	Quotient form of the first step of the Peaceman-Rachford method	19
2.26	Quotient form of the second step of the Peaceman-Rachford method	19
2.27	Truncation error for the first step of the Peaceman-Rachford method	20
2.28	Truncation error for the first step of the Peaceman-Rachford method	20
2.29	Order of the truncation errors	20
2.30	Definition of $v_{i,j}^{m+1/2}$	20
2.31	Definition of $v_{i,j}^{m+1}$	20
2.32	Initial condition $v_{i,j}^0$	20
2.33	Boundary condition for $v_{i,j}^m$	20
2.34	Boundary condition for $v_{i,j}^{m+1/2}$	20
2.35	Definition of $w_{i,j}^{m+1/2}$	21
2.36	Definition of $w_{i,j}^{m+1}$	21
2.37	Initial condition $w_{i,j}^0$	21

2.38	Boundary condition for $w_{i,j}^m$	21
2.39	Boundary condition for $w_{i,j}^{m+1/2}$	21
2.40	An expression for $v_{i,j}^{m+1/2}$	22
2.41	An expression for $v_{i,j}^{m+1}$	22
2.42	Inequality to bound $v_{i,j}^{m+1/2}$	22
2.43	Inequality to bound $v_{i,j}^{m+1}$	22
2.44	Order of V^m	23
2.45	Order of the maximum absolute error along the boundary, E^*	24
2.46	Inequalities to bound the values W_{\max} and W_{\min}	24
2.47	Assumption used to create proof by contradiction for bound on W_{\max}	24
2.48	Comparing W_{\max} to the values of its neighboring gridpoints	24
2.49	Order of W^*	26
2.50	Partial timestep of the Dyakonov method	26
2.51	Next full timestep of the Dyakonov method	26
2.52	Boundary conditions for the partial timestep of the Dyakonov method	27
2.53	Boundary conditions without perturbation terms for the partial timestep of the Dyakonov method	28
3.1	Two part definition for the domain of the general 2D problem	32
3.2	Initial condition for the general 2D problem	32
3.3	Boundary condition for the general 2D problem	32
3.4	Spacing constant for the x -direction	33
3.5	Spacing constant for the y -direction	33
3.6	Boundary curves in y	35

3.7	Boundary curves in x	38
3.8	First step of the Peaceman-Rachford method on a general 2D region	41
3.9	Second step of the Peaceman-Rachford method on a general 2D region	41
3.10	Third step of the Peaceman-Rachford method on a general 2D region	41
3.11	Initial condition for the Peaceman-Rachford method on a general 2D region	41
3.12	Boundary value at the first point in row i on the partial timestep	42
3.13	Boundary value at the last point in row i on the partial timestep	42
3.14	Step 1 of the general Peaceman-Rachford method for a column that contains a single interior point	42
3.15	The tridiagonal matrix $B^{(i)}$	43
3.16	Matrix-vector form of the first equation in the general Peaceman-Rachford method	43
3.17	Second equation in the general Peaceman-Rachford method in the special case where row j contains a single interior point	44
3.18	Matrix-vector equation for the second step in the general Peaceman-Rachford method	46
3.19	Third step of the general Peaceman-Rachford method in the special case where column i contains a single interior point	46
3.20	Matrix-vector form of the third step in the general Peaceman-Rachford method	47
4.1	The heat equation in three dimensions	60
4.2	The first partial timestep of the Douglas method	61
4.3	The second partial timestep of the Douglas method	61
4.4	The third partial timestep of the Douglas method, where we find the next full timestep of the approximate solution	61
4.5	Initial condition for the Douglas method	61
4.6	Boundary conditions for the integer timesteps of the Douglas method	61

4.7	Interior values for the two-thirds timestep of the Douglas method	62
4.8	Boundary values for the two-thirds timestep of the Douglas method	62
4.9	Boundary values for the one-third timestep of the Douglas method	62
4.10	Three-dimension test function u_1	63
4.11	Three-dimension test function u_2	63
4.12	Three-dimension test function u_3	63
4.13	Spacing constants used on the unit cube	63
4.14	Boundary condition without perturbation for the one-third timestep of the Douglas method	65
4.15	Boundary condition without perturbation for the two-thirds timestep of the Douglas method	65
4.16	Boundary condition with partial perturbation for the one-third timestep of the Douglas method	67
5.1	Three-part definition of Ω , the domain of the general 3D problem	70
5.2	Initial condition for the general 3D problem	70
5.3	Boundary condition for the general 3D problem	70
5.4	Spacing constant for the x -direction	71
5.5	Spacing constant for the y -direction	71
5.6	Spacing constant for the z -direction	71
5.7	Boundary surfaces in x for the domain of the general 3D problem	72
5.8	Boundary surfaces in y for the domain of the general 3D problem	74
5.9	Boundary surfaces in z for the domain of the general 3D problem	76

CHAPTER 1
INTRODUCTION

We seek numerical solutions to the heat equation

$$\frac{\partial u}{\partial t} - \nabla^2 u = f(\mathbf{x}, t) \tag{1.1}$$

defined for \mathbf{x} in a bounded domain Ω of dimension two or three. The temporal domain is the interval $[0, T]$. The problem is subject to an initial condition

$$u(\mathbf{x}, 0) = g_1(\mathbf{x}) \quad \text{for } \mathbf{x} \in \overline{\Omega}$$

and a Dirichlet boundary condition

$$u(\mathbf{x}, t) = g_2(\mathbf{x}, t) \quad \text{for } \mathbf{x} \in \partial\Omega, \quad t \in (0, T].$$

Note that we allow a nonzero, time-dependent function f on the right-hand side of equation (1.1). The boundary condition is also nonzero and time-dependent.

The heat equation is the simplest example of a parabolic partial differential equation (PDE). These equations model a large class of physical phenomena involving diffusion processes, where a gradient of temperature, pressure, or concentration causes a transport of matter or energy. In addition to the classical problem of heat conduction in a solid, parabolic PDEs are also used to model chemical mass transport in porous media, thermal oxidation of silicon, and the motion of a plate through a viscous fluid. These applications are described in detail in Chapter 6 of Selvadurai [8].

Because of the importance of these applications, the numerical solution of parabolic problems has been widely studied. Alternating direction implicit (ADI) methods are one class of finite difference methods for solving these problems in two and three dimensions. These methods are very efficient since they reduce solving parabolic PDEs to the problem of solving tridiagonal linear systems along the coordinate directions. The convergence of ADI

methods on rectangular domains is well-understood. In this study, we will show that ADI methods can also be applied successfully to arbitrary regions in two and three dimensions.

First, we examine a method due to Peaceman & Rachford [6] on a two-dimensional box. Next, we drop perturbation terms for intermediate approximations on the boundary of the box. A theoretical analysis of the modified method shows that it converges with order two under the discrete maximum norm. This is confirmed by numerical tests that show the same order of convergence under the discrete maximum norm and the discrete L^2 norm.

The modifications to Peaceman-Rachford allow us to extend the ADI approach to general regions in the plane. We demonstrate numerically that the order of convergence on these shapes is still two under the discrete maximum norm.

A similar approach is used in three dimensions. One standard ADI method in 3D is a scheme described by Douglas [2]. We first develop the method on a 3D box, and then we modify some of the boundary conditions. The modified method is then tested on general 3D regions. We show numerically that we achieve convergence of order two under the discrete L^2 norm but order one under the discrete maximum norm.

In this study, we look only at the heat equation. However, the methods developed here should extend to more complicated parabolic problems in two and three dimensions. In particular, ADI methods can be used to solve parabolic PDEs with variable coefficients depending on \mathbf{x} and t .

CHAPTER 2

ADI METHODS ON A TWO-DIMENSIONAL BOX

In two dimensions, the heat equation is

$$u_t - u_{xx} - u_{yy} = f(x, y, t). \quad (2.1)$$

We begin by considering this equation on a unit square. So the spatial domain is $\Omega = (0, 1) \times (0, 1)$. For the temporal domain, let $t \in [0, T]$. The problem has an initial condition

$$u(x, y, 0) = g_1(x, y) \quad \text{for } (x, y) \in \overline{\Omega},$$

and a nonzero Dirichlet boundary condition

$$u(x, y, t) = g_2(x, y, t) \quad \text{for } (x, y) \in \partial\Omega \text{ and } t \in (0, T].$$

To get the discrete form of the PDE, begin by defining $M + 1$ evenly spaced time values

$$0 = t_0, t_1, \dots, t_M = T.$$

So then the temporal step size is $\tau = T/M$. The value of each temporal coordinate is $t_m = m\tau$ for $m = 0, 1, \dots, M$.

Similarly, there are $N + 1$ equally spaced points in both x - and y -directions,

$$0 = x_0, x_1, \dots, x_N = 1$$

$$0 = y_0, y_1, \dots, y_N = 1.$$

Since they have the same length, the step size for both dimensions is $h = 1/N$. Then the discrete domain consists of points of the form (x_i, y_j) , where $x_i = ih$ and $y_j = jh$ for $i, j = 0, 1, \dots, N$.

Let $U_{i,j}^m \approx u(x_i, y_j, t_m)$ denote the discrete approximation to the PDE.

2.1 The Crank-Nicolson method in 2D

The standard approach to solving the heat equation in one dimension is the Crank-Nicolson method. This method converges with $\mathcal{O}(h^2 + \tau^2)$. It is computationally efficient because it finds the approximate solution by solving matrix-vector equations that contain only tridiagonal matrices.

The Crank-Nicolson method can be extended in a straightforward way to the 2D problem. In two dimensions we will require second order central difference quotients in both x and y . Denote these as

$$\begin{aligned}\delta_x^2 U_{i,j}^m &= \frac{U_{i-1,j}^m - 2U_{i,j}^m + U_{i+1,j}^m}{h^2} \\ \delta_y^2 U_{i,j}^m &= \frac{U_{i,j-1}^m - 2U_{i,j}^m + U_{i,j+1}^m}{h^2}\end{aligned}$$

for $i, j = 1, \dots, N-1$ and $m = 0, 1, \dots, M$.

Replacing the derivatives in (2.1) with the appropriate difference quotients gives

$$\frac{U_{i,j}^{m+1} - U_{i,j}^m}{\tau} - \frac{1}{2} (\delta_x^2 U_{i,j}^{m+1} + \delta_x^2 U_{i,j}^m) - \frac{1}{2} (\delta_y^2 U_{i,j}^{m+1} + \delta_y^2 U_{i,j}^m) = f_{i,j}^{m+1/2}. \quad (2.2)$$

The right-hand side of this equation is a time average for the function f ,

$$f_{i,j}^{m+1/2} = \frac{1}{2} (f(x_i, y_j, t_m) + f(x_i, y_j, t_{m+1})).$$

Rearranging equation (2.2) gives

$$\left(1 - \frac{\tau}{2} \delta_x^2 - \frac{\tau}{2} \delta_y^2\right) U_{i,j}^{m+1} = \left(1 + \frac{\tau}{2} \delta_x^2 + \frac{\tau}{2} \delta_y^2\right) U_{i,j}^m + \tau f_{i,j}^{m+1/2} \quad (2.3)$$

for $i, j = 1, \dots, N-1$ and $m = 0, \dots, M-1$. This formula can be used to define an implicit matrix-vector equation that gives the value of the approximate solution at timestep t_{m+1} .

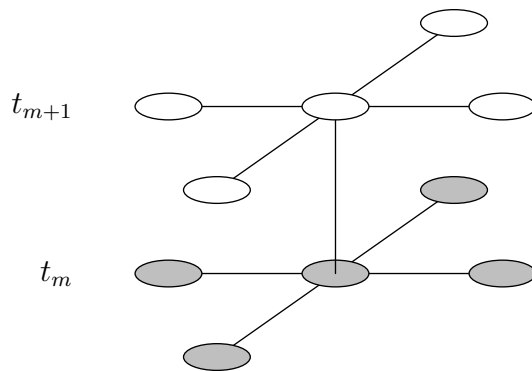


Figure 2.1: Ten point stencil for one timestep of the 2D Crank-Nicolson method. Known values are shaded, and unknown values are white.

In this method, the discrete form of the initial condition is

$$U_{i,j}^0 = g_1(x_i, y_j) \quad \text{for } i, j = 0, 1, \dots, N, \quad (2.4)$$

and the the boundary condition is

$$U_{i,j}^m = g_2(x_i, y_j, t_m) \quad (2.5)$$

for $i = 0, N, j = 0, 1, \dots, N, m = 1, \dots, M$ and for $j = 0, N, i = 0, 1, \dots, N, m = 1, \dots, M$.

In Section 4.3 of Thomas [9], the author concludes that this 2D method will converge with order $h^2 + \tau^2$ under the discrete maximum norm. This is the same performance as in the one-dimensional case, however, the matrices for the 2D method are no longer tridiagonal.

The geometry of the gridpoints is shown in Figure 2.1. Finding the approximate solution at one point depends on five known values and five unknown values. The unknown values go in both the x - and y -directions. This makes it impossible to arrange the system of equations into tridiagonal matrices. Instead, the matrix equation is sparse, with a banded structure. Solving a matrix-vector equation of this type is more time consuming than solving a tridiagonal system.

Alternating direction implicit (ADI) methods modify Crank-Nicolson to keep the order of convergence but reintroduce the more efficient tridiagonal matrices. Begin with the Crank-Nicolson method in equation (2.3). Add the term $\frac{\tau^2}{4} \delta_x^2 \delta_y^2 U_{i,j}^{m+1}$ to the left-hand side

of the equation and $\frac{\tau^2}{4}\delta_x^2\delta_y^2U_{i,j}^m$ to the right-hand side. The result is

$$\left(1 - \frac{\tau}{2}\delta_x^2 - \frac{\tau}{2}\delta_y^2 + \frac{\tau^2}{4}\delta_x^2\delta_y^2\right)U_{i,j}^{m+1} = \left(1 + \frac{\tau}{2}\delta_x^2 + \frac{\tau}{2}\delta_y^2 + \frac{\tau^2}{4}\delta_x^2\delta_y^2\right)U_{i,j}^m + \tau f_{i,j}^{m+1/2}.$$

Factor the operators on each side of this equation.

$$\left(1 - \frac{\tau}{2}\delta_x^2\right)\left(1 - \frac{\tau}{2}\delta_y^2\right)U_{i,j}^{m+1} = \left(1 + \frac{\tau}{2}\delta_x^2\right)\left(1 + \frac{\tau}{2}\delta_y^2\right)U_{i,j}^m + \tau f_{i,j}^{m+1/2} \quad (2.6)$$

The ADI methods are based on this factored form. The factors are used to split each full timestep into two partial timesteps. In section 4.4.2.2 of Thomas [9], he discusses the properties of this new scheme in detail. He concludes that the terms we added to make equation (2.6) are of small enough order that the new method still converges with $\mathcal{O}(h^2 + \tau^2)$.

2.2 The Peaceman-Rachford method

In 1955, Peaceman & Rachford [6] introduced one numerical method for the heat conduction problem that is based on equation (2.6). The factored equation is split into two fractional timesteps. One version of this method is given in Thomas [9] as equations (4.4.70) and (4.4.71),

$$\left(1 - \frac{\tau}{2}\delta_x^2\right)U_{i,j}^{m+1/2} = \left(1 + \frac{\tau}{2}\delta_y^2\right)U_{i,j}^m + \frac{\tau}{2}f_{i,j}^{m+1/2} \quad (2.7)$$

$$\left(1 - \frac{\tau}{2}\delta_y^2\right)U_{i,j}^{m+1} = \left(1 + \frac{\tau}{2}\delta_x^2\right)U_{i,j}^{m+1/2} + \frac{\tau}{2}f_{i,j}^{m+1/2} \quad (2.8)$$

for $i, j = 1, \dots, N - 1$ and $m = 0, 1, \dots, M - 1$. Boundary values, those gridpoints where $i = 0, N, j = 1, \dots, N - 1$ or $i = 1, \dots, N - 1, j = 0, N$, are discussed separately below. Equation (2.7) is an implicit equation for the partial step $U_{i,j}^{m+1/2}$, and the left-hand side contains only a derivative in y . Equation (2.8) is an implicit equation for the next full timestep $U_{i,j}^{m+1}$, and here the left-hand side has only a derivative in x .

The layout of the gridpoints for this method is shown in Figure 2.2. The unknown values for the first partial timestep all lie in the same row. The unknown values for the second timestep all lie in the same column. Since the unknown values lie in the same direction, we

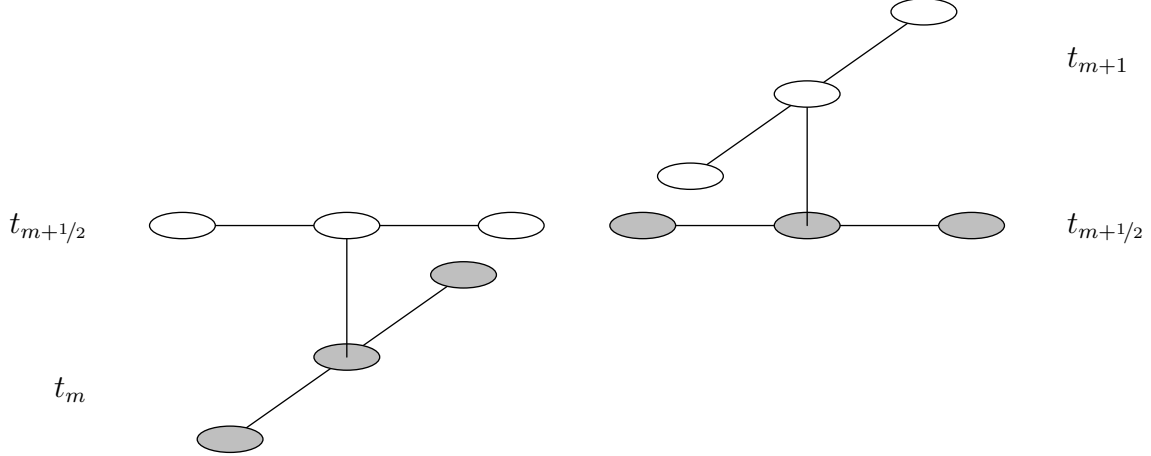


Figure 2.2: Stencil for two partial steps of the Peaceman-Rachford method. Each partial step uses six points.

will be able to assemble the equations into vector equations with tridiagonal matrices. On the first partial step, we need one such equation for each row of gridpoints in the domain. On the second partial step, we solve a tridiagonal matrix equation for every column of gridpoints in the domain.

We implement a version of the Peaceman-Rachford method given in a 1999 paper by Bialecki & Fernandes [1]. There are three steps for each interior gridpoint,

$$V_{i,j}^m = \left(1 + \frac{\tau}{2}\delta_y^2\right) U_{i,j}^m \quad (2.9)$$

$$\left(1 - \frac{\tau}{2}\delta_x^2\right) U_{i,j}^{m+1/2} = V_{i,j}^m + \frac{\tau}{2}f_{i,j}^{m+1/2} \quad (2.10)$$

$$\left(1 - \frac{\tau}{2}\delta_y^2\right) U_{i,j}^{m+1} = 2U_{i,j}^{m+1/2} - V_{i,j}^m \quad (2.11)$$

for $i, j = 1, \dots, N-1$ and $m = 0, \dots, M-1$. Equation (2.9) is an explicit equation for a new gridpoint function V . This step corresponds to finding the first term on the right-hand side of equation (2.7). So then the second step, equation (2.10), is equivalent to (2.7). It is an implicit equation for the partial timestep with a derivative in x .

Finally, equation (2.11) is an implicit equation for the next full timestep with the derivative in y . It is equivalent to finding the next full timestep using (2.8). To see this,

begin by subtracting equation (2.7) from equation (2.8) to get

$$\begin{aligned} \left(1 - \frac{\tau}{2}\delta_y^2\right) U_{i,j}^{m+1} &= 2U_{i,j}^{m+1/2} - \left(1 + \frac{\tau}{2}\delta_y^2\right) U_{i,j}^m \\ &= 2U_{i,j}^{m+1/2} - V_{i,j}^m \end{aligned}$$

for $i, j = 1, \dots, N - 1$ and $m = 0, \dots, M - 1$.

The three-step version of the Peaceman-Rachford method is a superior implementation because there are fewer difference quotients to calculate. The two-step implementation in equations (2.7) and (2.8) requires the calculation of four difference quotients at each interior gridpoint for each full timestep. The three-step approach uses only three difference quotients at each gridpoint at each step.

As we progress through the scheme the direction of the derivative on the two implicit steps alternates, giving the ADI method its name.

The initial condition is the same as (2.4),

$$U_{i,j}^0 = g_1(x_i, y_j) \quad \text{for } i, j = 0, 1, \dots, N. \quad (2.12)$$

Boundary values for the Peaceman-Rachford method are more involved. Equations (2.9) and (2.11) are evaluated on a full timestep, and so the necessary boundary values come from the boundary conditions of the original PDE. The derivatives in equations (2.9) and (2.11) are in the y -direction, so we will need the first and last values from each column. These have the form

$$U_{i,j}^m = g_2(x_i, y_j, t_m) \quad \text{for } i = 1, \dots, N - 1, \quad j = 0, N, \quad m = 1, \dots, M,$$

using the boundary condition of the original PDE.

On the partial timestep of the Peaceman-Rachford method in equation (2.10), we use a derivative in the x -direction. So we need boundary values of the form

$$U_{i,j}^{m+1/2} \quad \text{for } i = 0, N, \quad j = 1, \dots, N - 1, \quad m = 1, \dots, M - 1.$$

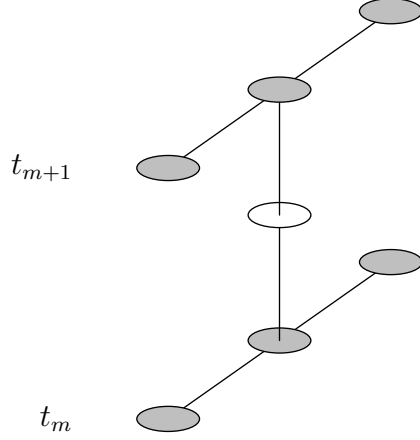


Figure 2.3: Stencil for boundary point on the partial timestep of the Peaceman-Rachford method. The calculation uses six known boundary values from neighboring integer timesteps.

