

1-1-2005

The method of fundamental solutions for solving wave equations

Lourdes Jeanette Rivera
University of Nevada, Las Vegas

Follow this and additional works at: <https://digitalscholarship.unlv.edu/rtds>

Repository Citation

Rivera, Lourdes Jeanette, "The method of fundamental solutions for solving wave equations" (2005).
UNLV Retrospective Theses & Dissertations. 1850.
<http://dx.doi.org/10.25669/bl58-7bw4>

This Thesis is protected by copyright and/or related rights. It has been brought to you by Digital Scholarship@UNLV with permission from the rights-holder(s). You are free to use this Thesis in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you need to obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/or on the work itself.

This Thesis has been accepted for inclusion in UNLV Retrospective Theses & Dissertations by an authorized administrator of Digital Scholarship@UNLV. For more information, please contact digitalscholarship@unlv.edu.

THE METHOD OF FUNDAMENTAL SOLUTIONS
FOR SOLVING WAVE EQUATIONS

by

Lourdes Jeanette Rivera

Bachelor of Arts
University of Guam
2003

A thesis submitted in partial fulfillment
of the requirements for the

Master of Science Degree in Mathematical Sciences
Mathematical Sciences Department
College of Sciences

Graduate College
University of Nevada, Las Vegas
August 2005

UMI Number: 1429724

Copyright 2005 by
Rivera, Lourdes Jeanette

All rights reserved.

INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

UMI®

UMI Microform 1429724

Copyright 2006 by ProQuest Information and Learning Company.

All rights reserved. This microform edition is protected against
unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

Copyright © 2005 by Lourdes Jeanette Rivera.
All rights reserved



Thesis Approval

The Graduate College
University of Nevada, Las Vegas

July 6, 05
_____, 20____

The Thesis prepared by

Lourdes Jeanette Rivera

Entitled

The Method of Fundamental Solutions for Solving Wave Equations

is approved in partial fulfillment of the requirements for the degree of

Master of Science in Mathematical Sciences

Malvane Amanda
for C.S. Chen

Examination Committee Chair

Chale H. King

Dean of the Graduate College

Zhonghai Ding

Examination Committee Member

Li

Examination Committee Member

Malvane Amanda
for Tao Pang

Graduate College Faculty Representative

ABSTRACT

The Method of Fundamental Solutions for Solving Wave Equations

by

Lourdes Jeanette Rivera

Dr. C. S. Chen, Examination Committee Chair
Professor of Mathematics
University of Nevada, Las Vegas

In recent years, the method of fundamental solutions (MFS) has emerged as a novel meshless method in the scientific computing community. In the past, the MFS was essentially restricted to solving homogeneous elliptic equations. Recently, the MFS has gradually extended to solving various types of elliptic and time-dependent problems through the uses of radial basis functions (RBFs).

In this thesis, we focus on solving wave equations through the MFS. Currently, there are two major approaches to solve the wave equation: (i) elimination of the time dependence by using the Laplace transform and (ii) discretization in time to approximate the time derivative. We propose to reduce the given wave equations to a series of inhomogeneous modified Helmholtz equations. The solution can then be split into evaluating both homogeneous and particular solutions. To evaluate the homogeneous solution, the MFS is adopted. Furthermore, a closed form particular solution is required for the proposed method. Intensive numerical tests are performed to compare the advantages and disadvantages of these two approaches.

TABLE OF CONTENTS

ABSTRACT.....	iii
LIST OF ABBREVIATIONS.....	vi
LIST OF FIGURES	vii
LIST OF TABLES.....	ix
ACKNOWLEDGEMENTS.....	x
CHAPTER 1 INTRODUCTION	1
1.1 Meshless Methods.....	1
1.2 Thesis Overview	3
CHAPTER 2 RADIAL BASIS FUNCTIONS	6
2.1 Definition	7
2.2 Interpolation and Approximation.....	8
2.2.1 Positive Definite vs. Conditionally Positive Definite.....	9
2.2.2 Eigenvalues and Condition Number	10
2.2.3 Compactly-Supported Radial Basis Functions	10
2.3 Surface Reconstruction	12
2.4 Concluding Remarks.....	16
CHAPTER 3 THE METHOD OF FUNDAMENTAL SOLUTIONS.....	18
3.1 Methodology and Fundamental Solutions	19
3.2 Numerical Implementation	21
3.3 Analysis of Accuracy.....	23
3.4 Concluding Remarks.....	25
CHAPTER 4 THE METHOD OF PARTICULAR SOLUTIONS	26
4.1 Methodology	27
4.2 Using the Thin Plate Spline	29
4.3 Particular Solutions for the Laplacian Operator, $L = \Delta$	30
4.4 Particular Solutions for Helmholtz-Type Operators, $L = \Delta \pm \lambda^2$	32
4.4.1 Higher Order Splines	33
4.5 Numerical Implementation	36
4.6 Analysis of Accuracy.....	39
4.7 Concluding Remarks.....	43

CHAPTER 5	THE WAVE EQUATION.....	44
5.1	The Laplace Transform.....	45
5.1.1	The Inverse Laplace Transform: Stehfest's Algorithm	46
5.1.2	Numerical Implementation	47
5.1.3	Analysis of Accuracy	51
5.2	The Time-Stepping Scheme.....	59
5.2.1	Numerical Implementation	62
5.2.2	Observation on Numerical Implementation.....	65
5.2.3	Analysis of Accuracy	66
5.3	Laplace Transform vs. Time-Stepping Scheme.....	74
5.4	Concluding Remarks.....	77
CHAPTER 6	CONCLUSIONS AND FUTURE RESEARCH	79
6.1	Conclusions.....	79
6.2	Recommendations for Future Research	80
REFERENCES	.....	82
VITA	.....	85

LIST OF ABBREVIATIONS

2D	two-dimension
3D	three-dimension
BEM	boundary element method
CPD	conditionally positive definite
CS-RBF	compactly supported radial basis function
DRM	dual reciprocity method
FDM	finite difference method
FEM	finite element method
MFS	method of fundamental solutions
MPS	method of particular solutions
MQ	multiquadric
PDE	partial differential equation
RBF	radial basis function
SPD	symmetric positive definite
TPS	thin plate spline

LIST OF FIGURES

1.1	Meshed domains in 2D and 3D.....	2
1.2	Meshless domains (using scattered nodes) in 2D and 3D.....	2
2.1	2D representation of distances among n centers ($n = 5$)	8
2.2	Graph of $f(x, y) = -xye^{-x^2-y^2}$	12
2.3	Uniform distribution of 144 x - y pairs	13
2.4	144 known data points (uniform distribution)	14
2.5	Absolute error of approximation.....	14
2.6	Random distribution of 144 x - y pairs.....	15
2.7	144 known data points (random distribution).....	15
2.8	Absolute error of approximation.....	16
3.1	Distribution of collocation and source points for the MFS.....	20
3.2	Distribution of collocation and source points on a simply-connected domain.....	22
3.3	Distribution of collocation and source points on a multiply-connected domain ...	23
4.1	Graph of $u(x, y) = e^x + e^y$	37
4.2	Distribution of interpolation, collocation, and source points.....	38
4.3	The effect of λ^2 on the accuracy of the MPS.....	39
4.4	The effect of the # of source points, m , on the accuracy of the MPS	41
4.5	The effect of the # of interpolation points, n , and the size of the fictitious radius, r , on the accuracy of the MPS.....	42
5.1	Test Example 1, exact solution at $t = 0.5$	48
5.2	Test Example 1, Laplace transform, absolute error at $t = 0.5$	49
5.3	Test Example 2, exact solution at $t = 2$	50
5.4	Test Example 2, Laplace transform, absolute error at $t = 2$	51
5.5	Test Example 1, exact solution at $t = 1$	52
5.6	Test Example 1, absolute error at $t = 1.5$ for $n_s = 20$ and $n_s = 22$	52
5.7	Test Example 1, absolute maximum error of $n_s = 20$ vs. $n_s = 22$ for $0 < t \leq 10$	54
5.8	Test Example 1, Laplace transform, exact solution and approximate solution of $(0.5, 0.5)$ for $0 < t \leq 10$	55
5.9	Test Example 1, Laplace transform, absolute error of approximation of $(0.5, 0.5)$ for $0 < t \leq 10$	55
5.10	Test Example 2, absolute maximum error of $n_s = 16, 18$, and 20 for $0 < t \leq 20$	57
5.11	Test Example 2, Laplace transform, exact solution and approximate solution of $(0.5, 0.5)$ for $0 < t \leq 20$	58
5.12	Test Example 2, Laplace transform, absolute error of approximation of $(0.5, 0.5)$ for $0 < t \leq 20$	55

5.13	Test Example 1, exact solution at $t = 1$	63
5.14	Test Example 1, time-stepping scheme ($\Delta t = 0.025$), absolute error at $t = 1$	63
5.15	Test Example 2, exact solution at $t = 3$	64
5.16	Test Example 2, time-stepping scheme ($\Delta t = 0.1$), absolute error at $t = 3$	65
5.17	The effect of Δt on the accuracy of the time-stepping scheme, Test Example 1, absolute maximum error for $0 < t \leq 10$	68
5.18	The effect of Δt on the accuracy of the time-stepping scheme, Test Example 2, absolute maximum error for $0 < t \leq 20$	70
5.19	Test Example 1, time-stepping scheme ($\Delta t = 0.01$), exact solution and approximate solution of $(0.5, 0.5)$ for $0 < t \leq 10$	71
5.20	Test Example 1, time-stepping scheme ($\Delta t = 0.01$), error of approximation of $(0.5, 0.5)$ for $0 < t \leq 10$	72
5.21	Test Example 2, time-stepping scheme ($\Delta t = 0.1$), exact solution and approximate solution of $(0.5, 0.5)$ for $0 < t \leq 20$	73
5.22	Test Example 2, time-stepping scheme ($\Delta t = 0.1$), error of approximation of $(0.5, 0.5)$ for $0 < t \leq 20$	74
5.23	Laplace transform vs. time-stepping scheme, Test Example 1, error of approximation of $(0.5, 0.5)$ for $0 < t \leq 10$	75
5.24	Laplace transform vs. time-stepping scheme, Test Example 2, error of approximation of $(0.5, 0.5)$ for $0 < t \leq 20$	76

LIST OF TABLES

2.1	Commonly used radial basis functions	7
2.2	Wendland's CS-RBFs.....	11
3.1	Fundamental solutions to commonly used differential operators.....	21
4.1	Particular solutions Φ to various CS-RBFs φ in 2D	32
4.2	Particular solutions Φ for polynomial terms φ	36
4.3	The effect of λ^2 on the accuracy of the MPS.....	38
4.4	The effect of the # of source points, m , on the accuracy of the MPS	40
4.5	The effect of the # of interpolation points, n , and the size of the fictitious radius, r , on the accuracy of the MPS.....	42
5.1	Test Example 1, absolute maximum error of $n_s = 20$ vs. $n_s = 22$ for $0 < t \leq 10$	53
5.2	Test Example 2, absolute maximum error of $n_s = 16, 18$, and 20 for $0 < t \leq 20$	56
5.3	The effect of Δt on the accuracy of the time-stepping scheme, Test Example 1, absolute maximum error for $0 < t \leq 10$	67
5.4	The effect of Δt on the accuracy of the time-stepping scheme, Test Example 2, absolute maximum error for $0 < t \leq 20$	69

ACKNOWLEDGEMENTS

My sincerest gratitude is extended to my thesis advisor, Dr. C. S. Chen. His intense dedication and guidance toward my growth as a graduate student and mathematician has been a valuable experience for me. I would not have finished this thesis in a timely manner without his persistence and concern in the progress of the procedural testing and technical writing herein. Dr. Chen's genuine commitment and passion toward the development of meshless methods and computational mathematics is admirable. I look forward to collaborating with him in prospective projects regarding meshless methods.

Secondly, I would like to thank the UNLV Mathematical Sciences Department, the UNLV Graduate College, and my thesis committee members Dr. Zhonghai Ding, Dr. Xin Li, and Dr. Tao Pang. Special thanks go to my classmates Sachin Deshmukh and Arthur Lee for their assistance in decoding and debugging programs throughout the year.

I wish to also show my appreciation toward those instructors whose enthusiasm and love for mathematics education have inspired me throughout the years: Ms. Julie Pascua, Mrs. Jane Hunt, Mr. David Hu, Dr. Peter Patacsil, and Dr. Alex Stanoyevitch.

Most importantly, thank you to my loving family – my parents, Peter and Lourdes Rivera, my sister Olivia, and my brother Peter James – for their endless support and faith in me as an aspiring, Chamorro woman. Thank you also to Paul Gutierrez for lending a listening ear and words of encouragement when setbacks and frustration dominated the thesis-writing process. Lastly, thank you to *Baby* for behaving during all of those long, sleepless nights.

CHAPTER 1

INTRODUCTION

1.1 Meshless Methods

Over the past four decades, the numerical solutions of partial differential equations (PDEs) in science and engineering problems have been dominated by the finite element methods (FEM), the finite difference methods (FDM), and boundary element method (BEM). In spite of their great success, these conventional numerical methods still have some drawbacks that impair their computational efficiency and even limit their applicability to more practical problems. In general, the FEM and FDM require a domain mesh generation to solve such problems. Consequently, numerical results become strongly dependent on mesh properties and low-order piecewise polynomial approximations, which impair computational accuracy and the convergence rates of the methods. Regarding the BEM, even though it only requires mesh generation on the boundary of the domain, it involves quite sophisticated mathematics and some difficult numerical integration of singular functions. Moreover, domain and boundary mesh generation impedes the computational process (making it more labor intensive and time-consuming) and poses an obstacle to solving more difficult, irregularly shape, and high-dimensional problems. Hence, meshless methods provide an attractive alternative for solving such problems.

In recent years, meshless methods have been under intensive research to overcome the difficulty of meshing the solution domain. Meshless methods require the approximation of given differential equations from a set of unstructured nodes; i.e., without any pre-defined connectivity or relationship among the nodes. Instead of generating mesh, meshless methods use scattered nodes, which can be randomly distributed, throughout the computational domain. Figures 1.1 and 1.2 show meshed and meshless domains in 2D and 3D, respectively.

Figure 1.1: Meshed domains in 2D and 3D

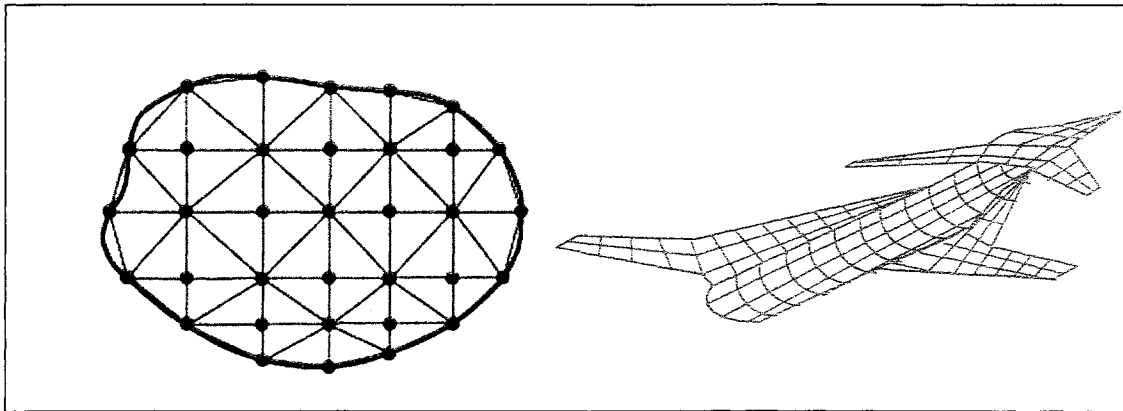
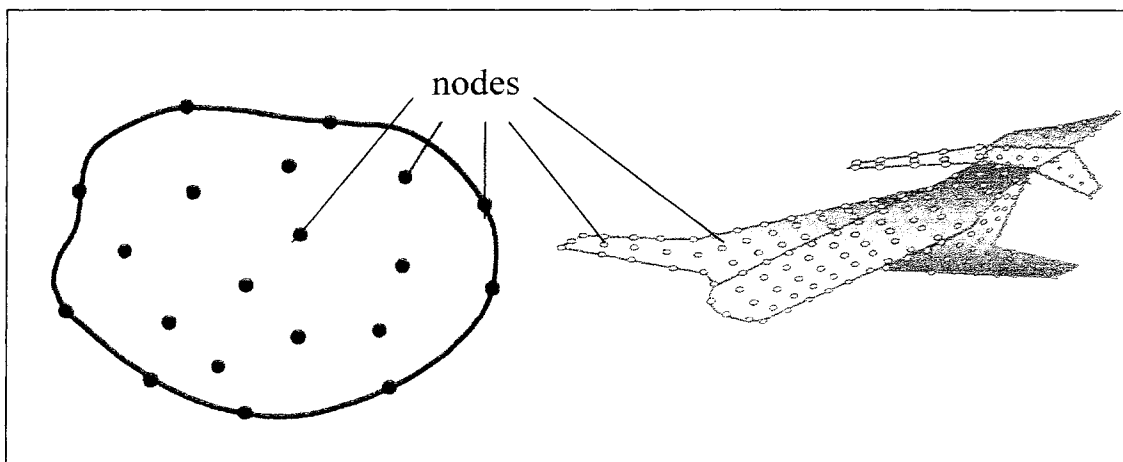


Figure 1.2: Meshless domains (using scattered nodes) in 2D and 3D



Meshless methods require neither surface nor domain discretization. It does not involve numerical integration, and the formulation is similar for problems in both 2D and 3D cases. Numerically, it is easy to code and easy to learn. Because there is no meshing required, it proves to be computationally cost effective [3].

1.2 Thesis Overview

This thesis focuses on the numerical solution of wave equations in 2D through the sole use of meshless methods. All numerical results and figures are obtained through the mathematical computer software MATLAB. Meanwhile, the mathematical symbols and fonts seen within the text are possible through the software MathType. Furthermore, all examples within this thesis are limited to second order linear differential operator problems in 2D. While this thesis focuses on solving equations in 2D, the process for solving problems in 3D is straightforward and can be solved as simple extensions to the techniques discussed herein.

Chapter 2 serves as an introduction to radial basis functions (RBFs) [16], a meshless technique used to approximate multivariate functions or surfaces. The methodology of RBF interpolation and approximation is discussed as well as considerations to ensure solution stability. One important way to ensure such stability is to appropriately select among the globally supported RBFs or compactly supported RBFs. An example in surface reconstruction is also presented to illustrate the usefulness and efficiency of RBF approximation. Additionally, the fundamental principles brought out in Chapter 2 are used in Chapters 4 and 5. In these chapters, RBFs are functional in the calculation of particular solutions to solve nonhomogeneous PDEs as well as time-dependent problems.

In Chapter 3, the method of fundamental solutions (MFS) [11, 15] – the primary meshless method for solving wave equations in this thesis – is explained. Two examples (one in a simply-connected domain and one in a multiply-connected domain) of the numerical implementation of the MFS are offered. The dominant factors that affect the accuracy of the MFS are also discussed. Moreover, the techniques of the MFS are used in Chapters 4 and 5.

In Chapter 4, the method of particular solutions (MPS) [1] – a method that solves nonhomogeneous PDEs – is described. Nonhomogeneous PDEs are split into two portions. One portion of this process deals with a homogeneous equation in which the MFS is employed to arrive at an approximate solution. The other portion deals with the calculation of a particular solution through RBF techniques. Numerical examples are further given as evidence to demonstrate the effectiveness of MPS. Lastly, an analysis of the accuracy of the MPS is surveyed.

In Chapter 5, we focus on solving wave equations by using the MFS in conjunction with the techniques discussed in previous chapters: RBFs and the MPS. Currently, there are two major approaches to solve the wave equation: (i) elimination of the time dependence by using the Laplace transform and (ii) using the method of discretization in time to approximate the time derivative. In this thesis, we propose to reduce the given wave equations to a series of inhomogeneous modified Helmholtz equations. The solution of these problems can then be further split into evaluating particular solutions and solving the related homogeneous equations (through the MPS). The MFS is adopted as the major numerical method to evaluate the homogeneous solution of the modified Helmholtz equation. Furthermore, the proposed method will require a closed form

particular solution for the modified Helmholtz equation. The difficulty of the first approach is that the inverse Laplace transform is ill-posed. For the second approach, high accuracy is difficult to achieve due to the low order time-stepping algorithm. Intensive numerical tests are performed to compare the advantages and disadvantages of these two approaches.

The final chapter, Chapter 6, recognizes the conclusions and contributions of this thesis. Potential research topics are suggested, and related topics that are under current investigation are noted.

CHAPTER 2

RADIAL BASIS FUNCTIONS

Radial basis functions (RBFs) are primarily used to reconstruct unknown, multivariate functions from known data. Such functions may be solutions to partial differential equations and subjected to additional conditions, defined on irregularly shaped domains rather than the straightforward square or rectangle, or rely on scattered data points as opposed to uniformly distributed grid points. While the more classical methods – finite element method (FEM), finite difference method (FDM), and boundary element method (BEM) – would require the tedious task of generating mesh on the domain and boundary, RBFs do not. Instead, RBFs reduce multivariate functions to scalar functions, which allow for a method that is efficient, accurate, stable, easy to implement, and truly meshless.

In recent decades, employing RBFs in the fields of numerical mathematics and scientific computing has been on the rise. Introduced by R.L. Hardy [19] in 1968, RBFs were first used for geophysical surface-fitting. By 1982, a mathematical appreciation toward RBFs resulted from the publication of a survey paper by R. Frank [14], who assessed 29 approximation methods concerning RBFs. Today, the implementation of RBF approximation is widespread: computer graphics, surface reconstruction, neural networks, picture processing and scratch removal, medical applications, science and engineering problems, etc.

2.1 Definition

A *radial basis function* $\varphi: \mathbb{R}^+ \rightarrow \mathbb{R}$ is a univariate real-valued function that is used via

$$\Phi(x, y) = \varphi(r) \text{ for all } x, y \in \mathbb{R}^d \quad (2.1)$$

as a symmetric multivariate function $\Phi: \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$. Φ is called the associated *kernel*, and r represents the *radius* (or Euclidean norm) among data points:

$$r = \|x - y\|. \quad (2.2)$$

For convenience, let us denote

$$\varphi_i(r) = \varphi(\|x - x_i\|). \quad (2.3)$$

A list of globally supported radial basis functions can be found in Table 2.1.

Table 2.1: Commonly used radial basis functions

Name	$\varphi(r)$	CPD order
Thin Plate Spline (TPS)	$r^2 \log r$	2
Cubic Spline	r^3	2
Multiquadric (MQ)	$\sqrt{r^2 + c^2}$	1
Inverse Multiquadric	$1/\sqrt{r^2 + c^2}$	0
Gaussian	e^{-cr^2}	0

In the table above, CPD is “conditionally positive definite” and c (the *shape parameter*) represents some constant that can be chosen to increase accuracy.

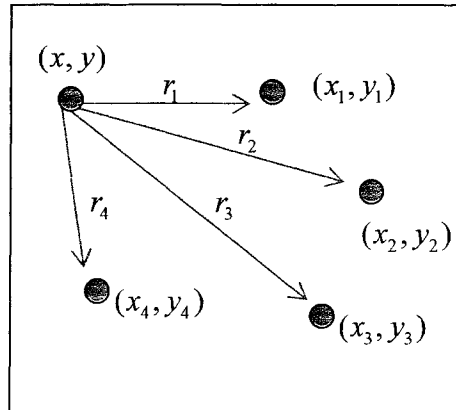
2.2 Interpolation and Approximation

Given a 2D finite set of scattered n data points (x_i, y_i) , often known as *centers* (or *interpolation points*), it is assumed that some function values $f(x_i, y_i)$ are known. Based on this known data, the task is to approximate a function that will fit the function values. Using RBFs, one can find a linear combination that closely approximates the function f :

$$f(x, y) \approx \hat{f}(x, y) = \sum_{i=1}^n \alpha_i \Phi(x, y) = \sum_{i=1}^n \alpha_i \phi_i(r), \quad (2.4)$$

where $\{\alpha_i\}$ are unknown coefficients that are to be determined. Simply put, there are measurable distances among the n centers (see Figure 2.1).

