THE METHOD OF FUNDAMENTAL SOLUTIONS FOR SOME DIRECT AND INVERSE PROBLEMS

by

THOMAS HENRY REEVE

A thesis submitted to
University of Birmingham
for the degree of
DOCTOR OF PHILOSOPHY

School of Mathematics
Engineering and Physical Sciences
University of Birmingham
June 2013

Abstract

We propose and investigate applications of the method of fundamental solutions (MFS) to several parabolic time-dependent direct and inverse heat conduction problems (IHCP). In particular, the two-dimensional heat conduction problem, the backward heat conduction problem (BHCP), the two-dimensional Cauchy problem, radially symmetric and axisymmetric BHCPs, the radially symmetric IHCP, inverse one and two-phase linear Stefan problems, the inverse Cauchy-Stefan problem, and the inverse two-phase one-dimensional nonlinear Stefan problem. The MFS is a collocation method therefore it does not require mesh generation or integration over the solution boundary, making it suitable for solving inverse problems, like the BHCP, an ill-posed problem. We extend the MFS proposed in Johansson and Lesnic (2008) for the direct one-dimensional heat equation, and Johansson and Lesnic (2009) for the direct one-phase one-dimensional Stefan problem, with source points placed outside the space domain of interest and in time. Theoretical properties, including linear independence and denseness, the placement of source points, and numerical investigations are included showing that accurate results can be efficiently obtained with small computational cost. Regularization techniques, in particular, Tikhonov regularization, in conjunction with the L-curve criterion, are used to solve the ill-conditioned systems generated by this method. In Chapters 6 and 8, investigating the linear and nonlinear Stefan problems, the MATLAB toolbox *lsqnonlin*, which is designed to minimize a sum of squares, is used.

ACKNOWLEDGEMENTS

This work is the culmination of over four years work (MSci and PhD) which was accomplished with my supervisor Tomas Johansson. I thank Tomas for his support and encouragement through the years, as well as being a genuine and kind person. I am also grateful to Daniel Lesnic in Leeds for his significant contributions and vast insight in the papers we have published together. I am grateful to my parents, family, and friends in the School of Mathematics at the University of Birmingham for their advice, help and ping pong sessions. Lastly, I very much appreciate the EPSRC and school funding I have received.

OUTCOME OF SCIENTIFIC WORK

This thesis is the culmination of seven published papers and two proceedings, listed below, as well as two submitted papers. The work is a result of collaboration and supervision by Dr B.T. Johansson (supervisor) and Professor D. Lesnic (University of Leeds).

- B.T. Johansson, D. Lesnic and T. Reeve. A method of fundamental solutions for two-dimensional heat conduction. *Int. J. Comput. Math.,* 88:1697–1713, 2011.

- T. Reeve, B. T. Johansson and D. Lesnic. The method of fundamental solutions for inverse Stefan problems, The Fifth International Conference *Inverse Problems: Modelling and Simulation* (Eds. A. Hasanov, B. Hofmann and S. I. Kabanikhin), Antalya, Turkey, (2010), 110–111.

- B.T. Johansson, D. Lesnic and T. Reeve. A comparative study on applying the method of fundamental solutions to the backward heat conduction problem. *Math. Comput. Modelling*, 54:403–416, 2011.

- T. Reeve and B.T. Johansson. The method of fundamental solutions for a time-dependent two-dimensional Cauchy heat conduction problem. *Eng. Anal. Boundary Elements*, 37:569–578 ,2013.

- B.T. Johansson, D. Lesnic and T. Reeve. A method of fundamental solutions for radially symmetric backward heat conduction problems. *Advances in Boundary Integral Methods - Proceedings of the Eight UK Conference on Boundary Integral Methods* (Eds. D. Lesnic), University of Leeds, UK, (2011), 9–16.

- B.T. Johansson, D. Lesnic and T. Reeve. A method of fundamental solutions for radially symmetric and axisymmetric backward heat conduction problems. *Int. J. Comput. Math.*, 89:1555–1568, 2012.

- B.T. Johansson, D. Lesnic and T. Reeve. A method of fundamental solutions for the radially symmetric inverse heat conduction problem. *Int. Commun. Heat Mass,* 39:887–895, 2012.