These boundary values do not come directly from the original boundary condition g_2 . Instead, we begin by rearranging equation (2.11) and then substituting in for $V_{i,j}^m$ using equation (2.9). The resulting equation for $U_{i,j}^{m+1/2}$ is

$$U_{i,j}^{m+1/2} = \frac{1}{2} \left(1 - \frac{\tau}{2} \delta_y^2 \right) U_{i,j}^{m+1} + \frac{1}{2} \left(1 + \frac{\tau}{2} \delta_y^2 \right) U_{i,j}^m$$

for $i, j = 1, \dots, N-1$ and $m = 0, 1, \dots, M-1$. Motivated by this expression, we use

$$U_{i,j}^{m+1/2} = \frac{1}{2} \left(1 - \frac{\tau}{2} \delta_y^2 \right) g_2(x_i, y_j, t_{m+1}) + \frac{1}{2} \left(1 + \frac{\tau}{2} \delta_y^2 \right) g_2(x_i, y_j, t_m) \quad (2.13)$$

to define boundary values for $i = 0, N, j = 1, \dots, N-1$, and $m = 0, 1, \dots, M-1$ in terms of g_2 , the boundary condition of the original PDE. Each of these boundary values uses six boundary values from adjacent integer timesteps. The layout is shown in Figure 2.3. We can always find all of these adjacent points since the domain of the problem is rectangular. This will change when we implement the Peaceman-Rachford method on more general regions.

2.3 Matrix-vector equations for the Peaceman-Rachford method

To implement the method, we will recast the operator equations in terms of linear algebra. The first step of the Peaceman-Rachford method is the explicit equation for $V_{i,j}^m$ in equation (2.9). If we consider one row of gridpoints in the domain, then we have a fixed

value of i . Then expanding the difference quotients leads to the system of equations

$$\begin{aligned}
V_{i,1}^m &= U_{i,1}^m + \frac{\tau}{2h^2} (U_{i,0}^m - 2U_{i,1}^m + U_{i,2}^m) \\
V_{i,2}^m &= U_{i,2}^m + \frac{\tau}{2h^2} (U_{i,1}^m - 2U_{i,2}^m + U_{i,3}^m) \\
&\vdots \\
V_{i,N-1}^m &= U_{i,N-1}^m + \frac{\tau}{2h^2} (U_{i,N-2}^m - 2U_{i,N-1}^m + U_{i,N}^m)
\end{aligned}$$

Now define the vectors in \mathbf{R}^{N-1}

$$\mathbf{U}_{i,\cdot}^m = \begin{bmatrix} U_{i,1}^m \\ U_{i,2}^m \\ \vdots \\ U_{i,N-1}^m \end{bmatrix} \quad \mathbf{V}_{i,\cdot}^m = \begin{bmatrix} V_{i,1}^m \\ V_{i,2}^m \\ \vdots \\ V_{i,N-1}^m \end{bmatrix},$$

and the tridiagonal matrix B in $\mathbf{R}^{N-1} \times \mathbf{R}^{N-1}$

$$B = \begin{bmatrix} -2 & 1 & 0 & 0 & & \\ & 1 & -2 & 1 & 0 & \dots \\ & 0 & 1 & -2 & 1 & \\ & 0 & 0 & 1 & -2 & \\ & & & & & \vdots \end{bmatrix}.$$

Then the system of equations can be rewritten as the matrix-vector equation

$$\mathbf{V}_{i,\cdot}^m = \left(I + \frac{\tau}{2h^2} B \right) \mathbf{U}_{i,\cdot}^m + \begin{bmatrix} V_{i,0}^m \\ 0 \\ \vdots \\ 0 \\ V_{i,N}^m \end{bmatrix}. \quad (2.14)$$

This is an explicit equation that is evaluated for each $i = 1, \dots, N - 1$.

On the second step of the Peaceman-Rachford method, we consider one column of

gridpoints in the domain. So we fix the value of j and expand the central difference operator in (2.10). This leads to the system of equations

$$\begin{aligned}
U_{1,j}^{m+1/2} - \frac{\tau}{2} \left(U_{0,j}^{m+1/2} - 2U_{1,j}^{m+1/2} + U_{2,j}^{m+1/2} \right) &= V_{1,j}^m + \frac{\tau}{2} f_{1,j}^{m+1/2} \\
U_{2,j}^{m+1/2} - \frac{\tau}{2} \left(U_{1,j}^{m+1/2} - 2U_{2,j}^{m+1/2} + U_{3,j}^{m+1/2} \right) &= V_{2,j}^m + \frac{\tau}{2} f_{2,j}^{m+1/2} \\
&\vdots \\
U_{N-1,j}^{m+1/2} - \frac{\tau}{2} \left(U_{N-2,j}^{m+1/2} - 2U_{N-1,j}^{m+1/2} + U_{N,j}^{m+1/2} \right) &= V_{N-1,j}^m + \frac{\tau}{2} f_{N-1,j}^{m+1/2}.
\end{aligned}$$

Define vectors in \mathbf{R}^{N-1} .

$$\mathbf{U}_{\cdot,j}^m = \begin{bmatrix} U_{1,j}^m \\ U_{2,j}^m \\ \vdots \\ U_{N-1,j}^m \end{bmatrix} \quad \mathbf{V}_{\cdot,j}^m = \begin{bmatrix} V_{1,j}^m \\ V_{2,j}^m \\ \vdots \\ V_{N-1,j}^m \end{bmatrix} \quad \mathbf{f}_{\cdot,j}^{m+1/2} = \begin{bmatrix} f_{1,j}^{m+1/2} \\ f_{2,j}^{m+1/2} \\ \vdots \\ f_{N-1,j}^{m+1/2} \end{bmatrix}.$$

The second step of the method can be recast as the matrix-vector equation

$$\left(I - \frac{\tau}{2h^2} B \right) \mathbf{U}_{\cdot,j}^{m+1/2} = \mathbf{V}_{\cdot,j}^m + \frac{\tau}{2} \mathbf{f}_{\cdot,j}^{m+1/2} + \frac{\tau}{2h^2} \begin{bmatrix} U_{0,j}^{m+1/2} \\ 0 \\ \vdots \\ 0 \\ U_{N,j}^{m+1/2} \end{bmatrix} \quad (2.15)$$

evaluated for $j = 1, \dots, N-1$. Here we solve an implicit equation for the vector $\mathbf{U}_{\cdot,j}^{m+1/2}$, but the matrix $\left(I - \frac{\tau}{2h^2} B \right)$ on the right-hand side is tridiagonal. So solving the equation will be computationally efficient.

On the third step of the Peaceman-Rachord method we again look at one equation for each row of gridpoints. So fix the value of index i to obtain the sequence of equations

$$\begin{aligned}
U_{i,1}^{m+1} - \frac{\tau}{2h^2} \left(U_{i,0}^{m+1} - 2U_{i,1}^{m+1} + U_{i,2}^{m+1} \right) &= 2U_{i,1}^{m+1/2} - V_{i,1}^m \\
U_{i,2}^{m+1} - \frac{\tau}{2h^2} \left(U_{i,1}^{m+1} - 2U_{i,2}^{m+1} + U_{i,3}^{m+1} \right) &= 2U_{i,1}^{m+1/2} - V_{i,1}^m
\end{aligned}$$

$$\begin{aligned} & \vdots \\ U_{i,N-1}^{m+1} - \frac{\tau}{2h^2} (U_{i,N-2}^{m+1} - 2U_{i,N-1}^{m+1} + U_{i,N}^{m+1}) &= 2U_{i,N-1}^{m+1/2} - V_{i,N-1}^m. \end{aligned}$$

Using the same vectors defined above, we can rewrite this sequence of equations in matrix-vector form,

$$\left(I - \frac{\tau}{2h^2}B\right) \mathbf{U}_{i,\cdot}^{m+1} = 2\mathbf{U}_{i,\cdot}^{m+1/2} - \mathbf{V}_{i,\cdot}^m + \frac{\tau}{2h^2} \begin{bmatrix} U_{i,0}^{m+1} \\ 0 \\ \vdots \\ 0 \\ U_{i,N}^{m+1} \end{bmatrix} \quad (2.16)$$

for $i = 1, \dots, N - 1$. This is another implicit equation with a tridiagonal matrix on the left-hand side.

Equations (2.14), (2.15), and (2.16) together with initial conditions and boundary conditions give the basis for implementing the Peaceman-Rachford method in MATLAB.

2.4 Numerical results for the Peaceman-Rachford method

The code for the Peaceman-Rachford method was tested in several different different ways. For the first round of testing, we use the exact solution

$$u_0(x, y, t) = x(1 - x)y(1 - y)e^{x+y+t}.$$

Since this function is zero on the boundary of the unit square, all of the boundary values (including those on the partial timestep) are identically zero. Since there is no error in the boundary terms, this is good test case to debug the formulas for interior points.

The exact solution is used to provide the initial condition, boundary condition, and forcing term f used in the equations. The initial condition is

$$\begin{aligned} g_1(x, y) &= u_0(x, y, 0) \\ &= x(1 - x)y(1 - y)e^{x+y} \quad \text{for } (x, y) \in \bar{\Omega}, \end{aligned}$$

and the boundary condition is

$$\begin{aligned} g_2(x, y, t) &= u_0(x, y, t) \\ &= x(1-x)y(1-y)e^{x+y+t} \quad \text{for } (x, y) \in \partial\Omega, \quad t \in (0, T]. \end{aligned}$$

The function f is obtained by substituting the exact solution u_0 into the left-hand side of the original PDE from equation (2.1). After taking the derivatives and combining like terms, we find

$$f_0(x, y, t) = -xy(xy + 3x + 3y + t)e^{x+y+t}.$$

This function is used to define $f_{i,j}^{m+1/2}$ in (2.10).

The test was run over five different values of the spatial grid,

$$h = 0.2 \times 2^{-n} \quad \text{for } n = 0, \dots, 4.$$

The temporal spacing on each trial was set as $\tau = h$. Every trial was run until final time $T = 1$.

Error for these tests was measured in two ways. At every interior gridpoint, the exact solution was compared with the approximate solution at final time T . The max norm is the largest absolute error at any internal gridpoint,

$$e_{\max} = \max_{i,j=1,\dots,N-1} |U_{i,j}^M - u(x_i, y_j, t_M)|. \quad (2.17)$$

The second measurement of error is the discrete L^2 norm,

$$e_{L^2} = \left(\sum_{i=1}^{N-1} \sum_{j=1}^{N-1} h^2 (U_{i,j}^M - u(x_i, y_j, t_M))^2 \right)^{1/2}. \quad (2.18)$$

For each error norm, the order of convergence is estimated with the formula

$$\text{numerical order of convergence}(\mathcal{O}) = \frac{\ln(e_n/e_{n+1})}{\ln(\tau_n/\tau_{n+1})}. \quad (2.19)$$

Here h_n, h_{n+1} are the temporal step sizes on two successive trials, and e_n, e_{n+1} are the corresponding error measurements.

The results for the first test function are shown in Table 2.1. We see that the Peaceman-Rachford method converges with order two under both the max norm and the L^2 norm. This result is a good check of the code, since it matches known results mentioned by Thomas [9].

For the next test, we use an exact solution that produces non-homogeneous boundary conditions on the unit square. The test function is

$$u_1(x, y, t) = e^{x+y+t}.$$

Initial condition, boundary condition, and forcing term f are derived in the same way as in the test case above. The approximation was tested for five different values of the spacing constant h . Results are shown in Table 2.2.

The next test solution is asymmetrical in the two spatial directions. One function of this type is

$$u_2(x, y, t) = e^{x+2y+3t}.$$

Using the same testing setup as above gives the results are shown in Table 2.3.

The next test used the exact solution

$$u_3(x, y, t) = e^{xyt}.$$

This function is an example of a solution that is not separable. Results are given in Table 2.4

For the final test function, we use an exact solution with many oscillations. The test function is

$$u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t).$$

The results are in Table 2.5.

All of the tests for the Peaceman-Rachford method shows that it converges with order two under both the max norm and the L^2 norm. This result matches the known results

Table 2.1: Results of numerical testing for the Peaceman-Rachford method using exact solution $u_0(x, y, t) = x(1-x)y(1-y)e^{x+y+t}$

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	3.269×10^{-2}		1.668×10^{-2}	
1/10	9.035×10^{-3}	1.855	4.456×10^{-3}	1.904
1/20	2.303×10^{-3}	1.972	1.133×10^{-3}	1.976
1/40	5.806×10^{-4}	1.988	2.843×10^{-4}	1.994
1/80	1.453×10^{-4}	1.998	7.115×10^{-5}	1.999

Table 2.2: Results of numerical testing for the Peaceman-Rachford method using exact solution $u_1(x, y, t) = e^{x+y+t}$.

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	1.933×10^{-5}		1.065×10^{-5}	
1/10	1.285×10^{-6}	3.912	7.111×10^{-7}	3.904
1/20	8.159×10^{-8}	3.977	4.519×10^{-8}	3.976
1/40	5.137×10^{-9}	3.989	2.836×10^{-9}	3.994
1/80	3.213×10^{-10}	3.999	1.774×10^{-10}	3.999

Table 2.3: Results of numerical testing for the Peaceman-Rachford method using exact solution $u_2(x, y, t) = e^{x+2y+3t}$.

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	1.478×10^{-1}		7.968×10^{-2}	
1/10	4.327×10^{-2}	1.772	2.355×10^{-2}	1.849
1/20	1.142×10^{-2}	1.922	6.153×10^{-3}	1.936
1/40	2.886×10^{-3}	1.988	1.556×10^{-3}	1.984
1/80	7.237×10^{-4}	1.998	3.900×10^{-4}	1.996

Table 2.4: Results of numerical testing for the Peaceman-Rachford method using exact solution $u_3(x, y, t) = e^{xyt}$.

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	7.144×10^{-2}		3.784×10^{-3}	
1/10	1.983×10^{-2}	1.849	1.056×10^{-3}	1.841
1/20	5.173×10^{-3}	1.939	2.718×10^{-4}	1.958
1/40	1.304×10^{-3}	1.988	6.846×10^{-5}	1.989
1/80	3.267×10^{-4}	1.997	1.715×10^{-5}	1.997

for the ADI method discussed in Chapter 4 of Thomas.

2.5 The Peaceman-Rachford method without perturbation terms on the boundary

In order to apply the Peaceman-Rachford method on shapes other than a box, we will need to modify the boundary conditions for the partial timestep used in equation (2.10). Recall that a boundary value on the partial timestep is calculated using six neighboring values on integer timesteps.

The effect of these neighboring points corresponds to some of the terms in equation (2.13). These terms are of higher order. We next look at this equation (2.13) as a perturbation of an underlying equation without the higher order terms,

$$U_{i,j}^{m+1/2} = \frac{1}{2} (g_2(x_i, y_j, t_{m+1}) + g_2(x_i, y_j, t_m)) \quad (2.20)$$

for $i = 0, N, j = 1, \dots, N - 1$. The Peaceman-Rachford method without perturbation terms on the boundary uses equation (2.20) in place of equation (2.13). All other parts of the method are unchanged.

To see if it is worthwhile to develop the Peaceman-Rachford method for general 2D shapes, we first test the modified method on a 2D box.

We use only a part of the suite of test functions described in Section 2.4, above,

$$\begin{aligned} u_1(x, y, t) &= e^{x+y+t} \\ u_2(x, y, t) &= e^{x+2y+3t} \\ u_3(x, y, t) &= e^{xyt} \\ u_4(x, y, t) &= 10 \cos(16x^2 + 4y^2 + t). \end{aligned}$$

For each test function, the Peaceman-Rachford method without perturbation terms on the boundary was run for the spacing constants

$$h = 0.2 \times 2^{-n} \quad \text{for } n = 0, \dots, 4.$$

On each trial the temporal step size τ was chosen to be equal to h , and the approximation solution was calculated until final time $T = 1$.

Error and order of convergence were again calculated using the formulas in equations (2.17) through (2.19).

Results are shown in Tables 2.6 through 2.9. Even without the perturbation terms, the Peaceman-Rachford method still converges with order two under both norms. This suggests that we should be able to successfully extend the approach to more general regions planar regions.

2.6 Convergence analysis for the Peaceman-Rachford method without perturbation terms on the boundary

To calculate the overall error of the approximation, we introduce a notation for the exact solution of the PDE at a gridpoint,

$$u_{i,j}^m = u(x_i, y_j, t_m)$$

for $i = 0, 1, \dots, N, j = 0, 1, \dots, N$ and $m = 0, 1, \dots, M$. On the partial timestep, we extend this notation to define the comparison function

$$u_{i,j}^{m+1/2} = \frac{1}{2} (u_{i,j}^{m+1} + u_{i,j}^m) - \frac{\tau}{4} (\delta_y^2 u_{i,j}^{m+1} - \delta_y^2 u_{i,j}^m) \quad (2.21)$$

for $i = 0, 1, \dots, N, j = 1, \dots, N - 1$ and $m = 0, 1, \dots, M - 1$. Since we are calculating the approximate solution without perturbation terms at the boundary, the extra terms on the right-hand side of equation (2.21) distinguish this formula from equation (2.20), used to find $U_{i,j}^{m+1/2}$ on the boundary.

Introduce a notation for the error of the approximation,

$$E_{i,j}^m = U_{i,j}^m - u_{i,j}^m. \quad (2.22)$$

for $i = 1, \dots, N - 1, j = 0, 1, \dots, N$, and $m = 0, 1, \dots, M$. On the partial timestep, we

Table 2.5: Results of numerical testing for the Peaceman-Rachford method using exact solution $u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t)$.

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
$1/5$	118.7		44.33	
$1/10$	19.13	2.634	4.052	3.452
$1/20$	3.267	2.550	0.7957	2.348
$1/40$	0.7665	2.092	0.1855	2.101
$1/80$	0.1889	2.021	0.04559	2.025

Table 2.6: Results of numerical testing for the Peaceman-Rachford method without perturbation terms on the boundary using exact solution $u_1(x, y, t) = e^{x+y+t}$.

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
$1/5$	7.500×10^{-2}		3.348×10^{-2}	
$1/10$	2.683×10^{-2}	1.483	1.042×10^{-2}	1.684
$1/20$	8.288×10^{-3}	1.695	2.890×10^{-3}	1.850
$1/40$	2.370×10^{-3}	1.806	7.595×10^{-4}	1.928
$1/80$	6.476×10^{-4}	1.872	1.946×10^{-4}	1.965

Table 2.7: Results of numerical testing for the Peaceman-Rachford method without perturbation terms on the boundary using exact solution $u_2(x, y, t) = e^{x+2y+3t}$.

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
$1/5$	10.32		4.499	
$1/10$	4.420	1.223	1.542	1.545
$1/20$	1.570	1.493	0.4476	1.784
$1/40$	0.4915	1.702	0.1202	1.897
$1/80$	0.1420	1.820	0.03108	1.951

Table 2.8: Results of numerical testing for the Peaceman-Rachford method without perturbation terms on the boundary using exact solution $u_3(x, y, t) = e^{xyt}$.

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
$1/5$	1.728×10^{-2}		6.048×10^{-3}	
$1/10$	8.130×10^{-3}	1.483	2.332×10^{-3}	1.375
$1/20$	2.887×10^{-3}	1.695	7.067×10^{-4}	1.723
$1/40$	8.871×10^{-4}	1.806	1.932×10^{-4}	1.871
$1/80$	2.512×10^{-4}	1.872	5.042×10^{-5}	1.938

define

$$E_{i,j}^{m+1/2} = U_{i,j}^{m+1/2} - u_{i,j}^{m+1/2},$$

for $i = 0, 1, \dots, N, j = 1, \dots, N - 1$, and $m = 0, 1, \dots, M - 1$. We have the following initial and boundary conditions for the error of the approximation,

$$\begin{aligned} E_{i,j}^0 &= 0 && \text{for } i = 1, \dots, N - 1, j = 0, 1, \dots, N \\ E_{i,j}^m &= 0 && \text{for } i = 1, \dots, N - 1, \quad j = 0, N, \quad m = 1, \dots, M \\ E_{i,j}^{m+1/2} &= \frac{\tau}{4} (\delta_y^2 u_{i,j}^{m+1} - \delta_y^2 u_{i,j}^m) && \text{for } i = 0, N, \quad j = 1, \dots, N - 1, \quad m = 0, 1, \dots, M - 1. \end{aligned} \quad (2.23)$$

By expanding the difference quotients using Taylor's theorem, we can show that boundary values for the partial timestep satisfy

$$E_{i,j}^{m+1/2} = \mathcal{O}(\tau^2) \quad \text{for } i = 0, N, \quad j = 1, \dots, N - 1, \quad m = 0, 1, \dots, M - 1. \quad (2.24)$$

Next we wish to define the truncation error for the steps of the Peaceman-Rachford method. It is helpful to rewrite equations (2.7) and (2.8) for the method as

$$\frac{U_{i,j}^{m+1/2} - U_{i,j}^m}{\tau/2} - \delta_x^2 U_{i,j}^{m+1/2} - \delta_y^2 U_{i,j}^m = f_{i,j}^{m+1/2} \quad (2.25)$$

for $i, j = 1, \dots, N - 1$ and $m = 0, 1, \dots, M - 1$, and

$$\frac{U_{i,j}^{m+1} - U_{i,j}^{m+1/2}}{\tau/2} - \delta_y^2 U_{i,j}^{m+1} - \delta_x^2 U_{i,j}^{m+1/2} = f_{i,j}^{m+1/2}, \quad (2.26)$$

for $i, j = 1, \dots, N - 1$, and $M = 0, 1, \dots, M - 1$. Then the truncation errors are defined by

$$T_{i,j}^{m+1/2} = \frac{E_{i,j}^{m+1/2} - E_{i,j}^m}{\tau/2} - \delta_x^2 E_{i,j}^{m+1/2} - \delta_y^2 E_{i,j}^m \quad (2.27)$$

for $i, j = 1, \dots, N - 1$ and $m = 0, 1, \dots, M - 1$, and

$$T_{i,j}^{m+1} = \frac{E_{i,j}^{m+1} - E_{i,j}^{m+1/2}}{\tau/2} - \delta_y^2 E_{i,j}^{m+1} - \delta_x^2 E_{i,j}^{m+1/2} \quad (2.28)$$

for $i, j = 1, \dots, N - 1$ and $M = 0, 1, \dots, M - 1$. Samarskii [7] shows on pp. 556-557 that,

$$T_{i,j}^{m+1/2} = T_{i,j}^{m+1} = \mathcal{O}(h^2 + \tau^2) \quad (2.29)$$

for $i, j = 1, \dots, N - 1$ and $m = 0, 1, \dots, M - 1$.