Figure 2.1: 2D representation of distances among n centers ($n = 5$)



These distances are then applied to a selected radial basis function (as indicated in Table 2.1) and written as n linear combination equations. The resulting system is

$$A\bar{\alpha} = \bar{f}, \quad (2.5)$$

where A is the $n \times n$ symmetric coefficient matrix of the linear equations, $\bar{\alpha}$ is the vector of corresponding unknown coefficients, and $\bar{f}$ is a vector of the associated function

values. Provided matrix A is nonsingular, the unknown coefficients $\{\alpha_i\}$ are uniquely solvable:

$$\bar{\alpha} = A^{-1} \bar{f} \quad (2.6)$$

2.2.1 Positive Definite vs. Conditionally Positive Definite

The nonsingular condition of the interpolation matrix A is a critical issue when attempting to solve for the unknown coefficients $\{\alpha_i\}$. If the matrix is positive definite, then it is nonsingular and invertible. Selecting RBFs denoted with a CPD order of zero (refer to Table 2.1) will ensure that the matrix is positive definite. Hence, RBFs with CPD order of zero are called *symmetric positive definite* (SPD).

For RBFs with a CPD order of m (e.g., the TPS or cubic spline), the nonsingular condition of matrix A is not met. Fortunately, this can be overcome by the addition of polynomial terms. In this case, such RBFs are called *conditionally positive definite* (CPD) with order m . The approximation of a function f then becomes

$$\hat{f}(x, y) = \sum_{i=1}^n \alpha_i \varphi_i(r) + \sum_{j=1}^M \alpha_j P_j(x), \quad (2.7)$$

along with the constraints

$$\sum_{i=1}^n \alpha_i P_j(x_i) = 0, \quad 1 \leq j \leq M, \quad (2.8)$$

where

$$M = \binom{m+d-1}{d}, \quad (2.9)$$

and d represents a d -variate polynomial, P . For example, if $\varphi(r) = r^2 \log r$ were chosen as the RBF, then $M = 3$, and the function f would be approximated by $n + 3$ equations:

$$\hat{f}(x, y) = \sum_{i=1}^n \alpha_i r^2 \log r + \alpha_{n+1} + \alpha_{n+2}x + \alpha_{n+3}y, \quad (2.10)$$

with the constraints

$$\sum_{i=1}^n \alpha_i = \sum_{i=1}^n \alpha_i x_i = \sum_{i=1}^n \alpha_i y_i = 0. \quad (2.11)$$

2.2.2 Eigenvalues and Condition Number

The eigenvalues and condition number of matrix A are also significant issues in RBF approximation. If A is an $n \times n$ matrix, then any scalar λ satisfying the equation

$$A\mathbf{x} = \lambda\mathbf{x}, \quad (2.12)$$

for some $m \times 1$ vector $\mathbf{x} \neq \mathbf{0}$, is called an *eigenvalue* of A . The matrix A can have many eigenvalues. Of particular interest are the maximum and minimum eigenvalues. These eigenvalues can tell us about the condition number of matrix A . The *condition number* of A is given by

$$\text{cond}(A) = \frac{\lambda_{\max}}{\lambda_{\min}}, \quad (2.13)$$

where $\lambda_{\max}$ and $\lambda_{\min}$ are the maximum and minimum eigenvalues of A .

The condition number of a matrix indicates the sensitivity of the solution. Very large condition numbers indicate an ill-conditioned problem and unreliable solutions, while small condition numbers point toward solution stability. The accuracy of the RBF is inversely related to the condition number of the interpolation matrix [28].

2.2.3 Compactly Supported Radial Basis Functions

The RBFs listed in Table 2.1 are globally supported, which can sometimes lead to dense interpolation matrices that are extremely ill-conditioned. This is especially the case when dealing with large-scale problems or a vast number (perhaps thousands) of

interpolation points. To overcome the dilemma of ill-conditioning, the employment of Wendland's [31] compactly supported RBFs (CS-RBFs) is needed. CS-RBFs, which are positive definite functions, allow for the interpolation matrix to be sparse and for the problem to be easily solved. CS-RBFs are piecewise polynomials defined as

$$(1-r)_+^n = \begin{cases} (1-r)^n, & \text{if } 0 \leq r \leq 1 \\ 0, & \text{if } r > 1 \end{cases} \quad (2.14)$$

where, $n \in \mathbb{N}$.

Table 2.2: Wendland's CS-RBFs

d	$\varphi(r)$	C^{2k}
1	$(1-r)_+$	C^0
	$(1-r)_+^3 (3r+1)$	C^2
	$(1-r)_+^5 (8r^2 + 5r + 1)$	C^4
2, 3	$(1-r)_+^2$	C^0
	$(1-r)_+^4 (4r+1)$	C^2
	$(1-r)_+^6 (35r^2 + 18r + 3)$	C^4
	$(1-r)_+^8 (32r^3 + 25r^2 + 8r + 1)$	C^6

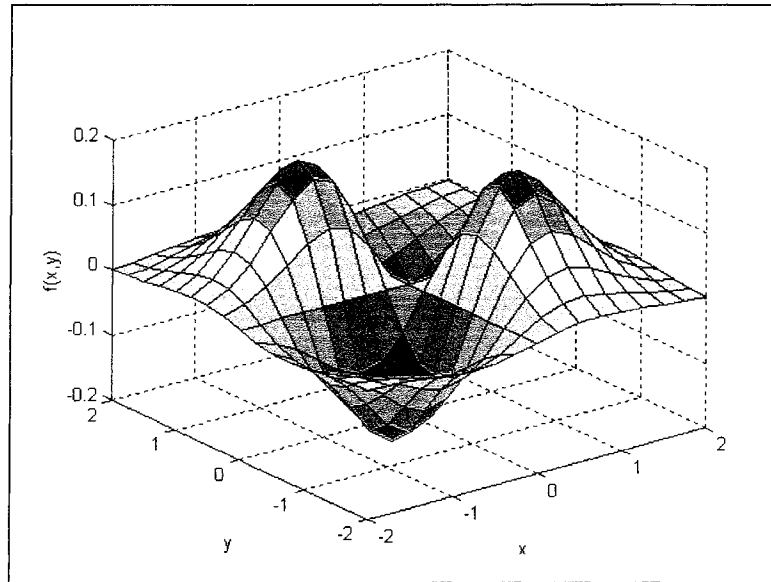
For any given dimension d and smoothness C^{2k} , $k \in \mathbb{N}$, there exists a positive definite polynomial $\varphi(r)$ with minimum degree that can be constructed. While CS-RBFs

reduce the ill-conditioning of an interpolation matrix, the accuracy relies significantly on the scale of the support. In Table 2.2, the supports of all of the functions have been normalized to $[0, 1]$. However, these functions can be rescaled with the support of radius α by using $\varphi\left(\frac{r}{\alpha}\right)$ for $\alpha > 0$. When dealing with a large amount of interpolation points that are compacted within the domain, α should be chosen to be small. For small enough α , the interpolation matrix will be sparse. When dealing with a small amount of interpolation points, a large α should be chosen. Consequently, a large α causes the CS-RBF to become globally supported.

2.3 Surface Reconstruction

Using the approximation method described above, let us consider the reconstruction of the following 2D surface: $f(x, y) = -xye^{-x^2-y^2}$ (Figure 2.2).

Figure 2.2: Graph of $f(x, y) = -xye^{-x^2-y^2}$



To approximate this function, we generate 144 known data points (or *interpolation points*) and select $\varphi(r) = r^3$. Since the cubic spline has a CPD order of 2, additional polynomial terms will be added to the approximation to ensure the solvability of the unknown coefficients $\{\alpha_i\}$. Hence, there will be 147 equations (or 147 unknown coefficients) to solve for the approximation:

$$\hat{f}(x, y) = \sum_{i=1}^{144} \alpha_i r^3 + \alpha_{145} + \alpha_{146}x + \alpha_{147}y \quad (2.15)$$

Let's consider two cases: (1) using 144 uniformly distributed x - y pairs, and (2) using 144 randomly distributed x - y pairs. Once the coefficients are solved, a new set of x - y pairs are generated and "fitted" into equation (2.15). Figures 2.3 and 2.4 are scatter plots of the uniformly distributed x - y pairs and the associated known data points, respectively. The approximation is then tested on 289 uniformly distributed points, and the resulting absolute error between the test points and the true graph of $f(x, y) = -xye^{-x^2-y^2}$ are shown in Figure 2.5.

Figure 2.3: Uniform distribution of 144 x - y pairs

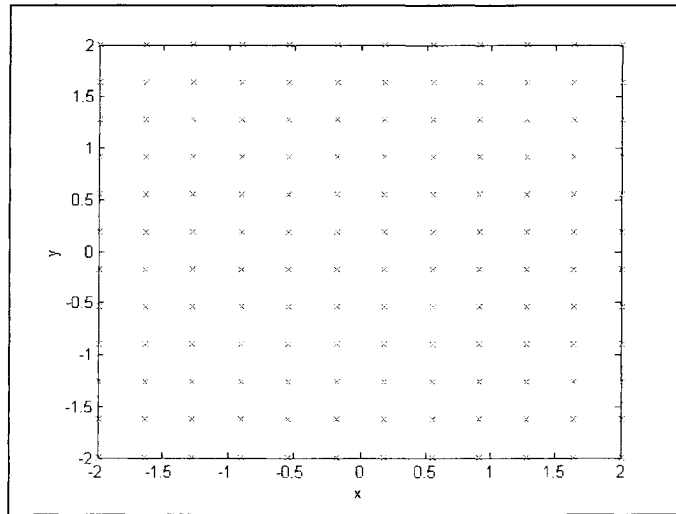


Figure 2.4: 144 known data points (uniform distribution)

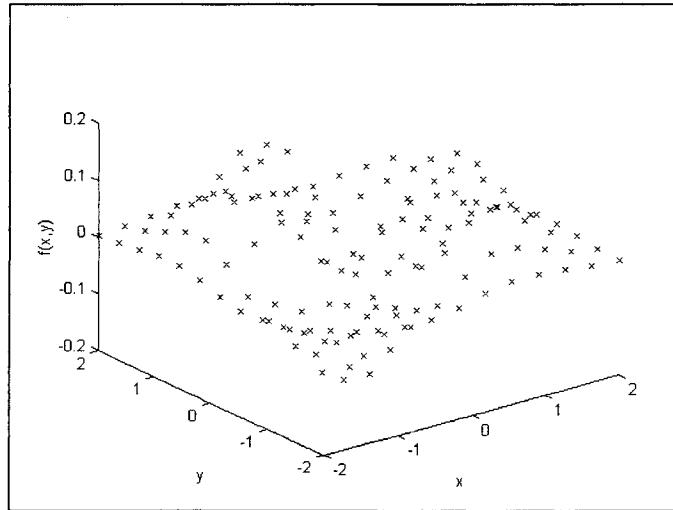
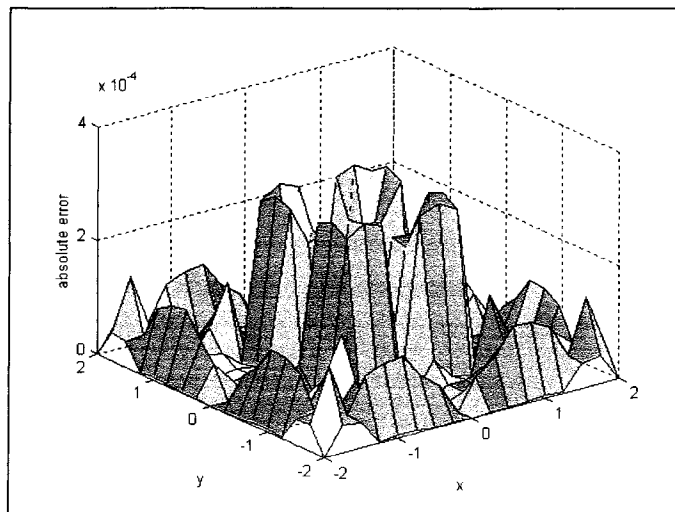


Figure 2.5: Absolute error of approximation



Using 144 uniformly distributed x - y pairs and the cubic spline as the RBF, the error between approximation and the true graph is as small as 10^{-4} . Meanwhile, using 144 randomly distributed x - y pairs (see below), the accuracy of approximation reaches 10^{-3} . Figures 2.6 to 2.7 are plots of the randomly distributed x - y pairs and the associated

known data points, respectively. Figure 2.8 is the absolute error of approximation between the 289 uniformly distributed test points and the true graph.

Figure 2.6: Random distribution of 144 x - y pairs

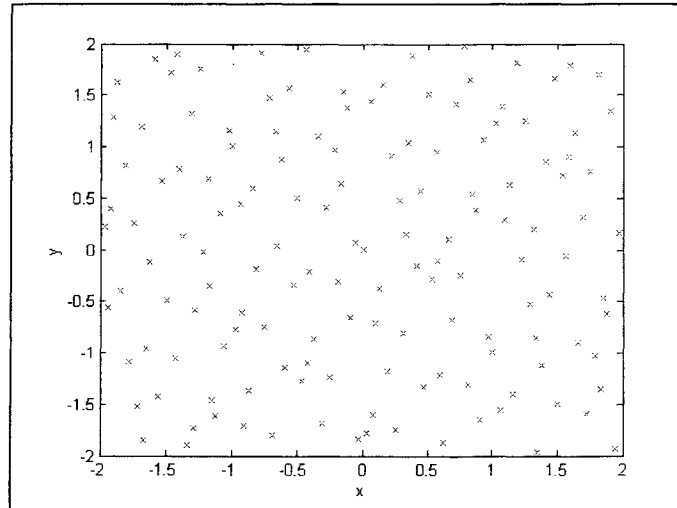


Figure 2.7: 144 known data points (random distribution)

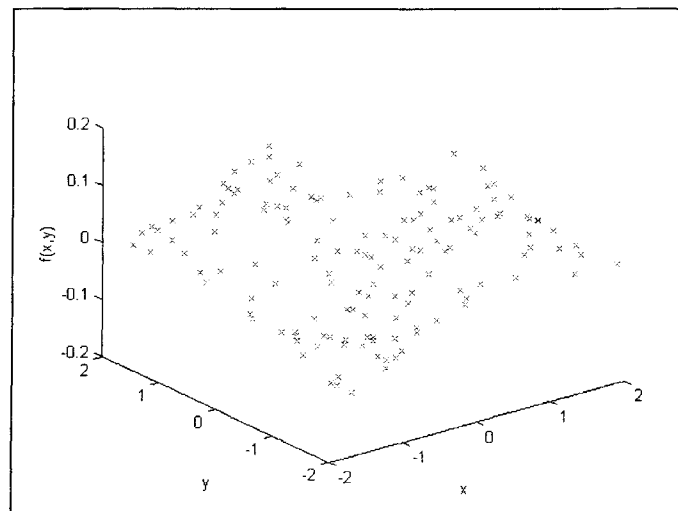
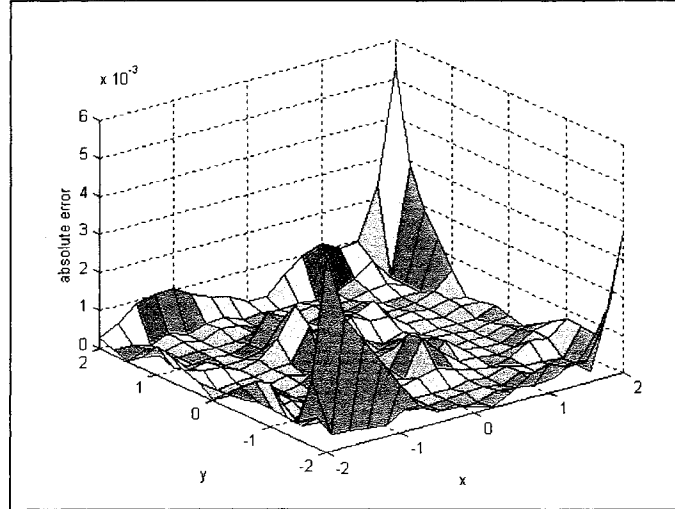


Figure 2.8: Absolute error of approximation



Whether uniformly distributed or randomly distributed data points are used, the approximations of $f(x,y) = -xye^{-x^2-y^2}$ are still fairly accurate, with accuracy reaching 10^{-3} or 10^{-4} . Of course, it is preferable to use uniformly distributed points when reconstructing surfaces since there is a slight increase in accuracy. Furthermore, it is important to note that accuracy will improve with an increase in interpolation points. Increasing the amount of interpolation points to 300 or 400 points results in an accuracy of 10^{-5} .

2.4 Concluding Remarks

It is easy to see that RBF approximation does not require a uniform grid system. The key feature to RBFs is the ability to approximate scattered, known data points on a meshless domain. The only information being manipulated is the distance among centers. Distances (merely scalars) are effortless to compute. For multidimensional problems, the degree of difficulty to compute scalars does not change. This component alone gives RBF

approximation an immense computational advantage over the classical methods.

Solution stability and computational efficiency should always be considered when implementing RBFs in the reconstruction of surfaces. RBFs can be adequately chosen to ensure a nonsingular interpolation matrix. The addition of polynomial terms for CPD RBFs, proper selection of the amount of interpolation points, or appropriate choice of compact support for CS-RBFs can ensure this condition. Furthermore, analysis of eigenvalues or the condition number of the interpolation matrix is beneficial.

RBF approximation is the technique that is most applicable to this thesis. In subsequent chapters, RBFs are used to solve nonhomogeneous partial differential equations (PDEs) as well as time-dependent problems. In Chapter 4, RBFs are used primarily to approximate inhomogeneous terms and particular solutions. In Chapter 5, the implementation of RBFs is an essential component for solving the wave equation.

CHAPTER 3

THE METHOD OF FUNDAMENTAL SOLUTIONS

Many boundary value problems in science or engineering involve domains that are irregular in shape. Such problems can be extremely difficult (and relatively expensive) to solve with domain meshing methods like the FEM or FDM. Instead, it is preferable to use the less costly BEM, since it only requires boundary meshing. However, this method also has its disadvantages: (1) it may involve the evaluation of singular integrals; (2) it becomes complicated when meshing surfaces in 3D; and (3) it has a slow rate of convergence due to the use of low order polynomial approximations.

Improving upon the discretization of the domain or boundary, meshless methods have evolved as the most advantageous means to solve boundary value problems (of any shape). As an indirect extension of the BEM, the *method of fundamental solutions* (MFS) has emerged as a powerful meshless method for solving boundary value problems. Initially proposed by Kupradze and Aleksidze [23, 24] in 1964, the MFS has been also known as the *superposition method*, *desingularized method*, and the *charge simulation method*. Extensively studied by Cheng [9], Katsurada and Okamoto [21], and Kitagawa [22] in the late 1980's, Fairweather and A. Karageorghis [13], Goldberg and Chen [15] in the 1990's, and Cho et al [11] in the 2000's, the MFS has been established as a useful method in solving homogeneous, elliptic PDEs.

3.1 Methodology and Fundamental Solutions

The MFS approximates the solution of a PDE by way of linear combinations of the fundamental solution of the governing differential operator. A generalized homogeneous PDE problem with mixed boundary conditions in 2D is

$$Lu(x, y) = 0, \quad x, y \in \Omega, \quad (3.1)$$

$$u(x, y) = g_1(x, y), \quad x, y \in \Gamma_1, \quad (3.2)$$

$$\frac{\partial}{\partial n} u(x, y) = g_2(x, y), \quad x, y \in \Gamma_2. \quad (3.3)$$

L is a second order linear differential operator with a known fundamental solution. Ω is a bounded, nonempty, connected domain with boundary $\partial\Omega$, where $\partial\Omega = \Gamma_1 \cup \Gamma_2$ and $\Gamma_1 \cap \Gamma_2 = \emptyset$. g_1 and g_2 are known functions and $\partial u / \partial n$ is the outward normal derivative of u to the boundary. A PDE problem having (3.2) as the only boundary condition on $\partial\Omega$ has *Dirichlet* boundary condition. The presence of $\partial u / \partial n$ as the only boundary condition is called the *Neumann* boundary condition. The *Robin* boundary condition has both (3.2) and (3.3).

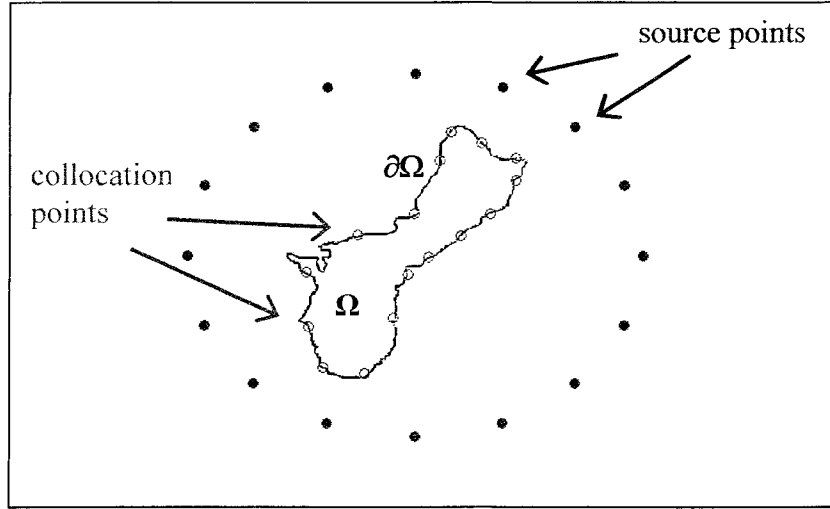
An approximate solution to the above PDE problem with Dirichlet boundary condition is represented by

$$\hat{u}(x, y) = \sum_{j=1}^m \beta_j G(r), \quad (3.4)$$

where $\{\beta_j\}$ are unknown coefficients, $G(r)$ is the associated *fundamental solution* of the governing differential operator, and r is the Euclidean norm as defined in (2.2). According to the MFS, m *collocation points* on the boundary $\partial\Omega$ and m *source points* on a fictitious boundary outside of the physical domain (see Figure 3.1) are used to

approximate (3.4). The rationale behind using a fictitious boundary is to ensure that there will be no singularities in the linear combinations of the selected fundamental solution. The Euclidean distance between a point and itself on the boundary $\partial\Omega$ would yield $r = 0$; this may pose a problem for calculations involving the fundamental solution.

Figure 3.1: Distribution of collocation and source points for the MFS



To approximate the solution to the PDE problem with mixed boundary conditions, $\{\beta_j\}$ must satisfy the following linear equations [15]:

$$\hat{g}_1 = \sum_{j=1}^{m^*} \beta_j G(r), \quad (3.5)$$

$$\hat{g}_2 = \sum_{j=m^*+1}^m \beta_j \frac{\partial}{\partial n} G(r), \quad (3.6)$$

where m^* is the number of collocation points on Γ_1 and m is the total number of source points. In the MFS, a fundamental solution $G(r)$ is required. Table 3.1 lists the fundamental solutions to the differential operators that are used in this thesis.