- B.T. Johansson, D. Lesnic and T. Reeve. A method of fundamental solutions for the one-dimensional inverse Stefan problem. *Appl. Math. Model. (AMM),* 35:4367–4378, 2011.

- B.T. Johansson, D. Lesnic and T. Reeve. A method of fundamental solutions for the two-dimensional inverse Stefan problem. Submitted to *Inverse Problems Sci. Eng.*

- B.T. Johansson, D. Lesnic and T. Reeve. Numerical approximation of the one-dimensional inverse Cauchy-Stefan problem using a method of fundamental solutions. *Inverse Problems Sci. Eng.*, 19:659–677, 2011.

- B.T. Johansson, D. Lesnic and T. Reeve. A meshless method for an inverse two-phase one-dimensional nonlinear Stefan problem. Revision submitted to *Math. Comput. Simulation.*

This thesis also contains a reference to a paper by the author, in collaboration with Dr B.T. Johansson and Professor D. Lesnic.

- B.T. Johansson, D. Lesnic and T. Reeve. A meshless method for an inverse two-phase one-dimensional linear Stefan problem. *Inverse Problems Sci. Eng.*, 21(1):17–33, 2013.

Additional papers I have also contributed to during my PhD, in collaboration with Dr B.T. Johansson, Professor D. Lesnic and Professor D.J. Needham.

- B.T. Johansson, D. Lesnic and T. Reeve. A meshless regularization method for a two-dimensional two-phase linear inverse Stefan problem. Revision submitted to *Adv. Appl. Math. Mech.*

- D.J. Needham, B.T. Johansson and T. Reeve. The development of a wax layer on the interior wall of a circular pipe transporting heated oil. Submitted to *Quart. J. Mech. Appl. Math.*

CONTENTS

GLOSSARY OF SYMBOLS AND NOTATION

Notation	Description
$\mathbb{R}$	Set of all real numbers
$\mathbb{R}^n$	Euclidean space
$\mathbb{C}$	Set of all complex numbers
$\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$	Column vector of dimension n, also a spatial variable
$\mathbf{x}_k, \mathbf{y}_j$	Discrete spatial coordinates
t, τ	Time variables
t_i, τ_m	Discrete time coordinates
(r, θ)	Polar coordinates
(r, θ, z)	Cylindrical coordinates
r_0, R	Location of inner and outer boundary in radial problems
$\|\cdot\|$	Vector norm
$\|\mathbf{x}\| = (\sum_{i=1}^n x_i^2)^{\frac{1}{2}}$	2-norm of a vector $\mathbf{x}$
A	Matrix
A_{ij}	Elements of a matrix A
A^{-1}	Inverse of a matrix A
A^T	Transpose of a matrix A
$A^\dagger$	Moore-Penrose pseudoinverse of a real matrix A
I	The identity matrix
$\mathbb{R}^{M,N}$	Set of real matrices of size $M \times N$
$\langle f, g\rangle = \int_a^b fg\,dx$	Inner product for the vector space of real functions over the closed interval $[a, b]$
$J_n(x)$	Bessel function of first kind of order n
$I_n(x)$	Modified Bessel function of first kind of order n
PDE	Partial differential equation
$\mathcal{L}$	Linear differential operator
$\frac{\partial}{\partial x}$, ∂_x	Partial derivative with respect to x
∇	$(\partial_{x_1}, \partial_{x_2}, \ldots, \partial_{x_n})$
Δ	$\partial_{x_1}^2 + \partial_{x_2}^2 + \ldots + \partial_{x_n}^2$
$\frac{\partial u}{\partial t} = \alpha\Delta u$	The heat equation
u, u_1, u_2	Temperature
$\alpha, \alpha_1, \alpha_2$	Thermal diffusivity