For the next step, we define gridpoint functions, $v_{i,j}^m, w_{i,j}^m$ for $i = 1, \dots, N - 1, j = 0, 1, \dots, N, m = 0, 1, \dots, M$ and functions $v_{i,j}^{m+1/2}, w_{i,j}^{m+1/2}$ for $i = 0, 1, \dots, N, j = 1, \dots, N - 1, m = 0, 1, \dots, M - 1$. The functions $v_{i,j}^m, v_{i,j}^{m+1/2}$ are defined by the equations

$$\frac{v_{i,j}^{m+1/2} - v_{i,j}^m}{\tau/2} - \delta_x^2 v_{i,j}^{m+1/2} - \delta_y^2 v_{i,j}^m = T_{i,j}^{m+1/2} \quad (2.30)$$

$$\frac{v_{i,j}^{m+1} - v_{i,j}^{m+1/2}}{\tau/2} - \delta_y^2 v_{i,j}^{m+1} - \delta_x^2 v_{i,j}^{m+1/2} = T_{i,j}^{m+1} \quad (2.31)$$

for $i, j = 1, \dots, N - 1$ and $m = 0, 1, \dots, M - 1$, with initial condition

$$v_{i,j}^0 = 0 \quad (2.32)$$

for $i = 1, \dots, N - 1, j = 0, 1, \dots, N$, and boundary conditions

$$v_{i,j}^m = 0 \quad (2.33)$$

for $i = 1, \dots, N - 1, j = 0, N, m = 1, \dots, M$, and

$$v_{i,j}^{m+1/2} = 0 \quad (2.34)$$

for $i = 0, N, j = 1, \dots, N - 1, m = 0, 1, \dots, M - 1$. The function $v_{i,j}^m$ has a nonzero right-hand side but zero initial conditions and boundary conditions, while the function $v_{i,j}^{m+1/2}$ has a nonzero right-hand side but zero boundary conditions.

The functions $w_{i,j}^m, w_{i,j}^{m+1/2}$ are defined by the equations

$$\frac{w_{i,j}^{m+1/2} - w_{i,j}^m}{\tau/2} - \delta_x^2 w_{i,j}^{m+1/2} - \delta_y^2 w_{i,j}^m = 0 \quad (2.35)$$

$$\frac{w_{i,j}^{m+1} - w_{i,j}^{m+1/2}}{\tau/2} - \delta_y^2 w_{i,j}^{m+1} - \delta_x^2 w_{i,j}^{m+1/2} = 0 \quad (2.36)$$

for $i, j = 1, \dots, N-1$ and $m = 0, 1, \dots, M-1$ with initial condition

$$w_{i,j}^0 = 0 \quad (2.37)$$

for $i = 1, \dots, N-1, j = 0, 1, \dots, N$, and boundary conditions

$$w_{i,j}^m = 0 \quad (2.38)$$

for $i = 1, \dots, N-1, j = 0, N, m = 1, \dots, M$ and

$$w_{i,j}^{m+1/2} = E_{i,j}^{m+1/2} \quad (2.39)$$

for $i = 0, N, j = 1, \dots, N-1, m = 0, 1, \dots, M-1$. The function $w_{i,j}^m$ has a zero right-hand side and zero initial and boundary conditions, while the function $w_{i,j}^{m+1/2}$ has a zero right-hand side but non-zero boundary conditions.

If we substitute $v_{i,j}^{m+1/2} + w_{i,j}^{m+1/2}$ in place of $E_{i,j}^{m+1/2}$ and $v_{i,j}^{m+1} + w_{i,j}^{m+1}$ in place of $E_{i,j}^{m+1}$ in equations (2.27) and (2.28) for $i, j = 1, \dots, N-1$ and $m = 0, 1, \dots, M-1$, we see by uniqueness that

$$E_{i,j}^{m+1/2} = v_{i,j}^{m+1/2} + w_{i,j}^{m+1/2}, \quad E_{i,j}^{m+1} = v_{i,j}^{m+1} + w_{i,j}^{m+1},$$

for $i, j = 1, \dots, N-1, m = 0, \dots, M-1$. So we can bound the overall error of the approximation by bounding the functions $v_{i,j}^m$ and $w_{i,j}^m$ separately.

We begin by bounding $v_{i,j}^m$. Expand difference quotients and rearrange equations

(2.30) and (2.31) to obtain

$$\left(1 + \frac{\tau}{h^2}\right) v_{i,j}^{m+1/2} = \frac{\tau}{2h^2} \left(v_{i-1,j}^{m+1/2} + v_{i+1,j}^{m+1/2}\right) + \left(1 - \frac{\tau}{h^2}\right) v_{i,j}^m + \frac{\tau}{2h^2} \left(v_{i,j-1}^m + v_{i,j+1}^m\right) + \frac{\tau}{2} T_{i,j}^{m+1/2} \quad (2.40)$$

$$\left(1 + \frac{\tau}{h^2}\right) v_{i,j}^{m+1} = \frac{\tau}{2h^2} \left(v_{i,j-1}^{m+1} + v_{i,j+1}^{m+1}\right) + \left(1 - \frac{\tau}{h^2}\right) v_{i,j}^{m+1/2} + \frac{\tau}{2h^2} \left(v_{i-1,j}^{m+1/2} + v_{i+1,j}^{m+1/2}\right) + \frac{\tau}{2} T_{i,j}^{m+1}, \quad (2.41)$$

for $i, j = 1, \dots, N-1$ and $m = 0, 1, \dots, M-1$. In the remainder of this section, we assume that

$$\frac{\tau}{h^2} \leq 1,$$

so that all of the coefficients on the right-hand side of equations (2.40) and (2.41) are non-negative. Then taking the absolute value and applying the triangle inequality produces

$$\begin{aligned} \left(1 + \frac{\tau}{h^2}\right) |v_{i,j}^{m+1/2}| &\leq \frac{\tau}{2h^2} \left(|v_{i-1,j}^{m+1/2}| + |v_{i+1,j}^{m+1/2}|\right) + \left(1 - \frac{\tau}{h^2}\right) |v_{i,j}^m| + \frac{\tau}{2h^2} \left(|v_{i,j-1}^m| + |v_{i,j+1}^m|\right) \\ &\quad + \frac{\tau}{2} |T_{i,j}^{m+1/2}| \end{aligned} \quad (2.42)$$

$$\begin{aligned} \left(1 + \frac{\tau}{h^2}\right) |v_{i,j}^{m+1}| &\leq \frac{\tau}{2h^2} \left(|v_{i,j-1}^{m+1}| + |v_{i,j+1}^{m+1}|\right) + \left(1 - \frac{\tau}{h^2}\right) |v_{i,j}^{m+1/2}| + \frac{\tau}{2h^2} \left(|v_{i-1,j}^{m+1/2}| + |v_{i+1,j}^{m+1/2}|\right) \\ &\quad + \frac{\tau}{2} |T_{i,j}^{m+1}|. \end{aligned} \quad (2.43)$$

for $i, j = 1, \dots, N-1$ and $m = 0, 1, \dots, M-1$. We define

$$V^m = \max_{i,j=1,\dots,N-1} |v_{i,j}^m|$$

for $m = 0, 1, \dots, M$ and

$$V^{m+1/2} = \max_{i,j=1,\dots,N-1} |v_{i,j}^{m+1/2}|, \quad T^{m+1/2} = \max_{i,j=1,\dots,N-1} |T_{i,j}^{m+1/2}|, \quad T^{m+1} = \max_{i,j=1,\dots,N-1} |T_{i,j}^{m+1}|$$

for $m = 0, 1, \dots, M-1$. Then equations (2.42), (2.43), (2.33), and (2.34) imply that

$$\begin{aligned} V^{m+1/2} &\leq V^m + \frac{\tau}{2} T^{m+1/2} \\ V^{m+1} &\leq V^{m+1/2} + \frac{\tau}{2} T^{m+1}, \end{aligned}$$

for $m = 0, 1, \dots, M - 1$, and so

$$V^{m+1} \leq V^m + \tau T^{m+1/2}$$

for $m = 0, 1, \dots, M - 1$. Therefore, by equations (2.29) and (2.32),

$$V^m = \mathcal{O}(h^2 + \tau^2) \tag{2.44}$$

for $m = 1, \dots, M$.

This completes the first half of the argument we use to bound the error. Next, we bound the gridpoint function $w_{i,j}^m$. Here we use a maximum principle argument similar to one given in Section 2.11 of Morton & Mayers [5] for the one-dimensional Crank-Nicolson method. Their argument applies to the heat equation with zero right-hand side.

We define a notation for the maximum and minimum value of the errors, $E_{i,j}^m, E_{i,j}^{m+1/2}$ taken along the boundary. From equation (2.23), the boundary values for E on an integer timestep are identically zero,

$$E_{i,j}^{m+1} = 0 \quad \text{for } i = 1, \dots, N - 1, \quad j = 0, N \quad m = 0, 1, \dots, M - 1.$$

So the maximum error on the boundary at either an integer or partial timestep is

$$E_{\max} = \max \left\{ 0; E_{i,j}^{m+1/2} \mid i = 0, N, j = 1, \dots, N - 1, m = 0, 1, \dots, M - 1 \right\},$$

and the minimum error along the boundary taken at any timestep is

$$E_{\min} = \min \left\{ 0; E_{i,j}^{m+1/2} \mid i = 0, N, j = 1, \dots, N - 1, m = 0, 1, \dots, M - 1 \right\}.$$

Next we define the maximum *absolute* error along the boundary,

$$E^* = \max \left\{ |E^{m+1/2}| \mid i = 0, N, j = 1, \dots, N - 1, m = 0, 1, \dots, M - 1 \right\}.$$

By equation (2.24),

$$E^* = \mathcal{O}(\tau^2). \quad (2.45)$$

We define a notation for the maximum and minimum values of the functions $w_{i,j}^{m+1}, w_{i,j}^{m+1/2}$, taken over all *internal* gridpoints. So let

$$\begin{aligned} W_{\max} &= \max \left\{ w_{i,j}^{m+1}, w_{i,j}^{m+1/2} \mid i, j = 1, \dots, N-1, m = 0, 1, \dots, M-1 \right\} \\ W_{\min} &= \min \left\{ w_{i,j}^{m+1}, w_{i,j}^{m+1/2} \mid i, j = 1, \dots, N-1, m = 0, 1, \dots, M-1 \right\}. \end{aligned}$$

We intend to show

$$W_{\max} \leq E_{\max}, \quad E_{\min} \leq W_{\min}. \quad (2.46)$$

We prove the first inequality in (2.46) by contradiction. So suppose that for some $i, j = 1, \dots, N-1$ and some $m = 0, 1, \dots, M-1$ we have

$$W_{\max} = w_{i,j}^{m+1} > E_{\max} \quad \text{or} \quad W_{\max} = w_{i,j}^{m+1/2} > E_{\max}. \quad (2.47)$$

In the first case, we must have $W_{\max} = w_{i,j-1}^{m+1}$, since otherwise equation (2.36) implies

$$\begin{aligned} \left(1 + \frac{\tau}{h^2}\right) W_{\max} &< \frac{\tau}{2h^2} (W_{\max} + w_{i,j+1}^{m+1}) + \left(1 - \frac{\tau}{h^2}\right) w_{i,j}^{m+1/2} + \frac{\tau}{2h^2} (w_{i-1,j}^{m+1/2} + w_{i+1,j}^{m+1/2}) \\ &\leq \frac{\tau}{2h^2} (W_{\max} + W_{\max}) + \left(1 - \frac{\tau}{h^2}\right) W_{\max} + \frac{\tau}{2h^2} (W_{\max} + W_{\max}) \\ &= \left(1 + \frac{\tau}{h^2}\right) W_{\max}, \end{aligned} \quad (2.48)$$

giving the contradiction

$$W_{\max} < W_{\max}.$$

Then, if $j-1 = 0$, we have shown that $W_{\max} = w_{i,0}^{m+1} = 0$. This contradicts the first inequality in (2.47). Or, if $j-1 > 0$, we repeat the above argument to show that $W_{\max} = w_{i,j-2}^{m+1}$. Continue as necessary until we reach the boundary, showing that $W_{\max} = w_{i,0}^{m+1}$. Again, we have obtained the desired contradiction.

For the second case given in (2.47), we must have $W_{\max} = w_{i-1,j}^{m+1/2}$, since otherwise

equation (2.35) implies

$$\begin{aligned}
\left(1 + \frac{\tau}{h^2}\right) W_{\max} &< \frac{\tau}{2h^2} \left(W_{\max} + w_{i+1,j}^{m+1/2}\right) + \left(1 - \frac{\tau}{h^2}\right) w_{i,j}^m + \frac{\tau}{2h^2} (w_{i,j-1}^m + w_{i,j+1}^m) \\
&\leq \frac{\tau}{2h^2} (W_{\max} + W_{\max}) + \left(1 - \frac{\tau}{h^2}\right) W_{\max} + \frac{\tau}{2h^2} (W_{\max} + W_{\max}) \\
&= \left(1 + \frac{\tau}{h^2}\right) W_{\max},
\end{aligned}$$

giving the contradiction

$$W_{\max} < W_{\max}.$$

If $i - 1 = 0$, we have shown that $W_{\max} = w_{0,j}^{m+1/2} = E_{0,j}^{m+1/2}$. This contradicts the second inequality in (2.47). Or, if $i - 1 > 0$, we repeat the above argument to show that $W_{\max} = w_{i-2,j}^{m+1/2}$. Continue as necessary until we show that $W_{\max} = w_{0,j}^{m+1/2}$. This gives the desired contradiction.

We have successfully shown that

$$W_{\max} \leq E_{\max}.$$

A similar argument demonstrates the second inequality in (2.46),

$$W_{\min} \geq E_{\min}.$$

We finish by bounding the absolute value of $w_{i,j}^{m+1}$. Let

$$W^* = \max \left\{ |w_{i,j}^{m+1}| \mid i, j = 1, \dots, N-1, m = 0, 1, \dots, M-1 \right\}.$$

Since we are taking the maximum of a finite set, there is some $m^* = 0, 1, \dots, M-1$ and some $i, j = 1, \dots, N-1$ such that

$$W^* = |w_{i,j}^{m^*}|.$$

If $w_{i,j}^{m^*} \geq 0$, then

$$W^* = w_{i,j}^{m^*} \leq W_{\max} \leq E_{\max} \leq E^*.$$

Or, if $w_{i,j}^{m^*} < 0$,

$$W^* = -w_{i,j}^{m^*} \leq -W_{\min} \leq -E_{\min} \leq E^*.$$

So, by equation (2.45),

$$W^* = \mathcal{O}(\tau^2) \tag{2.49}$$

Now we return to the overall error of the Peaceman-Rachford approximation without perturbation terms. For the error of the approximation, we have

$$\begin{aligned} |E_{i,j}^m| &= |v_{i,j}^m + w_{i,j}^m| \\ &\leq |v_{i,j}^m| + |w_{i,j}^m| \\ &\leq V^m + W^* \end{aligned}$$

for $i, j = 1, \dots, N-1$ and $m = 1, 2, \dots, M$. So, by equations (2.44) and (2.49),

$$|E_{i,j}^m| = \mathcal{O}(h^2 + \tau^2)$$

for $i, j = 1, \dots, N-1$ and $m = 1, 2, \dots, M$. This completes the proof that the Peaceman-Rachford method without perturbation terms converges with order two under the discrete maximum norm.

2.7 The Dyakonov method

There is second ADI method based on a different way to split the factored form of equation (2.6) into parts. The Dyakonov scheme is

$$\left(1 - \frac{\tau}{2}\delta_x^2\right) U_{i,j}^{m^*} = \left(1 + \frac{\tau}{2}\delta_x^2\right) \left(1 + \frac{\tau}{2}\delta_y^2\right) U_{i,j}^m + \tau f_{i,j}^{m+1/2}, \tag{2.50}$$

$$\left(1 - \frac{\tau}{2}\delta_y^2\right) U_{i,j}^{m+1} = U_{i,j}^{m^*} \tag{2.51}$$

for $i, j = 1, \dots, N-1$ and $m = 0, 1, \dots, M-1$. These equations are equations (4.4.77) and (4.4.78) in Thomas [9].

The initial condition is the same as that used in the Peaceman-Rachford method,

$$U_{i,j}^0 = g_1(x_i, y_j) \quad \text{for } i, j = 0, 1, \dots, N,$$

and the boundary values used on the right-hand side of equation (2.50) are given by

$$U_{i,j}^m = g_2(x_i, y_j, t_m) \quad \text{for } i = 0, N, \quad j = 0, 1, \dots, N, \quad m = 0, 1, \dots, M - 1,$$

and for $i = 1, \dots, N - 1, \quad j = 0, 1, \quad m = 0, 1, \dots, M - 1,$

On the left-hand side, there is a second-order derivative in x . So we will need boundary values for the first and last points in each row. These values occur on a partial timestep, and so we need to derive a formula for these points. Start with equation (2.51) and rearrange to get

$$U_{i,j}^{m*} = \left(1 - \frac{\tau}{2} \delta_y^2\right) U_{i,j}^{m+1}$$

for $i, j = 1, \dots, N - 1$ and $m = 0, 1, \dots, M - 1$. Motivated by this equation, we define the boundary values for the partial timestep,

$$U_{i,j}^{m*} = \left(1 - \frac{\tau}{2} \delta_y^2\right) g_2(x_i, y_j, t_{m+1}) \tag{2.52}$$

for $i = 0, N, j = 1, \dots, N - 1$, and $m = 0, 1, \dots, M - 1$. Unlike the corresponding formula from the Peaceman-Rachford method, this formula uses values on only one neighboring integer timestep. There is a y -derivative, and so the boundary value for the partial timestep depends on three adjacent boundary values from the next full timestep.

Finally, in equation (2.51), there is a derivative with respect to y on the left-hand side. Here we will need boundary values for the first and last point in each column. These values come from the boundary condition for the original PDE,

$$U_{i,j}^{m+1} = g_2(x_i, y_j, t_m)$$

for $i = 0, N, j = 1, \dots, N - 1$, and $m = 0, 1, \dots, M - 1$.

Deriving the matrix-vector form of this method is very similar to the steps used in Section 2.3 for the Peaceman-Rachford method. The details will not be repeated here.

We next test the Dyakonov method with some of the test functions used earlier. We look at

$$\begin{aligned} u_2(x, y, t) &= e^{3x+2y+t} \\ u_3(x, y, t) &= e^{xyt} \\ u_4(x, y, y) &= 10 \cos(16x^2 + 4y^2 + t). \end{aligned}$$

The results are given in Tables 2.10 through 2.12. The tests show that the Dyakonov method converges with order two under both norms. So the performance of the new method in this case is comparable to the performance of the Peaceman-Rachford method.

2.8 The Dyakonov method without perturbation terms on the boundary

We are interested in extending the ADI method to general 2D regions, and so we will to examine the performance of the Dyakonov method when the perturbation terms in equation (2.52) are removed. Then the boundary condition for the partial timestep becomes

$$U_{i,j}^{m+1/2} = g_2(x_i, y_j, t_{m+1}) \quad (2.53)$$

for $i = 0, N, j = 1, \dots, N - 1$ and $M = 0, 1, \dots, M - 1$.

This formula requires only a single boundary value from the next full timestep. No other steps in the Dyakonov method above are changed.

Using the standard testing setup described above, we examine the test functions

$$\begin{aligned} u_2(x, y, t) &= e^{3x+2y+t} \\ u_3(x, y, t) &= e^{xyt} \\ u_4(x, y, y) &= 2 \sin(10(x^2 + y^2 + t^2)). \end{aligned}$$

Table 2.9: Results of numerical testing for the Peaceman-Rachford method without perturbation terms on the boundary using exact solution $u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t)$

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
$1/5$	118.4		44.27	
$1/10$	19.15	2.628	4.046	3.452
$1/20$	3.267	2.551	0.7998	2.339
$1/40$	0.7801	2.066	0.1869	2.097
$1/80$	0.1911	2.030	0.04596	2.024

Table 2.10: Results of numerical testing for the Dyakonov method with exact solution $u_2(x, y, t) = e^{x+2y+3t}$

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
$1/5$	1.478×10^{-1}		7.968×10^{-2}	
$1/10$	4.327×10^{-2}	1.773	2.355×10^{-2}	1.759
$1/20$	1.142×10^{-2}	1.922	6.153×10^{-3}	1.936
$1/40$	2.886×10^{-3}	1.984	1.556×10^{-3}	1.984
$1/80$	7.237×10^{-4}	1.996	3.900×10^{-4}	1.996

Table 2.11: Results of numerical testing for the Dyakonov method with exact solution $u_3(x, y, t) = e^{x+2y+3t}$

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
$1/5$	7.144×10^{-3}		3.784×10^{-3}	
$1/10$	1.983×10^{-3}	1.849	1.056×10^{-3}	1.841
$1/20$	5.173×10^{-4}	1.939	2.718×10^{-4}	1.958
$1/40$	1.304×10^{-4}	1.988	6.846×10^{-5}	1.989
$1/80$	3.267×10^{-5}	1.997	1.715×10^{-5}	1.997

Table 2.12: Results of numerical testing for the Dyakonov method with exact solution $u_4(x, y, y) = 10 \cos(16x^2 + 4y^2 + t)$

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
$1/5$	118.7		44.33	
$1/10$	19.13	2.634	4.052	3.452
$1/20$	3.267	2.550	0.7957	2.348
$1/40$	0.7665	2.092	0.1855	2.101
$1/80$	0.1889	2.021	0.04559	2.025

Results are shown in Tables 2.13 through 2.15

These tests suggest that the Dyakonov method without perturbation terms only converges with order one (under both norms). So this method performs worse than Peaceman-Rachford under these conditions.

On the partial timestep of the ADI method, we will not have perturbation terms at the boundary on general 2D regions. So our tests show that Peaceman-Rachford is the correct method to extend to these general regions.