Table 3.1: Fundamental solutions to commonly used differential operators

Name of Operator	L	$G(r)$ in 2D	$G(r)$ in 3D
Laplacian	Δ	$-\frac{1}{2\pi} \log r$	$\frac{1}{4\pi r}$
Helmholtz	$\Delta + \lambda^2$	$-\frac{i}{4} H_0^{(2)}(\lambda r)$	$\frac{1}{4\pi r} e^{-i\lambda r}$
Modified Helmholtz	$\Delta - \lambda^2$	$\frac{1}{2\pi} K_0(\lambda r)$	$\frac{1}{4\pi r} e^{-\lambda r}$

In Table 3.1, $H_0^{(2)}$ denotes the Hankel function of the second kind of order zero, K_0 is the modified Bessel function of the second kind of order zero, and $i = \sqrt{-1}$.

3.2 Numerical Implementation

The numerical implementation of the MFS is similar for both simply-connected and multiply-connected domains. Let us consider the following PDE problem with Dirichlet boundary condition:

$$\Delta u = 0, \quad x, y \in \Omega, \quad (3.7)$$

$$u = \sin x \cosh y, \quad x, y \in \partial\Omega, \quad (3.8)$$

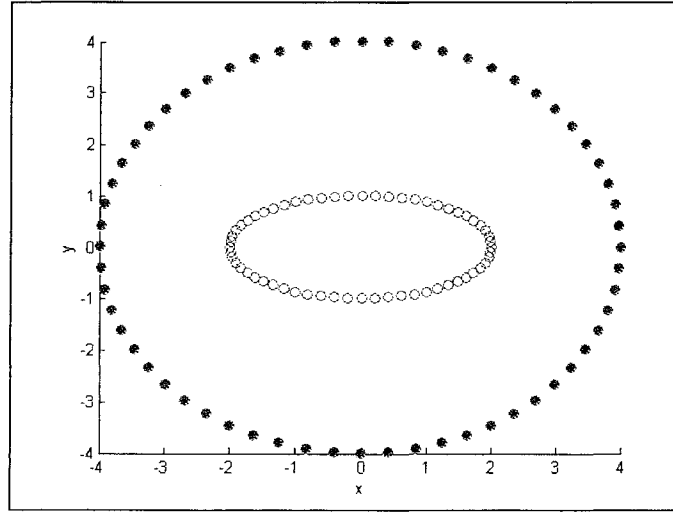
defined on a simply-connected domain (an ellipse):

$$\Omega \cup \partial\Omega = \left\{ (x, y) : \frac{x^2}{4} + y^2 \leq 1 \right\}. \quad (3.9)$$

For this problem, we use 60 uniformly distributed collocation points and 60 uniformly distributed source points on a fictitious circle with radius $r = 4$ (see Figure 3.2). The

approximation is then tested on 121 uniformly distributed collocation points, and the *absolute maximum error*, which refers to the greatest error among these test points and the exact solution u , is 6.8834×10^{-15} .

Figure 3.2: Distribution of collocation and source points
on a simply-connected domain



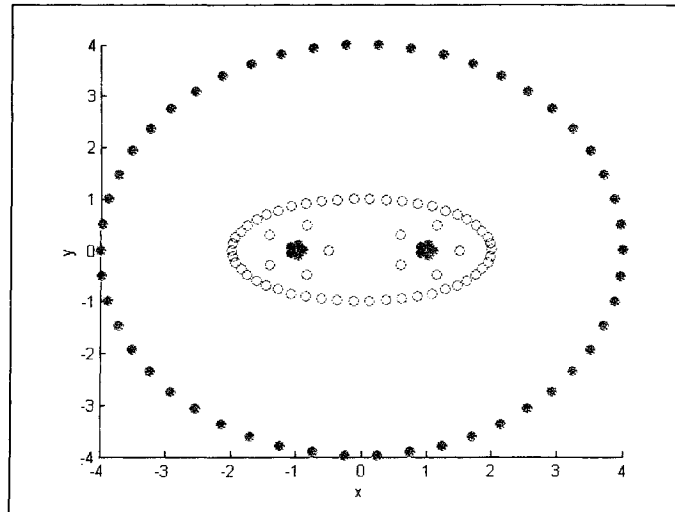
Let us also consider the same PDE problem defined on a multiply-connected domain (an ellipse with two holes in it):

$$\Omega \cup \partial\Omega = \left\{ (x, y) : \frac{x^2}{4} + y^2 \leq 1, (x-1)^2 + y^2 \geq \frac{1}{4}, (x+1)^2 + y^2 \geq \frac{1}{4} \right\}. \quad (3.10)$$

Using 60 uniformly distributed collocation points (50 on the ellipse and 5 each for the inner circles, 60 uniformly distributed source points (50 on the fictitious circle and 5 each inside the inner circles), and a fictitious circle, centered at $(0, 0)$, with radius $r = 4$ (see Figure 3.3), the absolute maximum error between the approximation $\hat{u}$ (utilizing the 121 test points) and the exact solution u is 7.5495×10^{-15} .

For this particular problem (whether the domain is simply-connected or multiply-connected), an increase in the radius of the fictitious circle results in a slight decrease in approximation accuracy. For radius $r = 20$, accuracy is 10^{-6} . Furthermore, a reduction in the total number of collocation/source points produces a slight decrease in accuracy. Using 20 collocation/source points, the accuracy is 10^{-6} . Accuracy is similar regardless of the placement of the collocation points in the multiply-connected domain (i.e., the amount of collocation points distributed among the inner circles and outer ellipse). An interesting point for the multiply-connected case is that the best accuracy was achieved with at least five collocation points for each inner circle.

Figure 3.3: Distribution of collocation and source points
on a multiply-connected domain



3.3 Analysis of Accuracy

The degree of accuracy of the MFS is contingent upon several factors: (1) the shape of the fictitious boundary; (2) the size of the radius for the fictitious boundary (using a

circle); (3) the quantity of collocation and source points (which should always be the same quantity); and (4) the placement of the collocation points and source points (uniformly distributed or randomly distributed).

For the MFS, the shape of the fictitious boundary can mimic the shape of the original boundary. However, for irregularly shaped boundaries, it may be difficult to construct a fictitious boundary with the same shape. For simplicity in computer coding, it is easiest to use a circle as the fictitious boundary. Despite their shapes, fictitious boundaries serve the same purpose: to cancel out singularities that may occur on the original boundaries. Therefore, it is best to use a fictitious circle in 2D and a fictitious sphere in 3D [15].

When the domain and the fictitious boundary are both circles, research by Bogomolny [2] and Cheng [9] has proven that the accuracy of the MFS improves as the fictitious radius enlarges. Work by Katsurada [20], involving arbitrary analytic curves in 2D, has confirmed this result. However, when the domain is not circular, this result is not necessarily applicable. Note that in the two examples from the previous section, an increase in the size of the fictitious radius resulted in a decrease in accuracy. For many boundary value problems, the condition number of the collocation matrix increases exponentially and the MFS equations can become highly ill-conditioned as the fictitious radius increases [15, 22]. When testing for MFS accuracy, it appears that there will be a critical fictitious radius that will minimize absolute maximum error. At present, however, there is no specific way to predict the value of radius r that will minimize such error [10].

As a rule of thumb, the amount of collocation points and source points must be the same. Generally, the more collocation/source points used in the MFS equations, the more accurate the approximation. Accuracy will increase with an increase in collocation/source

points, but there is a critical amount of points after which accuracy will no longer improve. Regarding the previous examples, accuracy was relatively high for as little as 60 collocation/source points.

The placement of collocation and source points is another significant factor in the accuracy of the MFS. So far, there is no theoretical result regarding the optimal selection of these points. In general, we choose uniformly distributed source and collocation points. Source points are normally equally spaced on the fictitious circle. Because the domain can be oddly-shaped, it may seem tricky to have equally-spaced collocation points. However, uniform distribution can be accomplished by placing collocation points along the same angles at which source points are placed.

3.4 Concluding Remarks

From a numerical standpoint, the MFS has many benefits. The MFS is truly meshless; neither domain nor boundary discretization is needed. Calculations within this method are singularity-free due to the application of a fictitious boundary. Requiring no numerical integration, the MFS produces low-cost computation and works well for irregularly shaped domains or high dimension problems. Furthermore, accuracy for this method can be optimized by taking into account the number and placement of collocation points and the size of the fictitious radius.

In the next chapter, the method of particular solutions (MPS) – a method that solves nonhomogeneous PDEs – is discussed. A major part of this process deals with a homogeneous equation in which the MFS is employed to arrive at an approximate solution.

CHAPTER 4

THE METHOD OF PARTICULAR SOLUTIONS

Introduced by Nardini and Brebbia [26] in 1982, the *dual reciprocity method* (DRM) has been particularly valuable in solving nonhomogeneous PDEs. In the past, engineering literature exclusively used the ad-hoc choice basis function $1 + r$ for approximations within the DRM. In the mid 1990's, however, Golberg and Chen [15] submitted a series of papers on RBFs to mathematically justify the use of suitable RBFs for the DRM. For example, the TPS, MQ, inverse MQ, and CS-RBFs [31, 32] are preferable basis functions for use in the DRM.

The theoretical foundation of RBFs in mathematical literature has contributed toward advances on the DRM development. Choosing an appropriate RBF can significantly improve both accuracy and efficiency for the DRM. An enhanced calculation of the particular solution – a crucial step in the DRM process – is also brought about by proper selection of an RBF. Furthermore, it is essential to note that CS-RBFs allow for large-scale PDE problems to be solved.

The main idea of the DRM for solving nonhomogeneous PDEs, $Lu = f$, is to expand the nonhomogeneous term f in terms of its values at the interpolation nodes so that a particular solution can be obtained [4]. This is done by RBF approximation. The governing equation can then be reduced to a homogeneous equation through the *method of particular solutions* (MPS).

4.1 Methodology

Stemming from the DRM, the MPS is used to solve nonhomogeneous PDEs [1]. Let us consider the following nonhomogeneous problem in 2D:

$$Lu(x, y) = f(x, y), \quad x, y \in \Omega, \quad (4.1)$$

$$u(x, y) = g_1(x, y), \quad x, y \in \Gamma_1, \quad (4.2)$$

$$\frac{\partial}{\partial n} u(x, y) = g_2(x, y), \quad x, y \in \Gamma_2, \quad (4.3)$$

where L , Ω , Γ_1 , Γ_2 are defined as in (3.1) – (3.3) of the previous chapter, and f , g_1 , and g_2 are given functions. The key idea to solving a nonhomogeneous PDE through the MPS is to split the solution u into the sum of its *homogeneous solution*, u_h , and its *particular solution*, u_p :

$$u = u_h + u_p, \quad (4.4)$$

where u_p satisfies the nonhomogeneous equation

$$Lu_p = f \quad (4.5)$$

but does not necessarily satisfy the boundary conditions (4.2) and (4.3). Rearranging the terms in (4.4), we can define u_h as

$$u_h = u - u_p. \quad (4.6)$$

By linearity,

$$Lu_h = L(u - u_p) = Lu - Lu_p = f - f = 0. \quad (4.7)$$

Thus, u_h satisfies

$$Lu_h = 0, \quad x, y \in \Omega, \quad (4.8)$$

$$u_h = g_1 - u_p, \quad x, y \in \Gamma_1, \quad (4.9)$$

$$\frac{\partial}{\partial n} u_h = g_2 - \frac{\partial}{\partial n} u_p, \quad x, y \in \Gamma_2. \quad (4.10)$$

Once u_p and $\partial u_p / \partial n$ are known, (4.8) – (4.10) can be solved through the MFS. The final solution for (4.1) – (4.3) is then obtained by (4.4).

The critical step for the MPS is how to obtain a particular solution u_p . Theoretically, particular solutions are not unique, but finding one particular solution is sufficient for numerical implementation. In general, when f is a simple function – such as a polynomial or finite trigonometric sum – u_p can be obtained analytically. For example, if $f = x$ and the given PDE was $\Delta u_p = x$, then possible particular solutions could be: $u_p = \frac{x^3}{6}$,

$$u_p = \frac{x^3}{6} + xy, \quad u_p = \frac{x^3}{6} + xy + 2, \dots, \text{etc.}$$

Realistically, the nonhomogeneous term f may not turn out to be simple. When this is the case, f can be approximated by a finite linear combination of basis functions (refer to Chapter 2):

$$f \approx \hat{f} = \sum_{i=1}^n \alpha_i \varphi_i(r), \quad (4.11)$$

where $\{\alpha_i\}$ are unknown coefficients that are to be determined and $\varphi_i(r)$ is as defined in (2.3). Once $\hat{f}$ is established, an approximate particular solution $\hat{u}_p$ to u_p is similarly obtained. According to the DRM, u_p can be evaluated by a series of approximate particular solutions.

$$\hat{u}_p = \sum_{i=1}^n \alpha_i \Phi_i(r), \quad (4.12)$$

where Φ satisfies

$$L\Phi_i(r) = \varphi_i(r). \quad (4.13)$$

The normal derivative of the particular solution is approximated by

$$\frac{\partial \hat{u}_p}{\partial n} = \sum_{i=1}^n \alpha_i \frac{\partial}{\partial n} \Phi(r). \quad (4.14)$$

4.2 Using the Thin Plate Spline

In order to approximate the particular solution, a closed-form Φ must be obtained. Recall that when implementing the TPS $\varphi(r) = r^2 \log r$ for interpolating the nonhomogeneous term in 2D, there will be additional polynomial terms in the approximation:

$$f \approx \hat{f} = \sum_{i=1}^n \alpha_i r^2 \log r + \alpha_{n+1} + \alpha_{n+2}x + \alpha_{n+3}y \quad (4.15)$$

with constraints

$$\sum_{i=1}^n \alpha_i = \sum_{i=1}^n \alpha_i x_i = \sum_{i=1}^n \alpha_i y_i = 0. \quad (4.16)$$

Likewise, when approximating the particular solution u_p , the additional polynomial terms must be considered [7]. By linearity, it suffices to solve (4.13):

$$L\Phi_i(r) = r^2 \log r, \quad 1 \leq i \leq n, \quad (4.17)$$

$$L\Psi_1 = 1, \quad (4.18)$$

$$L\Psi_2 = x, \quad (4.19)$$

$$L\Psi_3 = y. \quad (4.20)$$

Hence, the particular solution $\hat{u}_p$ is given by

$$\hat{u}_p = \sum_{i=1}^n \alpha_i \Phi + \alpha_{n+1} \Psi_1 + \alpha_{n+2} \Psi_2 + \alpha_{n+3} \Psi_3. \quad (4.21)$$

The same process can be applied to higher order splines (i.e. $r^4 \log r, r^6 \log r, r^8 \log r, \dots$), which will be addressed in Section 4.4.1.

4.3 Particular Solutions for the Laplacian Operator, $L = \Delta$

Note that the Laplacian operator $L = \Delta$ is both translationally and rotationally invariant. To obtain Φ in (4.13) for the 2D case, we can rewrite $\Delta\Phi$ in polar form:

$$\Delta\Phi = \frac{1}{r} \frac{d}{dr} \left(r \frac{d\Phi}{dr} \right). \quad (4.22)$$

Depending on the selected RBF φ , Φ can be obtained by repeated integration. Symbolic computer software (like Mathematica) can conveniently obtain Φ analytically. However, careful attention to the integration constants should be taken into consideration (since such software tends to omit these items).

Using the TPS $\varphi(r) = r^2 \log r$ as a basis function, by direct integration of (4.22),

$$\Delta\Phi = \frac{1}{r} \frac{d}{dr} \left(r \frac{d\Phi}{dr} \right) = r^2 \log r, \quad (4.23)$$

and choosing integration constants $C_1 = C_2 = 0$, we get the particular solution [15, 16]

$$\Phi(r) = \frac{r^4 \log r}{16} - \frac{r^4}{32}. \quad (4.24)$$

Using MQ $\varphi(r) = \sqrt{r^2 + c^2}$, by direct integration of (4.22) and proper selection of the integration constants, the particular solution becomes [15, 16, 17]:

$$\Phi(r) = \frac{1}{9} (4c^2 + r^2) \sqrt{r^2 + c^2} - \frac{c^3}{3} \log \left(c + \sqrt{r^2 + c^2} \right). \quad (4.25)$$

Undergoing a similar process, the inverse MQ $\varphi(r)=1/\sqrt{r^2+c^2}$ yields a particular solution of [15]

$$\Phi(r)=\sqrt{r^2+c^2}-c\log\left(c+\sqrt{r^2+c^2}\right). \quad (4.26)$$

For CS-RBFs (which is an alternate choice when dealing with a large number of interpolation points and the resulting ill-conditioned, dense interpolation matrix), the derivation of a particular solution is the same. However, it is important to take into account the compact support α . From (4.13), we get

$$\Delta\Phi\left(\frac{r}{\alpha}\right)=\varphi\left(\frac{r}{\alpha}\right). \quad (4.27)$$

By straightforward integration the particular solution is [4]:

$$\Phi\left(\frac{r}{\alpha}\right)=\begin{cases} \int_0^r \frac{1}{s} \left[\int_0^s t \varphi\left(\frac{t}{\alpha}\right) dt \right] ds, & r \leq \alpha, \\ \int_0^\alpha \frac{1}{s} \left[\int_0^s t \varphi\left(\frac{t}{\alpha}\right) dt \right] ds + \int_0^r \frac{1}{s} \left[\int_0^s t \varphi\left(\frac{t}{\alpha}\right) dt \right] ds, & r > \alpha. \end{cases} \quad (4.28)$$

Table 4.1 shows a list of particular solutions Φ corresponding to various CS-RBFs φ [4].

Table 4.1: Particular solutions Φ to various CS-RBFs φ in 2D

φ	Φ
$\left(1 - \frac{r}{\alpha}\right)_+^2$	$\begin{cases} \frac{r^4}{16\alpha^2} - \frac{2r^3}{9\alpha} + \frac{r^2}{4}, & r \leq \alpha \\ \frac{5\alpha^2}{36} + \frac{\alpha^2}{6} \log\left(\frac{r}{\alpha}\right), & r > \alpha \end{cases}$
$\left(1 - \frac{r}{\alpha}\right)_+^4 \left(4\frac{r}{\alpha} + 1\right)$	$\begin{cases} \frac{4r^7}{49\alpha^5} - \frac{5r^6}{12\alpha^4} + \frac{4r^5}{5\alpha^3} - \frac{5r^4}{8\alpha^2} + \frac{r^2}{4}, & r \leq \alpha \\ \frac{529\alpha^2}{5880} + \frac{\alpha^2}{14} \log\left(\frac{r}{\alpha}\right), & r > \alpha \end{cases}$
$\left(1 - \frac{r}{\alpha}\right)_+^6 \left(35\left(\frac{r}{\alpha}\right)^2 + 18\frac{r}{\alpha} + 3\right)$	$\begin{cases} \frac{7r^{10}}{20\alpha^8} - \frac{64r^9}{27\alpha^7} + \frac{105r^8}{16\alpha^6} - \frac{64r^7}{7\alpha^5} \\ + \frac{35r^6}{6\alpha^4} - \frac{7r^4}{4\alpha^2} + \frac{3r^2}{4}, & r \leq \alpha \\ \frac{3517\alpha^2}{15120} + \frac{\alpha^2}{6} \log\left(\frac{r}{\alpha}\right), & r > \alpha \end{cases}$
$\left(1 - \frac{r}{\alpha}\right)_+^8 \left(32\left(\frac{r}{\alpha}\right)^3 + 25\left(\frac{r}{\alpha}\right)^2 + 8\frac{r}{\alpha} + 1\right)$	$\begin{cases} \frac{32r^{13}}{169\alpha^{11}} - \frac{77r^{12}}{48\alpha^{10}} + \frac{64r^{11}}{11\alpha^9} - \frac{231r^{10}}{20\alpha^8} + \frac{352r^9}{27\alpha^7} \\ + \frac{352r^9}{27\alpha^7} - \frac{231r^8}{32\alpha^6} + \frac{11r^6}{6\alpha^4} - \frac{11r^4}{16\alpha^2} + \frac{r^2}{4}, & r \leq \alpha \\ \frac{541961\alpha^2}{8030880} + \frac{7\alpha^2}{156} \log\left(\frac{r}{\alpha}\right), & r > \alpha \end{cases}$

4.4 Particular Solutions for Helmholtz-Type Operators, $L = \Delta \pm \lambda^2$

Based on a theorem by Duchon [12], it has been suggested by Golberg and Chen [16] to use the TPS $\varphi(r) = r^2 \log r$ in the approximation of the nonhomogeneous term and particular solution for Helmholtz-type operators. Their derivation of the particular solution for the modified Helmholtz operator $L = \Delta - \lambda^2$ in 2D is [7]:

$$\Phi(r) = \begin{cases} -\frac{4}{\lambda^4} - \frac{4 \log r}{\lambda^4} - \frac{r^2 \log r}{\lambda^2} - \frac{4K_0(\lambda r)}{\lambda^4}, & r \neq 0, \\ -\frac{4}{\lambda^4} + \frac{4\gamma}{\lambda^4} + \frac{4}{\lambda^4} \log\left(\frac{\lambda}{2}\right), & r = 0. \end{cases} \quad (4.29)$$

Here, K_0 is the modified Bessel function of the second kind of order zero and $\gamma \approx 0.5772156649015328$ is *Euler's constant*. The particular solution for the Helmholtz operator $L = \Delta + \lambda^2$ in 2D is [7]:

$$\Phi(r) = \begin{cases} -\frac{4}{\lambda^4} - \frac{4 \log r}{\lambda^4} + \frac{r^2 \log r}{\lambda^2} + \frac{2\pi Y_0(\lambda r)}{\lambda^4}, & r \neq 0, \\ -\frac{4}{\lambda^4} - \frac{4\gamma}{\lambda^4} - \frac{4}{\lambda^4} \log\left(\frac{\lambda}{2}\right), & r = 0, \end{cases} \quad (4.30)$$

where Y_0 is the Bessel function of the second kind of order zero. Also, from (4.18)-(4.20), it can be shown that

$$\Psi_1 = -\frac{1}{\lambda^2}, \quad \Psi_2 = -\frac{x}{\lambda^2}, \quad \text{and} \quad \Psi_3 = -\frac{y}{\lambda^2}. \quad (4.31)$$

4.4.1 Higher Order Splines

To achieve better approximations of $\hat{f}$ and $\hat{u}_p$, it is ideal to use *higher order splines*

$$\varphi(r) = r^{2n} \log r, \quad n \geq 1, \quad (4.32)$$

where n is the order of the spline. Using higher order splines, particular solutions for the modified Helmholtz operator $L = \Delta - \lambda^2$ can be found by using the general formula [25]:

$$\Phi(r) = AI_0(\lambda r) + BK_0(\lambda r) + \sum_{k=1}^{n+1} c_k r^{2k-2} \log r + \sum_{k=1}^{n+1} d_k r^{2k-2}, \quad (4.33)$$

where A is arbitrary (and can be chosen to be zero), I_0 and K_0 are Bessel functions of order zero, and

$$\left\{ \begin{array}{l} B = -\frac{[(2n)!!]^2}{\lambda^{2n+2}}, \\ c_k = -\frac{[(2n)!!]^2}{[(2k-2)!!]^2} \lambda^{2k-2n-4}, \quad 1 \leq k \leq n+1, \\ d_k = c_k \sum_{j=k}^n \left(\frac{1}{j} \right), \quad 1 \leq k \leq n. \end{array} \right. \quad (4.34)$$

Specifically, the particular solutions for $\varphi(r) = r^4 \log r$ is

$$\Phi(r) = \left\{ \begin{array}{ll} -\frac{96}{\lambda^6} - \frac{8r^2}{\lambda^4} - \frac{64 \log r}{\lambda^6} - \frac{16r^2 \log r}{\lambda^4} - \frac{r^4 \log r}{\lambda^2} - \frac{64K_0(\lambda r)}{\lambda^6}, & r \neq 0, \\ -\frac{96}{\lambda^6} + \frac{64\gamma}{\lambda^6} + \frac{64}{\lambda^6} \log\left(\frac{\lambda}{2}\right), & r = 0, \end{array} \right. \quad (4.35)$$

and for $\varphi(r) = r^6 \log r$ is

$$\Phi(r) = \left\{ \begin{array}{ll} -\frac{4224}{\lambda^8} - \frac{480r^2}{\lambda^6} - \frac{12r^4}{\lambda^4} - \frac{2304 \log r}{\lambda^8} - \frac{576r^2 \log r}{\lambda^6} \\ \quad - \frac{36r^4 \log r}{\lambda^4} - \frac{r^6 \log r}{\lambda^2} - \frac{2304K_0(\lambda r)}{\lambda^8}, & r \neq 0, \\ -\frac{4224}{\lambda^8} + \frac{2304\gamma}{\lambda^8} + \frac{2304}{\lambda^8} \log\left(\frac{\lambda}{2}\right), & r = 0. \end{array} \right. \quad (4.36)$$

The general formula for the particular solutions of the Helmholtz operator $L = \Delta + \lambda^2$ is [25]:

$$\Phi(r) = AJ_0(\lambda r) + BY_0(\lambda r) + \sum_{k=1}^{n+1} c_k r^{2k-2} \log r + \sum_{k=1}^n d_k r^{2k-2}, \quad (4.37)$$

where A can be chosen as zero (since it is arbitrary), Y_0 is the Bessel function of the second kind of order zero, and

$$\left\{ \begin{array}{l} B = -\frac{[(2n)!!!]^2}{\lambda^{2n+2}}, \\ c_k = -\frac{[(2n)!!!]^2}{[(2k-2)!!!]^2} \lambda^{2k-2n-4}, \quad 1 \leq k \leq n+1, \\ d_k = c_k \sum_{j=k}^n \left(\frac{1}{j} \right), \quad 1 \leq k \leq n. \end{array} \right. \quad (4.38)$$

Using $\varphi(r) = r^4 \log r$, we get

$$\Phi(r) = \begin{cases} \frac{96}{\lambda^6} - \frac{8r^2}{\lambda^4} + \frac{64 \log r}{\lambda^6} - \frac{16r^2 \log r}{\lambda^4} + \frac{r^4 \log r}{\lambda^2} - \frac{32\pi Y_0(\lambda r)}{\lambda^6}, & r \neq 0, \\ \frac{96}{\lambda^6} - \frac{64\gamma}{\lambda^6} - \frac{64}{\lambda^6} \log\left(\frac{\lambda}{2}\right), & r = 0, \end{cases} \quad (4.39)$$

and using $\varphi(r) = r^6 \log r$, we get

$$\Phi(r) = \begin{cases} -\frac{4224}{\lambda^8} + \frac{480r^2}{\lambda^6} - \frac{12r^4}{\lambda^4} - \frac{2304 \log r}{\lambda^8} - \frac{576r^2 \log r}{\lambda^6} \\ \quad + \frac{36r^4 \log r}{\lambda^4} - \frac{r^6 \log r}{\lambda^2} + \frac{1152\pi Y_0(\lambda r)}{\lambda^8}, & r \neq 0, \\ -\frac{4224}{\lambda^8} - \frac{2304\gamma}{\lambda^8} - \frac{2304}{\lambda^8} \log\left(\frac{\lambda}{2}\right), & r = 0. \end{cases} \quad (4.40)$$

Theoretically, it is preferable to use higher order splines due to higher convergence rate.