Notation	Description
$s(t), s(y,t)$	Free boundary used in Stefan problems
k_i	Thermal conductivity
ρ_i	Density
c_i	Heat capacity
L	Latent heat
BHCP	Backward heat conduction problem
IHCP	Inverse heat conduction problem
FEM	Finite element method
BEM	Boundary element method
FDM	Finite difference method
RBF	Radial basis function
MFS	Method of fundamental solutions
$\delta(x)$	The Dirac delta function
$H(x)$	The Heaviside function
$F(\mathbf{x}, \mathbf{y})$	Fundamental solution with source point $\mathbf{y}$
$F(\mathbf{x}, t; \mathbf{y}, \tau)$	Fundamental solution with source point $(\mathbf{y}, \tau)$
F_1, F_2	Fundamental solutions of radially symmetric and axisymmetric heat equations
$u_M, u_{M,N}$	MFS approximation of the temperature
$c_m^{(j)}$	Constant coefficients appearing in MFS approximation constructed as a linear combination of fundamental solutions
$\mathbf{c}$	Vector of constant coefficients $c_m^{(j)}$
$\mathbf{g}$	Vector of boundary values at collocation points
h	Measure of the distance source points are located from the domain boundary
Dynamic MFS	Requires the determination of the source point locations as well as $\mathbf{c}$ when applying the MFS
Static MFS	Position of source points are fixed when applying the MFS
$V(\mathbf{y}, \tau)$	Single-layer potential
$K(\mathbf{y}, \tau)$	Double-layer potential
D	Solution domain
$\overline{D}$	Closure of D
Γ, ∂D	Boundary of the domain D

Notation	Description
$\boldsymbol{\nu}$	Outward pointing unit normal to Γ
D_E	Open domain containing $\overline{D}$
Γ_E	Boundary of D_E on which source points are placed
T	Final time point
D_T, Γ_T	$D \times (0,T]$ and $\Gamma \times (0,T]$
$u_0(\mathbf{x})$, $u_T(\mathbf{x})$	Temperature data at the initial and final time points
$C^0(D)$	The space of continuous functions on D
$C^j(D)$	The space of functions with all derivatives continuous up to order j on D
$C^{j,k}(D)$	The space of functions with all derivatives continuous up to order j for the variable $\mathbf{x}$ (for example), and order k for another variable $\mathbf{t}$ on D
$L^2(D)$	The space of square-integrable functions on D
H^2_{loc}	Local Sobolev space of order 2
λ	The Tikhonov regularization parameter
L_k	Matrix operators constructed as approximations of derivatives ($k=0$ continuity, $k=1$ first order smoothness, etc.)
$N(\mu, \sigma^2)$	Normal distribution with mean μ and standard deviation σ
$\text{erf}(\xi) = \frac{2}{\sqrt{\pi}} \int_0^\xi e^{-\sigma^2}\, d\sigma$	The error function
$\text{erfc}(\xi) = 1 - \text{erf}(\xi)$	The complementary error function
δ, p_n	Relative noise level, percentage of random noise
u^δ, u_x^δ	Boundary data (temperature, flux) containing measurement errors
rand	Uniformly distributed pseudo-random numbers generated in MATLAB on the interval $(0,1)$
randn	Normally distributed pseudo-random numbers generated in MATLAB drawn from the standard normal distribution
RMSE	Root mean square error
RRMSE	Relative root mean square error

GENERAL DESCRIPTION OF THE METHOD OF FUNDAMENTAL SOLUTIONS

We begin this thesis by giving a brief description of the method of fundamental solutions (MFS), so that potential users can get a flavour for the method.

The MFS is a numerical method for solving partial differential equations (PDEs) when the fundamental solution of the PDE is known. To illustrate how we could implement the MFS, assume we have a bounded domain D, with boundary Γ, and a PDE, which we denote by

$$\mathcal{L}(u) = 0, \tag{1}$$

where $\mathcal{L}$ is a linear differential operator with constant coefficients and $u : \mathbb{R}^n \to \mathbb{R}$. A typical problem might involve recovering information, including finding an approximation to the solution u in the domain of interest, unknown boundary data, etc., from data given on the boundary Γ (or a part of the boundary). We write the boundary condition as

$$\mathcal{B}u(\mathbf{x}) = f(\mathbf{x}), \quad \mathbf{x} \in \Gamma, \tag{2}$$

where $f : \mathbb{R}^n \to \mathbb{R}$. Here, the boundary condition may be:

- Dirichlet: $u(\mathbf{x}) =$ known for $\mathbf{x} \in \Gamma$,
- Neumann: $\frac{du}{d\boldsymbol{\nu}}(\mathbf{x}) =$ known for $\mathbf{x} \in \Gamma$,
- Robin: $\frac{du}{d\boldsymbol{\nu}}(\mathbf{x}) + \alpha u(\mathbf{x}) =$ known for α being a constant and $\mathbf{x} \in \Gamma$,

with $\boldsymbol{\nu}$ being the outward unit normal to the boundary. We mainly concentrate on the first two conditions in the problems that we study. However, more complicated boundary conditions such as prescribed derivatives at certain angles on the boundary (oblique conditions) and different types on different parts of the boundary (mixed conditions) can be handled as well. In the MFS we construct an approximation which satisfies both the governing PDE and the boundary condition (at a finite number of points). The fundamental solution $F(\mathbf{x}, \mathbf{y})$ of equation (1) satisfies

$$\mathcal{L}(F) = \delta(\mathbf{x} - \mathbf{y}), \tag{3}$$

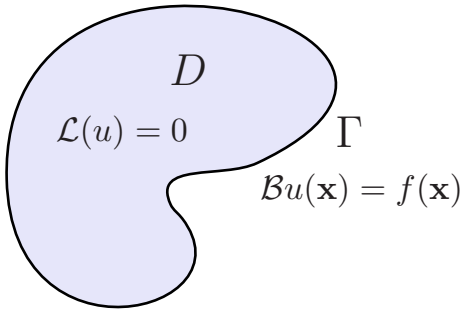

Figure 1: Representation of an example domain D, boundary Γ, and equations (1) and (2).

where δ is the Dirac delta function, and $\mathbf{y}$ is called a source point. The Dirac delta function can be thought (informally) as having infinite value at the point $\mathbf{y}$ (a singularity) and taking the value zero everywhere else, i.e.

$$\delta(\mathbf{x}-\mathbf{y}) = \begin{cases} \infty, & \mathbf{x} = \mathbf{y}, \\ 0, & \mathbf{x} \neq \mathbf{y}, \end{cases}$$

with

$$\int_{\mathbb{R}^n} \delta(\mathbf{x}-\mathbf{y})\, d\mathbf{x} = 1,$$

which makes it a unit source. If the source point $\mathbf{y}$ is placed outside the domain D then the fundamental solution $F(\mathbf{x}, \mathbf{y})$ (and any linear combination of fundamental solutions) satisfies equation (1). Hence, we construct an approximation to (1) in the form

$$u_N(\mathbf{x}) = \sum_{j=1}^{N} c_j F(\mathbf{x}, \mathbf{y}_j),$$

where the $\mathbf{y}_j$ represent N source points placed outside the domain D, see Figure 2, and c_j are constant coefficients.

We then need to determine the constant coefficients c_j, which is accomplished by collocation (imposing the boundary condition (2)) at a finite number of points (called collocation points). If we place M collocation points on the boundary, see Figure 2, then we obtain a system with M equations and N unknowns. In the best case scenario we can solve this problem using Gaussian elimination, however, the MFS often generates matrices which are highly ill-conditioned, and regularization needs to be incorporated.

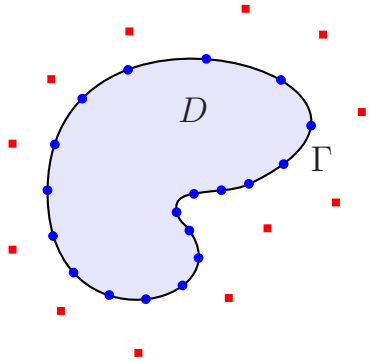

Figure 2: Representation of a possible placement of source (■) and collocation points (●).

The main advantages and disadvantages of the MFS are listed below.

Advantages:

- The method is relatively easy to program and computationally inexpensive.
- The MFS is a collocation method, therefore no complicated meshes need to be generated.
- It has produced accurate and stable results (in conjunction with regularization techniques) for different types of problems (both direct and inverse) including elliptic, time-dependent parabolic, free boundary, coefficient and boundary inverse problems, etc.

Disadvantages:

- The MFS can only be applied when the fundamental solution of the governing linear PDE is known.
- The position and number of source and collocation points can affect the accuracy greatly, it is possible to solve a nonlinear minimization problem to determine the position of the source points, however, this can significantly increase the computation time.

This thesis will focus on investigating and implementing the MFS for various heat conduction type problems.