Table 2.13: Results of numerical testing for the Dyakonov method without perturbation terms on the boundary using exact solution $u_2(x, y, t) = e^{x+2y+3t}$

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	8.270		3.380	
1/10	5.759	0.522	2.018	0.744
1/20	3.458	0.736	1.081	0.901
1/40	1.951	0.826	0.556	0.961
1/80	1.055	0.887	0.281	0.984

Table 2.14: Results of numerical testing for the Dyakonov method without perturbation terms on the boundary using exact solution $u_3(x, y, y) = e^{xyt}$

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	4.802×10^{-2}		1.790×10^{-2}	
1/10	3.719×10^{-2}	0.368	1.217×10^{-2}	0.557
1/20	2.305×10^{-2}	0.691	6.850×10^{-3}	0.829
1/40	1.311×10^{-2}	0.814	3.597×10^{-3}	0.929
1/80	7.104×10^{-3}	0.884	1.837×10^{-3}	0.969

Table 2.15: Results of numerical testing for the Dyakonov method without perturbation terms on the boundary using exact solution $u_4(x, y, y) = 10 \cos(16x^2 + 4y^2 + t)$

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	119.43		44.14	
1/10	21.52	2.473	5.630	2.971
1/20	26.95	-0.325	3.805	0.565
1/40	26.19	0.042	2.781	0.452
1/80	19.55	0.422	1.673	0.733

CHAPTER 3

ADI METHODS ON A GENERAL 2D REGION

As in the previous chapter, we seek solutions to equation (2.1), the heat equation in two dimensions. However, the spatial domain is now allowed to be a general region in the plane. Let $\Omega \subset \mathbf{R}^2$ be a domain that satisfies the two-part definition,

$$\begin{aligned}\Omega &= \{(x, y) | A < x < B, \phi_1(x) < y < \phi_2(x)\} \\ &= \{(x, y) | C < y < D, \psi_1(y) < x < \psi_2(x)\},\end{aligned}\tag{3.1}$$

where A, B are real numbers with $A < B$, and C, D are real numbers with $C < D$. The functions $\phi_1, \phi_2 : [A, B] \rightarrow \mathbf{R}$ are bounded and piecewise continuous with $\phi_1(x) \leq \phi_2(x)$ for $x \in [A, B]$. So the domain Ω is the region between these curves.

Similarly, $\psi_1, \psi_2 : [C, D] \rightarrow \mathbf{R}$ are bounded and piecewise continuous with $\psi_1(y) \leq \psi_2(y)$ for $y \in [C, D]$. This means that the same domain Ω can also be described as the region between this second pair of curves. We will see that many familiar shapes can be described in this way, including circles, ellipses and polygons. The details are given in Section 3.5, below. The temporal domain is the interval $[0, T]$.

The initial condition is

$$u(x, y, 0) = g_1(x, y) \quad \text{for } (x, y) \in \overline{\Omega},\tag{3.2}$$

and the Dirichlet boundary condition is

$$u(x, y, t) = g_2(x, y, t) \quad \text{for } (x, y) \in \partial\Omega, \quad t \in (0, T].\tag{3.3}$$

Because of the geometry of the rectangular domains, the boundary points of the unit square were consistently spaced from their neighboring interior points. More general regions may have curved boundaries, and so this distance may change from one row of gridpoints to

the next. Therefore we develop the boundary conditions for the discrete problem in detail.

3.1 The discrete problem on a general 2D region

Discrete time values are straightforward. There are $M+1$ evenly spaced time values

$$0 = t_0, t_1, \dots, t_M = T.$$

The temporal step size is $\tau = T/M$, and the value of each temporal coordinate is $t_m = m\tau$ for $m = 0, 1, \dots, M$.

For the discrete spatial domain, begin by dividing both x - and y -directions into N steps. Then the step size in the x -direction is

$$h_x = \frac{B - A}{N}, \tag{3.4}$$

and the step size in the y -direction is

$$h_y = \frac{C - D}{N}. \tag{3.5}$$

Unlike the grid for the unit square in Chapter 2, the two directions in a general domain can have different spacing constants. Use h_x, h_y to define an array of gridpoints (x_i, y_j) , where

$$x_i = A + ih_x \quad \text{for } i = 0, 1, \dots, N$$

and

$$y_j = C + jh_y \quad \text{for } j = 0, 1, \dots, N.$$

The domain of the original PDE will lie inside this array of gridpoints. Then let Ω_N be the set of all of these gridpoints that are in Ω , the domain of the original PDE. Points in Ω_N are called interior points, where we evaluate the formulas of the Peaceman-Rachford method.

For example, the elliptic domain shown in Figure 3.1 generates large box of gridpoints. This box extends to the minimum and maximum points of the ellipse in both

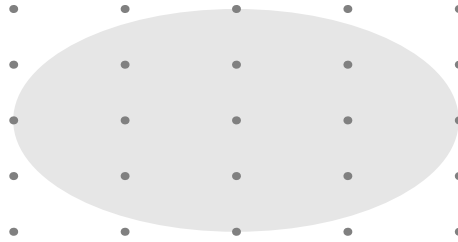


Figure 3.1: A large array of regularly-spaced gridpoints that cover the irregular domain Ω shown in gray

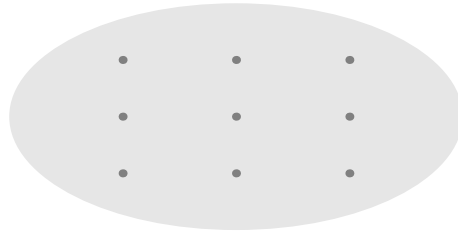


Figure 3.2: Ω_N , the gridpoints that lie in the interior of the original domain. We call these points interior points.

directions. But when using the Peaceman-Rachford method we work on only the gridpoints that lie inside the ellipse. Those interior points are shown in Figure 3.2. Notice in particular that gridpoints that lie exactly on the boundary of the original domain are excluded. These points are considered boundary points, discussed separately below.

We saw in Chapter 2 that the Peaceman-Rachford method on a box leads to tridiagonal systems of linear equations. There is one such system for each row and columns of gridpoints in the domain. On arbitrary shapes, we will use the same structure. But here the number of points in each column is not necessarily uniform across the domain.

A column is made of boundary points and interior points. We begin by describing the interior points.

Each $i = 1, \dots, N - 1$ corresponds to one possible column of interior gridpoints in Ω_N . Gridpoints with $i = 0, N$ can only be boundary points. All points in column i share the same x -coordinate, $x_i = A + ih_x$. So all of these points lie on the vertical line

$$x = A + ih_x.$$

It is possible for a vertical line to intersect the domain in only one point. For

example, on the left-hand side of Figure 3.1 there is a column of points that meets the domain in a single point. In a column of this type the upper and lower boundary points are identical. There are no interior points. It is also possible that there are two distinct boundary points that lie so close together that there are no interior points in between them.

In either of these two special cases, we describe the corresponding column as an empty column. If column i is not empty, we call it nonempty. In this case there are one or more gridpoints (x_i, y_j) that lie between the lower and upper boundary. Then the y -coordinates of these points satisfy

$$\phi_1(x_i) < y_j < \phi_2(x_i). \quad (3.6)$$

These points are interior points.

If column i contains one or more interior points, then we define the constant $j_{\min}(i)$ to be the smallest $j = 1, \dots, N - 1$ such that y_j satisfies equation (3.6). Similarly, we define the constant $j_{\max}(i)$ to be the largest $j = 1, \dots, N - 1$ such that y_j satisfies (3.6). So then the complete set of interior points in column i is

$$(x_i, y_{j_{\min}(i)}), (x_i, y_{j_{\min}(i)+1}), \dots, (x_i, y_{j_{\max}(i)}).$$

It is possible for a column to contain only a single interior point. In this case, $j_{\min}(i)$ and $j_{\max}(i)$ are equal.

The boundary points for column i are the points where the vertical line

$$x = x_i$$

intersects the boundary curves ϕ_1 and ϕ_2 . To describe these points of intersection, define the constants

$$\phi_{i,\ell} = \phi_1(x_i)$$

$$\phi_{i,u} = \phi_2(x_i).$$

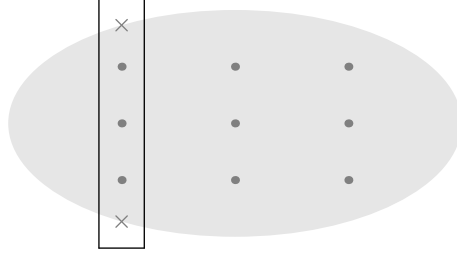


Figure 3.3: A nonempty column showing interior points and boundary points

The constant $\phi_{i,\ell}$ is the y -coordinate of the lower boundary point, and $\phi_{i,u}$ is the y -coordinate of the upper boundary point. So the lower boundary point in column i has coordinates $(x_i, \phi_{i,\ell})$, and the upper boundary point in column i has coordinates $(x_i, \phi_{i,u})$.

One example of a column is shown in Figure 3.3. In this column, $i = 1$. There are three interior points and two boundary points. Interior points in the column begin in the first row, so we have $j_{\min(1)} = 1$. There are three interior points total, and so $j_{\max(1)} = 3$. This means that the interior points in column 1 are

$$(x_1, y_1), (x_1, y_2), (x_1, y_3).$$

In addition, there are upper and lower boundary points with the coordinates

$$(x_1, \phi_{1,\ell}) \text{ and } (x_1, \phi_{1,u}).$$

The y -coordinates come from the boundary curves used to define the domain of the original PDE.

In calculating the steps of the Peaceman-Rachford method, we need a value for the approximate solution U at every interior point and boundary point. Let

$$U_{i,j}^m \approx u(x_i, y_j, t_m)$$

denote the approximate solution to the PDE at an interior gridpoint. In column i , we will use $U_{i,\ell}^m$ to denote the boundary value of the solution u at the lower boundary. So then

$$U_{i,\ell}^m = g_2(x_i, \phi_{i,\ell}),$$

from the boundary condition of the original PDE. Similarly, $U'_{i,u}{}^m$ is the value of the solution u at the upper boundary, and so

$$U'_{i,u}{}^m = g_2(x_i, \phi_{i,u}),$$

The difference quotients in the general Peaceman-Rachford method also use the spacing constants for neighboring pairs of gridpoints. For a general domain in the plane these quotients may be defined on a column with irregularly spaced points. In particular, the distance between a boundary point and an interior point can be different than the distance between two interior points. Figure 3.3 shows an example of this irregular spacing in the left column.

The distance between two neighboring interior points in the same column is always h_y . For every nonempty column, we also calculate two spacing constants, one at each boundary point. Let

$$h'_{i,\ell} = y_{j_{\min}(i)} - \phi_{i,\ell}$$

be the spacing constant at the lower boundary, and let

$$h'_{i,u} = \phi_{i,u} - y_{j_{\max}(i)}$$

be the spacing constant at the upper boundary. We will find one pair of spacing constants for every column that contains at least one interior point.

Now we have a description of column i that will enable us to expand the difference quotients in the y -direction from the Peaceman-Rachford method. In general, different columns will contain different numbers of points.

A similar set of constants is used to describe the rows of Ω_N . Every $j = 1, \dots, N-1$ corresponds to one possible row of interior gridpoints. Points in the same row share a y -coordinate, $y_j = C + jh_y$, and so lie on the same horizontal line,

$$y = C + hh_y.$$

If a horizontal line intersects the domain Ω in only one point, then the lower boundary point and the upper boundary point are identical. It is also possible that the upper and lower boundary points are distinct, but so close together that there are no interior points between them. A row of either of these two types is called an empty row.

Otherwise, we say that row j is nonempty. In a row of this type there is at least one interior gridpoint (x_i, y_j) that lies between the left and right boundary points. This point has x -coordinate satisfying

$$\psi_1(y_j) < x_i < \psi_2(y_j). \quad (3.7)$$

For every row j with one or more interior points, we define the constant $i_{\min}(j)$ to be the smallest $i = 1, \dots, N - 1$ such that x_i satisfies equation (3.7). Similarly, we define the constant $i_{\max}(j)$ to be the largest $i = 1, \dots, N - 1$ such that x_i satisfies (3.7). So the complete list of points in row j is

$$(x_{i_{\min}(j)}, y_j), (x_{i_{\min}(j)+1}, y_j), \dots, (x_{i_{\max}(j)}, y_j).$$

It is possible that a nonempty row contains only a single interior point. In this case, the constants will be equal.

The boundary points for row j are the points where the horizontal line $y = y_j$ meets the boundary curves ψ_1 and ψ_2 given in equation (3.1). For the x -coordinates of these boundary points, define the constants

$$\begin{aligned} \psi_{j,\ell} &= \psi_1(y_j) \\ \psi_{j,u} &= \psi_2(y_j). \end{aligned}$$

Then the left boundary point in row j has coordinates $(\psi_{j,\ell}, y_j)$, and the right boundary point has coordinates $(\psi_{j,u}, y_j)$.

The difference quotients in Peaceman-Rachford use the spacing between neighboring pairs of gridpoints. The spacing in the x -direction can be irregular near the boundary.

So we define the constants

$$\begin{aligned} h''_{j,\ell} &= y_{j_{\min}(i)} - \psi_{j,\ell} \\ h''_{j,u} &= \psi_{j,u} - y_{j_{\max}(i)} \end{aligned}$$

for the grid spacings at the left and right boundaries, respectively. We have one such pair of boundary spacing constants for every row that contains at least one interior point.

Now we have a complete description of both the rows and columns of the domain that will enable us to implement the Peaceman-Rachford method.

3.2 Difference quotients on irregular grids

On a general region, the Peaceman-Rachford method depends on difference quotients defined over gridpoints that are irregularly spaced. A difference quotient in y is evaluated at three neighboring points that lie in the same column. The first possible geometry for this quotient is a column i that contains only a single interior point $(x_i, y_j) \in \Omega_N$. Then the difference quotient in y about this points is

$$\delta_y^2 U_{i,j}^m = \frac{2}{h'_{i,u} + h'_{i,\ell}} \left(\frac{U'_{i,u}{}^m - U_{i,j}^m}{h'_{i,u}} - \frac{U_{i,j}^m - U'_{i,\ell}{}^m}{h'_{i,\ell}} \right)$$

Note that this formula uses that values associated with both the upper and lower gridpoints of column i .

The other possibility for column i is that it contains two or more interior points. Here we have three possible formulas for the difference quotient in y . The point $(x_i, y_{j_{\min}(i)})$ is the interior point in column i that lies adjacent to the lower boundary. For this point, the difference quotient is

$$\delta_y^2 U_{i,j_{\min}(i)}^m = \frac{2}{h_y + h'_{i,\ell}} \left(\frac{U_{i,j_{\min}(i)+1}^m - U_{i,j_{\min}(i)}^m}{h_y} - \frac{U_{i,j_{\min}(i)}^m - U'_{i,\ell}{}^m}{h'_{i,\ell}} \right).$$

There is a similar formula at $(x_i, y_{j_{\max}(i)})$, the point nearest the upper boundary. This

difference quotient is

$$\delta_y^2 U_{i,j_{\max}(i)}^m = \frac{2}{h'_{i,u} + h_y} \left(\frac{U_{i,u}^{m'} - U_{i,j_{\max}(i)}^m}{h'_{i,u}} - \frac{U_{i,j_{\max}(i)}^m - U_{i,j_{\max}(i)-1}^m}{h_y} \right).$$

Finally, column i may also contain interior points that do not lie next to either boundary in y . For a point (x_i, y_j) of this type, the difference quotient is

$$\delta_y^2 U_{i,j}^m = \frac{U_{i,j+1}^m - 2U_{i,j}^m + U_{i,j-1}^m}{h_y^2}.$$

This is the same central difference quotient that we used on the unit square in Chapter 2.

The Peaceman-Rachford method also uses second order central difference quotients in the x -direction. A quotient of the type is defined on three neighboring points in the same row. In the special case where row j contains only one interior point (x_i, y_j) , the difference quotient is

$$\delta_x^2 U_{i,j}^m = \frac{2}{h''_{j,u} + h''_{j,\ell}} \left(\frac{U_{j,u}^{m''} - U_{i,j}^m}{h''_{j,u}} - \frac{U_{i,j}^m - U_{j,\ell}^{m''}}{h''_{j,\ell}} \right).$$

If row j contains two or more interior points, then we will use one of three formulas for the difference quotient. If $(x_{i_{\min}(j)}, y_j)$ is the point nearest the lower boundary, then

$$\delta_x^2 U_{i_{\min}(j),j}^m = \frac{2}{h_x + h''_{j,\ell}} \left(\frac{U_{i_{\min}(j)+1,j}^m - U_{i_{\min}(j),j}^m}{h_x} - \frac{U_{i_{\min}(j),j}^m - U_{j,\ell}^{m''}}{h''_{j,\ell}} \right).$$

The point nearest the upper boundary is $(x_{i_{\max}(j)}, y_j)$. Here the difference quotient is

$$\delta_x^2 U_{i_{\max}(j),j}^m = \frac{2}{h''_{j,u} + h_x} \left(\frac{U_{j,u}^{m''} - U_{i_{\max}(j),j}^m}{h''_{j,u}} - \frac{U_{i_{\max}(j),j}^m - U_{i_{\max}(j)-1,j}^m}{h_x} \right).$$

Row j may also contain one or more points that do not lie next to either boundary. For a point (x_i, y_j) of this type, the difference quotient is

$$\delta_x^2 U_{i,j}^m = \frac{U_{i+1,j}^m - 2U_{i,j}^m + U_{i-1,j}^m}{h_x^2}.$$

3.3 The Peaceman-Rachford method on a general region

Expressed in terms of operators, our method looks identical to equations (2.9) through (2.11) in Chapter 2. There are three steps at every interior point $(x_i, y_j) \in \Omega_N$,

$$V_{i,j}^m = (1 + \frac{\tau}{2}\delta_y^2)U_{i,j}^m \quad (3.8)$$

$$(1 - \frac{\tau}{2}\delta_x^2)U_{i,j}^{m+1/2} = V_{i,j}^m + \frac{\tau}{2}f_{i,j}^{m+1/2} \quad (3.9)$$

$$(1 - \frac{\tau}{2}\delta_y^2)U_{i,j}^{m+1} = 2U_{i,j}^{m+1/2} - V_{i,j}^m \quad (3.10)$$

$m = 0, 1, \dots, M - 1$. When we expand the difference quotients in these equations, we use the appropriate definition of δ_x^2, δ_y^2 from Section 3.2.

As in Chapter 2, $f_{i,j}^{m+1/2}$ is defined as a time average,

$$f_{i,j}^{m+1/2} = \frac{1}{2} (f(x_i, y_j, t_{m+1}) + f(x_i, y_j, t_m))$$

for $(x_i, y_j) \in \Omega_N$ and $m = 0, 1, \dots, M - 1$.

The discrete form of the initial condition is

$$U_{i,j}^0 = g_1(x_i, y_j) \quad (3.11)$$

for $(x_i, y_j) \in \Omega_N$.

In equations (3.8) and (3.10) we have difference quotients in the y -direction taken on an integer timestep. For these quotients we need boundary values for the first and last points in each column. In column i , these boundary values are $U_{i,\ell}^m$ and $U_{i,u}^m$ for $i = 0, 1, \dots, N$ where column i is nonempty.

In equation (3.9), we evaluate a difference quotient in the x on the partial timestep. We need boundary values for the first and last points in row j . We use a formula for these points based on equation (2.20), the formula for boundary values without perturbation terms in Chapter 2,

$$U_{j,\ell}^{m+1/2} = \frac{1}{2} (g_2(\psi_{j,\ell}, y_j, t_{m+1}) + g_2(\psi_{j,\ell}, y_j, t_m)) \quad (3.12)$$

$$U_{j,u}^{m+1/2} = \frac{1}{2} (g_2(\psi_{j,u}, y_j, t_{m+1}) + g_2(\psi_{j,u}, y_j, t_m)) \quad (3.13)$$

for $j = 1, \dots, N-1$ where row j is nonempty. The values on the right-hand side of these definitions are boundary values for each row on an integer timestep and have already been defined.

3.4 Matrix-vector equations for the Peaceman-Rachford method on a general region

To rewrite the operator equations in terms of linear algebra, we begin by defining the vectors

$$\mathbf{U}_{i,\cdot}^m = \begin{bmatrix} U_{i,j_{\min}(i)}^m \\ U_{i,j_{\min}(i)+1}^m \\ U_{i,j_{\min}(i)+2}^m \\ \vdots \\ U_{i,j_{\max}(i)}^m \end{bmatrix} \quad \mathbf{V}_{i,\cdot}^m = \begin{bmatrix} V_{i,j_{\min}(i)}^m \\ V_{i,j_{\min}(i)+1}^m \\ V_{i,j_{\min}(i)+2}^m \\ \vdots \\ V_{i,j_{\max}(i)}^m \end{bmatrix}$$

for $i = 0, 1, \dots, N$ where column i is nonempty. In general, different columns will have vectors of different sizes.

Next, we expand the difference quotient in equation (3.8). If column i contains only one interior point (x_i, y_j) , then the vector form of equation (3.8) is just the scalar equation

$$\begin{aligned} V_{i,j}^m &= \left(1 + \frac{\tau}{2} \delta_y^2\right) U_{i,j}^m \\ &= U_{i,j}^m + \frac{\tau}{h'_{i,u} + h'_{i,\ell}} \left(\frac{U_{i,u}^{m'} - U_{i,j}^m}{h'_{i,u}} - \frac{U_{i,j}^m - U_{i,\ell}^{m'}}{h'_{i,\ell}} \right). \end{aligned} \quad (3.14)$$

Otherwise, if column i contains more than one interior point, we expand all the difference quotients and get the system of equations

$$\begin{aligned} V_{i,j_{\min}(i)}^m &= U_{i,j_{\min}(i)}^m + \frac{\tau}{h_y + h'_{i,\ell}} \left(\frac{U_{i,j_{\min}(i)+1}^m - U_{i,j_{\min}(i)}^m}{h_y} - \frac{U_{i,j_{\min}(i)}^m - U_{i,\ell}^{m'}}{h'_{i,\ell}} \right) \\ V_{i,j_{\min}(i)+1}^m &= U_{i,j_{\min}(i)+1}^m + \frac{\tau}{h_y^2} (U_{i,j_{\min}(i)+2}^m - 2U_{i,j_{\min}(i)+1}^m + U_{i,j_{\min}(i)}^m) \end{aligned}$$

The next step in our implementation of the Peaceman-Rachford method is (3.9). Since this equation contains a derivative with regard to x , we will solve one vector equation for each row $j = 1, 2, \dots, N - 1$ where row j is nonempty. For these equations, we define the vectors

$$\mathbf{U}_{\cdot,j}^{m+1/2} = \begin{bmatrix} U_{i_{\min}(j),j}^{m+1/2} \\ U_{i_{\min}(j)+1,j}^{m+1/2} \\ U_{i_{\min}(j)+2,j}^{m+1/2} \\ \vdots \\ U_{i_{\max}(j),j}^{m+1/2} \end{bmatrix} \quad \mathbf{V}_{\cdot,j}^m = \begin{bmatrix} V_{i_{\min}(j),j}^m \\ V_{i_{\min}(j)+1,j}^m \\ V_{i_{\min}(j)+2,j}^m \\ \vdots \\ V_{i_{\max}(j),j}^m \end{bmatrix} \quad \mathbf{f}_{\cdot,j}^{m+1/2} = \begin{bmatrix} f_{i_{\min}(j),j}^{m+1/2} \\ f_{i_{\min}(j)+1,j}^{m+1/2} \\ f_{i_{\min}(j)+2,j}^{m+1/2} \\ \vdots \\ f_{i_{\max}(j),j}^{m+1/2} \end{bmatrix}.$$

If row j contains a single interior point (x_i, y_j) , then the vector equation for (3.9) is the same as the scalar equation

$$U_{i,j}^{m+1/2} - \frac{\tau}{h_{j,u}'' + h_{j,\ell}''} \left(\frac{U_{j,u}''^{m+1/2} - U_{i,j}^{m+1/2}}{h_{j,u}''} - \frac{U_{i,j}^{m+1/2} - U_{j,\ell}''^{m+1/2}}{h_{j,\ell}''} \right) = V_{i,j}^m + \frac{\tau}{2} f_{i,j}^{m+1/2}$$

when we expand the difference quotient. Rearrange to get

$$\left(1 + \frac{\tau}{h_{j,u}'' h_{j,\ell}''} \right) U_{i,j}^{m+1/2} = V_{i,j}^m + \frac{\tau}{2} f_{i,j}^{m+1/2} + \frac{\tau}{h_{j,u}'' + h_{j,\ell}''} \left(\frac{U_{j,u}''^{m+1/2}}{h_{j,u}''} + \frac{U_{j,\ell}''^{m+1/2}}{h_{j,\ell}''} \right), \quad (3.17)$$

an explicit formula for $U_{i,j}^{m+1/2}$.