However, computationally, higher order splines may cause numerical round-off errors.

When using higher order splines, the additional polynomial terms in the approximation must reach the same order as the spline being used. Table 4.2 lists the specific polynomial terms and their corresponding particular solutions.

Table 4.2: Particular solutions Φ for polynomial terms φ

φ	Φ	φ	Φ
1	$-\frac{1}{\lambda^2}$	y^2	$-\frac{y^2}{\lambda^2} - \frac{2}{\lambda^4}$
x	$-\frac{x}{\lambda^2}$	x^3	$-\frac{x^3}{\lambda^2} - \frac{6x}{\lambda^4}$
y	$-\frac{y}{\lambda^2}$	x^2y	$-\frac{x^2y}{\lambda^2} - \frac{2y}{\lambda^4}$
x^2	$-\frac{x^2}{\lambda^2} - \frac{2}{\lambda^4}$	xy^2	$-\frac{xy^2}{\lambda^2} - \frac{2x}{\lambda^4}$
xy	$-\frac{xy}{\lambda^2}$	y^3	$-\frac{y^3}{\lambda^2} - \frac{6y}{\lambda^4}$

4.5 Numerical Implementation

Let us consider the following modified Helmholtz equation

$$\Delta u - \lambda^2 u = (e^x + e^y)(1 - \lambda^2), \quad x, y \in \Omega, \quad (4.41)$$

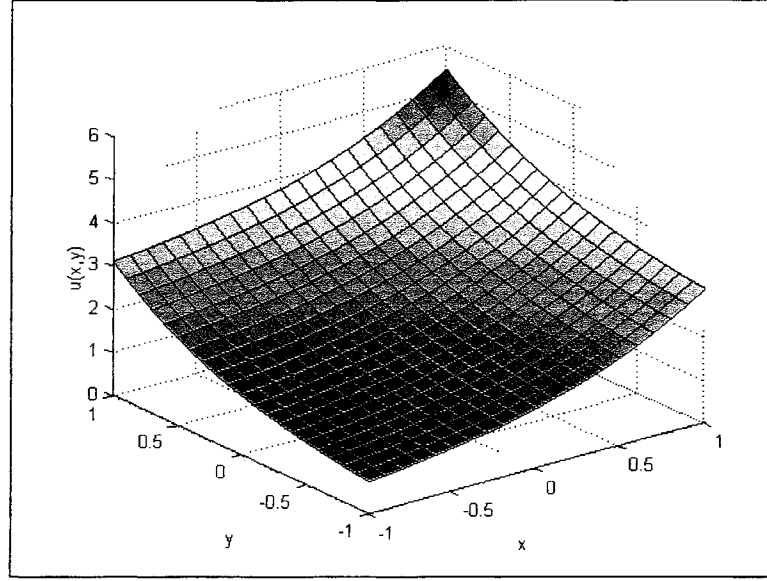
$$u = e^x + e^y, \quad x, y \in \partial\Omega, \quad (4.42)$$

defined on a circular domain

$$\Omega \cup \partial\Omega = \{(x, y) : x^2 + y^2 \leq 1\}. \quad (4.43)$$

The exact solution is given by $u = e^x + e^y$. For convenience of graphing, the exact solution shown in Figure 4.1 is in the extended domain $[-1, 1]^2$.

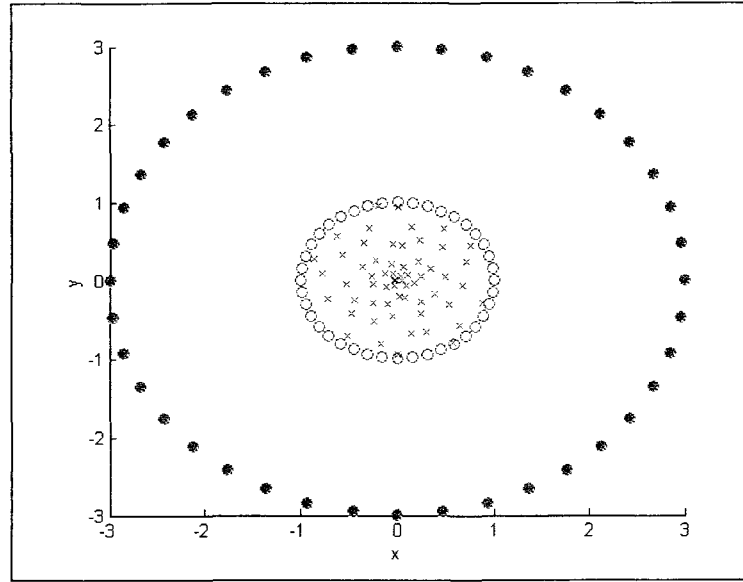
Figure 4.1: Graph of $u(x, y) = e^x + e^y$



Sixty randomly placed points within the domain and the TPS $\phi(r) = r^2 \log r$ are used to interpolate the nonhomogeneous term f and to evaluate the particular solution. Equation (4.29) and the particular solutions for the additional polynomial terms (as specified in Table 4.2) are used in the calculation of the particular solution. After subtracting $\hat{u}_p$ in (4.8) and (4.9), the problem then becomes homogeneous, and 40 evenly distributed collocation and source points are used to implement the MFS.

$G(r) = \frac{1}{2\pi} K_0(\lambda r)$ is the appropriate fundamental solution (from Table 3.1) and the radius of the fictitious circle, centered at $(0, 0)$, is $r = 3$. Figure 4.2 shows the distribution of all relevant points. Table 4.3 and Figure 4.3 demonstrate the effect of λ^2 on the accuracy of the MPS in the modified Helmholtz problem. For this analysis, the approximation is tested on 50 randomly placed points within the domain. The *absolute maximum error* refers to the greatest error among these test points and the exact solution.

Figure 4.2: Distribution of interpolation, collocation, and source points

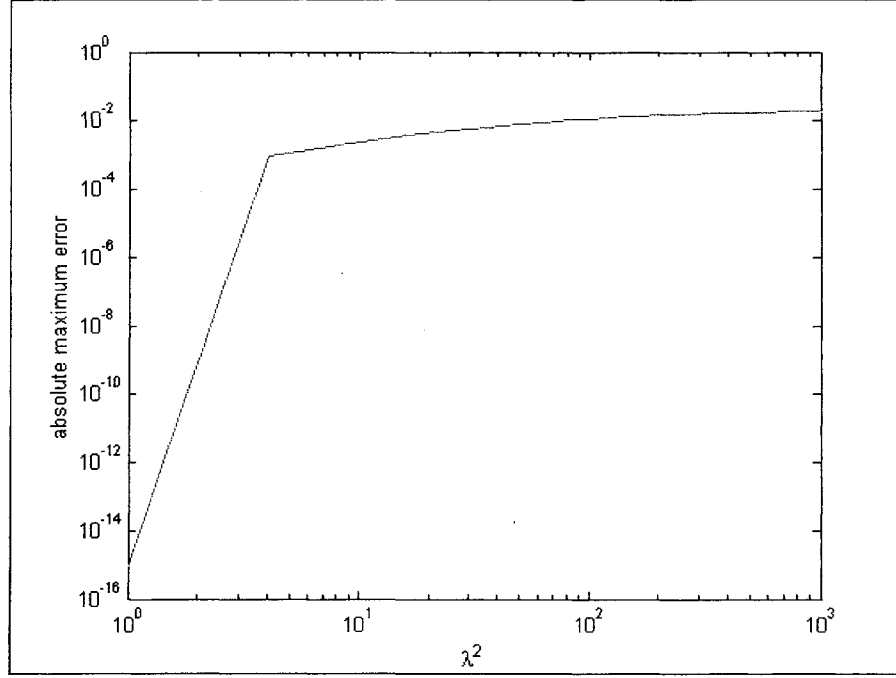


In the table below, note that “E” represents a power of “10.”

Table 4.3: The effect of λ^2 on the accuracy of the MPS

λ^2	absolute maximum error	λ^2	absolute maximum error
1	8.88E-16	324	1.51E-02
4	8.92E-04	361	1.54E-02
9	2.10E-03	400	1.56E-02
16	3.50E-03	441	1.58E-02
25	4.90E-03	484	1.60E-02
36	6.30E-03	529	1.62E-02
49	7.60E-03	576	1.64E-02
64	8.70E-03	625	1.65E-02
81	9.80E-03	676	1.67E-02
100	1.07E-02	729	1.68E-02
121	1.15E-02	784	1.69E-02
144	1.23E-02	841	1.70E-02
169	1.29E-02	900	1.70E-02
196	1.35E-02	961	1.71E-02
225	1.39E-02	1024	1.72E-02
256	1.44E-02	1089	1.73E-02
289	1.47E-02	1156	1.73E-02

Figure 4.3: The effect of λ^2 on the accuracy of the MPS



The best accuracy is achieved when $\lambda^2 = 1$; the accuracy of the approximation $\hat{u}$ is as small as 10^{-16} with the absolute maximum error 8.88×10^{-16} . For increasing values of λ^2 , the accuracy declines. At $\lambda^2 = 1225$, the accuracy is only 10^{-2} with absolute maximum error 1.73×10^{-2} . It is obvious that for $\lambda^2 = 1$, the problem in (4.1) – (4.3) becomes a homogeneous PDE, which is solved solely through the MFS; this accounts for the high degree of accuracy. For $\lambda^2 > 1$, the problem is a nonhomogeneous modified Helmholtz equation.

4.6 Analysis of Accuracy

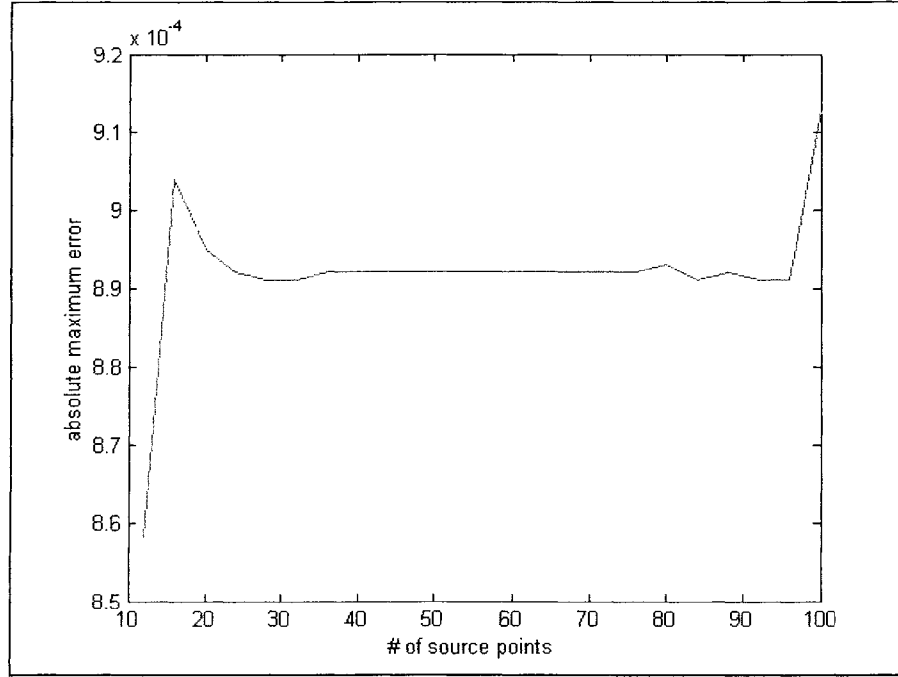
The accuracy of the MPS relies on the approximations of the homogeneous solution $\hat{u}_h$ and the particular solution $\hat{u}_p$. Since $\hat{u}_h$ is solely determined by the MFS, the main

factors that can affect this part of the MPS are those mentioned in Chapter 3: the quantity of source points, m , and the size of the fictitious radius, r . The only factor that can directly affect the calculation of $\hat{u}_p$ is the quantity of interpolation points, n . Using the example of the modified Helmholtz problem in the previous section (with the default parameter settings $n = 60$, $r = 3$, $m = 12$, and $\lambda^2 = 4$), Table 4.4 and Figure 4.4 demonstrate the effect of m , while Table 4.5 and Figure 4.5 demonstrate the effect of n and r . It is important to note that these same factors should be considered when analyzing the accuracy of other nonhomogeneous PDEs with second order linear differential operators. For Tables 4.4 and 4.5, the approximation is again tested on 50 randomly placed points within the domain, and the absolute maximum error is based on such points.

Table 4.4: The effect of the # of source points, m ,
on accuracy of the MPS

m	absolute maximum error	m	absolute maximum error
12	8.58E-04	60	8.92E-04
16	9.04E-04	64	8.92E-04
20	8.95E-04	68	8.92E-04
24	8.92E-04	72	8.92E-04
28	8.91E-04	76	8.92E-04
32	8.91E-04	80	8.93E-04
36	8.92E-04	84	8.91E-04
40	8.92E-04	88	8.92E-04
44	8.92E-04	92	8.91E-04
48	8.92E-04	96	8.91E-04
52	8.92E-04	100	9.13E-04
56	8.92E-04		

Figure 4.4: The effect of the # of source points, m ,
on accuracy of the MPS

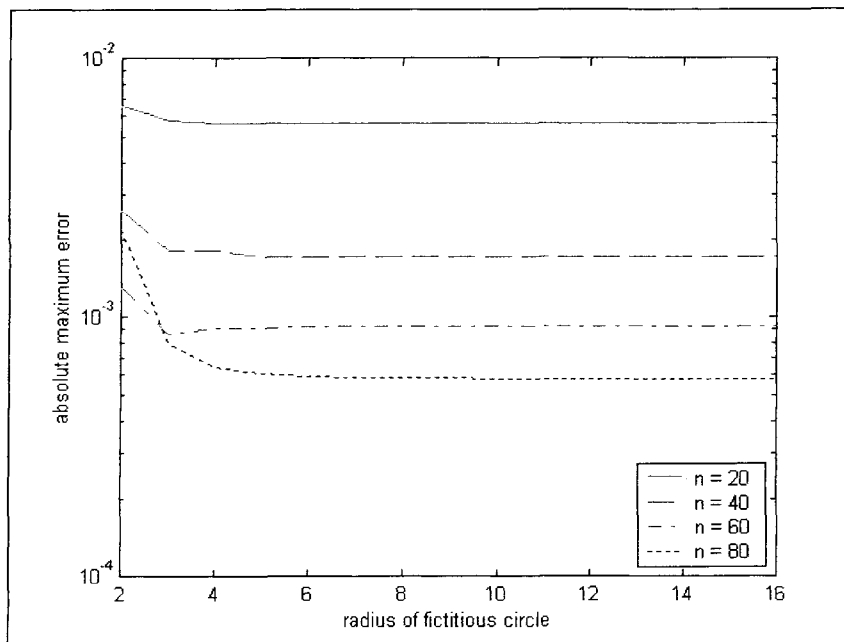


The degree of accuracy stays at 10^{-4} for the amount of source points $12 \leq m \leq 100$. This shows that increasing the number of source points does not significantly improve approximation accuracy. This occurrence is explained by the exponential convergence of the MFS (which deals solely with source points). There is no significant difference between the absolute maximum errors 8.58×10^{-4} (for $m = 12$) and 9.13×10^{-4} (for $m = 100$). Hence, choosing a specific amount of source points is not an issue that ultimately affects the accuracy of the modified Helmholtz equation and the MPS. Utilizing the least amount of source points for the approximation is adequate. The best amount of source points to choose for this example is $m = 12$, where accuracy is 8.58×10^{-4} .

Table 4.5 The effect of the # of interpolation points, n ,
and the size of the fictitious radius, r , on the accuracy of the MPS

r	absolute maximum error			
	n = 20	n = 40	n = 60	n = 80
2	6.50E-03	2.60E-03	1.30E-03	2.20E-03
3	5.70E-03	1.80E-03	8.58E-04	7.98E-04
4	5.60E-03	1.80E-03	9.00E-04	6.48E-04
5	5.60E-03	1.70E-03	9.10E-04	6.10E-04
6	5.60E-03	1.70E-03	9.14E-04	5.96E-04
7	5.60E-03	1.70E-03	9.15E-04	5.89E-04
8	5.60E-03	1.70E-03	9.16E-04	5.86E-04
9	5.60E-03	1.70E-03	9.17E-04	5.84E-04
10	5.60E-03	1.70E-03	9.17E-04	5.83E-04
11	5.60E-03	1.70E-03	9.17E-04	5.82E-04
12	5.60E-03	1.70E-03	9.18E-04	5.82E-04
13	5.60E-03	1.70E-03	9.18E-04	5.82E-04
14	5.60E-03	1.70E-03	9.18E-04	5.81E-04
15	5.60E-03	1.70E-03	9.18E-04	5.81E-04
16	5.60E-03	1.70E-03	9.18E-04	5.81E-04

Figure 4.5 The effect of the # of interpolation points, n ,
and the size of the fictitious radius, r , on the accuracy of the MPS



For this example, an increase in the amount of interpolation points does not necessarily improve the accuracy of the approximation. As seen in Table 4.5, accuracy is of the same degree for a larger amount of points. In general, however, a greater amount of interpolation points will improve the accuracy of the approximation. For this particular example, the optimal accuracy of the MPS can be achieved with as small amount of points as possible; $n = 60$ or $n = 80$ is adequate. Computationally, the lesser the amount of points being used in the calculation, the lesser the amount of computational time.

The size of the fictitious radius does not ultimately affect the accuracy of the MPS either. From Figure 4.5, it is clear that for $r \geq 3$, the accuracy remains the same at 10^{-3} or 10^{-4} for all cases.

4.7 Concluding Remarks

The MPS solves nonhomogeneous PDEs by splitting the problem into two portions; one that involves the homogeneous solution u_h and one that involves the particular solution u_p . While the homogeneous solution can be easily solved through the highly effective MFS, the particular solution can be solved through RBF approximation. In the end, the sum of the two solutions gives us the overall approximate solution $\hat{u} = \hat{u}_h + \hat{u}_p$.

The factors that may affect the overall accuracy of the MPS include: the number of collocation/source points m , the size of the fictitious radius r , and the number of interpolation points n . However, these parameters are not sensitive to the accuracy of the MPS; in spite of changing the quantities of such parameters, the approximations still have the absolute maximum errors that are of the same degree. In the next chapter, the time-dependent wave equations are solved with the MPS approach.

CHAPTER 5

THE WAVE EQUATION

The focus of this thesis is to couple the MFS and the MPS in solving wave equations. In the preceding chapters, the approximate solutions to PDEs were produced through the meshless techniques of RBFs, the MFS, and the MPS. While these examples did not involve the time variable, time-dependent PDEs (such as the wave equation) can nevertheless be solved through similar means. The main idea is to reduce the time-dependent problem into a time-independent problem. Elimination of the time variable is possible through either the Laplace transform or finite differencing in time. Once the time variable is removed, the resultant wave equation is reduced to a series of inhomogeneous modified Helmholtz equations, which can be solved through the aforementioned methods.

Let us consider the following wave equation in 2D

$$u_{tt}(\bar{x}, t) = \Delta u(\bar{x}, t) + f(\bar{x}, t), \quad \bar{x} \in \Omega, \quad t > 0, \quad (5.1)$$

with Dirichlet boundary condition

$$u(\bar{x}, t) = g(\bar{x}, t), \quad \bar{x} \in \partial\Omega, \quad t > 0, \quad (5.2)$$

and initial conditions

$$u(\bar{x}, 0) = u_0(\bar{x}), \quad u_t(\bar{x}, 0) = w_0(\bar{x}), \quad (5.3)$$

where $\bar{x} = (x, y)$.

5.1 The Laplace Transform

To remove the time variable temporarily, we define the Laplace transform of $u(\bar{x}, t)$, when it exists, as

$$\mathcal{L}[u(\bar{x}, t)] = U(\bar{x}; s) = \int_0^\infty u(\bar{x}, t) e^{-st} dt \quad (5.4)$$

where the transform parameter s is positive and real. By integration by parts, we obtain

$$\mathcal{L}[u_t(\bar{x}, t)] = \int_0^\infty u_t(\bar{x}, t) e^{-st} dt \quad (5.5)$$

$$= -u_t(\bar{x}, 0) + s \int_0^\infty u_t(\bar{x}, t) e^{-st} dt \quad (5.6)$$

$$= -u_t(\bar{x}, 0) + s[sU(\bar{x}; s) - u_0(\bar{x})] \quad (5.7)$$

$$= s^2 U(\bar{x}; s) - su_0(\bar{x}) - w_0(\bar{x}). \quad (5.8)$$

By substituting (5.4) and (5.8) into (5.1) and (5.2) and taking the Laplace transforms of $f(\bar{x}, t)$ and $g(\bar{x}, t)$, the wave equation can be reduced to the following inhomogeneous modified Helmholtz equation:

$$\Delta U(\bar{x}; s) - s^2 U(\bar{x}; s) = -su_0 - sF(\bar{x}; s) - w_0, \quad \bar{x} \in \Omega, \quad (5.9)$$

$$U(\bar{x}; s) = G(\bar{x}; s), \quad \bar{x} \in \partial\Omega, \quad (5.10)$$

where $F(\bar{x}; s) = \mathcal{L}[f(\bar{x}, t)]$ and $G(\bar{x}; s) = \mathcal{L}[g(\bar{x}, t)]$.