CHAPTER 0

INTRODUCTION

Solving partial differential equations (PDEs), which are used to model real world problems, is an important area of research in many disciplines of mathematics, physics, chemistry, etc. The focus of this thesis will be the heat equation, which models many engineering applications, from heat exchangers, mathematical finance (due to the relation with the Black-Scholes equation), and various chemical and biological systems, including diffusion and transportation problems. For a derivation of the heat equation, which can be found in most introductory books on partial differential equations, see, for example, [111]. Various numerical techniques have been used to solve heat conduction problems, however, there are certain limitations with some of these methods, in particular when applied to inverse problems, that we wish to overcome, some of which are detailed in Section 0.4.

We apply a numerical technique, known as the method of fundamental solutions (MFS), which has been used to produce accurate numerical solutions to linear PDEs, with little computational effort, see the reviews [36, 40]. The MFS has predominantly been applied to stationary heat flow problems governed by elliptic PDEs, see [5, 13]. Recently, in [62], an MFS for the time-dependent linear one-dimensional heat equation was proposed and investigated, which included theoretical properties and numerical results. This method was extended by the same authors to heat conduction in one-dimensional layered materials in [63] and to free surface Stefan problems in [27]. In this thesis, we extend the MFS given in [62] to the two-dimensional heat conduction problem, the backward heat conduction problem (BHCP), the two-dimensional Cauchy problem, radially symmetric and axisymmetric BHCPs, the radially symmetric inverse heat conduction problem (IHCP), the inverse one and two-phase linear Stefan problems, the inverse Cauchy-Stefan problem, the two-dimensional inverse Stefan problem, and the inverse

two-phase one-dimensional nonlinear Stefan problem. We note that other formulations of the MFS, for example, different placements of source points and applying the MFS after using the Laplace transform, for the parabolic heat equation were given in [26, 41, 71, 103, 110] without theoretical justification. In the next section we describe the different types of Stefan problems that will be considered in the rest of this thesis.

0.1 Inverse and ill-posed problems

Most of the work in this thesis focuses on developing and applying an MFS to inverse heat conduction problems, which are ill-posed in the sense that the solution does not depend continuously on the data (i.e. measurement errors can magnify greatly and blur the sought solution). Applications of inverse problems can be found in physics, geophysics, electrodynamics, tomography, acoustics, financial mathematics, ecology, medicine, astronomy, etc. For surveys of inverse and ill-posed problems see [51, 68].

Informally, a direct problem involves solving a system where input data, in the form of boundary conditions, etc., is given on the entire boundary, and the resulting output data is found analytically or numerically. Inverse problems quite often require determining missing input data, including boundary data, coefficients appearing in the partial differential equation, etc., when some additional output data is given (for example data is overspecified on a part of the boundary). These so called inverse problems are generally ill-posed - a problem which does not satisfy one or more of Hadamard's definition of well-posedness. Hadamard's definition states that a solution for a model must exist, be unique and the solution must depend continuously on the data (i.e. small perturbations in the input data do not result in large perturbations in the solution). The ill-posedness of the problems we consider in this thesis will be due to problems with stability, which can be resolved using regularization techniques, see Section 0.5.

In the next section we describe and classify various types of direct and inverse Stefan problems.

0.2 Stefan problems

The classical Stefan problem is the name given to an initial-boundary value problem, which involves both fixed and moving phase boundaries. G. Lamé and B. P. Clapeyron are believed to be the first people to publish a paper [75] related to the area, however, it is named after Joseph Stefan, who, in 1889, studied the rate at which ice freezes on the ground [99], see [94] for a detailed analysis and history of the Stefan problem.

Stefan problems model many real world and engineering situations in which there is freezing or melting causing a boundary to change in time. Examples include solidification of metals, freezing of water and food, crystal growth, casting, welding, melting, etc. The direct Stefan problem requires determining both the temperature and the moving boundary interface when the initial and boundary conditions, and the thermal properties of the heat conducting body are known, [94]. Conversely, inverse Stefan problems require determining the initial and/or boundary conditions, and/or thermal properties from additional information which may involve the partial knowledge or measurement of the moving boundary interface position, its velocity in a normal direction, or the temperature at selected interior thermocouples of the domain, [42].