If row j contains two or more interior points we get a system of equations,

$$\begin{aligned} \left(1 - \frac{\tau}{2} \delta_x^2 \right) U_{i_{\min}(j),j}^{m+1/2} &= V_{i_{\min}(j),j}^m + \frac{\tau}{2} f_{i_{\min}(j),j}^{m+1/2} \\ \left(1 - \frac{\tau}{2} \delta_x^2 \right) U_{i_{\min}(j)+1,j}^{m+1/2} &= V_{i_{\min}(j)+1,j}^m + \frac{\tau}{2} f_{i_{\min}(j)+1,j}^{m+1/2} \\ \left(1 - \frac{\tau}{2} \delta_x^2 \right) U_{i_{\min}(j)+2,j}^{m+1/2} &= V_{i_{\min}(j)+2,j}^m + \frac{\tau}{2} f_{i_{\min}(j)+2,j}^{m+1/2} \\ &\vdots \\ \left(1 - \frac{\tau}{2} \delta_x^2 \right) U_{i_{\max}(j),j}^{m+1/2} &= V_{i_{\max}(j),j}^m + \frac{\tau}{2} f_{i_{\max}(j),j}^{m+1/2} \end{aligned}$$

with one equation for each interior point in the column. Expand the difference quotients

In matrix-vector form, the system of equations becomes

$$(I + C^{(j)}) \mathbf{U}_{\cdot,j}^{m+1/2} = \mathbf{V}_{\cdot,j}^m + \frac{\tau}{2} \mathbf{f}_{\cdot,j}^{m+1/2} + \begin{bmatrix} \frac{\tau}{h'_{j,\ell}(h_x+h'_{j,\ell})} U_{j,\ell}''^{m+1/2} \\ 0 \\ \vdots \\ 0 \\ \frac{\tau}{h'_{j,u}(h_x+h'_{j,u})} U_{j,u}''^{m+1/2} \end{bmatrix}. \quad (3.18)$$

This is an implicit equation with a tridiagonal matrix for the vector on the left-hand side.

The final step of the Peaceman-Rachford method is equation (3.10). This equation has a difference quotient y and so we solve a vector equation for each nonempty column $i = 0, 1, \dots, N$.

If column i contains a single interior point (x_i, y_j) , then the vector equation for (3.10) is the scalar equation

$$U_{i,j}^{m+1} - \frac{\tau}{h'_{i,u} + h'_{i,\ell}} \left(\frac{U_{i,u}'^{m+1} - U_{i,j}^{m+1}}{h'_{i,u}} - \frac{U_{i,j}^{m+1} - U_{i,\ell}'^{m+1}}{h'_{i,\ell}} \right) = 2U_{i,j}^{m+1/2} - V_{i,j}^m.$$

Rearrange to get an explicit formula for the approximate solution at the next full timestep,

$$\left(1 + \frac{\tau}{h'_{i,u}h'_{i,\ell}} \right) U_{i,j}^{m+1} = 2U_{i,j}^{m+1/2} - V_{i,j}^m + \frac{\tau}{h'_{i,u} + h'_{i,\ell}} \left(\frac{U_{i,u}'^{m+1}}{h'_{i,u}} + \frac{U_{i,\ell}'^{m+1}}{h'_{i,\ell}} \right) \quad (3.19)$$

In the case where column i contains two or more interior points, we get a system of equations.

$$\begin{aligned} \left(1 - \frac{\tau}{2} \delta_y^2 \right) U_{i,j_{\min}(i)}^{m+1} &= 2U_{i,j_{\min}(i)}^{m+1/2} - V_{i,j_{\min}(i)}^m \\ \left(1 - \frac{\tau}{2} \delta_y^2 \right) U_{i,j_{\min}(i)+1}^{m+1} &= 2U_{i,j_{\min}(i)+1}^{m+1/2} - V_{i,j_{\min}(i)+1}^m \\ &\vdots \\ \left(1 - \frac{\tau}{2} \delta_y^2 \right) U_{i,j_{\max}(i)-1}^{m+1} &= 2U_{i,j_{\max}(i)-1}^{m+1/2} - V_{i,j_{\max}(i)-1}^m \\ \left(1 - \frac{\tau}{2} \delta_y^2 \right) U_{i,j_{\max}(i)}^{m+1} &= 2U_{i,j_{\max}(i)}^{m+1/2} - V_{i,j_{\max}(i)}^m \end{aligned}$$

Expand the difference quotients and rearrange.

$$\begin{aligned}
U_{i,j_{\min}(i)}^{m+1} + \frac{\tau}{h_y h'_{i,\ell}} U_{i,j_{\min}(i)}^{m+1} - \frac{\tau}{h_y (h_y + h'_{i,\ell})} U_{i,j_{\min}(i)+1}^{m+1} &= 2U_{i,j_{\min}(i)}^{m+1/2} - V_{i,j_{\min}(i)}^m \\
&\quad + \frac{\tau}{h'_{i,\ell} (h_y + h'_{i,\ell})} U_{i,\ell}'^{m+1} \\
U_{i,j_{\min}(i)+1}^{m+1} - \frac{\tau}{2h_y^2} \left(U_{i,j_{\min}(i)+2}^{m+1} - 2U_{i,j_{\min}(i)+1}^{m+1} + U_{i,j_{\min}(i)}^{m+1} \right) &= 2U_{i,j_{\min}(i)+1}^{m+1/2} - V_{i,j_{\min}(i)+1}^m \\
U_{i,j_{\min}(i)+2}^{m+1} - \frac{\tau}{2h_y^2} \left(U_{i,j_{\min}(i)+3}^{m+1} - 2U_{i,j_{\min}(i)+2}^{m+1} + U_{i,j_{\min}(i)+1}^{m+1} \right) &= 2U_{i,j_{\min}(i)+2}^{m+1/2} - V_{i,j_{\min}(i)+2}^m \\
&\quad \vdots \\
U_{i,j_{\max}(i)}^{m+1} - \frac{\tau}{h_y (h_y + h'_{i,u})} U_{i,j_{\max}(i)-1}^{m+1} + \frac{\tau}{h_y h'_{i,u}} U_{i,j_{\max}(i)}^{m+1} &= 2U_{i,j_{\max}(i)}^{m+1/2} - V_{i,j_{\max}(i)}^m \\
&\quad + \frac{\tau}{h'_{i,u} (h_y + h'_{i,u})} U_{i,u}'^{m+1}
\end{aligned}$$

This system of equations can be written in vector form as

$$\left(I + \frac{\tau}{2h_y^2} B^{(i)} \right) \mathbf{U}_{i,\cdot}^{m+1} = 2\mathbf{U}_{i,\cdot}^{m+1/2} - \mathbf{V}_{i,\cdot}^m + \begin{bmatrix} \frac{\tau}{h'_{i,\ell} (h_y + h'_{i,\ell})} U_{i,\ell}'^{m+1} \\ 0 \\ \vdots \\ 0 \\ \frac{\tau}{h'_{i,u} (h_y + h'_{i,u})} U_{i,u}'^{m+1} \end{bmatrix} \quad (3.20)$$

where $B^{(i)}$ is the tridiagonal matrix defined for column i in equation (3.15). Equation 3.20 is an implicit matrix-vector equation for the approximate solution in column i at the next full timestep. The equation holds for $m = 0, 1, \dots, M - 1$ in every column $i = 0, 1, \dots, N$ containing more than one interior point.

Equations (3.14), (3.16), (3.17), (3.18), (3.19), and (3.20) together with initial conditions and boundary conditions give basis for implementing the method in MATLAB.

3.5 Numerical results for the Peaceman-Rachford method on a general 2D region

We use a number of different tests to evaluate the performance of the method. To see if we can extend the Peaceman-Rachford method to general 2D regions, we need to test the method on different kinds of shapes in the plane. Each shape must be described using the two-part given above in equation (3.1).

We will use the group of test functions from Chapter 2,

$$u_0(x, y, t) = x(1-x)y(1-y)e^{x+y+t}$$

$$u_1(x, y, t) = e^{x+y+t}$$

$$u_2(x, y, t) = e^{3x+2y+t}$$

$$u_3(x, y, t) = e^{xyt}$$

$$u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t).$$

On a given test, the test function is used to define the initial condition, boundary condition, and forcing function f . Then the method was used over a range of different grids. In both directions, we will use N steps, where

$$N = 5 \times 2^p \quad \text{for } p = 0, \dots, 4.$$

We will use the same number of temporal steps, setting

$$\tau = \frac{1}{N}.$$

In every test, the Peaceman-Rachford method is run to find the approximate solution at final time $T = 1$. So we use M timesteps, where

$$M = N.$$

The error of the approximation is calculated in two ways. The discrete maximum

norm e_{\max} is

$$e_{\max} = \max_{(x_i, y_j) \in \Omega_N} |U_{i,j}^M - u(x_i, y_j, t_M)|,$$

the maximum of the absolute error taken over all internal gridpoints. The discrete L^2 norm is

$$e_{L^2} = \left(\sum_{(x_i, y_j) \in \Omega_N} h_x h_y (U_{i,j}^M - u(x_i, y_j, t_M))^2 \right)^{1/2}.$$

For each norm, the order of convergence was estimated with equation (5.3).

Example 1: The unit square

To make sure that the code is consistent with our earlier results, we first consider the problem defined on the unit square Ω_S , the same domain used in Chapter 2. Possible x -values for points in Ω_S go from $A = 0$ to $B = 1$. The associated boundary functions defined on $[A, B]$ are

$$\phi_1(x) = 0, \quad \phi_2(x) = 1 \quad \text{for } A < x < B.$$

The definition of the domain in terms of y is similar. The minimum and maximum values for the y -coordinate of points in Ω_S are $C = 0$ and $D = 1$, respectively. The corresponding boundary functions are

$$\psi_1(y) = 0, \quad \psi_2(y) = 1 \quad \text{for } C < y < D.$$

Therefore, two-part definition of Ω_S needed to matches equation (3.1) is

$$\begin{aligned} \Omega_S &= \{(x, y) \in \mathbf{R}^2 | 0 < x < 1, 0 < y < 1\} \\ &= \{(x, y) \in \mathbf{R}^2 | 0 < y < 1, 0 < x < 1\}. \end{aligned}$$

The general Peaceman-Rachford method on Ω_S should match the performance of the Peaceman-Rachford method without perturbation terms on the boundary described in Section 2.5, above, since the calculations needed for the two methods are identical.

Results of the tests on the unit square Ω_S are shown in Tables 3.1 through 3.5. As expected, the order of convergence is two under both norms. This matches the results we obtained on a square domain in Chapter 2.

Example 2: The unit circle

Next we evaluate the method on a domain that is non-rectangular. Perhaps the simplest such domain is the unit circle. The minimum x -value for this domain is $A = -1$, and the maximum x -value is $B = 1$. Then, for $A < x < B$, we have boundary functions

$$\begin{aligned}\phi_1(x) &= -\frac{1}{2}\sqrt{1-x^2} \\ \phi_2(x) &= \frac{1}{2}\sqrt{1-x^2}.\end{aligned}$$

In the y -direction the minimum value is $C = -1$ and the maximum value is $D = 1$. The corresponding boundary functions are

$$\begin{aligned}\psi_1(y) &= -\sqrt{1-y^2} \\ \psi_2(y) &= \sqrt{1-y^2}.\end{aligned}$$

with two-part definition

$$\begin{aligned}\Omega_C &= \{(x, y) \in \mathbf{R}^2 \mid -1 < x < 1, -\sqrt{1-x^2} < y < \sqrt{1-x^2}\} \\ &= \{(x, y) \in \mathbf{R}^2 \mid -1 < y < 1, -\sqrt{1-y^2} < x < \sqrt{1-y^2}\}.\end{aligned}$$

The test functions u_2, u_3, u_4 defined above were used as exact solutions. Again, the error of the approximation was calculated for different grids. The results of the tests on the unit circle are shown in Tables 3.6 through 3.8. Convergence is still order two under the discrete maximum norm. This series of tests demonstrates that the Peaceman-Rachford method can be successfully implemented on general regions in 2D.

Table 3.1: Results of numerical testing for the general Peaceman-Rachford method on the square domain Ω_S using exact solution $u_0(x, y, t) = x(1-x)y(1-y)e^{x+y+t}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$1/5$	3.269×10^{-2}		1.668×10^{-2}	
10	$1/10$	9.035×10^{-3}	1.855	4.456×10^{-3}	1.904
20	$1/20$	2.303×10^{-3}	1.972	1.133×10^{-3}	1.976
40	$1/40$	5.806×10^{-4}	1.988	2.843×10^{-4}	1.994
80	$1/80$	1.453×10^{-4}	1.998	7.115×10^{-5}	1.999

Table 3.2: Results of numerical testing for the general Peaceman-Rachford method on the square domain Ω_S using exact solution $u_1(x, y, t) = e^{x+y+t}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$1/5$	3.765×10^{-2}		1.680×10^{-2}	
10	$1/10$	1.343×10^{-2}	1.487	5.216×10^{-3}	1.688
20	$1/20$	4.145×10^{-3}	1.696	1.445×10^{-3}	1.851
40	$1/40$	1.185×10^{-3}	1.806	3.798×10^{-4}	1.928
80	$1/80$	3.238×10^{-4}	1.872	9.729×10^{-5}	1.965

Table 3.3: Results of numerical testing for the general Peaceman-Rachford method on the square domain Ω_S using exact solution $u_2(x, y, t) = e^{x+2y+3t}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$1/5$	6.570		2.874	
10	$1/10$	2.779	1.241	0.9741	1.561
20	$1/20$	0.9832	1.499	0.2817	1.790
40	$1/40$	0.3074	1.677	0.07554	1.899
80	$1/80$	0.08877	1.792	0.01953	1.952

Table 3.4: Results of numerical testing for the general Peaceman-Rachford method on the square domain Ω_S using exact solution $u_3(x, y, t) = e^{xyt}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$1/5$	1.541×10^{-2}		5.409×10^{-3}	
10	$1/10$	7.247×10^{-3}	1.088	2.110×10^{-3}	1.358
20	$1/20$	2.534×10^{-3}	1.516	6.408×10^{-4}	1.719
40	$1/40$	7.678×10^{-4}	1.723	1.753×10^{-4}	1.870
80	$1/80$	2.152×10^{-4}	1.835	4.573×10^{-5}	1.938

Table 3.5: Results of numerical testing for the general Peaceman-Rachford method on the square domain Ω_S using exact solution $u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t)$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$1/5$	118.4		44.27	
10	$1/10$	19.14	2.629	4.045	3.452
20	$1/20$	3.266	2.551	0.799	2.339
40	$1/40$	0.7796	2.066	0.187	2.097
80	$1/80$	0.1909	2.030	0.04594	2.024

Table 3.6: Results of numerical testing for the general Peaceman-Rachford method on the circular domain Ω_C using exact solution $u_2(x, y, t) = e^{x+2y+3t}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	3.074		2.365	
10	$1/5$	1.150	1.418	0.7754	1.609
20	$1/10$	0.3619	1.669	0.2176	1.833
40	$1/20$	0.1028	1.815	0.05756	1.918
80	$1/40$	0.02722	1.917	0.01482	1.958

Table 3.7: Results of numerical testing for the general Peaceman-Rachford method on the circular domain Ω_C using exact solution $u_3(x, y, t) = e^{xyt}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	5.629×10^{-3}		5.173×10^{-3}	
10	$1/5$	3.299×10^{-3}	0.771	1.955×10^{-3}	1.404
20	$1/10$	1.250×10^{-3}	1.401	5.992×10^{-4}	1.706
40	$1/20$	3.529×10^{-4}	1.824	1.604×10^{-4}	1.901
80	$1/40$	9.412×10^{-5}	1.907	4.128×10^{-5}	1.959

Table 3.8: Results of numerical testing for the general Peaceman-Rachford method on the circular domain Ω_C using exact solution $u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t)$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	237.7		247.0	
10	$1/5$	114.3	1.056	90.15	1.454
20	$1/10$	6.819	4.067	3.907	4.528
40	$1/20$	2.631	1.374	1.101	1.827
80	$1/40$	0.6738	1.965	0.2922	1.914

Example 3: An ellipse

For the remaining tests, we consider several domains that are special cases. One such case is a domain with different sizes in x and y . Let Ω_E be the interior of the ellipse given by

$$x^2 + 4y^2 < 1.$$

The minimum x -value is $A = -1$, and the maximum x -value is $B = 1$. Then, for $A < x < B$, we have boundary functions

$$\begin{aligned}\phi_1(x) &= -\frac{1}{2}\sqrt{1-x^2} \\ \phi_2(x) &= \frac{1}{2}\sqrt{1-x^2}.\end{aligned}$$

In the y -direction the minimum value is $C = -\frac{1}{2}$ and the maximum value is $D = \frac{1}{2}$. The corresponding boundary functions are

$$\begin{aligned}\psi_1(y) &= -\sqrt{1-4y^2} \\ \psi_2(y) &= \sqrt{1-4y^2}.\end{aligned}$$

So Ω_E can be described with the two-part definition in equation (3.1),

$$\begin{aligned}\Omega_E &= \left\{ (x, y) \in \mathbf{R}^2 \mid -1 < x < 1, -\frac{1}{2}\sqrt{1-x^2} < y < \frac{1}{2}\sqrt{1-x^2} \right\} \\ &= \left\{ (x, y) \in \mathbf{R}^2 \mid -\frac{1}{2} < y < \frac{1}{2}, -\sqrt{1-4y^2} < x < \sqrt{1-4y^2} \right\}.\end{aligned}$$

The two spacing constants for this domain are distinct,

$$h_x = \frac{2}{N} \qquad h_y = \frac{1}{N}.$$

We use the same test parameters described above for unit circle. Results are shown in Tables 3.9 through 3.11.

Table 3.9: Results of numerical testing for the general Peaceman-Rachford method on the elliptic domain Ω_E using exact solution $u_2(x, y, t) = e^{3x+2y+t}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	1.099		5.325×10^{-1}	
10	$1/5$	0.4886	1.170	1.782×10^{-1}	1.580
20	$1/10$	0.1805	1.437	5.274×10^{-2}	1.756
40	$1/20$	0.05595	1.690	1.431×10^{-2}	1.882
80	$1/40$	0.01566	1.837	3.722×10^{-3}	1.942

Table 3.10: Results of numerical testing for the general Peaceman-Rachford method on the elliptic domain Ω_E using exact solution $u_3(x, y, t) = e^{xyt}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	3.316×10^{-3}		2.269×10^{-3}	
10	$1/5$	2.042×10^{-3}	-0.009	8.914×10^{-3}	1.348
20	$1/10$	8.355×10^{-4}	1.221	2.818×10^{-3}	1.662
40	$1/20$	2.644×10^{-4}	1.593	7.719×10^{-4}	1.868
80	$1/40$	7.462×10^{-4}	1.789	2.011×10^{-4}	1.940

Table 3.11: Results of numerical testing for the general Peaceman-Rachford method on the elliptic domain Ω_E using exact solution $u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t)$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	140.5		99.67	
10	$1/5$	68.95	1.027	33.71	1.564
20	$1/10$	4.765	3.855	2.165	3.961
40	$1/20$	2.213	1.106	0.6493	1.737
80	$1/40$	0.5652	1.969	0.1608	2.014

Example 4: A diamond

Next, we test the general Peaceman-Rachford method on the diamond-shaped domain Ω_D shown in Figure 3.4. The minimum x -value for this domain is $A = -1$, and the maximum is $B = 1$. Then the boundary functions are

$$\begin{aligned}\phi_1(x) &= \begin{cases} -\frac{1}{2}x - \frac{1}{2} & -1 \leq x \leq 0 \\ \frac{1}{2}x - \frac{1}{2} & 0 < x \leq 1 \end{cases} \\ \phi_2(x) &= \begin{cases} \frac{1}{2}x + \frac{1}{2} & -1 \leq x \leq 0 \\ -\frac{1}{2}x + \frac{1}{2} & 0 < x \leq 1 \end{cases}.\end{aligned}$$

In the y -direction the minimum value is $C = -\frac{1}{2}$ and the maximum value is $D = \frac{1}{2}$. The associated boundary functions are

$$\begin{aligned}\psi_1(y) &= \begin{cases} -2y - 1 & -\frac{1}{2} \leq y \leq 0 \\ 2y - 1 & 0 < y \leq \frac{1}{2} \end{cases} \\ \psi_2(y) &= \begin{cases} 2y + 1 & -\frac{1}{2} \leq y \leq 0 \\ -2y + 1 & 0 < y \leq \frac{1}{2} \end{cases}.\end{aligned}$$

So the domain has two-part definition

$$\begin{aligned}\Omega_D &= \{(x, y) \in \mathbf{R}^2 \mid -1 < x < 1, \phi_1(x) < y < \phi_2(x)\} \\ &= \{(x, y) \in \mathbf{R}^2 \mid -\frac{1}{2} < y < \frac{1}{2}, \psi_1(y) < x < \psi_2(y)\}.\end{aligned}$$

This test domain is important because of the way its corners are oriented relative to the axes. Near these corners, the discrete domain of gridpoints Ω_N will often contain rows and columns with a single interior point. The difference quotients for rows and columns of this type are a special case, as discussed above in Section 3.2.