The wave equation is now in the Laplace space and can be solved by the MPS and the MFS. As seen in equations (4.4) – (4.7), the approximate solution

$$\hat{U} = \hat{U}_h + \hat{U}_p, \quad (5.11)$$

where $\hat{U}_p$ satisfies

$$(\Delta - s^2)\hat{U}_p = -su_0 - sF(\bar{x}; s) - w_0, \quad (5.12)$$

but does not necessarily satisfy the boundary condition in (5.10). Meanwhile, $\hat{U}_h$ satisfies

$$(\Delta - s^2)\hat{U}_h = 0, \quad \bar{x} \in \Omega, \quad (5.13)$$

$$\hat{U}_h = \hat{G}(\bar{x}; s), \quad \bar{x} \in \partial\Omega. \quad (5.14)$$

To approximate the particular solution $\hat{U}_p$, we refer to the closed form particular solution

$\Phi(r)$ in equation (4.29). To approximate the homogeneous solution $\hat{U}_h$, we refer to the

fundamental solution $G(r) = \frac{1}{2\pi} K_0(\lambda r)$ found in Table 3.1. In both cases, $\lambda = s$.

5.1.1 The Inverse Laplace Transform: Stehfest's Algorithm

After the approximate solution $\hat{U}(\bar{x}; s)$ is found in the Laplace space, the solution must be inverted back to the original time-space. Among all of the numerical inverse transform schemes, *Stehfest's algorithm* [29] has been successfully implemented in the context of BEM [8, 33] and the MFS [5, 6]. For the first part of Stehfest's algorithm, n_s distinct parameters s must be chosen for a given observation time t ; i.e.,

$$s_v = \frac{\ln 2}{t} v, \quad v = 1, 2, \dots, n_s, \quad (5.15)$$

where n_s is the number of terms in Stehfest's algorithm and must be an even number. At each s_v , an approximate solution $\hat{U}(\bar{x}; s_v)$ from (5.9) and (5.10) must be obtained for any given point $\bar{x} \in \Omega \cup \partial\Omega$.

The second part of Stehfest's algorithm is to invert the solution from the Laplace space to the time domain at any given time with the inversion procedure

$$\hat{u}(\bar{x}, t) = \frac{\ln 2}{t} \sum_{v=1}^{n_s} W_v \cdot \hat{U}(\bar{x}; s_v), \quad (5.16)$$

where

$$W_v = (-1)^{n_s+v} \sum_{k=\lfloor \frac{1}{2}(v+1) \rfloor}^{\min\{v, \frac{n_s}{2}\}} \frac{k^{\frac{n_s}{2}} (2k)!}{\left(\frac{n_s}{2} - k\right)! k! (k-1)! (v-k)! (2k-v)!}, \quad (5.17)$$

and $\lfloor \frac{1}{2}(v+1) \rfloor$ is the largest integer less than or equal to $\frac{1}{2}(v+1)$.

The accuracy of Stehfest's algorithm depends on the correct choice of n_s , the number of terms in equation (5.16). As n_s increases, the accuracy improves and then round-off error becomes a factor and the accuracy eventually declines. The optimal n_s has a significant impact on the final solution for our proposed approach.

5.1.2 Numerical Implementation

To demonstrate the effectiveness of the Laplace transform in solving the wave equation, let us consider two test examples: one with a *forcing term* $f(x, y, t)$ and one without a forcing term. Both test examples are considered in the domain $\Omega \cup \partial\Omega = [0, 1]^2$ for $t > 0$.

Test Example 1

$$u_{tt} = \Delta u, \quad x, y \in \Omega, \quad (5.18)$$

$$u(x, y, t) = 0, \quad x, y \in \partial\Omega, \quad (5.19)$$

$$u(x, y, 0) = \sin \pi x \sin \pi y, \quad x, y \in \Omega, \quad (5.20)$$

$$u_t(x, y, 0) = 0, \quad x, y \in \Omega, \quad (5.21)$$

where the exact solution is given by

$$u(x, y, t) = \sin \pi x \sin \pi y \cos(\sqrt{2}\pi t). \quad (5.22)$$

To numerically evaluate the particular solution $\hat{U}_p$ and to interpolate the nonhomogeneous term (the right side of the equation), 121 uniformly distributed points are placed within the domain $[0,1]^2$, and the TPS $\varphi(r)=r^2 \log r$ is used (refer to Equation 4.29). To approximate the homogeneous solution $\hat{U}_h$ by the MFS, 24 equally distributed collocation points on the boundary and the same number of source points on a circle with center $(0.5, 0.5)$ and radius $r = 1.5$ are used. The number of terms n_s for the inversion of the Laplace transform is $n_s = 20$. Profiles of the exact solution and the absolute error of approximation at time $t = 0.5$ are shown in Figures 5.1 and 5.2, respectively. The absolute maximum error for the approximation, which is based on 121 uniformly distributed test points, is 1.20×10^{-3} .

Figure 5.1: Test Example 1, exact solution at $t = 0.5$

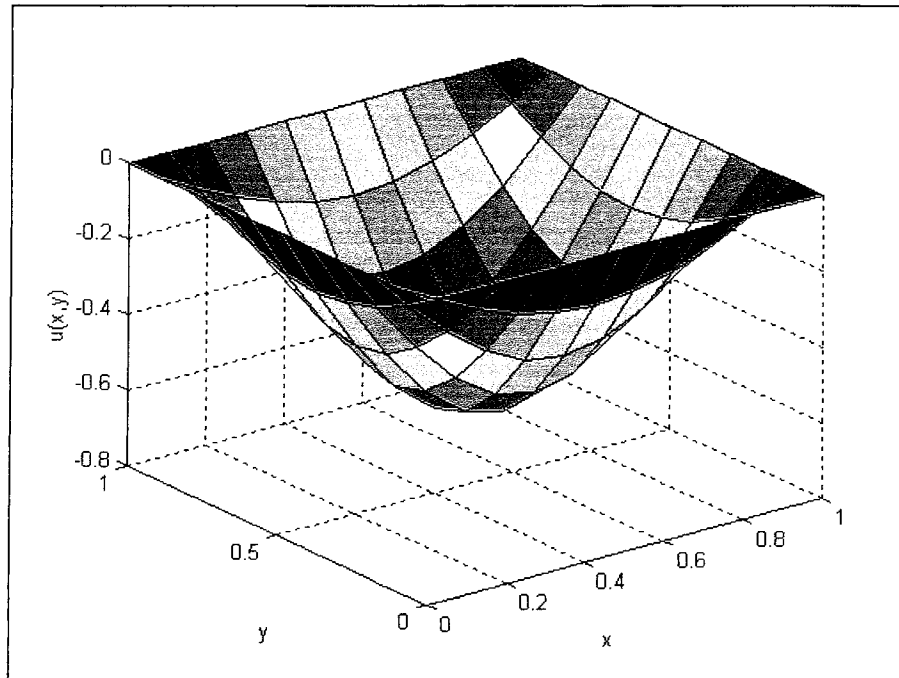
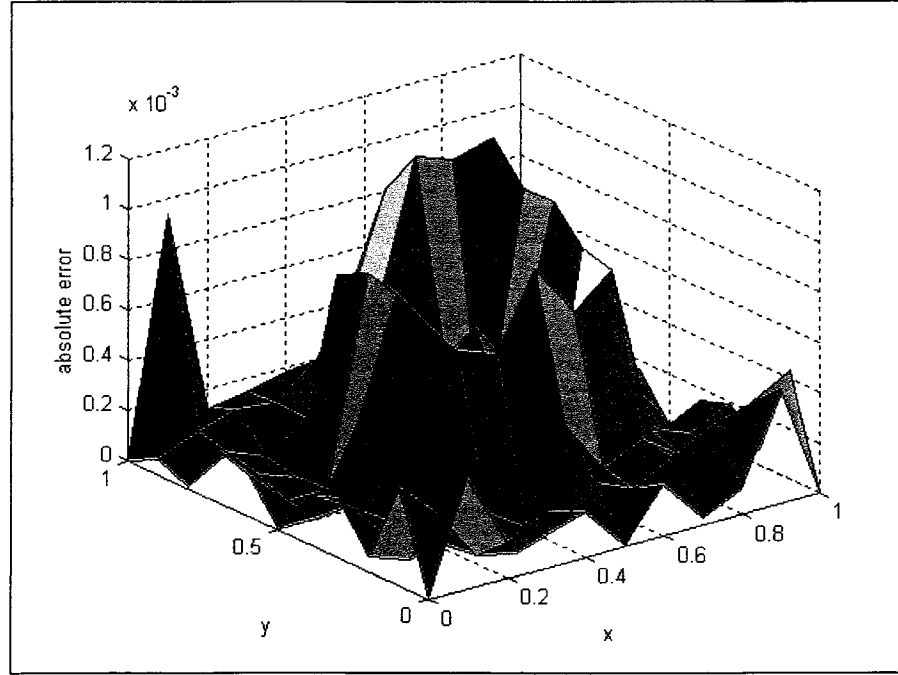


Figure 5.2: Test Example 1, Laplace transform,
absolute error at $t = 0.5$



Test Example 2

$$u_{tt} = \Delta u + f(x, y, t), \quad x, y \in \Omega, \quad (5.23)$$

$$u(x, y, t) = 0, \quad x, y \in \partial\Omega, \quad (5.24)$$

$$u(x, y, 0) = x(1-x)y(1-y), \quad x, y \in \Omega, \quad (5.25)$$

$$u_t(x, y, 0) = 0, \quad x, y \in \Omega, \quad (5.26)$$

where

$$f(x, y, t) = [2x(1-x) + 2y(1-y) - x(1-x)y(1-y)] \cos t, \quad (5.27)$$

and the exact solution is

$$u(x, y, t) = x(1-x)y(1-y) \cos t. \quad (5.28)$$

Similar to the previous example, 121 uniformly distributed points are used to approximate the nonhomogeneous term, and the TPS $\varphi(r) = r^2 \log r$ is used in the calculation of the particular solution $\hat{U}_p$ (4.29). To approximate the homogeneous solution $\hat{U}_h$, 32 equally distributed collocation points on the boundary and the same number of source points on a circle with center $(0.5, 0.5)$ and radius $r = 2$ are used. The number of terms n_s for the inversion of the Laplace transform is $n_s = 18$. Profiles of the approximate solution and absolute error of approximation at time $t = 2$ are shown in Figures 5.3 and 5.4, respectively. The absolute maximum error for the approximation, which is based on 121 uniformly distributed test points, is 7.1484×10^{-6} .

Figure 5.3: Test Example 2, exact solution at $t = 2$

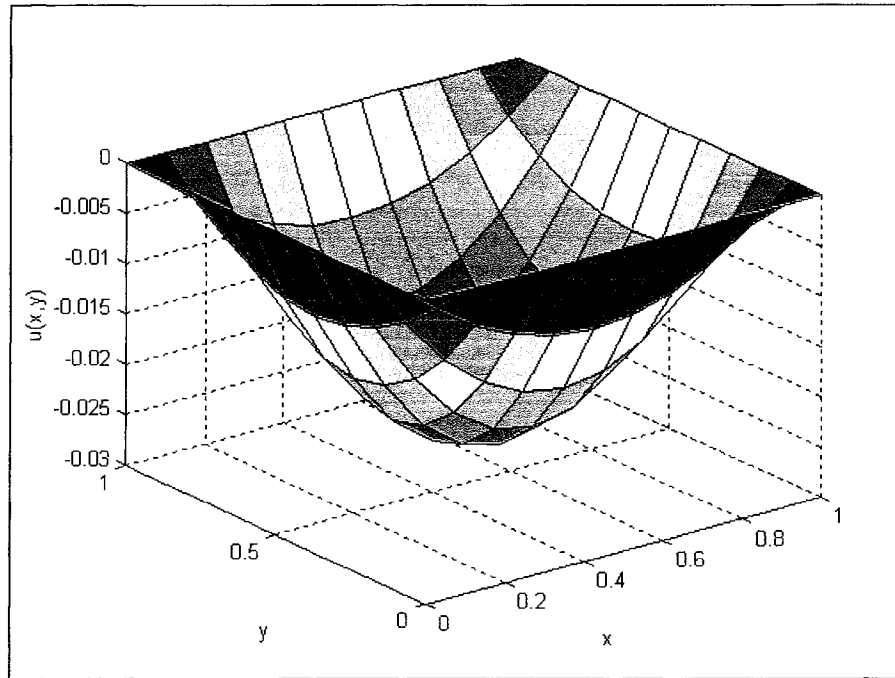
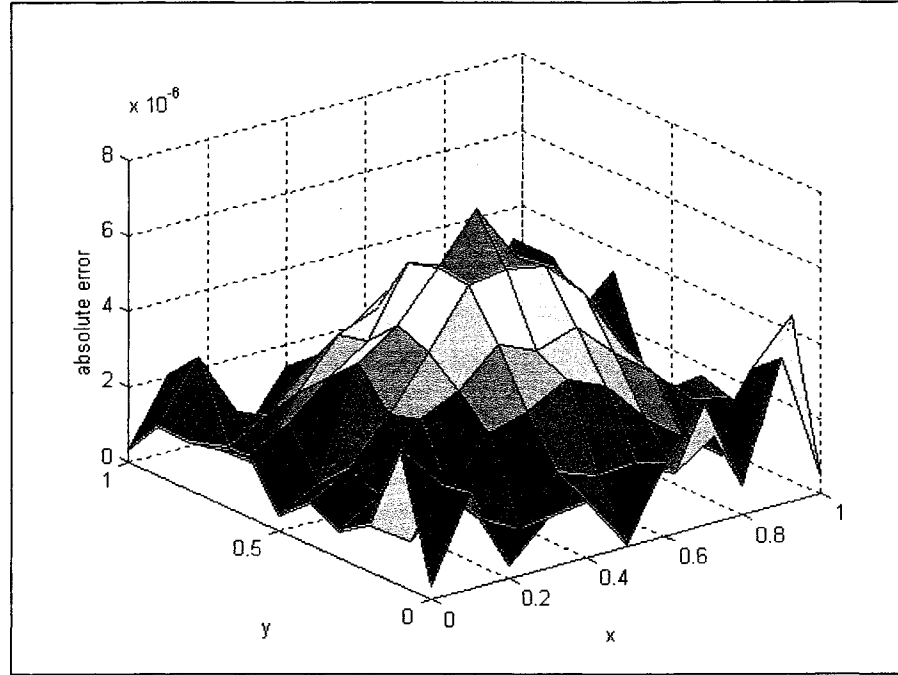


Figure 5.4: Test Example 2, Laplace transform,
absolute error at $t = 2$



5.1.3 Analysis of Accuracy

As stated in Section 5.1.1, the accuracy of the approximation for the wave equation is dependent on the proper choice of n_s , the number of terms in equation (5.16). A particular choice of n_s may yield optimal accuracy for a certain time t , but another choice of n_s may yield optimal accuracy for another time t . Referring to Test Example 1, and using the same parameter settings ($n = 121$, $m = 24$, and $r = 1.5$), the exact solution at time $t = 1.5$ is shown in Figure 5.5. The corresponding absolute error is shown for both $n_s = 20$ and $n_s = 22$ (in Figure 5.6). The absolute maximum errors for $n_s = 20$ and $n_s = 22$ are 1.009×10^{-1} and 8.20×10^{-3} , respectively. It is obvious that at this specific time, $n_s = 22$ produces better accuracy than $n_s = 20$.

Figure 5.5: Test Example 1, exact solution at $t = 1.5$

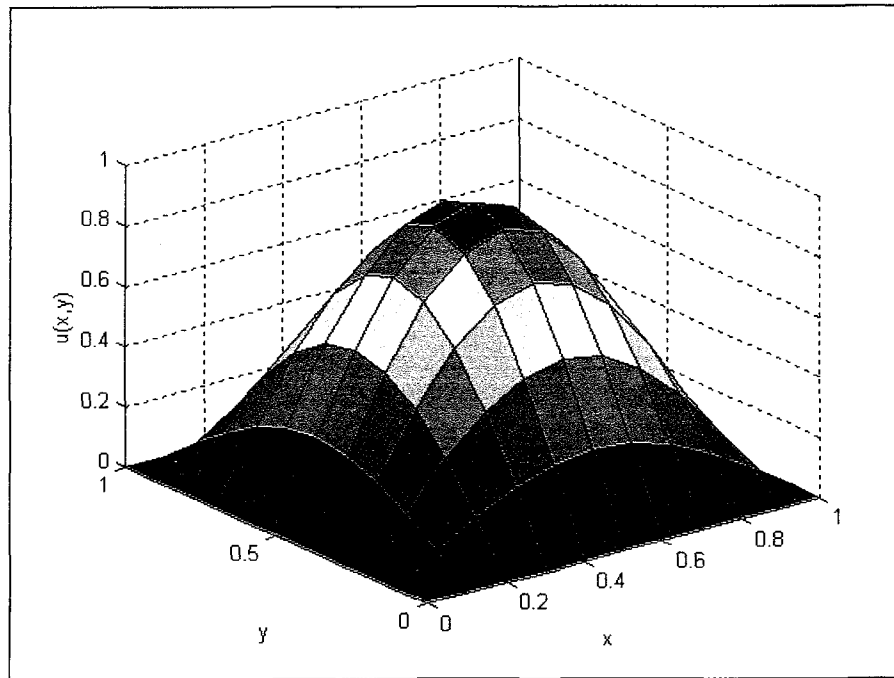
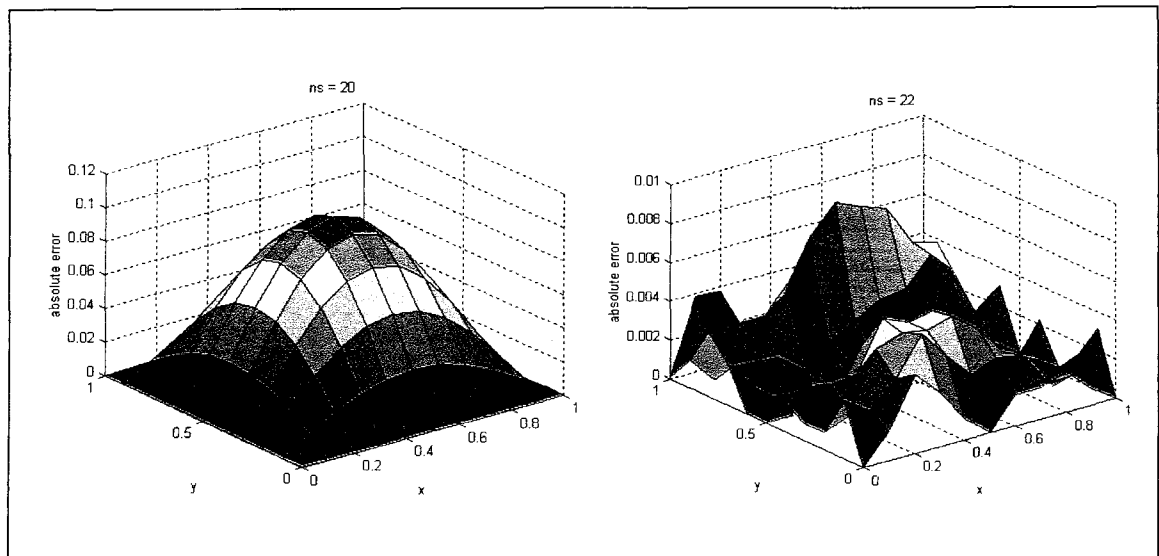


Figure 5.6: Test Example 1, absolute error at $t = 1.5$

for $n_s = 20$ and $n_s = 22$



For this example, $n_s = 20$ and $n_s = 22$ give way to optimal accuracy for different times. However, the degree of accuracy is similar for both choices (refer to Table 5.1 and Figure 5.7) as time increases. In the table below, the absolute maximum error for the entire domain is noted. A visual representation of the data is given in Figure 5.7

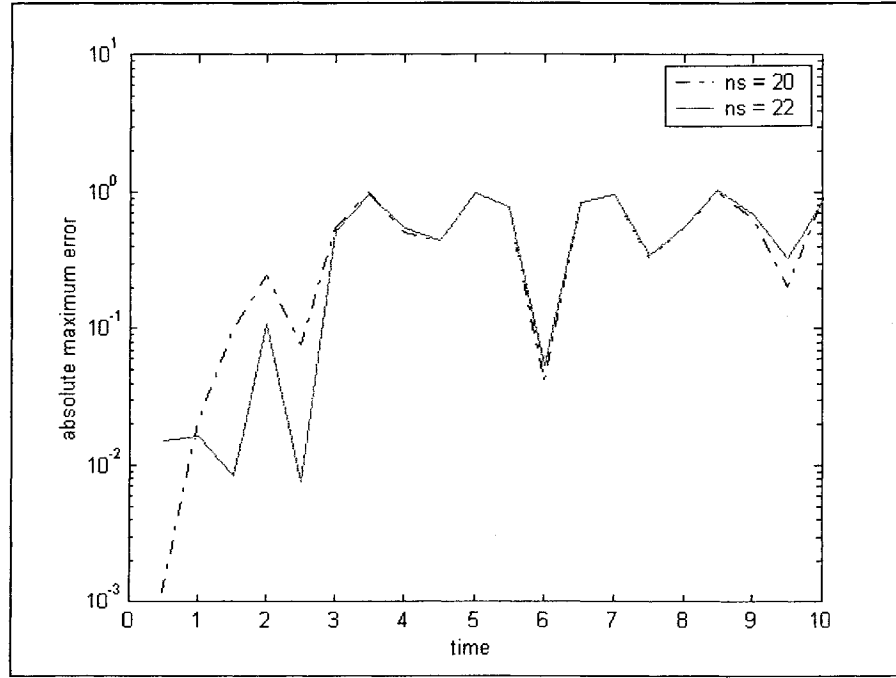
Table 5.1: Test Example 1, absolute maximum error
of $n_s = 20$ vs. $n_s = 22$ for $0 < t \leq 10$

time	ns = 20	ns = 22
0.5	1.20E-03	1.48E-02
1.0	2.13E-02	1.63E-02
1.5	1.01E-01	8.20E-03
2.0	2.48E-01	1.08E-01
2.5	7.70E-02	7.40E-03
3.0	5.53E-01	5.07E-01
3.5	9.96E-01	9.38E-01
4.0	5.13E-01	5.40E-01
4.5	4.38E-01	4.35E-01
5.0	9.71E-01	9.78E-01
5.5	7.64E-01	7.57E-01
6.0	4.20E-02	5.26E-02
6.5	8.27E-01	8.25E-01
7.0	9.50E-01	9.35E-01
7.5	3.30E-01	3.41E-01
8.0	5.53E-01	5.51E-01
8.5	9.96E-01	1.04E+00
9.0	6.57E-01	6.85E-01
9.5	2.03E-01	3.22E-01
10.0	9.08E-01	8.46E-01

For small enough times, the approximation has accuracy that is either 10^{-2} or 10^{-3} . For $t \geq 3$, the approximation pretty much stays at 10^{-1} accuracy.