0.2.1 Mathematical formulation

In [94] Rubinsteĭn gives a general definition for the single-phase Stefan problem, and in [43] one and two-phase ill-posed inverse Stefan problems for parabolic equations are defined. Here we only state different versions of the Stefan problem for the heat equation. We note that all physical constants have been normalized.

0.2.2 The direct one-phase Stefan problem

In the direct one-dimensional Stefan problem we wish to determine the free boundary (sufficiently smooth) given by $x = s(t)$ and the temperature solution $u_1(x,t)$ in the heat conduction domain $(0, s(t)) \times (0, T]$, where $T > 0$ is a given arbitrary final time of interest, and we have a fixed boundary at $x = 0$. The free boundary represents a phase transition boundary, where the medium changes from one form to another (i.e. ice to liquid, for example). For the direct one-phase Stefan problem we seek a solution $(u_1(x,t), s(t))$, which satisfies the one-dimensional heat

equation

$$\frac{\partial u_1}{\partial t} - \frac{\partial^2 u_1}{\partial x^2} = 0, \quad \text{in } 0 < x < s(t),\ 0 < t \leq T, \tag{0.1}$$

subject to the initial condition

$$u_1(x,0) = u_1^0(x), \ \ 0 \leq x \leq s(0), \tag{0.2}$$

where $s(0) > 0$ is known, the Stefan interface conditions on the moving boundary $x = s(t)$

$$u_1(s(t),t) = u^*(t), \ \ t \in (0,T], \tag{0.3}$$

$$\frac{\partial u_1}{\partial x}(s(t),t) = -s'(t), \ \ t \in (0,T], \tag{0.4}$$

where (0.4) is a heat balance condition, which states that the rate of change of the moving boundary (denoted by $s'(t)$) is equal to the amount of heat entering/leaving through the boundary interface (energy conservation).

Additionally we impose a Neumann (heat flux) boundary condition for $x = 0$

$$-\frac{\partial u_1}{\partial x}(0,t) = g(t), \ \ t \in (0,T], \tag{0.5}$$

which is introduced to generate the freezing/melting process. In (0.2), $u_1^0 \in C^1([0, s(0)])$ and in (0.4), $s \in C^1([0,T])$ is a positive function.

In (0.3) u^* is the freezing/melting temperature (often constant) at the interface $s(t)$ (and (0.3) is natural to impose since it represents a change of phase at $s(t)$). When $u^* \equiv 0$ the problem represents, for example, the heating of an ice block via a heat flux $g(t)$ on the fixed boundary. We note that in our model problem the thermal diffusivity constant does not appear in equation (0.1) due to applying a change of variables. The Neumann boundary condition in (0.5) can be replaced by a Dirichlet (temperature) boundary condition

$$u_1(0,t) = h(t), \ \ t \in (0,T], \tag{0.6}$$

or the energy specification

$$\int_0^{s(t)} u_1(x,t)\,dx = E(t), \ \ t \in (0,T], \tag{0.7}$$

which are introduced to generate the freezing/melting process. The initial condition u_1^0 satisfies the compatibility condition at $x = s(0)$, namely,

$$u_1^0(s(0)) = u^*(0). \tag{0.8}$$

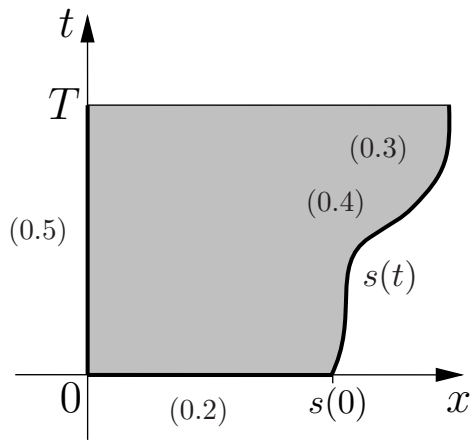

Figure 1: Representation of the one-phase direct Stefan problem, with locations of the initial and boundary conditions (0.2)–(0.5).