Table 3.12: Results of numerical testing for the general Peaceman-Rachford method on the diamond-shaped domain Ω_D using exact solution $u_2(x, y, t) = e^{3x+2y+t}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	8.778×10^{-1}		4.719×10^{-1}	
10	$1/5$	2.713×10^{-1}	1.694	1.451×10^{-1}	1.701
20	$1/10$	8.200×10^{-2}	1.726	4.134×10^{-2}	1.812
40	$1/20$	2.340×10^{-2}	1.809	1.092×10^{-2}	1.921
80	$1/40$	6.440×10^{-3}	1.861	2.825×10^{-3}	1.950

Table 3.13: Results of numerical testing for the general Peaceman-Rachford method on the diamond-shaped domain Ω_D using exact solution $u_3(x, y, t) = e^{xyt}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	3.386×10^{-3}		1.228×10^{-3}	
10	$1/5$	1.302×10^{-3}	1.380	4.234×10^{-4}	1.536
20	$1/10$	4.143×10^{-4}	1.652	1.096×10^{-4}	1.949
40	$1/20$	1.174×10^{-4}	1.820	2.895×10^{-5}	1.921
80	$1/40$	3.189×10^{-5}	1.880	7.299×10^{-6}	1.988

Table 3.14: Results of numerical testing for the general Peaceman-Rachford method on the diamond-shaped domain Ω_D using exact solution $u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t)$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	68.21		29.55	
10	$1/5$	15.60	2.128	3.956	2.901
20	$1/10$	2.904	2.425	0.6765	2.548
40	$1/20$	0.7521	1.949	0.1610	2.071
80	$1/40$	0.1952	1.943	0.04026	2.000

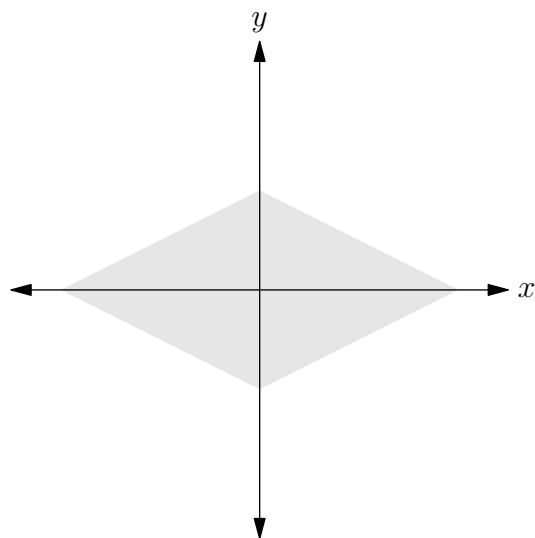


Figure 3.4: The diamond-shaped domain Ω_D

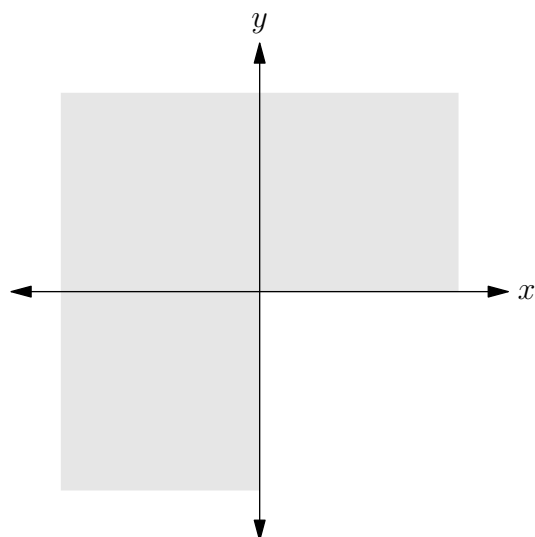


Figure 3.5: Ω_L , a domain with discontinuous boundary functions

Example 5: An L-shape

For the final test domain, we consider the L-shaped domain shown in Figure 3.5. The minimum x -value for this domain is $A = -1$ and the maximum is $B = 1$. Then the boundary functions are

$$\phi_1(x) = \begin{cases} -1 & -1 \leq x \leq 0 \\ 0 & 0 < x \leq 1 \end{cases}$$
$$\phi_2(x) = 1.$$

In the y -direction the minimum value is $C = -1$ and the maximum value is $D = 1$. The associated boundary functions are

$$\psi_1(y) = -1$$
$$\psi_2(y) = \begin{cases} 0 & -1 \leq y \leq 0 \\ 1 & 0 < y \leq 1 \end{cases}.$$

Then the domain has two-part definition

$$\begin{aligned} \Omega^L &= \{(x, y) \in \mathbf{R}^2 \mid -1 < x < 1, \phi_1(x) < y < \phi_2(x)\} \\ &= \{(x, y) \in \mathbf{R}^2 \mid -1 < y < 1, \psi_1(x) < x < \psi_2(y)\}. \end{aligned}$$

This domain is used as a test case to show that the general Peaceman-Rachford method can be used to find solutions on domains with discontinuous boundary conditions. We use the standard testing procedure, with results shown in Tables 3.15 through 3.17.

All of the different test cases support the contention that the Peaceman-Rachford method can be successfully extended to general regions in the plane. The generalized method demonstrates order of convergence of two under the discrete maximum norm. This is the same convergence as the standard Peaceman-Rachford method defined on a rectangular domain.

Table 3.15: Results of numerical testing for the general Peaceman-Rachford method on the L-shaped domain Ω_L using exact solution $u_2(x, y, t) = e^{3x+2y+t}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	5.179		3.373	
10	$1/5$	2.128	1.283	1.080	1.643
20	$1/10$	0.7971	1.417	0.3132	1.786
40	$1/20$	0.2627	1.601	0.08464	1.887
80	$1/40$	0.07914	1.731	0.02203	1.942

Table 3.16: Results of numerical testing for the general Peaceman-Rachford method on the L-shaped domain Ω_L using exact solution $u_3(x, y, t) = e^{xyt}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	6.113×10^{-3}		4.226×10^{-3}	
10	$1/5$	4.454×10^{-3}	0.457	2.058×10^{-3}	1.038
20	$1/10$	1.912×10^{-3}	1.220	6.828×10^{-4}	1.592
40	$1/20$	6.494×10^{-4}	1.558	1.952×10^{-4}	1.806
80	$1/40$	1.940×10^{-4}	1.743	5.211×10^{-4}	1.905

Table 3.17: Results of numerical testing for the general Peaceman-Rachford method on the L-shaped domain Ω_L using exact solution $u_4(x, y, t) = 10 \cos(16x^2 + 4y^2 + t)$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	219.6		143.1	
10	$1/5$	135.2	0.699	107.9	0.407
20	$1/10$	19.05	2.828	7.025	3.941
40	$1/20$	3.279	2.538	1.410	2.318
80	$1/40$	0.7767	2.077	0.3286	2.101

CHAPTER 4

ADI METHODS ON A THREE-DIMENSIONAL BOX

In three dimensions, the heat equation is

$$u_t - u_{xx} - u_{yy} - u_{zz} = f(x, y, z, t) \quad (4.1)$$

for $(x, y, z) \in \Omega$. We begin with the unit cube as the spatial domain, that is, $\Omega = (0, 1) \times (0, 1) \times (0, 1)$. For the temporal domain, let $t \in (0, T]$. The initial condition is

$$u(x, y, z, 0) = g_1(x, y, z) \quad \text{for } (x, y, z) \in \overline{\Omega},$$

and the Dirichlet boundary condition is

$$u(x, y, z, t) = g_2(x, y, z, t) \quad \text{for } (x, y, z) \in \partial\Omega, \quad t \in (0, T].$$

As in Chapter 2, we begin defining the discrete problem by dividing the temporal domain into $M + 1$ evenly spaced timesteps,

$$0 = t_0, t_1, \dots, t_M = T.$$

The temporal step size is $\tau = T/M$, and timestep t_m occurs at time τm .

Discretize the spatial domain with $N + 1$ equally spaced gridpoints in all three directions. Since the unit cube has the same length in every dimension, the step size $h = 1/N$ is also the same in every dimension. Then gridpoints have the form

$$(x_i, y_j, z_k)$$

where $x_i = ih$, $y_j = jh$, and $z_k = kh$ for $i, j, k = 0, 1, \dots, N$. Denote the approximate solution by

$$U_{i,j,k}^m \approx u(x_i, y_j, z_k, t_m).$$

4.1 The Douglas method

Although the Peaceman-Rachford method is successful for parabolic problems in 2D, the equivalent setup is not unconditionally stable in 3D. Details are given in Section 4.4.6 of Thomas [9] and will not be considered here.

Instead, we consider an ADI method for the 3D box given in equations (2.7) and (2.69) through (2.72) from the 1964 paper of Douglas & Gunn [3]. These equations refer to an earlier method devised by Douglas [2], and so we will call this the Douglas method,

$$\left(1 - \frac{\tau}{2}\delta_x^2\right) V_{i,j,k}^{m+1/3} = \left(1 + \frac{\tau}{2}\delta_x^2 + \tau\delta_y^2 + \tau\delta_z^2\right) U_{i,j,k}^m + \tau f_{i,j,k}^{m+1/2} \quad (4.2)$$

$$\left(1 - \frac{\tau}{2}\delta_y^2\right) V_{i,j,k}^{m+2/3} = V_{i,j,k}^{m+1/3} - \frac{\tau}{2}\delta_y^2 U_{i,j,k}^m \quad (4.3)$$

$$\left(1 - \frac{\tau}{2}\delta_z^2\right) U_{i,j,k}^{m+1} = V_{i,j,k}^{m+2/3} - \frac{\tau}{2}\delta_z^2 U_{i,j,k}^m \quad (4.4)$$

for $i, j, k = 1, \dots, N-1$ and $m = 0, \dots, M-1$. In this method $V_{i,j,k}^{m+1/3}, V_{i,j,k}^{m+2/3}$ are gridpoint functions that store approximate values for the two partial timesteps.

The discrete initial condition is

$$U_{i,j,k}^0 = g_1(x_i, y_j, z_k), \quad (4.5)$$

for $i, j, k = 0, 1, \dots, N$. On the right-hand side of equation (4.2), we need boundary values defined on integer timesteps. These are given by the boundary condition of the original PDE,

$$U_{i,j,k}^m = g_2(x_i, y_j, z_k, t_m) \quad (4.6)$$

for $i = 0, N, j, k = 1, \dots, N-1, m = 1, \dots, M$ or $j = 0, N, i, k = 1, \dots, N-1, m = 1, \dots, M$ or $k = 0, N, i, j = 1, \dots, N-1, m = 1, \dots, M$.

The boundary conditions for the partial timesteps $V_{i,j,k}^{m+1/3}, V_{i,j,k}^{m+2/3}$ are derived in a manner similar to that used for partial steps in Section 2.2. The form of the boundary condition used on the left-hand side of equation (4.3) is suggested by rearranging (4.4) to solve for $V_{i,j,k}^{m+2/3}$,

$$V_{i,j,k}^{m+2/3} = \left(1 - \frac{\tau}{2}\delta_z^2\right) U_{i,j,k}^{m+1} + \frac{\tau}{2}\delta_z^2 U_{i,j,k}^m. \quad (4.7)$$

for $i, j, k = 1, \dots, N - 1$ and $m = 0, 1, \dots, M - 1$. Since equation (4.3) involves δ_y^2 on the left-hand side, we will need boundary values for $j = 0$ and $j = N$. These are given by

$$V_{i,j,k}^{m+2/3} = \left(1 - \frac{\tau}{2}\delta_z^2\right) g_2(x_i, y_j, z_k, t_{m+1}) + \frac{\tau}{2}\delta_z g_2(x_i, y_j, z_k, t_m) \quad (4.8)$$

for $j = 0, N, i, k = 1, \dots, N - 1$ and $m = 0, 1, \dots, M - 1$. The values on the right-hand side come from the boundary condition of the original PDE.

We also need boundary values for the one-third partial timestep given in equation (4.2). The left-hand side of this equation involves δ_x^2 , and so we will need boundary values for $i = 0$ and $i = N$. For this step, the form of the boundary conditions is suggested by rearranging (4.3) to solve for the partial step $V_{i,j,k}^{m+1/3}$ and then applying equation (4.7),

$$\begin{aligned} V_{i,j,k}^{m+1/3} &= \left(1 - \frac{\tau}{2}\delta_y^2\right) V_{i,j,k}^{m+2/3} + \frac{\tau}{2}\delta_y^2 U_{i,j,k}^m \\ &= \left(1 - \frac{\tau}{2}\delta_y^2\right) \left(\left(1 - \frac{\tau}{2}\delta_z^2\right) U_{i,j,k}^{m+1} + \frac{\tau}{2}\delta_z^2 U_{i,j,k}^m \right) + \frac{\tau}{2}\delta_y^2 U_{i,j,k}^m \\ &= \left(1 - \frac{\tau}{2}\delta_y^2 - \frac{\tau}{2}\delta_z^2 + \frac{\tau^2}{4}\delta_y^2\delta_z^2\right) U_{i,j,k}^{m+1} + \left(\frac{\tau}{2}\delta_y^2 + \frac{\tau}{2}\delta_z^2 - \frac{\tau^2}{4}\delta_y^2\delta_z^2\right) U_{i,j,k}^m \end{aligned}$$

for $i, j, k = 1, \dots, N - 1$ and $m = 0, 1, \dots, M - 1$. This motivates the definition

$$\begin{aligned} V_{i,j,k}^{m+1/3} &= \left(1 - \frac{\tau}{2}\delta_y^2 - \frac{\tau}{2}\delta_z^2 + \frac{\tau^2}{4}\delta_y^2\delta_z^2\right) g_2(x_i, y_j, z_k, t_{m+1}) \\ &\quad + \left(\frac{\tau}{2}\delta_y^2 + \frac{\tau}{2}\delta_z^2 - \frac{\tau^2}{4}\delta_y^2\delta_z^2\right) g_2(x_i, y_j, z_k, t_m) \end{aligned} \quad (4.9)$$

for $i = 0, N, j, k = 1, \dots, N - 1$, and $m = 0, 1, \dots, M - 1$. In this expression the boundary condition for the one-third timestep is expressed in terms of g_2 , the boundary condition of the original PDE. Equations (4.8) and (4.9) are given as equations (4.2) and (4.3) in Fairweather & Mitchel [4].

We now have a complete set of equations with which to implement the Douglas method. The linear algebra for the implementation is very similar to the 2D case discussed in Section 2.3. Those details will not be repeated here.

4.2 Numerical results for the Douglas method

This method was tested with a group of different test functions,

$$u_1(x, y, z, t) = e^{x+2y+3z+4t} \quad (4.10)$$

$$u_2(x, y, z, t) = e^{xyzt} \quad (4.11)$$

$$u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t). \quad (4.12)$$

Each test function was used to give the initial condition, boundary condition, and forcing function f . Every function was tested with the sequence of grid spacings

$$h = 0.2 \times 2^n \quad \text{for } n = 0, \dots, 4. \quad (4.13)$$

For each trial, the timestep τ was set equal to h and the approximation was calculated at final time $T = 1$. Error was measured in two ways at the final timestep t_M . The maximum norm is

$$e_{\max} = \max_{i,j,k=1,\dots,N-1} |U_{i,j,k}^M - u(x_i, y_j, z_k, t_M)|,$$

and the discrete L^2 norm for a 3D domain is

$$e_{L^2} = \left(\sum_{i=1}^{N-1} \sum_{j=1}^{N-1} \sum_{k=1}^{N-1} h^3 (U_{i,j,k}^M - u(x_i, y_j, z_k, t_M))^2 \right)^{1/2}.$$

The order of convergence for each norm was calculated using equations (5.3).

Results of the tests are given in Tables 4.1 through 4.3. As expected, the Douglas method on the unit cube converges with order two under both norms.

4.3 Douglas method without perturbation terms on the boundary

Our goal is the extension of the Douglas method to general regions in 3D. To do this, we will need to adjust the calculation of the boundary values on the partial timesteps. On domains with curved boundaries, for example, we will not necessarily be able to find neighboring gridpoints on the boundary to evaluate at partial timesteps. In consequence, we

Table 4.1: Results of numerical testing for the Douglas method using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	23.68		7.929	
1/10	11.37	1.058	3.376	1.243
1/20	3.574	1.670	1.015	1.734
1/40	0.9714	1.880	0.2732	1.893
1/80	0.2522	1.946	0.07057	1.953

Table 4.2: Results of numerical testing for the Douglas method using exact solution $u_2(x, y, z, t) = e^{xyzt}$

τ	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	1.959×10^{-3}		7.749×10^{-4}	
1/10	1.019×10^{-3}	0.943	3.549×10^{-4}	1.126
1/20	3.281×10^{-4}	1.635	1.089×10^{-4}	1.705
1/40	9.008×10^{-5}	1.865	2.944×10^{-5}	1.887
1/80	2.341×10^{-5}	1.944	7.607×10^{-6}	1.952

Table 4.3: Results of numerical testing for the Douglas method using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$

τ	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	97.03		27.57	
1/10	18.89	2.360	3.573	2.948
1/20	3.232	2.548	0.7235	2.304
1/40	0.7632	2.082	0.1702	2.087
1/80	0.1881	2.020	0.04194	2.021

need to discard the higher-order perturbation terms in equations (4.9) and (4.8).

Equation (4.2) calculates the one-third partial step of the Douglas method. Here we use boundary values for the first and last point in each column. Without perturbation terms, these values are given by

$$V_{i,j,k}^{m+1/3} = g_2(x_i, y_j, z_k, t_{m+1}) \quad \text{for } j = 0, N \quad i, k = 1, \dots, N - 1, \quad (4.14)$$

the value at the same boundary point on the next integer timestep. This comes from g_2 , the boundary condition of the original PDE.

Similarly, for the two-thirds partial step given in equation (4.3), we use the boundary values at the first and last point in each stack. (A stack is the equivalent of a row or column, but extending in the z -direction.) Without the perturbation terms, these boundary values are

$$V_{i,j,k}^{m+2/3} = g_2(x_i, y_j, z_k, t_{m+1}) \quad \text{for } k = 0, N \quad i, j = 1, \dots, N - 1. \quad (4.15)$$

On both of these partial steps, no values at neighboring gridpoints are used. So we will be able to extend this approach to domains that do not have flat boundaries.

Aside from the calculation of boundary values on the partial timesteps in equations (4.14) and (4.15), the steps of the Douglas method are unchanged from those used in Section 4.3.

We evaluate the performance of this method by applying the same group of test functions given in equations(4.10) through (4.12), above. Each test was run until final time $T = 1$ using the grids given in equation (4.13). The order of convergence was calculated for both the max norm and the discrete L^2 norm. Results are shown in Tables 4.4 through 4.6.

The numerical results for the Douglas method in 3D show that removing the perturbation terms on the boundary changes the convergence of the method for the worse. Without the perturbation terms, convergence drops to order one under the max norm but remains two under the discrete L^2 norm.

Table 4.4: Results of numerical testing for the Douglas method without perturbation on the boundary terms using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	277.4		53.49	
1/10	225.7	0.298	25.09	1.092
1/20	130.0	0.796	8.345	1.588
1/40	65.93	0.971	2.348	1.830
1/80	32.36	1.027	0.6102	1.944

Table 4.5: Results of numerical testing for the Douglas method without perturbation terms on the boundary using exact solution $u_2(x, y, z, t) = e^{xyz t}$

τ	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	2.221×10^{-2}		4.014×10^{-2}	
1/10	1.753×10^{-2}	0.341	1.991×10^{-2}	1.011
1/20	9.908×10^{-3}	0.824	6.649×10^{-3}	1.582
1/40	5.047×10^{-3}	0.973	1.859×10^{-3}	1.839
1/80	2.510×10^{-3}	1.008	4.792×10^{-4}	1.956

Table 4.6: Results of numerical testing for the Douglas method without perturbation terms on the boundary using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$

τ	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	96.59		27.48	
1/10	18.92	2.352	3.576	2.942
1/20	3.258	2.538	0.7293	2.249
1/40	0.7786	2.065	0.1722	2.083
1/80	0.1907	2.029	0.04245	2.020

4.4 The Douglas method with partial perturbation on the boundary

It is also possible to define the Douglas method with perturbation terms in only one spatial direction along the boundary. Assume that we can always find neighboring boundary values in the z -direction. No such assumption is made about the x - or y -directions.

To calculate the one-third partial step in equation (4.2), the original Douglas method uses the boundary values given in equation (4.9). To match our assumption, we will keep only those perturbation terms that contain only δ_z^2 . We will discard any terms that contain δ_x^2 . So the formula for the boundary points becomes

$$V_{i,j,k}^{m+1/3} = \left(1 - \frac{\tau}{2}\delta_z^2\right) g_2(x_i, y_j, z_k, t_{m+1}) - \frac{\tau}{2}\delta_z^2 g_2(x_i, y_j, z_k, t_m) \quad (4.16)$$

for $j = 0, N, i, k = 1, \dots, N - 1, m = 0, 1, \dots, M - 1$. The boundary values on the two-thirds timestep in the Douglas method are given in equation (4.8). This formula contains only difference quotients in the z -direction to begin with, and so it can be used without change.

All other steps of the Douglas method are unchanged from Section 4.3.

The method with partial perturbation on the boundary was tested over the standard group of test functions and grids used above. Results are given in Tables 4.7 through 4.9.

Including perturbation terms in only one direction on the boundary improves the performance of the Douglas method. It now converges with order two under the max norm. This should translate into better performance on some types of cylindrical domains.