Figure 5.7: Test Example 1, absolute maximum error

of $n_s = 20$ vs. $n_s = 22$ for $0 < t \leq 10$



For this equation, it is evident that error increases as time increases. This can be explained either by the nature of the ill-posed inverse Laplace transform [6] or the instability of the wave problem itself. As time progresses, a rise in error between the approximate solution and the exact solution suggest that the approximate solution is tending toward zero as time increases. Figures 5.8 and 5.9 demonstrate this tendency for the approximate solution to go to zero with an increase in time. The single test point $(0.5, 0.5)$ and the parameters $n = 121$, $m = 24$, $r = 1.5$, and $n_s = 22$ are used to implement the Laplace transform for Test Example 1.

Figure 5.8: Test Example 1, Laplace transform, exact solution
and approximate solution of $(0.5, 0.5)$ for $0 < t \leq 10$

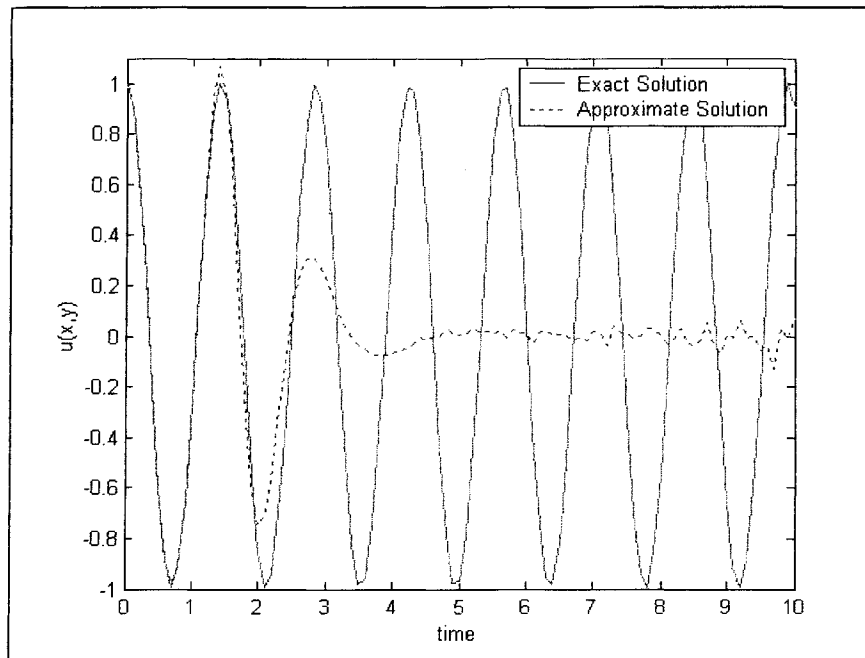
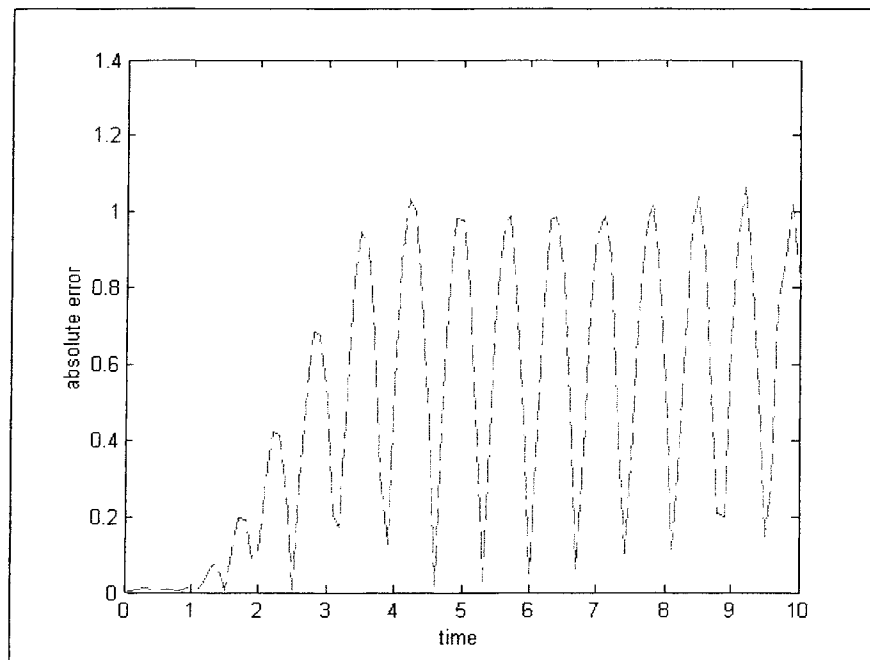


Figure 5.9: Test Example 1, Laplace transform, absolute error
of approximation of $(0.5, 0.5)$ for $0 < t \leq 10$



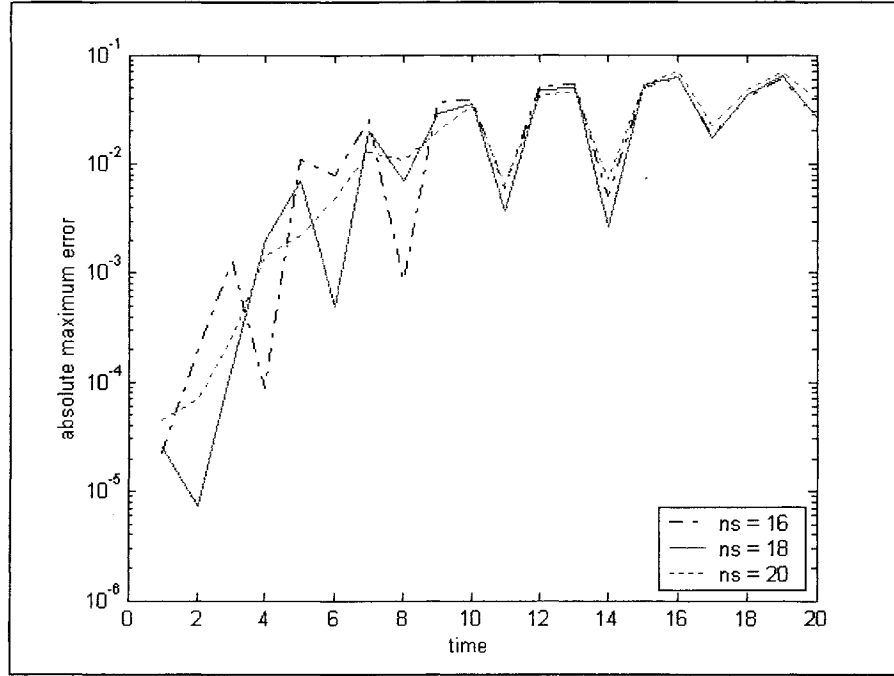
Similar to Test Example 1, Test Example 2 can achieve optimal accuracy at different times for different choices of n_s ; that is, one choice for n_s does not necessarily guarantee optimal accuracy for all times being tested. For this case, $n_s = 16, 18$, or 20 will generate the best possible results. Using the parameter settings $n = 121$, $m = 32$, and $r = 2$, the table below lists the absolute maximum error for the entire domain. A visual representation of the data is given in Figure 5.10.

Table 5.2: Test Example 2, absolute maximum error

of $n_s = 16, 18$, and 20 for $0 < t \leq 20$

time	ns = 16	ns = 18	ns = 20
1	2.277E-05	2.490E-05	4.416E-05
2	2.016E-04	7.148E-06	6.765E-05
3	1.300E-03	1.372E-04	2.578E-04
4	8.839E-05	2.000E-03	1.400E-03
5	1.090E-02	6.900E-03	2.200E-03
6	7.900E-03	4.891E-04	4.700E-03
7	2.560E-02	1.970E-02	1.270E-02
8	8.502E-04	6.900E-03	1.050E-02
9	3.750E-02	2.840E-02	1.970E-02
10	3.820E-02	3.440E-02	3.300E-02
11	6.100E-03	3.600E-03	6.500E-03
12	5.250E-02	4.760E-02	4.120E-02
13	5.370E-02	4.910E-02	4.590E-02
14	5.000E-03	2.600E-03	7.200E-03
15	5.050E-02	5.160E-02	5.240E-02
16	6.200E-02	6.070E-02	6.930E-02
17	1.850E-02	1.700E-02	2.220E-02
18	4.060E-02	4.230E-02	4.730E-02
19	6.150E-02	6.330E-02	6.880E-02
20	2.560E-02	2.600E-02	3.940E-02

Figure 5.10: Test Example 2, absolute maximum error
of $n_s = 16, 18$, and 20 for $0 < t \leq 20$



Comparable to Test Example 1, the error for Test Example 2 also increases with time. This, again, is attributed to the problem being unstable or the ill-posed inverse Laplace transform. The presence of a forcing term, however, allows for the approximate solution to be closer to the exact solution (for a short time period), keeping the error smaller than the wave equation without a forcing term (as Test Example 1). Nevertheless, error still increases over time, and the approximate solution goes to zero as time gets larger (see Figures 5.11 and 5.12). The single test point $(0.5, 0.5)$ and the parameters $n = 121$, $m = 32$, $r = 2$, and $n_s = 18$ are used.

Figure 5.11: Test Example 2, Laplace transform, exact solution
and approximate solution of $(0.5, 0.5)$ for $0 < t \leq 20$

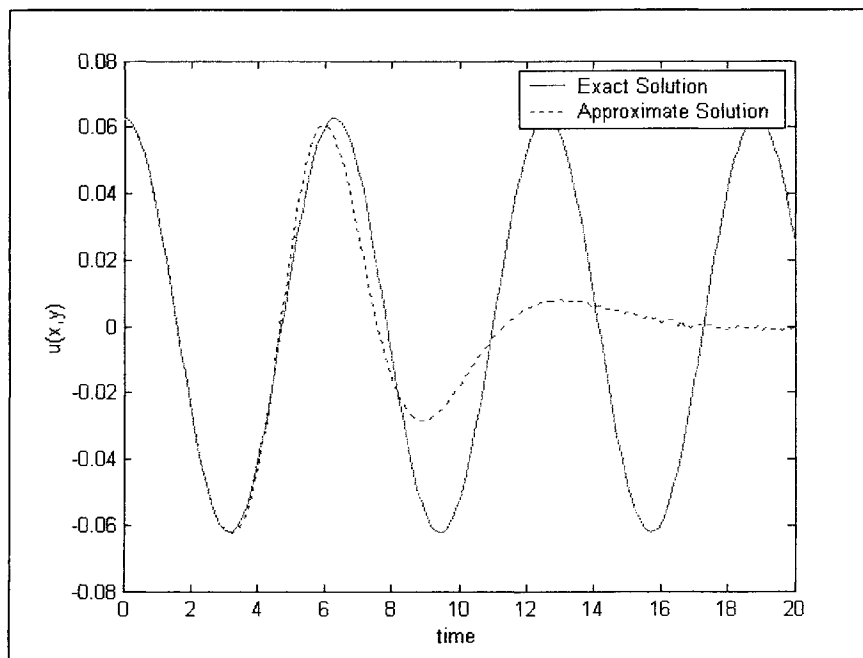
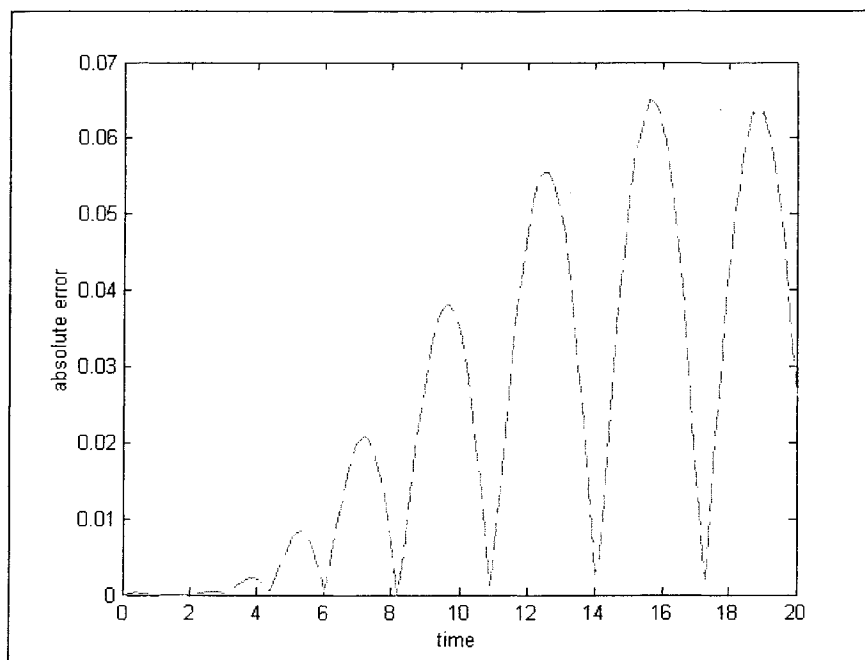


Figure 5.12: Test Example 2, Laplace transform, absolute error
of approximation of $(0.5, 0.5)$ for $0 < t \leq 20$



For both Test Example 1 and 2, the Laplace transform works well for very small time-frames. For Test Example 1 and the results in Figures 5.8 and 5.9, the Laplace transform has optimal accuracy for $t \leq 2.5$. For Test Example 2 and the results in Figures 5.11 and 5.12, optimal accuracy is achieved for $t \leq 6$. The decline in accuracy after the given time-frames is attributed to the ill-posed nature of the inverse Laplace transform.

The difficulty of implementing the Laplace transform is the proper selection of n_s . The question remains: how do we accurately pick n_s ? It may be very time-consuming to determine the appropriate combination of parameter values r (radius), m (the number of source points, and n (the number of interpolation points) as well as n_s . Once the suitable n_s is selected, approximate solutions will be stable for changes in the parameters r , m , and n . Hence, it is easiest to pick the smallest parameter values (that will allow for the most favorable accuracy) since these parameter values make no significant difference in accuracy. This is the reason the parameters for Test Example 1 and 2 were chosen.

Another factor to consider when choosing n_s is the sensitivity of the approximate solution at certain times. Sometimes, it is preferable to use an n_s that is one value higher or lower to achieve optimal accuracy for certain times. This was clear in the fluctuations of accuracy for $n_s = 20$ or 22 in Test Example 1 and $n_s = 16, 18$, or 20 in Test Example 2.

5.2 The Time-Stepping Scheme

The solution of the wave equation through the Laplace transform algorithm may not be effective for all problems since the numerical inversion of the Laplace transform is ill-

posed. Instead, it may be suitable to use the finite differences in time to solve the wave equation. A popular class of methods for doing this is the θ -method, defined as follows [27, 30]: let the *time-step* $\Delta t > 0$ and define the mesh $t^n = n\Delta t$, $n \geq 0$. For $t^n \leq t \leq t^{n+1}$, $u(\bar{x}, t)$ and $f(\bar{x}, t)$ in equation (5.1) are approximated by

$$u(\bar{x}, t) \approx \theta u(\bar{x}, t^{n+1}) + (1 - \theta) u(\bar{x}, t^n), \quad (5.29)$$

$$f(\bar{x}, t) \approx \theta f(\bar{x}, t^{n+1}) + (1 - \theta) f(\bar{x}, t^n), \quad (5.30)$$

where $0 < \theta \leq 1$. It follows that

$$\Delta u(\bar{x}, t) \approx \theta \Delta u(\bar{x}, t^{n+1}) + (1 - \theta) \Delta u(\bar{x}, t^n). \quad (5.31)$$

Furthermore, u_{tt} is approximated by the *central difference formula*

$$u_{tt} \approx \frac{u^{n+1} - 2u^n + u^{n-1}}{(\Delta t)^2}. \quad (5.32)$$

Here, the superscripts $n-1$, n , and $n+1$ denote the corresponding time-step. Substituting (5.29) – (5.32) into (5.1), we get

$$\frac{u^{n+1} - 2u^n + u^{n-1}}{(\Delta t)^2} = \theta \Delta u^{n+1} + (1 - \theta) \Delta u^n + \theta f^{n+1} + (1 - \theta) f^n. \quad (5.33)$$

To avoid evaluating Δu^{n+1} at each time-step, it is necessary to use the transformation

$$v^{n+1} = u^{n+1} + \frac{1 - \theta}{\theta} u^n. \quad (5.34)$$

Substituting the transform into (5.33) yields

$$\frac{v^{n+1} - \frac{1 - \theta}{\theta} u^n - 2u^n + u^{n-1}}{\theta (\Delta t)^2} = \Delta v^{n+1} + f^{n+1} + \frac{(1 - \theta)}{\theta} f^n, \quad (5.35)$$

and rearranging the terms produces the following time-stepping scheme:

$$(\Delta - \lambda^2)v^{n+1} = -\frac{1+\theta}{\theta^2(\Delta t)^2}u^n + \frac{1}{\theta(\Delta t)^2}u^{n-1} - f^{n+1} - \frac{1-\theta}{\theta}f^n, \quad \bar{x} \in \Omega, \quad (5.36)$$

$$v^{n+1} = g^{n+1}, \quad \bar{x} \in \partial\Omega, \quad (5.37)$$

where

$$\lambda^2 = \frac{1}{\theta(\Delta t)^2}, \quad (5.38)$$

and the boundary condition

$$g^{n+1} = g(\bar{x}, t^{n+1}). \quad (5.39)$$

Using the approximation

$$u_t \approx \frac{u^{n+1} - u^n}{\Delta t}, \quad (5.40)$$

we get the initial conditions

$$u^0(\bar{x}) = u_0(\bar{x}), \quad \bar{x} \in \Omega \cup \partial\Omega, \quad (5.41)$$

$$u^1(\bar{x}) = u_0(\bar{x}) + \Delta t w_0(\bar{x}), \quad \bar{x} \in \Omega \cup \partial\Omega, \quad (5.42)$$

where $u_0(\bar{x})$ and $w_0(\bar{x})$ are the given initial conditions as prescribed in (5.3).

At each time-step, after v^{n+1} is evaluated, u^{n+1} can be recovered from (5.34). Using this time-step scheme, the wave equation is converted into a series of nonhomogeneous modified Helmholtz equations. The tricky part about solving the series of equations is to properly obtain an approximate particular solution $\hat{v}_p^{n+1}$. Recall that $\hat{v}_p^{n+1}$ satisfies

$$(\Delta - \lambda^2)\hat{v}_p^{n+1} = -\frac{1+\theta}{\theta^2(\Delta t)^2}u^n + \frac{1}{\theta(\Delta t)^2}u^{n-1} - f^{n+1} - \frac{1-\theta}{\theta}f^n, \quad (5.43)$$

but does not essentially satisfy the boundary conditions. At each step, the right side is entirely known and can be approximated through the TPS or higher order splines.

Additionally, the approximate homogeneous solution $\hat{v}_h^{n+1}$ satisfies

$$(\Delta - \lambda^2) \hat{v}_h^{n+1} = 0, \quad \bar{x} \in \Omega, \quad (5.44)$$

$$\hat{v}^{n+1} = \hat{g}^{n+1} - \hat{v}_p^{n+1}, \quad \bar{x} \in \partial\Omega. \quad (5.45)$$

Again, the closed form particular solution $\Phi(r)$ in equation (4.29) is used to approximate

the particular solution $\hat{v}_p^{n+1}$. The fundamental solution $G(r) = \frac{1}{2\pi} K_0(\lambda r)$ in Table 3.1 is

used to approximate the homogeneous solution $\hat{v}_h^{n+1}$. In both cases, $\lambda = \frac{1}{\sqrt{\theta(\Delta t)}}$.

5.2.1 Numerical Implementation

Let us again consider Test Example 1. To numerically evaluate the particular solution $\hat{v}_p$ and to interpolate the nonhomogeneous term (the right side of the time-stepping scheme), 121 uniformly distributed points are placed inside the domain $[0,1]^2$, and the TPS $\varphi(r) = r^2 \log r$ is used (4.29). To approximate the homogeneous solution $\hat{v}_h$ by the MFS, 24 equally distributed collocation points along the interior of the boundary and the same number of source points on a circle with center $(0.5, 0.5)$ and radius $r = 1.5$ are used. The values $\Delta t = 0.025$ and $\theta = 1$ are chosen. Profiles of the exact solution and the absolute error of approximation at time $t = 1$ are shown in Figures 5.13 and 5.14, respectively. The absolute maximum error for the approximation, which is based on 81 uniformly distributed test points within the interior domain $[0.1, 0.9]$, is 1.75×10^{-2} .

Figure 5.13: Test Example 1, exact solution at $t = 1$

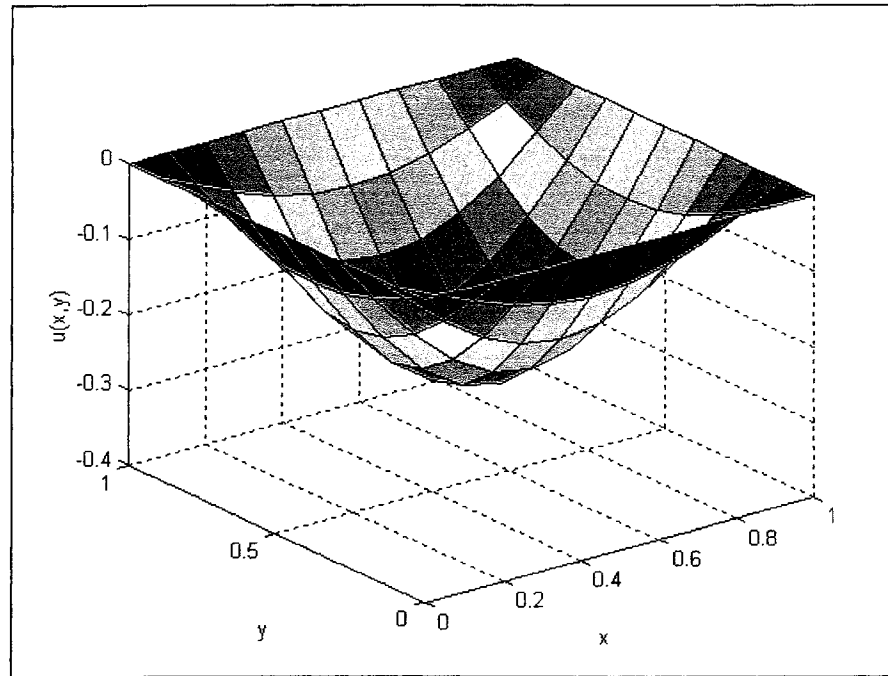
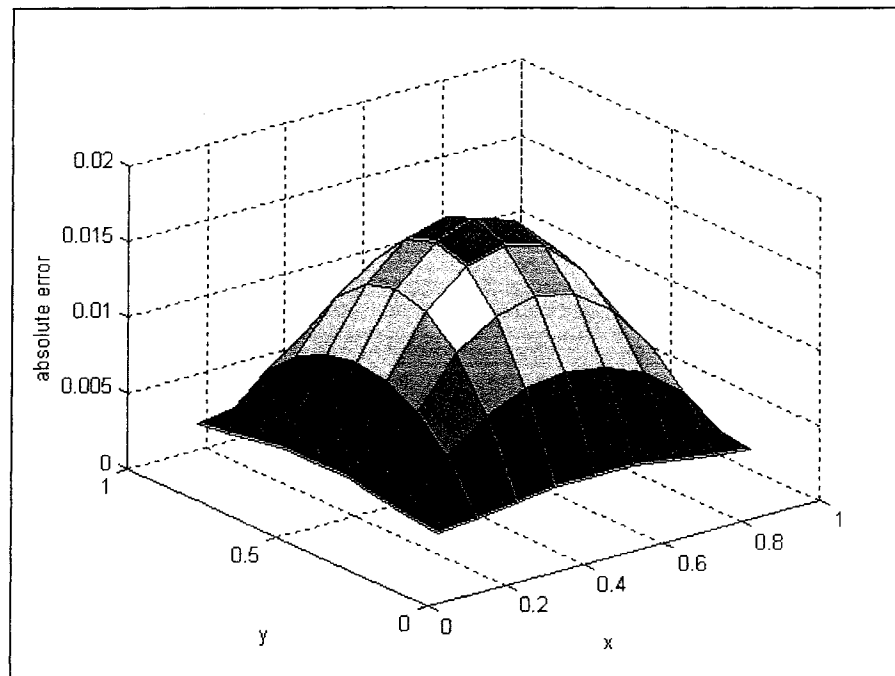


Figure 5.14: Test Example 1, time-stepping scheme ($\Delta t = 0.025$),
absolute error at $t = 1$



For Test Example 2, 121 uniformly distributed points and the TPS $\varphi(r) = r^2 \log r$ are used to approximate the nonhomogeneous term and the particular solution $\hat{v}_p$ (4.29). To approximate the homogeneous solution $\hat{v}_h$, 32 equally distributed collocation points along the interior of the boundary and the same number of source points on a circle with center $(0.5, 0.5)$ and radius $r = 2$ are used. The values $\Delta t = 0.1$ and $\theta = 1$ are also chosen. Profiles of the exact solution and the absolute error of approximation at time $t = 3$ are shown in Figures 5.15 and 5.16, respectively. The absolute maximum error for the approximation, which is based on 81 uniformly distributed test points within the interior domain $[0.1, 0.9]$, is 2.00×10^{-4} .