Existence and uniqueness of a solution, and continuous dependence on the data, i.e. well-posedness, for $u_1(x,t)$ and $s(t)$, hold for direct one-phase Stefan problems with Neumann condition on the fixed boundary (0.1)–(0.5), Dirichlet condition on the fixed boundary (0.1)–(0.4) and with energy specification (0.6), (0.1)–(0.4) and (0.7), see [22, 17, 24], respectively. Note that the boundary conditions given by equations (0.3) and (0.4), if they can be determined, could be replaced by the more general boundary conditions

$$u_1(s(t),t) \;=\; h_1(t), \;\; t \in (0,T], \tag{0.9}$$

$$\frac{\partial u_1}{\partial x}(s(t),t) \;=\; h_2(t), \;\; t \in (0,T], \tag{0.10}$$

where $h_1, h_2 \in C^1([0,T])$ satisfy the compatibility conditions $h_1(0) = u_1^0(s(0)), h_2(0) = u_1^{0\prime}(s(0))$.

0.2.3 The direct two-phase Stefan problem

The direct one-dimensional one-phase Stefan problem can be extended to the direct two-phase Stefan problem, where we now have two regions in a slab of thickness $l > 0$, consisting partly of

water and partly of ice separated by the moving interface $x = s(t)$, where

$$s(t) \in (0, l) \quad \text{for } t \in [0, T]. \tag{0.11}$$

The problem requires determining two functions u_1 and u_2 satisfying the one-dimensional linear heat equations given by

$$\frac{\partial u_1}{\partial t} = \alpha_1 \frac{\partial^2 u_1}{\partial x^2} \quad \text{in } 0 < x < s(t),\ 0 < t \leq T, \tag{0.12a}$$

$$\frac{\partial u_2}{\partial t} = \alpha_2 \frac{\partial^2 u_2}{\partial x^2} \quad \text{in } s(t) < x < l,\ 0 < t \leq T, \tag{0.12b}$$

subject to the initial conditions at $t = 0$ given by (0.2) and

$$u_2(x, 0) = u_2^0(x), \quad s(0) \leq x \leq l, \tag{0.13}$$

the Stefan interface conditions along the curve $x = s(t)$

$$u_1(s(t), t) = u_2(s(t), t) = u^*(t), \quad 0 < t \leq T, \tag{0.14a}$$

$$s'(t) = -K_1 \frac{\partial u_1}{\partial x}(s(t), t) + K_2 \frac{\partial u_2}{\partial x}(s(t), t), \quad 0 < t \leq T, \tag{0.14b}$$

and a Dirichlet or Neumann boundary condition at $x = l$, i.e.

$$u_2(l, t) = f_l(t) \qquad 0 < t \leq T, \tag{0.15a}$$

or

$$k_2 \frac{\partial u_2}{\partial x}(l, t) = g_l(t), \quad 0 < t \leq T. \tag{0.15b}$$

In the above model $\alpha_i = k_i/(\rho_i c_i)$ are the thermal diffusivities of water ($i = 1$) and ice ($i = 2$), k_i, ρ_i, c_i are the thermal conductivities, densities and heat capacities, respectively, $K_i = k_i/(\rho_2 L)$ where L is the latent heat, and all the preceding physical quantities are constant, positive and given. The functions u_1^0 and u_2^0 represent the given initial data and satisfy the compatibility conditions at $x = s(0)$, namely,

$$u_1^0(s(0)) = u_2^0(s(0)) = u^*(0), \tag{0.16}$$

where $s(0)$ is assumed to be in the interval $(0, l)$. We also require the compatibility condition at $x = l$, namely

$$u_2^0(l) = f_l(0) \quad \text{or} \quad k_2 \frac{du_2^0}{dx}(l) = g_l(0), \tag{0.17}$$

in the case of the Dirichlet or Neumann data in (0.15), respectively.

The Stefan boundary conditions (0.14a) and (0.14b) reflect, respectively, the facts that the temperature at the interface must be equal to the initial temperature $u^*(t)$ (usually taken to be zero in melting processes, for simplicity) and that the rate of melting is proportional to the rate of absorption of heat energy at the interface, [16].