Table 4.7: Results of numerical testing for the Douglas method with partial perturbation on the boundary terms using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$

h	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	51.92		12.050	
1/10	55.85	-0.105	8.180	0.559
1/20	31.28	0.836	3.232	1.340
1/40	12.85	1.283	1.008	1.681
1/80	4.375	1.555	0.2806	1.844
1/160	1.327	1.721	0.07399	1.923

Table 4.8: Results of numerical testing for the Douglas method with partial perturbation terms on the boundary using exact solution $u_2(x, y, z, t) = e^{xyzt}$

τ	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	2.692×10^{-3}		9.192×10^{-4}	
1/10	1.387×10^{-3}	0.957	3.055×10^{-4}	1.589
1/20	1.043×10^{-3}	0.410	1.263×10^{-4}	1.274
1/40	4.843×10^{-4}	1.107	4.251×10^{-5}	1.571
1/80	1.744×10^{-4}	1.474	1.231×10^{-5}	1.787
1/160	5.412×10^{-5}	1.688	3.308×10^{-6}	1.896

Table 4.9: Results of numerical testing for the Douglas method with partial perturbation terms on the boundary using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$

τ	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
1/5	96.60		27.50	
1/10	18.89	2.355	3.250	2.946
1/20	3.250	2.539	0.7285	2.293
1/40	0.7760	2.066	0.1719	2.084
1/80	0.1902	2.029	0.04236	2.020

CHAPTER 5

ADI METHODS ON A GENERAL 3D REGION

We again seek solutions to equation (4.1), the heat equation in three dimensions. But now the domain Ω is a general shape in three dimensions. In order to create a rectangular array of gridpoints to cover Ω , we need to find real numbers $A_x, B_x, A_y, B_y, A_z, B_z$ such that

$$A_x < B_x, \quad A_y < B_y, \quad A_z < B_z$$

and

$$\Omega \subset [A_x, B_x] \times [A_y, B_y] \times [A_z, B_z].$$

So we use these constants to define the maximum and minimum coordinates of the domain in the three coordinate directions.

For ADI methods, we solve systems of linear equations corresponding to each direction. In Chapter 3, we solved the 2D problem by using the two-part definition of the domain given in equation (3.1). Each part of the definition let us find boundary points in one of the coordinate directions. For the 3D problem, we need a three-part description for the domain so that we can find boundary points in all three directions. Each direction is described with a pair of boundary surfaces, one for the lower boundary and one for the upper.

Let D_{xy} be the projection of Ω in the xy -plane. Then let the surfaces

$$\sigma_1, \sigma_2 : D_{xy} \rightarrow \mathbf{R}$$

be such that $\sigma_1(x, y) \leq \sigma_2(x, y)$ for $(x, y) \in D_{xy}$. Similarly, D_{yz} is the two-dimensional projection in the yz -plane. Let

$$\zeta_1, \zeta_2 : D_{yz} \rightarrow \mathbf{R}$$

be surfaces with $\zeta_1(y, z) \leq \zeta_2(y, z)$ for $(y, z) \in D_{yz}$. Finally, D_{xz} is the projection in the

xz -plane, with corresponding surfaces

$$\xi_1, \xi_2 : D_{xz} \rightarrow \mathbf{R}$$

where $\xi_1(x, z) \leq \xi_2(x, z)$ for $(x, z) \in D_{xz}$. Then the spatial domain of the problem $\Omega \subset \mathbf{R}^2$ satisfies the three-part definition,

$$\begin{aligned} \Omega &= \{(x, y, z) \mid (x, y) \in D_{xy}, \sigma_1(x, y) < z < \sigma_2(x, y)\} \\ &= \{(x, y, z) \mid (y, z) \in D_{yz}, \zeta_1(y, z) < x < \zeta_2(y, z)\} \\ &= \{(x, y, z) \mid (x, z) \in D_{xz}, \xi_1(x, z) < y < \xi_2(x, z)\} \end{aligned} \quad (5.1)$$

We will see that many familiar shapes in 3D can be described with this kind of definition, including a cube and a sphere.

The temporal domain of the problem is the interval $[0, T]$. The initial condition is

$$u(x, y, 0) = g_1(x, y, z) \quad \text{for } (x, y) \in \overline{\Omega}, \quad (5.2)$$

and the Dirichlet boundary condition is

$$u(x, y, t) = g_2(x, y, t) \quad \text{for } (x, y) \in \partial\Omega, \quad t \in (0, T]. \quad (5.3)$$

5.1 The discrete problem on a general 3D region

The first step in discretizing the domain is creating $M + 1$ evenly spaced time values

$$0 = t_0, t_1, \dots, t_M = T.$$

The temporal step size is $\tau = T/M$, and the value of each temporal coordinate is $t_m = m\tau$ for $m = 0, 1, \dots, M$.

For the discrete spatial domain, begin by dividing the x -, y -, and z -directions into

N steps. So the step size in the x -direction is

$$h_x = \frac{B_x - A_x}{N}, \quad (5.4)$$

the step size in the y -direction is

$$h_y = \frac{B_y - A_y}{N}, \quad (5.5)$$

and the step size in the z -direction is

$$h_z = \frac{B_z - A_z}{N}. \quad (5.6)$$

In general, the three directions have different spacing constants. These values h_x, h_y, h_z define an array of gridpoints (x_i, y_j, z_k) , where

$$\begin{aligned} x_i &= A_x + ih_x && \text{for } i = 0, 1, \dots, N \\ y_j &= A_y + jh_y && \text{for } j = 0, 1, \dots, N \\ z_k &= A_z + kh_z && \text{for } k = 0, 1, \dots, N. \end{aligned}$$

The domain of the original PDE lies inside the large rectangular array of gridpoints. Let Ω_N denote the set of all of these gridpoints that lie inside Ω , the domain of the PDE. Points in Ω_N are interior points, where we evaluate equations (4.2) through (4.4), the formulas of the Douglas method.

We saw in Chapter 4 that the Douglas method leads to tridiagonal systems of linear equations. At every timestep, we solve one such system for each row, each column, and each stack of gridpoints in the domain. (A stack of points is like a row, but extending in the z -direction.) We begin describing the discrete domain by creating a description of the rows, which extend in the x -direction.

A row is made of boundary points and interior points. We first describe the interior points. Each $i, k = 1, \dots, N - 1$ corresponds to one possible row of interior gridpoints in Ω_N . Points where $i = 0, N$ or $j = 0, N$ must be boundary points. Note that in three dimensions, we need two coordinates to specify a row of points. The points in row (j, k) share the same

y - and z -coordinates, $y_j = A_y + jh_y$ and $z_k = A_z + kh_z$. So all of these points lie on the horizontal line defined by the equations

$$y = y_j, \quad z = z_k.$$

It is possible for a horizontal line to intersect the domain in only one point, in which case the corresponding row contains no interior points. It is also possible that there are two distinct boundary points that lie so close together that there are no interior points in between them.

In either of these two cases, we describe the row as an empty row. If row (j, k) is not empty, we call it nonempty. In this case there are one or more gridpoints (x_i, y_j, z_k) that lie between the lower and upper boundary. So the x -coordinates of these points satisfy

$$\zeta_1(y_j, z_k) < x_i < \zeta_2(y_j, z_k). \quad (5.7)$$

These points are interior points of Ω_N .

If row (j, k) contains one or more interior points, then we define the constant $i_{\min}(j, k)$ to be the smallest $i = 1, \dots, N - 1$ such that x_i satisfies equation (5.7). Similarly, we define the constant $i_{\max}(j, k)$ to be the largest $i = 1, \dots, N - 1$ such that x_i satisfies (5.7). So then the complete set of interior points in row (j, k) is

$$(x_{i_{\min}(j,k)}, y_j, z_k), (x_{i_{\min}(j,k)+1}, y_j, z_k), \dots, (x_{i_{\max}(j,k)}, y_j, z_k).$$

It is possible for a row to contain only a single interior point. In this case, $i_{\min}(j, k)$ and $i_{\max}(j, k)$ are equal.

The boundary points for row (j, k) are the points where the horizontal line with

$$y = y_j, \quad z = z_k$$

intersects the boundary surfaces ζ_1 and ζ_2 . To describe these points of intersection, define

the constants

$$\begin{aligned}\zeta_{j,k(\ell)} &= \zeta_1(y_j, z_k) \\ \zeta_{j,k(u)} &= \zeta_2(y_j, z_k).\end{aligned}$$

The constant $\zeta_{j,k(\ell)}$ is the x -coordinate of the lower boundary point, and $\zeta_{j,k(u)}$ is the x -coordinate of the upper boundary point. Then the lower boundary point in row (j, k) has coordinates $(\zeta_{j,k(\ell)}, y_j, z_k)$, and the upper boundary point in row (j, k) has coordinates $(\zeta_{j,k(u)}, y_j, z_k)$.

To calculate the steps of the general Douglas method, we need values for the approximate solution U at every interior point and boundary point. Let

$$U_{i,j,k}^m \approx u(x_i, y_j, z_k, t_m)$$

denote the approximate solution to the PDE at an interior gridpoint. In row (j, k) , we will use $U'_{j,k(\ell)}{}^m$ to denote the boundary value at the lower boundary. So then

$$U'_{j,k(\ell)}{}^m = g_2(\zeta_{j,k(\ell)}, y_j, z_k),$$

using the boundary condition of the original PDE. Similarly, $U'_{j,k(u)}{}^m$ is the value of the solution at the upper boundary, and so

$$U'_{j,k(u)}{}^m = g_2(\zeta_{j,k(u)}, y_j, z_k).$$

The difference quotients in the general Douglas method also depend on the spacing constants between neighboring pairs of gridpoints. If Ω is a general domain in 3D, constants may be irregular. In particular, the distance between a boundary point and an interior point can be different than the distance between two interior points.

The distance between two neighboring interior points in the same row is always h_x . For every row containing at least one interior point, we also calculate two spacing constants,

one at each boundary. Let

$$h'_{j,k(\ell)} = x_{i_{\min}(j,k)} - \zeta_{j,k(\ell)}$$

be the spacing constant at the lower boundary, and let

$$h'_{j,k(u)} = \zeta_{j,k(u)} - x_{i_{\max}(j,k)}$$

be the spacing constant at the upper boundary.

Now we have a description of row (j, k) that will enable us to expand the difference quotients in the x -direction as needed for the Douglas method.

A similar set of constants is used to describe the columns of Ω_N . Every $i, k = 1, \dots, N - 1$ corresponds to one possible column of interior gridpoints. Points in the same column share an x -coordinate $x_i = A_x + ih_x$ and a z -coordinate, $z_k = A_z + kh_z$. So they lie on the same vertical line, defined by

$$x = x_i, \quad z = z_k.$$

If a vertical line intersects the domain Ω in only one point, then the lower boundary point and the upper boundary point are identical. It is also possible that the upper and lower boundary points are distinct, but so close together that there are no interior points between them. A column of either of these two types is called an empty column.

Otherwise, we say that column (i, k) is nonempty. In a column of this type there is at least one interior gridpoint (x_i, y_j, z_k) that lies between the lower and upper boundary surfaces. This point has y -coordinate satisfying

$$\xi_1(x_i, z_k) < y_j < \xi_2(x_i, z_k). \tag{5.8}$$

For every column (i, k) with one or more interior points, we define the constant $j_{\min}(i, k)$ to be the smallest $j = 1, \dots, N - 1$ such that y_j satisfies equation (5.8). Similarly, we define the constant $j_{\max}(i, k)$ to be the largest $j = 1, \dots, N - 1$ such that y_j satisfies

(5.8). It is possible that a nonempty row contains only a single interior point. In this case, the constants will be equal.

The boundary points for column (i, k) are the points where the vertical line with $x = x_i, z = z_k$ meets the boundary curves ξ_1 and ξ_2 given in equation (5.1). For the y -coordinates of these boundary points, define the constants

$$\begin{aligned}\xi_{i,k(\ell)} &= \xi_1(x_i, z_k) \\ \xi_{i,k(u)} &= \xi_2(x_i, z_k).\end{aligned}$$

Then the lower boundary point in column (i, k) has coordinates $(x_i, \xi_{i,k(\ell)}, z_k)$, and the upper boundary point has coordinates $(x_i, \xi_{i,k(u)}, z_k)$.

To evaluate the difference quotients in the Douglas method, we need the value of the approximate solution at both boundary points in column (i, k) . Let

$$U''_{i,k(\ell)}{}^m = g_2(x_i, \xi_{i,k(\ell)}, z_k, t_m)$$

be the value of the solution at the lower boundary in the column. Similarly, let

$$U''_{i,k(u)}{}^m = g_2(x_i, \xi_{i,k(u)}, z_k, t_m)$$

be the value of the solution at the upper boundary.

The spacing of gridpoints can be irregular near the boundary, and so we define the constants

$$\begin{aligned}h''_{i,k(\ell)} &= y_{j_{\min}(i,k)} - \xi_{i,k(\ell)} \\ h''_{i,k(u)} &= \xi_{i,k(u)} - y_{j_{\max}(i,k)}\end{aligned}$$

for the grid spacings at the lower and upper boundaries, respectively. We have one such pair of boundary spacing constants for column with at least one interior point.

The equivalent of a row or column for the z -direction is called a stack. Every $i, j = 1, \dots, N - 1$ corresponds to one possible stack of interior gridpoints. Points in the

same stack share an x -coordinate, $x_i = A_x + ih_x$ and a y -coordinate, $y_j = A_y + jh_y$. So they lie on a line parallel to the z -axis, defined by

$$x = x_i, \quad y = y_j.$$

If such a line intersects the domain Ω in only one point, then the lower boundary point and the upper boundary point are identical. It is also possible that the upper and lower boundary points are distinct, but so close together that there are no interior points between them. A stack of either of these two types is called an empty stack.

Otherwise, we call stack (i, j) nonempty. In a stack of this type there is at least one interior gridpoint (x_i, y_j, z_k) that lies between the lower and upper boundary points. This point has z -coordinate satisfying

$$\sigma_1(x_i, y_j) < z_k < \sigma_2(x_i, y_j). \quad (5.9)$$

For every stack (i, j) with one or more interior points, we define the constant $k_{\min}(i, j)$ to be the smallest $k = 1, \dots, N - 1$ such that z_k satisfies equation (5.9). Similarly, we define the constant $k_{\max}(i, j)$ to be the largest $k = 1, \dots, N - 1$ such that z_k satisfies (5.9). It is possible that a nonempty stack contains only a single interior point, and then the two constants will be equal.

The boundary points for stack (i, j) are the points where the line with

$$x = x_i, y = y_j$$

meets the boundary curves σ_1 and σ_2 given in equation (5.1). For the z -coordinates of these boundary points, define the constants

$$\begin{aligned} \sigma_{i,j(\ell)} &= \sigma_1(x_i, y_j) \\ \sigma_{i,j(u)} &= \sigma_2(x_i, y_j). \end{aligned}$$

Then the lower boundary point in stack (i, j) has coordinates $(x_i, y_j, \sigma_{i,j(\ell)})$, and the upper boundary point has coordinates $(x_i, y_j, \sigma_{i,j(u)})$.

To evaluate the difference quotients in the Douglas method, we need the value of the approximate solution at every interior point and boundary point in stack (i, j) . Let

$$U_{i,j(\ell)}'''^m = g_2(x_i, y_j, \sigma_{i,j,\ell}, t_m)$$

be value at the lower boundary of the stack. Similarly,

$$U_{j,k(u)}'''^m = g_2(x_i, y_j, \sigma_{i,j,u}, t_m)$$

is the value at the upper boundary.

The difference quotients in Douglas method also use the spacing between neighboring pairs of gridpoints. The spacing in the z -direction can be irregular near the boundary. So we define the constants

$$\begin{aligned} h_{i,j(\ell)}''' &= z_{k_{\min}(i,j)} - \sigma_{i,j(\ell)} \\ h_{i,j(u)}''' &= \sigma_{i,j(u)} - z_{k_{\max}(i,j)} \end{aligned}$$

for the grid spacings at the lower and upper boundaries, respectively. We have one such pair of boundary spacing constants for every stack with at least one interior point.

Now we have a complete description of the rows, columns, and stacks of the domain that will enable us to implement the Douglas method.

5.2 The Douglas method for a general 3D region

The method is based on three partial timesteps given in equations (4.2) through (4.4). The initial condition is given in equation (4.5), and the boundary conditions on integer timesteps is given in equation (4.6). For equation (4.2) we solve one tridiagonal matrix-vector equation for each row of gridpoints in the domain. This requires boundary values for the first and last points in the row on the partial timestep. Based on equation (4.14) for the

Douglas method without perturbation, we define these boundary values for the partial step,

$$\begin{aligned} U'_{j,k(\ell)}{}^{m+1/3} &= g_2(\zeta_{j,k,(\ell)}, y_j, z_k, t_{m+1}) \\ U'_{j,k(u)}{}^{m+1/3} &= g_2(\zeta_{j,k,(u)}, y_j, z_k, t_{m+1}) \end{aligned}$$

for $j, k = 1, \dots, N - 1$ where row (j, k) is nonempty.

The second partial step in the Douglas method is equation (4.3). Here we solve a tridiagonal equation for each column of gridpoints. So we need the boundary values for the first and last boundary points in each column on the second partial timestep. Based on equation (4.15), these boundary values are

$$\begin{aligned} U''_{i,k(\ell)}{}^{m+2/3} &= g_2(x_i, \xi_{i,k,(\ell)}, z_k, t_{m+1}) \\ U''_{i,k(u)}{}^{m+2/3} &= g_2(x_i, \xi_{i,k,(u)}, z_k, t_{m+1}) \end{aligned}$$

for $i, k = 1, \dots, N - 1$ where column (i, k) is nonempty.

The third step in the Douglas method (4.4). We solve an equation for each stack of gridpoints. The boundary values for these equations occur at integer time values, and so they come from the boundary condition of the original PDE in equation (4.6).

When expanding the difference quotients in the Douglas method, it is important to remember that gridpoints on arbitrary domains may be irregularly spaced. In this case, we need use the formulas for second-order central difference quotients given in Section 3.2.

5.3 Numerical results for the Douglas method on a general 3D region

We evaluate the extended Douglas method numerically. For each test, the first step is creating a description of the discrete domain, as described above. Divide all three directions into N steps, where

$$N = 5 \times 2^p \quad \text{for } p = 0, \dots, 4.$$

We use the same number of temporal steps. Since we want the approximate solution at final time $T = 1$, the temporal step size is

$$\tau = \frac{1}{N}.$$

In every test, the Douglas method is run to find the approximate solution at final time $T = 1$. So we use M timesteps, where

$$M = N.$$

We use the same test functions defined in Chapter 4,

$$u_1(x, y, t) = e^{x+2y+3z+4t}$$

$$u_2(x, y, t) = e^{xyzt}$$

$$u_3(x, y, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t).$$

The test function is used to define the initial condition, boundary condition, and forcing function f .

The error of the approximate solution is measured in two ways, the discrete maximum norm

$$e_{\max} = \max_{(x_i, y_j, z_k) \in \Omega_N} |U_{i,j,k}^M - u(x_i, y_j, z_k, t_M)|$$

and the discrete L^2 norm

$$e_{L^2} = \left(\sum_{(x_i, y_j, z_k) \in \Omega_N} h_x h_y h_z (U_{i,j,k}^M - u(x_i, y_j, z_k, t_M))^2 \right)^{1/2}.$$

For each norm, the order of convergence was estimated by

$$\text{numerical order of convergence}(\mathcal{O}) = \frac{\ln(e_n/e_{n+1})}{\ln(h_n/h_{n+1})}$$

where h_n, h_{n+1} are the maximum spatial step sizes on successive trials and e_n, e_{n+1} are the corresponding error measurements.

Example 1: The unit cube

Our first test domain is the unit cube, Ω_C . Since examined the Douglas method without perturbation terms on the unit cube in Chapter 4, we should see the same results. So this test case is a good check of the new code.

We begin by expressing the unit cube using the three-part definition given in equation (5.1). This definition is straightforward for a cube. The three domains in the planes defined by the coordinate axes are

$$\begin{aligned} D_{xy} &= \{(x, y) \mid 0 < x < 1, 0 < y < 1\} \\ D_{yz} &= \{(y, z) \mid 0 < y < 1, 0 < z < 1\} \\ D_{xz} &= \{(x, z) \mid 0 < x < 1, 0 < z < 1\}, \end{aligned}$$

and then the three-part definition is

$$\begin{aligned} \Omega_C &= \{(x, y, z) \mid (x, y) \in D_{xy}, 0 < z < 1\} \\ &= \{(x, y, z) \mid (y, z) \in D_{yz}, 0 < x < 1\} \\ &= \{(x, y, z) \mid (x, z) \in D_{xz}, 0 < y < 1\}. \end{aligned}$$

Results of the tests on the unit cube are shown in Tables 5.1 through 5.3.

In general, the method converges with order one under the discrete maximum norm, although some particular test functions may show better performance. The method converges with order two under the discrete L^2 norm. This is consistent with the performance of the Douglas method without perturbation from Chapter 4.

Example 2: The unit sphere

Although the results so far are good, we already knew that the Douglas method without perturbation terms works on a 3D box. The main purpose of this study is to look at its performance on non-rectangular domains. One simple non-rectangular domain is the unit sphere, Ω_S .

For this shape, the domains in the coordinate planes are circles,

$$D_{xy} = \{(x, y) \mid x^2 + y^2 < 1\}$$

Table 5.1: Results of numerical testing for the general Douglas method on the unit cube Ω_C using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	277.4		53.49	
10	$1/5$	225.7	0.298	25.09	1.092
20	$1/10$	130.0	0.796	8.345	1.588
40	$1/20$	65.94	0.979	2.348	1.830
80	$1/40$	32.36	1.027	0.6103	1.944

Table 5.2: Results of numerical testing for the general Douglas method on the unit cube Ω_C using exact solution $u_2(x, y, z, t) = e^{xyzt}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	2.221×10^{-2}		4.014×10^{-3}	
10	$1/5$	1.753×10^{-2}	0.341	1.991×10^{-3}	1.011
20	$1/10$	9.908×10^{-3}	0.824	6.649×10^{-4}	1.582
40	$1/20$	5.047×10^{-3}	0.973	1.859×10^{-4}	1.839
80	$1/40$	2.510×10^{-3}	1.008	4.792×10^{-5}	1.956

Table 5.3: Results of numerical testing for the general Douglas method on the unit cube Ω_C using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	96.59		27.48	
10	$1/5$	18.92	2.352	3.576	2.942
20	$1/10$	3.258	2.538	0.7293	2.294
40	$1/20$	0.7786	2.065	0.1722	2.083
80	$1/40$	0.1907	2.029	0.04245	2.020

$$D_{yz} = \{(y, z) \mid y^2 + z^2 < 1\}$$

$$D_{xz} = \{(x, z) \mid x^2 + z^2 < 1\}.$$

Then the unit sphere has three-part definition

$$\begin{aligned} \Omega_S &= \left\{ (x, y, z) \mid (x, y) \in D_{xy}, -\sqrt{1 - x^2 - y^2} < z < \sqrt{1 - x^2 - y^2} \right\} \\ &= \left\{ (x, y, z) \mid (y, z) \in D_{yz}, -\sqrt{1 - y^2 - z^2} < x < \sqrt{1 - y^2 - z^2} \right\} \\ &= \left\{ (x, y, z) \mid (x, z) \in D_{xz}, -\sqrt{1 - x^2 - z^2} < y < \sqrt{1 - x^2 - z^2} \right\}. \end{aligned}$$

Results are shown in Tables 5.4 through 5.6.