Figure 5.15: Test Example 2, exact solution at $t = 3$

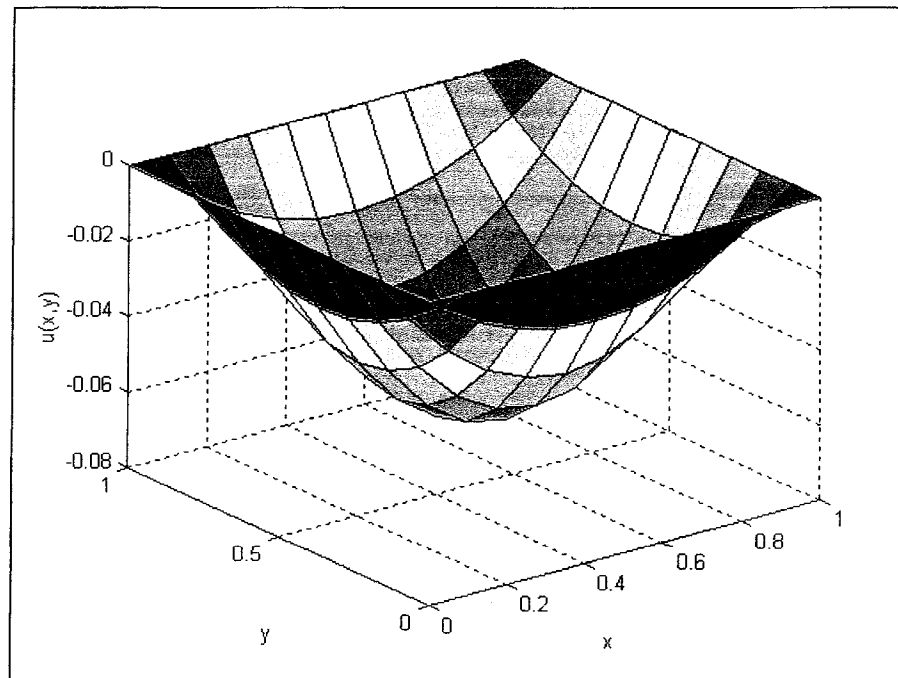
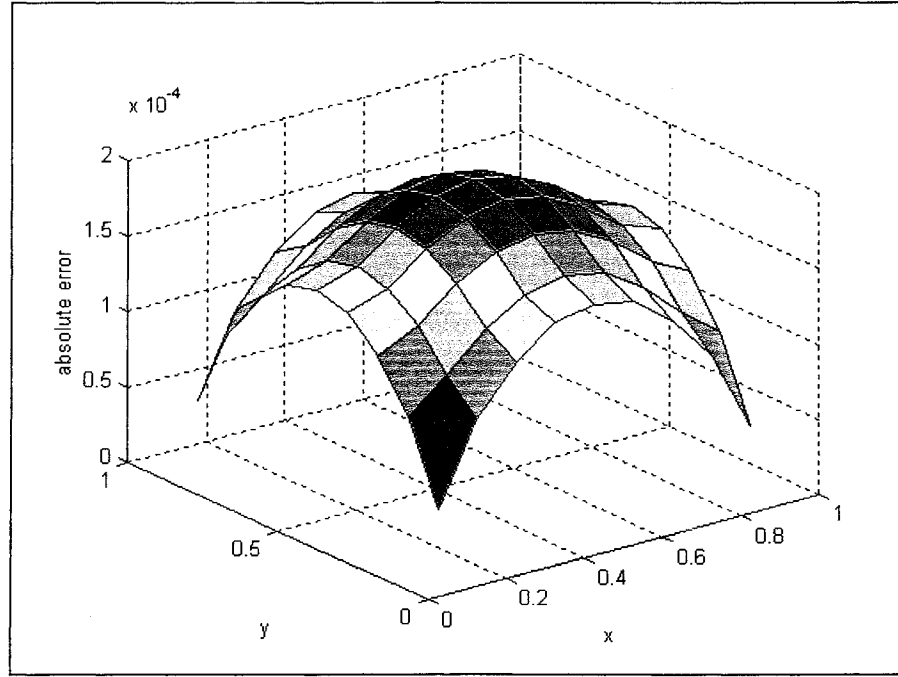


Figure 5.16: Test Example 2, time-stepping scheme ($\Delta t = 0.1$),
absolute error at $t = 3$



For both Test Example 1 and 2, it should be noted that the time-stepping scheme defined in (5.36) utilizes the previous two time-steps u^{n-1} and u^n to compute the next time-step u^{n+1} . Initially, this seems impossible to do. However, from (5.42), the initial condition $u_t = 0$ implies that $u^0 = u^1$.

5.2.2 Observation on Numerical Implementation

A critical observation in the numerical implementation of the time-stepping scheme is that extremely high absolute maximum error (that eventually diverges) can be avoided by forcing the boundary condition $u(x, y, t) = 0$ at each time-step. This is due to the reason that the maximum error normally occurs on the boundary. The ill-conditioning of the MFS tends to magnify the small errors on the boundary at each time-step.

For small values of Δt , errors along the boundary tend to escalate while errors within the interior are still fairly accurate. This creates the illusion of an ill-conditioned, unstable approximation. Consequently, the absolute maximum error will yield errors that are much higher than what they truly are (for small enough Δt , the errors will even diverge), even though it is already known that boundary conditions have a value of zero for Test Examples 1 and 2. Therefore, the approximations of Test Example 1 and 2 are tested on 81 uniformly distributed test points within the interior domain $[0.1, 0.9]$. As noticed in Figures 5.14 and 5.16 (figures of the absolute error of approximation), the boundary is excluded from the graphs.

5.2.3 Analysis of Accuracy

The accuracy of the time-stepping scheme can be improved upon with the correct size of the time-step Δt . Ideally, for smaller time-steps, the accuracy of the approximation increases. This is due to the notion that finite differences are taken over smaller intervals, allowing for closer approximation. To demonstrate the effect of the time-step Δt , let us consider Test Examples 1 and 2. Table 5.3 and Figure 5.17 demonstrate the effect of Δt on Test Example 1, and Table 5.4 and Figure 5.18 display the effect of Δt on Test Example 2. The absolute maximum error is based on 81 uniformly distributed test points within the interior domain $[0.1, 0.9]$ for each time (where $\theta = 1$). The parameter settings are as follows: $n = 121$, $m = 24$, and $r = 1.5$ (for Test Example 1) and $n = 121$, $m = 32$, and $r = 2$ (for Test Example 2).

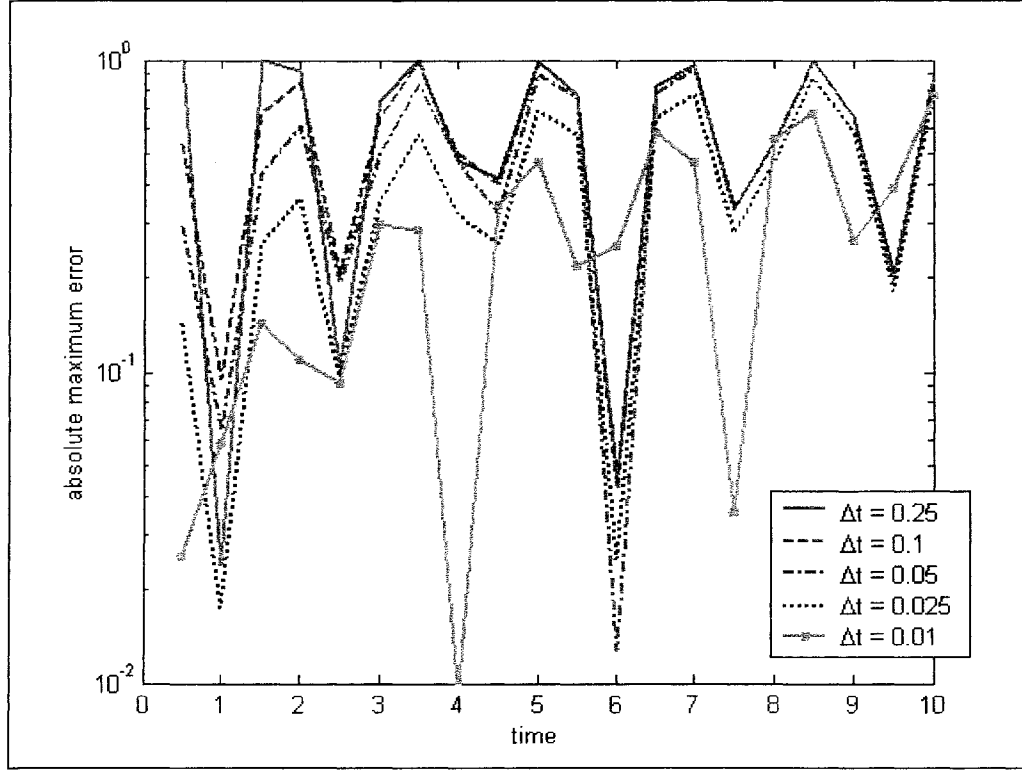
Table 5.3: The effect of Δt on the accuracy of the time-stepping scheme,

Test Example 1, absolute maximum error for $0 < t \leq 10$

time	$\Delta t = 0.25$	$\Delta t = 0.1$	$\Delta t = 0.05$	$\Delta t = 0.025$	$\Delta t = 0.01$
0.5	1.05E+00	5.34E-01	2.92E-01	1.43E-01	2.55E-02
1	2.39E-02	9.45E-02	6.44E-02	1.75E-02	5.89E-02
1.5	9.95E-01	6.71E-01	4.33E-01	2.60E-01	1.43E-01
2	9.13E-01	8.41E-01	6.10E-01	3.61E-01	1.09E-01
2.5	1.03E-01	2.05E-01	1.91E-01	9.68E-02	9.15E-02
3	7.35E-01	6.57E-01	4.98E-01	3.51E-01	2.97E-01
3.5	9.87E-01	9.84E-01	8.31E-01	5.76E-01	2.83E-01
4	4.71E-01	4.98E-01	4.69E-01	3.19E-01	9.20E-03
4.5	4.15E-01	3.97E-01	3.22E-01	2.59E-01	3.41E-01
5	9.75E-01	9.74E-01	8.88E-01	6.88E-01	4.67E-01
5.5	7.67E-01	7.73E-01	7.45E-01	5.78E-01	2.18E-01
6	4.61E-02	4.18E-02	1.27E-02	2.51E-02	2.53E-01
6.5	8.23E-01	8.23E-01	7.79E-01	6.53E-01	5.76E-01
7	9.51E-01	9.52E-01	9.31E-01	7.84E-01	4.66E-01
7.5	3.29E-01	3.30E-01	3.38E-01	2.82E-01	3.50E-02
8	5.52E-01	5.52E-01	5.32E-01	4.73E-01	5.55E-01
8.5	9.98E-01	9.98E-01	9.85E-01	8.76E-01	6.73E-01
9	6.56E-01	6.57E-01	6.57E-01	5.87E-01	2.61E-01
9.5	2.03E-01	2.03E-01	1.94E-01	1.83E-01	3.89E-01
10	9.02E-01	9.02E-01	8.94E-01	8.26E-01	7.69E-01

Figure 5.17: The effect of Δt on the accuracy of the time-stepping scheme,

Test Example 1, absolute maximum error for $0 < t \leq 10$



According to the results, the accuracy of the approximation fluctuates between 10^{-2} and 10^{-1} . Choosing $\Delta t = .01$ would be the best choice when approximating Test Example 1 for $0 < t \leq 10$. Results also show that smaller time-steps offer better accuracy. It is important to note that pushing the time-step to $\Delta t = 0.005$, the approximate solution diverges. This is due to the treatment of the fundamental solution $G(r) = \frac{1}{2\pi} K_0(\lambda r)$, where $\lambda = \frac{1}{\sqrt{\theta(\Delta t)}}$. The modified Bessel function K_0 (of the second kind of order zero) has exponential decay as λr gets larger. Therefore, for small enough time-steps Δt , λ gets larger, and makes the value of $G(r)$ extremely small which causes the resultant

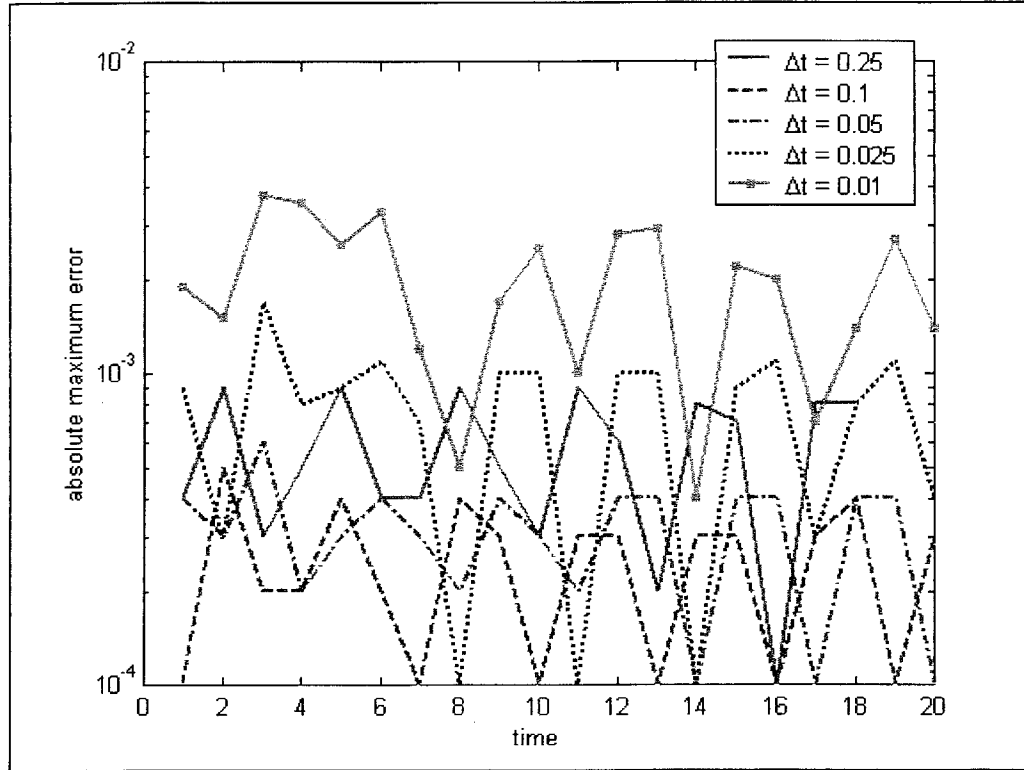
matrix to become very ill-conditioned. Subsequently, approximation through the time-stepping scheme will diverge for Δt sufficiently small.

A similar problem for small time-steps also occurred for Test Example 2. For $\Delta t = 0.001$, the contents of the MFS matrix experienced an “underflow”. Using time-step $\Delta t = 0.005$ yielded absolute maximum errors with degree 10^{-2} or 10^{-1} , but these results are omitted from Table 5.4.

Table 5.4: The effect of Δt on the accuracy of the time-stepping scheme,
Test Example 2, absolute maximum error for $0 < t \leq 20$

time	$\Delta t = 0.25$	$\Delta t = 0.1$	$\Delta t = 0.05$	$\Delta t = 0.025$	$\Delta t = 0.01$
1	4.00E-04	1.00E-04	4.00E-04	9.00E-04	1.90E-03
2	9.00E-04	5.00E-04	3.00E-04	3.00E-04	1.50E-03
3	3.00E-04	2.00E-04	6.00E-04	1.70E-03	3.70E-03
4	5.00E-04	2.00E-04	2.00E-04	8.00E-04	3.50E-03
5	9.00E-04	4.00E-04	3.00E-04	9.00E-04	2.60E-03
6	4.00E-04	2.00E-04	4.00E-04	1.10E-03	3.30E-03
7	4.00E-04	1.00E-04	3.00E-04	7.00E-04	1.20E-03
8	9.00E-04	4.00E-04	2.00E-04	1.00E-04	5.00E-04
9	5.00E-04	3.00E-04	4.00E-04	1.00E-03	1.70E-03
10	3.00E-04	1.00E-04	3.00E-04	1.00E-03	2.50E-03
11	9.00E-04	3.00E-04	2.00E-04	1.00E-04	1.00E-03
12	6.00E-04	3.00E-04	4.00E-04	1.00E-03	2.80E-03
13	2.00E-04	1.00E-04	4.00E-04	1.00E-03	2.90E-03
14	8.00E-04	3.00E-04	1.00E-04	1.00E-04	4.00E-04
15	7.00E-04	3.00E-04	4.00E-04	9.00E-04	2.20E-03
16	1.00E-04	1.00E-04	4.00E-04	1.10E-03	2.00E-03
17	8.00E-04	3.00E-04	1.00E-04	3.00E-04	7.00E-04
18	8.00E-04	4.00E-04	4.00E-04	8.00E-04	1.40E-03
19	0	1.00E-04	4.00E-04	1.10E-03	2.70E-03
20	7.00E-04	3.00E-04	1.00E-04	4.00E-04	1.40E-03

Figure 5.18: The effect of Δt on the accuracy of the time-stepping scheme,
Test Example 2, absolute maximum error for $0 < t \leq 20$



Unlike Test Example 1, smaller time-steps do not necessarily offer better approximation than larger time steps. The results indicate that $\Delta t = 0.1$ is sufficient for the time-step, since it yields the best accuracy for $0 < t \leq 20$. It also seems that the time-stepping scheme works better for the wave equation with a forcing term as opposed to the wave equation without a forcing term; Test Example 2 has better approximation accuracy than Test Example. The presence of the forcing term is the reason for the better accuracy in approximation, and it can be explained through the study of physics.

Let us now consider the trend of the approximation error as time increases. The single test point $(0.5, 0.5)$ will be considered. Using the parameters $n = 121$, $m = 24$, and $r = 1.5$ for Test Example 1, as well as the time-step $\Delta t = 0.01$, the following two figures show the exact solution with the approximate solution (Figure 5.19) and the error of the approximation (Figure 5.20) for $0 < t \leq 10$.

Figure 5.19: Test Example 1, time-stepping scheme ($\Delta t = 0.01$), exact solution and approximate solution of $(0.5, 0.5)$ for $0 < t \leq 10$

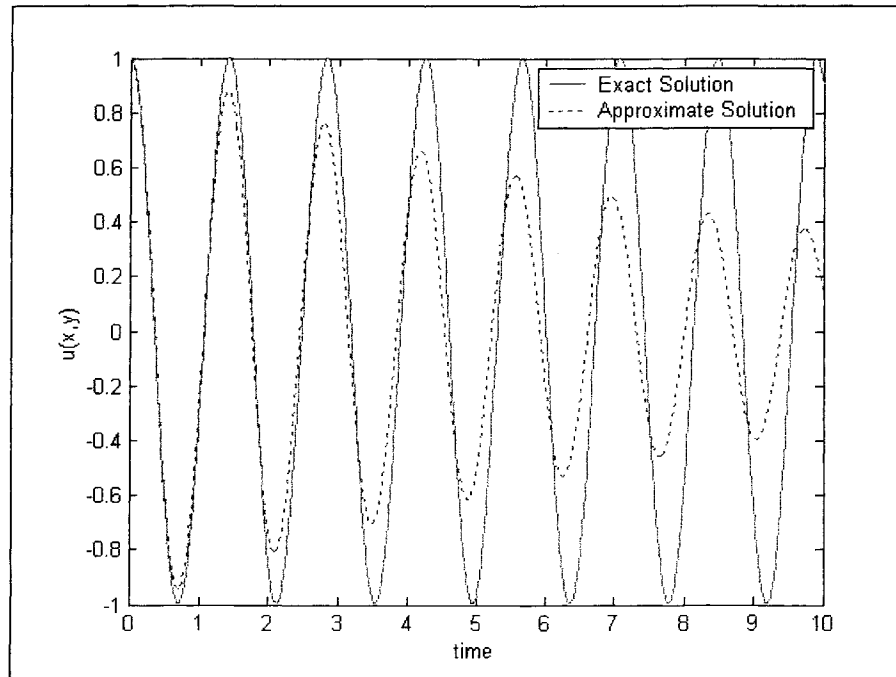
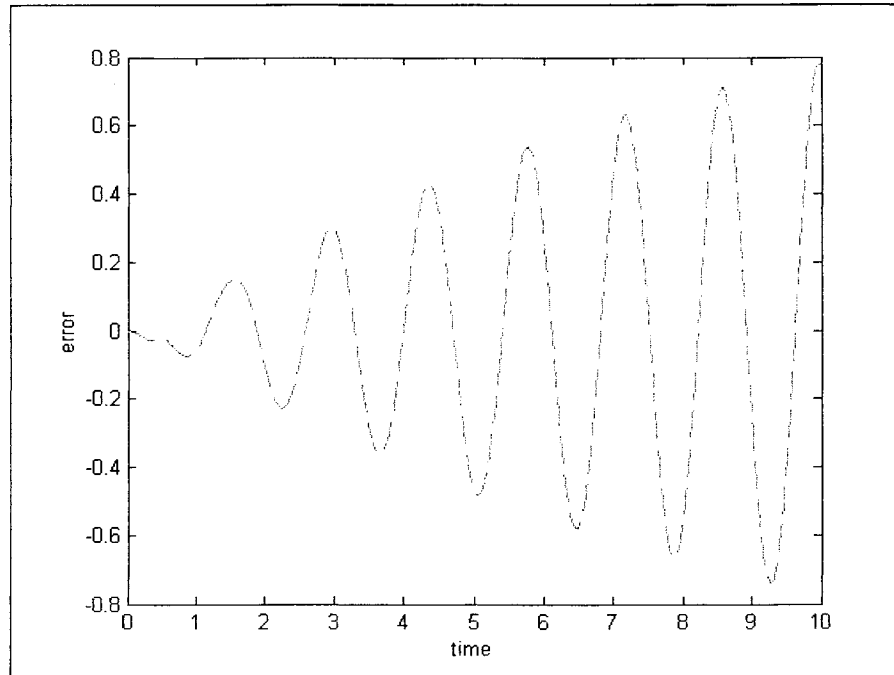


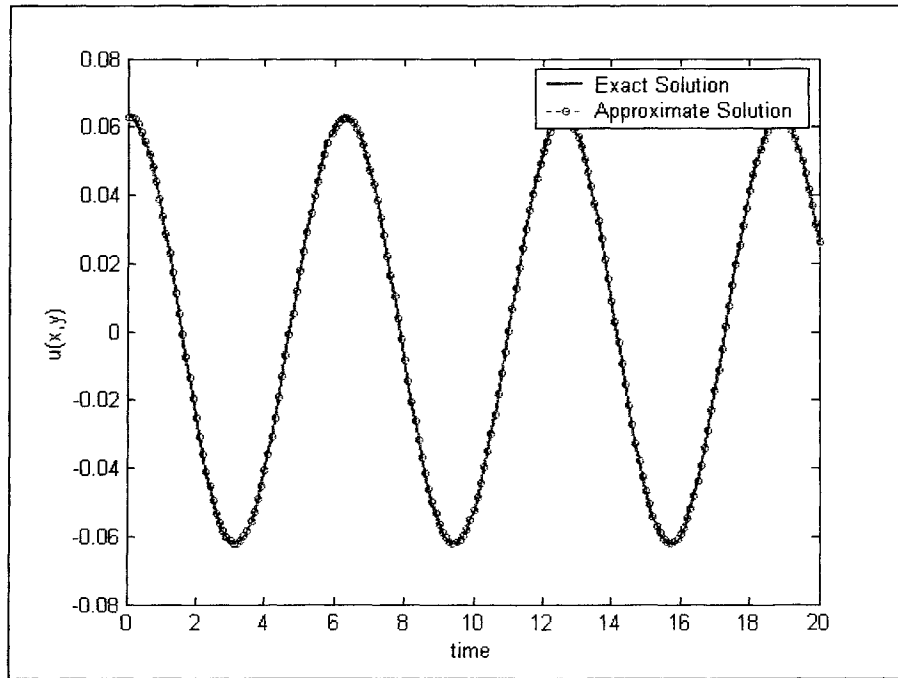
Figure 5.20: Test Example 1, time-stepping scheme ($\Delta t = 0.01$),
error of approximation of $(0.5, 0.5)$ for $0 < t \leq 10$



For this example, results show the error between the approximate solution and the exact solution grows as time increases. The explanation is straightforward. As time increases, error accumulates because approximations at current time-steps rely on the approximations of previous time-steps. At each time-step, an approximation may be erroneous, and that error is used in the computation of subsequent approximations. It is noticeable that when the exact solution of the wave equation has high amplitude, the approximate solution will have the greatest error (especially when the approximation is near zero). For periods of low amplitude, the approximation may appear to be highly accurate, even at larger times.