Example 3: An ellipsoid

Our next domain is the ellipsoid consisting of solutions to the inequality

$$x^2 + 4y^2 + 16z^2 < 1.$$

This is a simple example of a domain that has different sizes in the three coordinate directions.

The domains in the coordinate planes are

$$D_{xy} = \{(x, y) \mid x^2 + 4y^2 < 1\}$$

$$D_{yz} = \{(y, z) \mid 4y^2 + 16z^2 < 1\}$$

$$D_{xz} = \{(x, z) \mid x^2 + 16z^2 < 1\}.$$

Then the ellipsoid has three-part definition

$$\begin{aligned} \Omega_E &= \left\{ (x, y, z) \mid (x, y) \in D_{xy}, -\frac{1}{4}\sqrt{1 - x^2 - 4y^2} \leq z \leq \frac{1}{4}\sqrt{1 - x^2 - 4y^2} \right\} \\ &= \left\{ (x, y, z) \mid (y, z) \in D_{yz}, -\sqrt{1 - 4y^2 - 16z^2} \leq x \leq \sqrt{1 - 4y^2 - 16z^2} \right\} \\ &= \left\{ (x, y, z) \mid (x, z) \in D_{xz}, -\frac{1}{2}\sqrt{1 - x^2 - 16z^2} \leq y \leq \frac{1}{2}\sqrt{1 - x^2 - 16z^2} \right\}. \end{aligned}$$

Results are shown in Tables 5.7 through 5.9.

Table 5.4: Results of numerical testing for the general Douglas method on the unit sphere Ω_S using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	15.51		7.691	
10	$1/5$	9.070	0.774	3.493	1.138
20	$1/10$	5.535	0.713	1.203	1.538
40	$1/20$	3.046	0.862	0.3554	1.758
80	$1/40$	1.464	1.057	0.09820	1.856

Table 5.5: Results of numerical testing for the general Douglas method on the unit sphere Ω_S using exact solution $u_2(x, y, z, t) = e^{xyzt}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	3.953×10^{-3}		2.848×10^{-3}	
10	$1/5$	1.370×10^{-3}	1.529	7.956×10^{-4}	1.840
20	$1/10$	1.430×10^{-3}	-0.062	2.995×10^{-4}	1.410
40	$1/20$	7.707×10^{-4}	0.892	9.829×10^{-5}	1.607
80	$1/40$	3.691×10^{-4}	1.062	2.885×10^{-5}	1.769

Table 5.6: Results of numerical testing for the general Douglas method on the unit sphere Ω_S using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	158.6		144.3	
10	$1/5$	87.83	0.853	52.98	1.446
20	$1/10$	6.577	3.739	3.650	3.859
40	$1/20$	2.500	1.396	0.9007	2.019
80	$1/40$	0.6454	1.953	0.2296	1.972

Table 5.7: Results of numerical testing for the general Douglas method on the ellipsoid Ω_E using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	1.663		0.3033	
10	$1/5$	1.246	0.417	0.1432	1.083
20	$1/10$	0.7474	0.737	0.05515	1.376
40	$1/20$	0.3920	0.931	0.01839	1.584
80	$1/40$	0.1921	1.029	0.005622	1.710

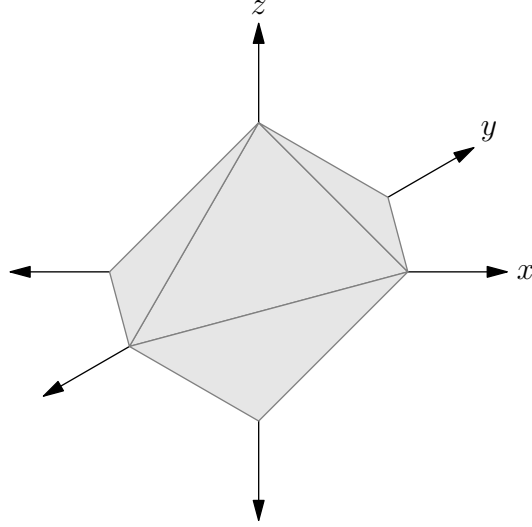


Figure 5.1: Ω_O , an octahedron

Example 4: An octahedron

The next domain is the octahedron Ω_O with vertices at coordinates $(\pm 1, 0, 0)$, $(0, \pm 1, 0)$, $(0, 0, \pm 1)$, pictured in Figure 5.1. This shape is important because the corners are positioned so that the discrete domain has some rows, columns, and stacks that contain a single interior point. So the octahedron tests the part of the code that deals with that special case.

To define the domains in the coordinate plans, first let

$$\phi_1(x) = \begin{cases} -x - 1 & -1 \leq x \leq 0 \\ x - 1 & 0 < x \leq 1 \end{cases}$$

$$\phi_2(x) = \begin{cases} x + 1 & -1 \leq x \leq 0 \\ -x + 1 & 0 < x \leq 1, \end{cases}$$

and then in the xy -plane, the octahedron has projection

$$D_{xy} = \{(x, y) \mid -1 < x < 1, \phi_1(x) < y < \phi_2(x)\}.$$

The definitions are symmetric in the other two planes. So

$$D_{yz} = \{(y, z) \mid -1 < y < 1, \psi_1(y) < z < \phi_2(y)\},$$

where

$$\psi_1(y) = \begin{cases} -y - 1 & -1 \leq y \leq 0 \\ y - 1 & 0 < y \leq 1 \end{cases}$$

$$\psi_2(y) = \begin{cases} y + 1 & -1 \leq y \leq 0 \\ -y + 1 & 0 < y \leq 1, \end{cases}$$

and

$$D_{xz} = \{(x, z) \mid -1 < z < 1, \chi_1(z) < x < \chi_2(z)\}.$$

where

$$\chi_1(z) = \begin{cases} -z - 1 & -1 \leq z \leq 0 \\ z - 1 & 0 < z \leq 1 \end{cases}$$

$$\chi_2(z) = \begin{cases} z + 1 & -1 \leq z \leq 0 \\ -z + 1 & 0 < z \leq 1. \end{cases}$$

We use these domains to construct the three-part definition of the octahedron,

$$\begin{aligned} \Omega_O &= \{(x, y, z) \mid (x, y) \in D_{xy}, \sigma_1(x, y) < z < \sigma_2(x, y)\} \\ &= \{(x, y, z) \mid (y, z) \in D_{yz}, \zeta_1(y, z) < x < \zeta_2(y, z)\} \\ &= \{(x, y, z) \mid (x, z) \in D_{xz}, \xi_1(x, z) < y < \xi_2(x, z)\}. \end{aligned}$$

where $\sigma_1, \sigma_2 : D_{xy} \rightarrow \mathbf{R}$,

$$\sigma_1(x, y) = \begin{cases} -x - y + 1 & -1 < x \leq 0, \phi_1(x) < y \leq 0 \\ -x + y + 1 & -1 < x \leq 0, 0 < y < \phi_2(x) \\ x - y + 1 & 0 < x < 1, \phi_1(x) < y \leq 0 \\ x + y + 1 & 0 < x < 1, 0 < y < \phi_2(x) \end{cases}$$

$$\sigma_2(x, y) = \begin{cases} x + y + 1 & -1 < x \leq 0, \phi_1(x) < y \leq 0 \\ x - y + 1 & -1 < x \leq 0, 0 < y < \phi_2(x) \\ -x + y + 1 & 0 < x < 1, \phi_1(x) < y \leq 0 \\ -x - y + 1 & 0 < x < 1, 0 < y < \phi_2(x), \end{cases}$$

$\zeta_1, \zeta_2 : D_{yz} \rightarrow \mathbf{R}$,

$$\zeta_1(y, z) = \begin{cases} -y - z + 1 & -1 < y \leq 0, \psi_1(y) < z \leq 0 \\ -y + z + 1 & -1 < y \leq 0, 0 < z < \psi_2(y) \\ y - z + 1 & 0 < y < 1, \psi_1(y) < z \leq 0 \\ y + z + 1 & 0 < y < 1, 0 < z < \psi_2(y) \end{cases}$$

$$\zeta_2(y, z) = \begin{cases} y + z + 1 & -1 < y \leq 0, \psi_1(y) < z \leq 0 \\ y - z + 1 & -1 < y \leq 0, 0 < z < \psi_2(y) \\ -y + z + 1 & 0 < y < 1, \psi_1(y) < z \leq 0 \\ -y - z + 1 & 0 < y < 1, 0 < z < \psi_2(y), \end{cases}$$

and $\xi_1, \xi_2 : D_{xz} \rightarrow \mathbf{R}$,

$$\xi_1(x, z) = \begin{cases} -z - x + 1 & -1 < z \leq 0, \chi_1(z) < x \leq 0 \\ -z + x + 1 & -1 < z \leq 0, 0 < x < \chi_2(z) \\ z - x + 1 & 0 < z < 1, \chi_1(z) < x \leq 0 \\ z + x + 1 & 0 < z < 1, 0 < x < \chi_2(z) \end{cases}$$

$$\xi_2(x, z) = \begin{cases} z + x + 1 & -1 < z \leq 0, \chi_1(z) < x \leq 0 \\ z - x + 1 & -1 < z \leq 0, 0 < x < \chi_2(z) \\ -z + x + 1 & 0 < z < 1, \chi_1(z) < x \leq 0 \\ -z - x + 1 & 0 < z < 1, 0 < x < \chi_2(z). \end{cases}$$

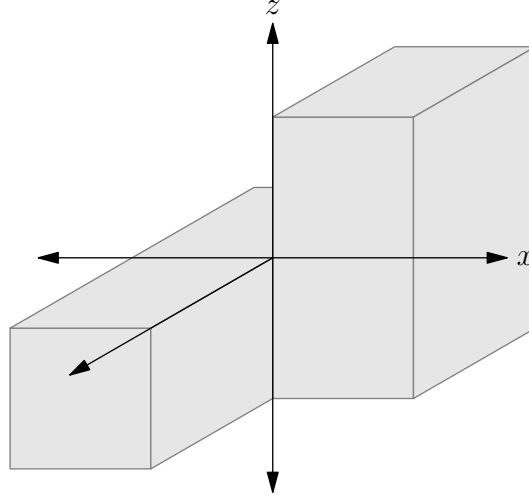


Figure 5.2: Ω_L , a domain with discontinuous boundary functions

Results of testing on the octahedron are shown in Tables 5.10 through 5.12.

Example 5: A twisted L-shape

The final domain is Ω_L , an L-shape that has been twisted into the third dimension. A picture of this domain is shown in Figure 5.2. This test case is interesting because the boundary functions that define the domain are discontinuous. Start by considering the projection in the xy -plane. Define

$$\phi_1(x) = \begin{cases} -1 & -1 \leq x \leq 0 \\ 0 & 0 < x \leq 1 \end{cases}$$

$$\phi_2(x) = 1.$$

Then in the xy -plane, the projection is

$$D_{xy} = \{(x, y) \mid -1 < x < 1, \phi_1(x) < y < \phi_2(x)\}.$$

The steps are similar for the other two coordinate planes. Let

$$\psi_1(y) = -1$$

Table 5.8: Results of numerical testing for the general Douglas method on the ellipsoid Ω_E using exact solution $u_2(x, y, z, t) = e^{xyz t}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	1.956×10^{-4}		5.035×10^{-5}	
10	$1/5$	1.489×10^{-4}	0.392	2.492×10^{-5}	1.015
20	$1/10$	9.503×10^{-5}	0.648	1.080×10^{-5}	1.207
40	$1/20$	5.002×10^{-5}	0.926	3.795×10^{-6}	1.508
80	$1/40$	2.464×10^{-5}	1.021	1.203×10^{-6}	1.657

Table 5.9: Results of numerical testing for the general Douglas method on the ellipsoid Ω_E using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	35.27		10.76	
10	$1/5$	30.14	0.227	5.591	0.944
20	$1/10$	3.768	3.000	0.7844	2.900
40	$1/20$	1.639	1.201	0.2176	1.783
80	$1/40$	0.4118	1.992	0.05405	2.009

Table 5.10: Results of numerical testing for the general Douglas method on the octahedron Ω_O using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	3.689		1.248	
10	$1/5$	1.659	1.152	0.5032	1.311
20	$1/10$	0.5563	1.577	0.1259	1.999
40	$1/20$	0.2299	1.275	0.03221	1.967
80	$1/40$	0.09220	1.318	0.0082100	1.972

Table 5.11: Results of numerical testing for the general Douglas method on the octahedron Ω_O using exact solution $u_2(x, y, z, t) = e^{xyz t}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	3.403×10^{-5}		2.411×10^{-5}	
10	$1/5$	1.245×10^{-4}	-1.871	3.449×10^{-5}	-0.517
20	$1/10$	5.842×10^{-5}	1.092	1.334×10^{-5}	1.370
40	$1/20$	2.395×10^{-5}	1.286	4.260×10^{-6}	1.700
80	$1/40$	9.871×10^{-6}	1.279	1.213×10^{-6}	1.767

$$\psi_2(y) = \begin{cases} 0 & -1 \leq y \leq 0 \\ 1 & 0 < y \leq 1, \end{cases}$$

so that

$$D_{yz} = \{(y, z) \mid -1 < y < 1, \psi_1(y) < z < \phi_2(y)\},$$

and let

$$\chi_1(z) = \begin{cases} -1 & -1 \leq z \leq 0 \\ 0 & 0 < z \leq 1 \end{cases}$$

$$\chi_2(z) = 1,$$

so then

$$D_{xz} = \{(x, z) \mid -1 < z < 1, \chi_1(z) < x < \chi_2(z)\}.$$

Now we use the domains to create the three-part definition,

$$\begin{aligned} \Omega_L &= \{(x, y, z) \mid (x, y) \in D_{xy}, \sigma_1(x, y) < z < \sigma_2(x, y)\} \\ &= \{(x, y, z) \mid (y, z) \in D_{yz}, \zeta_1(y, z) < x < \zeta_2(y, z)\} \\ &= \{(x, y, z) \mid (x, z) \in D_{xz}, \xi_1(x, z) < y < \xi_2(x, z)\}. \end{aligned}$$

where $\sigma_1, \sigma_2 : D_{xy} \rightarrow \mathbf{R}$,

$$\sigma_1(x, y) = \begin{cases} -1 & -1 < x \leq 0, \phi_1(x) < y \leq 0 \\ -1 & -1 < x \leq 0, 0 < y < \phi_2(x) \\ -1 & 0 < x < 1, \phi_1(x) < y < \phi_2(x) \end{cases}$$

$$\sigma_2(x, y) = \begin{cases} 0 & -1 < x \leq 0, \phi_1(x) < y \leq 0 \\ 0 & -1 < x \leq 0, 0 < y < \phi_2(x) \\ 1 & 0 < x < 1, \phi_1(x) < y < \phi_2(x), \end{cases}$$

$\zeta_1, \zeta_2 : D_{yz} \rightarrow \mathbf{R}$,

$$\zeta_1(y, z) = \begin{cases} -1 & -1 < y \leq 0, \psi_1(y) < z < \psi_2(y) \\ -1 & 0 < y < 1, \psi_1(y) < z \leq 0 \\ 0 & 0 < y < 1, 0 < z < \psi_2(y) \end{cases}$$

$$\zeta_2(y, z) = \begin{cases} 0 & -1 < y \leq 0, \psi_1(y) < z < \psi_2(y) \\ 1 & 0 < y < 1, \psi_1(y) < z \leq 0 \\ 1 & 0 < y < 1, 0 < z < \psi_2(y), \end{cases}$$

and $\xi_1, \xi_2 : D_{xz} \rightarrow \mathbf{R}$,

$$\xi_1(x, z) = \begin{cases} -1 & -1 < z \leq 0, \chi_1(z) < x \leq 0 \\ 0 & -1 < z \leq 0, 0 < x < \chi_2(z) \\ 0 & 0 < z < 1, \chi_1(z) < x < \chi_2(z) \end{cases}$$

$$\xi_2(x, z) = \begin{cases} 0 & -1 < z \leq 0, \chi_1(z) < x \leq 0 \\ 1 & -1 < z \leq 0, 0 < x < \chi_2(z) \\ 1 & 0 < z < 1, \chi_1(z) < x < \chi_2(z). \end{cases}$$

Tests run on this domain give the results shown in Tables 5.13 through 5.15.

All of the results in this chapter are consistent with an order of convergence of one under the discrete maximum norm and two under the discrete L^2 norm. So we conclude that it is possible to extend the ADI approach to general domains in three dimensions. This approach is most successful with regards to the L^2 norm.

Based on the results in Section 4.5, the Douglas method with partial perturbation terms converges with order two under the discrete maximum norm. It should be possible to extend this method from a box to a cylinder, as long the sides of the cylinder are parallel to one of the coordinate axes. Then, in the direction of this axes, we will always be able to find perturbation terms along the boundary of the cylinder. So, on this cylinder, we expect

Table 5.12: Results of numerical testing for the general Douglas method on the octahedron Ω_O using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	18.46		13.21	
10	$1/5$	37.72	-1.031	8.561	0.625
20	$1/10$	3.725	3.340	0.9939	3.107
40	$1/20$	1.269	1.554	0.2323	2.097
80	$1/40$	0.3031	2.066	0.05685	2.031

Table 5.13: Results of numerical testing for the general Douglas method on the L-shaped domain Ω_L using exact solution $u_1(x, y, z, t) = e^{x+2y+3z+4t}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	137.7		50.36	
10	$1/5$	113.2	0.283	26.03	0.952
20	$1/10$	63.01	0.845	9.038	1.526
40	$1/20$	29.06	1.117	2.608	1.793
80	$1/40$	12.95	1.166	0.6928	1.912

Table 5.14: Results of numerical testing for the general Douglas method on the L-shaped domain Ω_L using exact solution $u_2(x, y, z, t) = e^{xyzt}$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	4.600×10^{-3}		2.454×10^{-3}	
10	$1/5$	6.190×10^{-3}	-0.428	1.799×10^{-3}	0.448
20	$1/10$	4.340×10^{-3}	0.512	7.165×10^{-4}	1.328
40	$1/20$	2.224×10^{-3}	0.965	2.193×10^{-4}	1.708
80	$1/40$	1.033×10^{-3}	1.106	5.929×10^{-5}	1.887

Table 5.15: Results of numerical testing for the general Douglas method on the L-shaped domain Ω_L using exact solution $u_3(x, y, z, t) = 10 \cos(16x^2 + 4y^2 + z^2 + t)$.

N	h_{\max}	e_{\max}	\mathcal{O}_{\max}	e_{L^2}	\mathcal{O}_{L^2}
5	$2/5$	96.13		64.25	
10	$1/5$	117.1	-0.003	74.77	-0.219
20	$1/10$	19.09	2.617	7.620	3.295
40	$1/20$	3.319	2.524	1.545	2.302
80	$1/40$	0.7845	2.081	0.3614	2.096

to see convergence of order two under the discrete maximum norm. This question is one possible topic for future research.

APPENDIX
ELECTRONIC FILES

Chapter 2

- u.m Test functions used for the exact solution to the PDE.
- f.m The functions for f on the right-hand side of the PDE.
- g1.m Initial conditions.
- g2.m Boundary conditions.
- PR.m The Peaceman-Rachford method on the unit square. Different test functions and different types of perturbation are selected with constants at the top of the file.

Chapter 3

- u.m Test functions used for the exact solution to the PDE.
- f.m The functions for f on the right-hand side of the PDE.
- g1.m Initial conditions.
- g2.m Boundary conditions.
- phi1.m Lower boundary curves in y for test domains.
- phi2.m Upper boundary curves in y for test domains.
- psi1.m Lower boundary curves in x for test domains.
- psi2.m Upper boundary curves in x for test domains.
- PR.m The Peaceman-Rachford method on general 2D domains. Different test functions and different spatial domains are selected with constants at the top of the file.
- Dyakonov.m The Dyakonov method on general 2D domains. Different test functions and different spatial domains are selected with constants at the top of the file.

Chapter 4

u3.m	Test functions used for the exact solution to the PDE.
f3.m	The functions for f on the right-hand side of the PDE.
g3.m	Initial conditions.
g4.m	Boundary conditions.
Douglas.m	The Douglas method on the unit cube. Different test functions and different types of perturbation are selected with constants at the top of the file.

Chapter 5

u3.m	Test functions used for the exact solution to the PDE.
f3.m	The functions for f on the right-hand side of the PDE.
g3.m	Initial conditions.
g4.m	Boundary conditions.
phi1.m	Lower boundary curves in the xy -plane for test domains.
phi2.m	Upper boundary curves in the xy -plane for test domains.
zeta1.m	Lower boundary surfaces in z for test domains.
zeta2.m	Upper boundary surfaces in z for test domains.
eta1.m	Lower boundary surfaces in x for test domains.
eta2.m	Upper boundary surfaces in x for test domains.
theta1.m	Lower boundary surfaces in y for test domains.
theta2.m	Upper boundary surfaces in y for test domains.
Douglas.m	The Douglas method on general 3D domains. Different test functions and different spatial domains are selected with constants at the top of the file.

REFERENCES CITED

- [1] B. Bialecki and R. I. Fernandes. Orthogonal spline collocation Laplace-modified and alternating direction methods for parabolic problems on rectangles. *Mathematics of Computation*, 60:545–573, 1993.
- [2] J. Douglas. Alternating direction methods for three space variables. *Numerische Mathematik*, 4(1):41–63, 1962.
- [3] J. Douglas and J. E. Gunn. A general formulation of alternating direction methods. I. Parabolic and hyperbolic problems. *Numerische Mathematik*, 6:428–453, 1964.
- [4] G. Fairweather and A. R. Mitchell. A new computational procedure for A.D.I. methods. *SIAM Journal on Numerical Analysis*, 4(2):163–170, 1967.
- [5] K. W. Morton and D. F. Mayers. *Numerical solution of partial differential equations: An introduction*. Cambridge University Press, Cambridge, 1994.
- [6] D. W. Peaceman and H. H. Rachford. The Numerical Solution of Parabolic and Elliptic Differential Equations. *Journal of the Society for Industrial and Applied Mathematics*, 3(1):28–41, 1955.
- [7] A. A. Samarskii. *The theory of difference schemes*. Marcel Dekker, New York, 2001.
- [8] A. P. S. Selvadurai. *Partial differential equations in mechanics 1: Fundamentals, Laplace's equation, diffusion equation, wave equation*. Springer-Verlag, Berlin, 2000.
- [9] J. W. Thomas. *Numerical partial differential equations: Finite difference methods*. Springer-Verlag, New York, 1995.