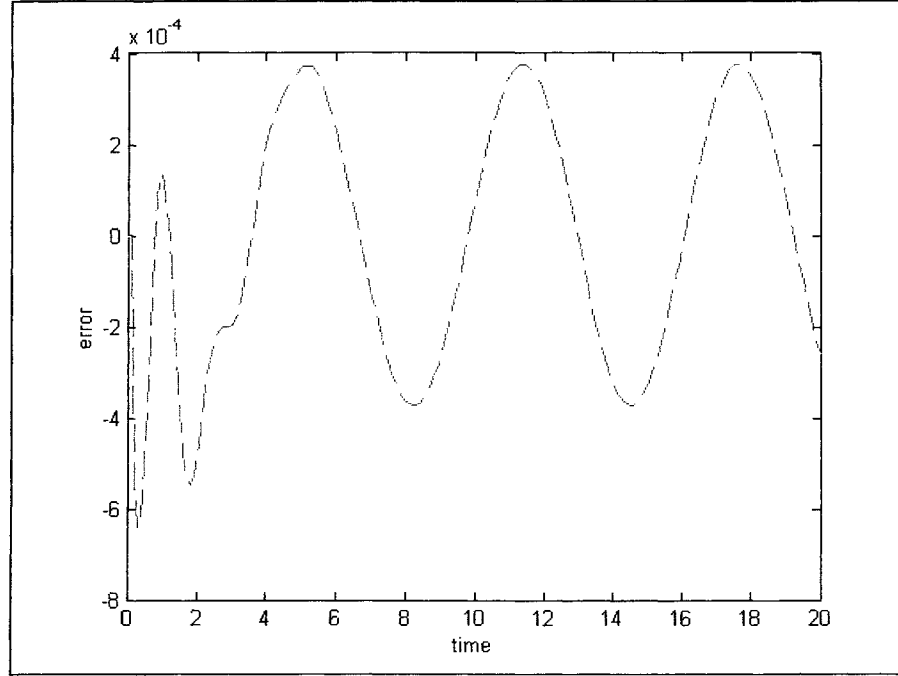
Regarding Test Example 2, the parameters $n = 121$, $m = 32$, and $r = 2$, as well as $\Delta t = 0.1$, are used to show the approximate solution in comparison to the exact solution (Figure 5.21) and the error of approximation (Figure 5.22).

Figure 5.21: Test Example 2, time-stepping scheme ($\Delta t = 0.1$), exact solution and approximate solution of $(0.5, 0.5)$ for $0 < t \leq 20$



For this case, the exact solution and the approximate solution are nearly identical. This can be verified by the results generated in Table 5.4, where the accuracy of approximation has a 10^{-4} degree of accuracy.

Figure 5.22: Test Example 2, time-stepping scheme ($\Delta t = 0.1$),
error of approximation of $(0.5, 0.5)$ for $0 < t \leq 20$.



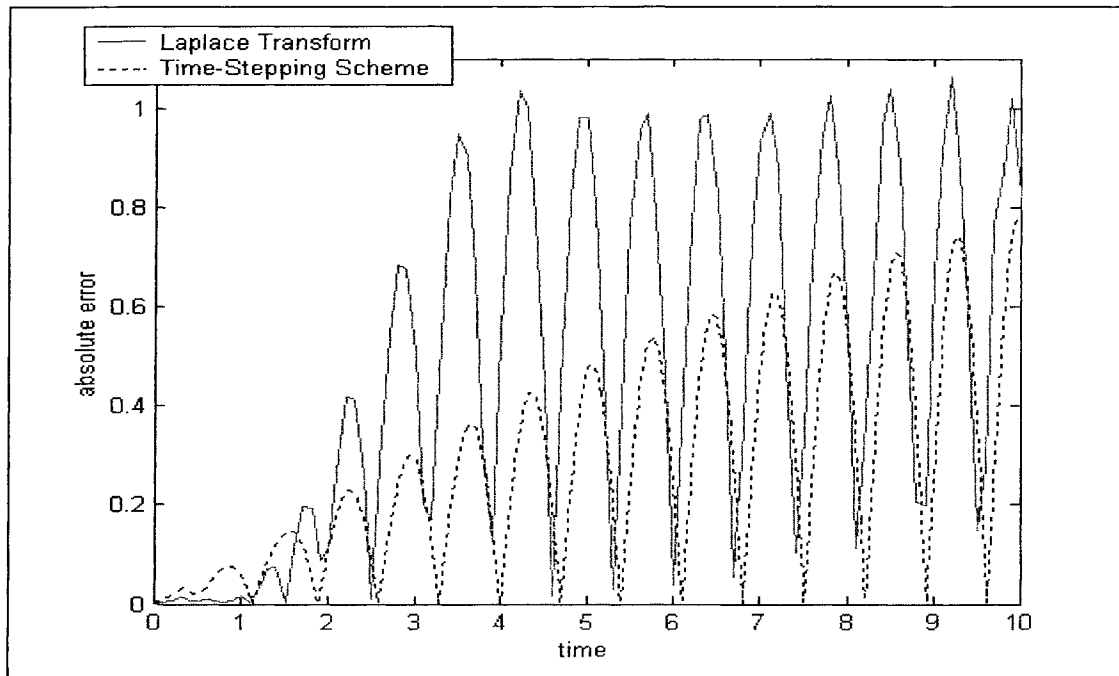
In contrast to the wave equation without a forcing term, there seems to be no drastic accumulation of error as time progresses. This, of course, can be physically explained by the presence of the forcing term.

5.3 Laplace Transform vs. Time-Stepping Scheme

Concerning Test Example 1, it is evident that both the Laplace transform and time-stepping scheme have a general accuracy of 10^{-1} for almost all time. For $t \leq 1$, both methods have accuracy that reaches 10^{-2} . Because of this, neither method is suitable to approximate the wave equation without a forcing term. Nevertheless, let us compare the two methods for the time-frame $0 < t \leq 10$. Below are figures showing the accuracy of approximation for the two methods at the single test point $(0.5, 0.5)$. The parameter

settings for the Laplace transform are: $n = 121$, $m = 24$, $r = 1.5$, and $n_s = 22$. The parameter settings for the time-stepping scheme are: $n = 121$, $m = 24$, $r = 1.5$, $\theta = 1$, and $\Delta t = 0.01$.

Figure 5.23: Laplace transform vs. time-stepping scheme, Test Example 1,
error of approximation of $(0.5, 0.5)$ for $0 < t \leq 10$

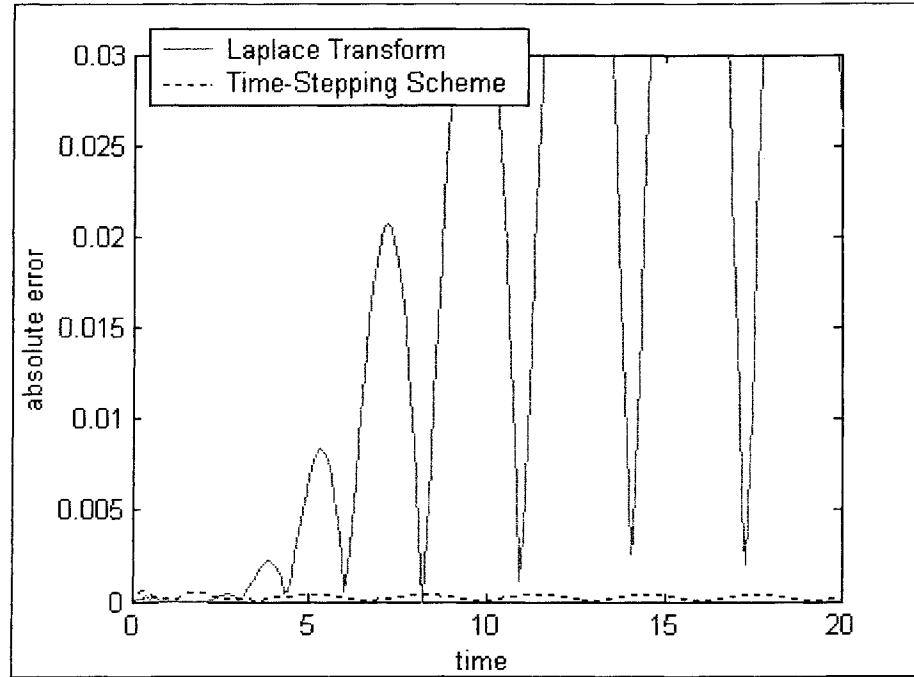


For $0 < t \leq 10$, the Laplace transform seems to start out having better accuracy than the time-stepping scheme. As time progresses, the time-stepping scheme eventually shows better accuracy. Recall, however, both methods generally have the same degree of accuracy throughout time. Furthermore, the absolute error for both methods intensifies as time advances.

Regarding Test Example 2, it is obvious that the time-stepping scheme yields much higher accuracy than the Laplace transform (refer to Tables 5.2 and 5.4). The time-

stepping scheme produces stable solutions at a steady 10^{-4} degree of accuracy for $0 < t \leq 20$. Meanwhile, the Laplace transform has suitable accuracy for very small time, but rises to 10^{-2} accuracy afterwards. The time-stepping scheme is highly preferred for the approximation of the wave equation with a forcing term. A visual comparison of the two methods is provided in Figure 5.24. The parameter settings for the Laplace transform are: $n = 121$, $m = 32$, $r = 2$, and $n_s = 18$. The parameter settings for the time-stepping scheme are: $n = 121$, $m = 32$, $r = 2$, $\theta = 1$, and $\Delta t = 0.1$.

Figure 5.24: Laplace transform vs. time-stepping scheme, Test Example 2,
error of approximation of $(0.5, 0.5)$ for $0 < t \leq 20$



When addressing the computational efficiency of the Laplace transform and the time-stepping scheme, the Laplace transform is favorable. The Laplace transform can quickly

and directly compute approximations for any given time. The time-stepping scheme, however, must begin at time $t = 0$ and use the approximations from several previous time-steps Δt in order to compute the approximation for a given time. As observed in Figures 5.23 and 5.24, the Laplace transform yields better accuracy than the time-stepping scheme for a very small time-frame. If the Laplace transform were to remain at a stable degree of accuracy throughout time, this method could be a more powerful approach than the time-stepping scheme, especially since the Laplace transform is more computationally efficient.

5.4 Concluding Remarks

The Laplace transform and the time-stepping scheme both have their advantages and disadvantages. The Laplace transform can prove to be advantageous to wave approximation provided that the appropriate parameters are selected. However, it may be a time-consuming effort to render the necessary preliminary testing to determine the suitable value of n_s . The proper amount of interpolation points, source points, and size of fictitious radius may also be considered when rendering such preliminary tests. Nevertheless, these parameters do not place a significant impact on the accuracy of the approximation when the right n_s is chosen.

The disadvantages of the Laplace transform are found in the sensitivity of the value of n_s at certain times. For Test Example 1 and 2, n_s varied for certain times. Furthermore, for examples, the Laplace transform generated sufficient accuracy for very small time. But, for larger time, the Laplace transform was a rather unfavorable method. This is especially so when dealing with wave equations without forcing terms. The troubles of

the Laplace transform lie in the ill-posed condition of the inverse Laplace transform. Small truncation errors can be greatly magnified in the numerical inversion process, and thus leading to poor numerical results [6].

The time-stepping scheme is excellent for approximating a wave equation with a forcing term. The opposite is true when approximating a wave equation with no forcing term. The troubles of this scheme lie in the exponential decay of the fundamental solution

$G(r) = \frac{1}{2\pi} K_0(\lambda r)$ in conjunction with the size of the time step Δt . Improvements to this

scheme can be made by appropriately choosing the size of the time-step. Another downfall to this method is the continual accumulation of error as time increases. This cannot be avoided, especially with the absence of a forcing term.

CHAPTER 6

CONCLUSIONS AND FUTURE RESEARCH

This thesis focused on the method of fundamental solutions for solving wave equations. Two wave equation examples were selected for testing under the analysis of two proposed approaches: the Laplace transform and the time-stepping scheme. Both approaches eliminated the time-dependency of the wave equations and treated the resultant series of inhomogeneous modified Helmholtz equations through separate algorithms.

6.1 Conclusions

The Laplace transform and the time-stepping scheme are equally valuable methods for the solution of the wave equation. For the two selected test examples in this thesis, however, one method performed better. Regarding Test Example 1 (the wave equation without a forcing term), both the Laplace transform and time-stepping scheme produced approximate solutions that only had a general degree of accuracy 10^{-1} ; this is not a favorable result. In addition, both methods produced increasing error as time increased. Concerning Test Example 2 (the wave equation with a forcing term), the time-stepping scheme was definitely the better approach. Approximate solutions were stable and highly accurate throughout time. On the other hand, the Laplace transform had

a similar performance to the case in Test Example 1; it yielded fairly minimal error for a very small time and growing error for increasing time.

6.2 Recommendations for Future Research

This thesis serves as a stepping stone toward a more improved solution to wave equations. When dealing with the Laplace transform for the solution of wave equations, the main challenge is to overcome the sensitivity and ill-posed condition of the inverse Laplace transform. This subject is an excellent topic for future research. An improvement to the Laplace transform can be made if a more stable inverse Laplace transform is available. An additional improvement to the Laplace transform is to find a way to extend the high accuracy within small time-frames to larger time-frames. As discussed in Chapter 5, the Laplace transform is preferable to the time-stepping scheme for very small times. However, accuracy of the Laplace rapidly declines as time grows.

Another recommendation for research is the implementation of a different scheme to remove the time dependency of the wave equation. Both the Laplace transform and the time-stepping scheme dealt with a series of inhomogeneous modified Helmholtz equations. Rather than working under the modified Helmholtz operator, the time-dependency can be reduced to a series of inhomogeneous Helmholtz equations. For this case, the fundamental solution would be $G(r) = -\frac{i}{4} H_0^{(2)}(\lambda r)$, which is the Hankel function of the second kind of order zero. The wave equation could then be treated under the Fourier transform and inverse Fourier transform instead.

Finally, the findings and methods discussed in this thesis can be extended to other time-dependent problems. Currently, there is extensive literature on the solution of the

heat equation through the Laplace transform and the time-stepping scheme [6, 15, 18]. However, more complicated wave problems (with mixed boundary conditions, initial conditions, or irregularly shaped domains) can be solved through the approaches mentioned herein. Higher-dimensional time-dependent problems or nonlinear PDEs can be solved through similar approaches as well.

REFERENCES

- [1] Atkinson, K.E. The numerical evaluation of particular solutions for Poisson's equation. *IMA Journal of Numerical Analysis*, 5: 319-338, 1985.
- [2] Bogomolny, A. Fundamental solutions method for elliptic boundary value problems. *SIAM Journal on Numerical Analysis*, 22: 644-669, 1985.
- [3] Chen, C.S. Meshless Methods for Scientific Computing, presentation, 2004.
- [4] Chen, C.S. and C.A. Brebbia. Dual reciprocity method using compactly supported radial basis functions. *Communications in Numerical Methods in Engineering*, 15: 137-150, 1999.
- [5] Chen, C.S., M.A. Golberg, and Y.C. Hon. The method of fundamental solutions and quasi-Monte Carlo method for diffusion equations. *International Journal of Numerical Methods in Engineering*, 43: 1421-1436, 1998.
- [6] Chen, C.S., M.A. Golberg, and Y.F. Rashed. A mesh free method for linear diffusion equations. *Numerical Heat Transfer, Part B*, 33: 469-486, 1998.
- [7] Chen, C.S. and Y.F. Rashed. Evaluation of thin plate spline based particular solutions for Helmholtz-type operators for the DRM. *Mechanics Research Communications*, 25(2): 195-201, 1998.
- [8] Cheng, A.H.-D., Y. Abousleiman, and T. Badmus. A Laplace transform BEM for axisymmetric diffusion utilizing pre-tabulated Green's function. *Engineering Analysis with Boundary Elements*, 9: 39-46, 1992.
- [9] Cheng, R.S.C. *Delta-Trigonometric and Spline Methods using the Single-Layer Potential Representation*, Ph.D. dissertation, University of Maryland, 1987.
- [10] Cho, H.A., C.S. Chen, and M.A. Golberg. Some comments on mitigating the ill-conditioning of the method of fundamental solutions, submitted.
- [11] Cho, H.A., M.A. Golberg, A.S. Muleshkov, and X. Li. Trefftz methods for time dependent partial differential equations. *Computers, Materials, and Continua*, 1: 1-37, 2004.

- [12] Duchon, J. Splines minimizing rotation invariant semi-norm in Sobolev spaces. *Constructive Theory of Functions of Several Variables, Lecture Notes in Mathematics 571*, ed. W. Schempp & K. Zeller, Springer-Verlag, Berlin, 85-110.
- [13] Fairweather and A. Karageorghis. The method of fundamental solutions for elliptic boundary value problems. *Advances in Computational Mathematics*, 9: 69-95, 1998.
- [14] Frank, R. Scattered data interpolation: tests of some methods. *Mathematics of Computation*, 48: 181-200, 1982.
- [15] Golberg, M.A. and C.S. Chen. The method of fundamental solutions for potential, Helmholtz and diffusion problems. *Boundary Integral Methods – Numerical and Mathematical Aspects*, ed. M.A. Golberg, Computational Mechanics Publications, 103-176, 1998.
- [16] Golberg, M.A. and C.S. Chen. The theory of radial basis functions applied to the BEM for inhomogeneous partial differential equation. *Boundary Elements Communications*, 5(2): 57-61, 1994.
- [17] Golberg, M.A., C.S. Chen, and S.R. Karur. Improved multiquadric interpolation for partial differential equations. *Engineering Analysis with Boundary Elements*, 18: 9-17, 1996.
- [18] Golberg, M.A., C.S. Chen, and A.S. Muleshkov. The method of fundamental solutions for time-dependent problems. *Boundary Element Technology XIII*, eds. C.S. Chen, C.A. Brebbia, and D.W. Pepper, Boston, Southampton, 377-386, 1999.
- [19] Hardy, R.L. Multiquadric equations of topography and other irregular surfaces. *Journal of Geophysical Research*, 176: 1905-1915, 1971.
- [20] Katsurada, M. Asymptotic error analysis of the charge simulation method in a Jordan region with an analytic boundary. *Journal of the Faculty of Science of Tokyo University, Section 1A*, 37:635-657, 1990.
- [21] Katsurada, M. and H. Okamoto. A mathematical study of the charge simulation method. *Journal of the Faculty of Science, University of Tokyo, Section 1A*, 35: 507-518, 1988.
- [22] Kitagawa, T. On the numerical stability of the method of fundamental solutions applied to the Dirichlet problem. *Japan Journal of Applied Mathematics*, 35: 507-518, 1988.

- [23] Kupradze, V.D. and M.A. Aleksidze. A method for the approximate solution of limiting problems in mathematical physics. *U.S.S.R. Computational Mathematics and Mathematical Physics*, 4: 199-205, 1964.
- [24] Kupradze, V.D. and M.A. Aleksidze. The method of functional equations for the approximate solution of certain boundary value problems. *U.S.S.R. Computational Mathematics and Mathematical Physics*, 4: 82-126, 1964.
- [25] Muleshkov, A.S., M.A. Golberg, and C.S. Chen. Particular solutions of Helmholtz-type operators using higher order polyharmonic splines. *Computational Mechanics*, 24: 411-419, 1999.
- [26] Nardini, D. and C.A. Brebbia. A new approach to free vibration analysis using boundary elements. *Boundary Element Methods in Engineering*, ed. C.A. Brebbia. Springer-Verlag, Berlin, 1982.
- [27] Partridge, P.W., C.A. Brebbia, and L.C. Wrobel. *The Dual Reciprocity Element Method*. Computation Mechanics Publications, Southampton and Elsevier, London, 1992.
- [28] Schaback, R. Error estimates and condition numbers for radial basis function interpolation. *Advances in Computational Mathematics*, 3(3): 251-264, 1995.
- [29] Stehfest, H. Algorithm 368: numerical inversion of Laplace transform. *Commun. ACM*, 13: 47-49, 1970.
- [30] Su, J. and B. Tabarrok. A time-marching integral equations method for unsteady state problems. *Computer Methods in Applied Mechanics and Engineering*, 142: 203-214, 1997.
- [31] Wendland, H. Piecewise polynomial, positive definite and compactly supported radial basis functions of minimal degree. *Advances in Computational Mathematics*, 4(4): 389-396, 1995.
- [32] Wu, Z. Multivariate compactly supported positive definite radial functions. *Advances in Computational Mathematics*, 4: 283-292, 1995.
- [33] Zhu, S., P. Satravaha, and X. Lu. Solving linear diffusion equations with the Dual Reciprocity Method in Laplace space. *Engineering Analysis with Boundary Elements*, 13: 1-10, 1994.

VITA

Graduate College
University of Nevada, Las Vegas

Lourdes Jeanette Rivera

Local Address:

5280 W. Hacienda Ave. #2024
Las Vegas, Nevada 89118

Home Address:

P.O. Box 4639 AAFB
Yigo, Guam 96929

Degrees:

Bachelor of Arts, Mathematics and Secondary Education, 2003
University of Guam

Special Honors and Awards:

Nevada Space Grant, 2005
Chancellor's List, 2004-2005
National Dean's List, 2003-2004
Magna Cum Laude, Spring 2003

Thesis Title:

The Method of Fundamental Solutions for Solving Wave Equations

Thesis Examination Committee:

Chairperson, Dr. C. S. Chen, Ph. D.
Committee Member, Dr. Zhonghai Ding, Ph. D.
Committee Member, Dr. Xin Li, Ph. D.
Graduate Faculty Representative, Dr. Tao Pang, Ph. D.