Denote by

$$Q_T = \{(x, t) \,|\, 0 < x < l, 0 < t \leq T\}$$

the two-phase rectangular solution domain $(0, l) \times (0, T]$ which is subdivided into two subdomains

$$D_T^1 = \{(x, t) \in Q_T \,|\, 0 < x < s(t),\ 0 < t \leq T\},$$
$$D_T^2 = \{(x, t) \in Q_T \,|\, s(t) < x < l,\ 0 < t \leq T\}.$$

The direct two-phase Stefan problem requires finding the temperatures $u_i \in C^{2,1}(D_T^i), i = 1, 2$, and the interface $s \in C([0, T]) \cap C^1((0, T])$ satisfying $s(t) \in (0, l)$ for $0 \leq t \leq T$, equations (0.12), (0.2), (0.13)–(0.17) and also a boundary condition at $x = 0$, namely

$$u_1(0, t) = f_0(t), \quad 0 < t \leq T, \tag{0.18a}$$

or

$$-k_1 \frac{\partial u_1}{\partial x}(0, t) = g_0(t), \quad 0 < t \leq T, \tag{0.18b}$$

or

$$\int_0^{s(t)} u_1(x, t)\, dx = E(t), \quad 0 < t \leq T, \tag{0.18c}$$

where f_0, or g_0, or E are given functions satisfying the compatibility conditions

$$u_1^0(0) = f_0(0), \quad \text{or} \quad -k_1 \frac{\partial u_1^0}{\partial x}(0) = g_0(0), \quad \text{or} \quad \int_0^{s(0)} u_1^0(x)\, dx = E(0). \tag{0.19}$$

The well-posedness of the above direct two-phase Stefan problems were accomplished by J.R. Cannon and his co-workers in [20, 21, 23]. The infinite differentiability of the free surface function

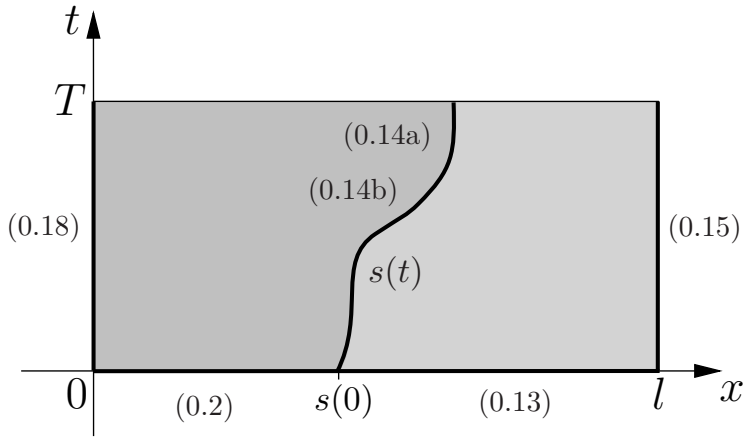

Figure 2: Representation of the two-phase direct Stefan problem, with locations of the initial and boundary conditions (0.2), (0.13)–(0.15) and (0.18).

$x = s(t)$ has been established in [19].

Several inverse Stefan–type problems can then be formulated as described in the following section.

0.2.4 Inverse Stefan problems

There are many classes of inverse Stefan problems that can be formulated, and in contrast to the previously formulated direct problems, these inverse problems are ill-posed, and regularization methods are usually applied to obtain a stable solution. The problems listed in this subsection can be found in [43], which also contains examples which display the ill-posedness of these problems, including non-uniqueness and solutions which do not depend continuously on the data.

One class of one-phase inverse Stefan problems requires finding the boundary condition at $x = 0$, where we apply an additional condition along a known boundary $\tilde{l}(t)$ $(0 < \tilde{l}(t) < s(t))$, given by

$$u_1(\tilde{l}(t), t) = u_{\tilde{l}}(t), \quad 0 \leq t \leq T. \tag{0.20}$$

Another class has additional information given at the final time point $t = T$ given by

$$u_1(x, T) = u_T(x), \quad 0 \leq x \leq s(T). \tag{0.21}$$

The problem is then to determine the functions $u_1(x,t)$, $h(t)$ and the boundary $s(t)$, satisfying the equations (0.1)–(0.4) and either (0.20) (when data is given on $\tilde{l}(t)$) or (0.21) (when data is given at T).

Similar to the one-phase case, a class of two-phase inverse Stefan problem requires the determination of the functions $u_1(x,t)$, $u_2(x,t)$, the Dirichlet condition at $x=0$, $f_0(t)$, and the boundary $s(t)$, where additional data is given at either the final time point $t=T$, given by

$$\begin{aligned} u_1(x,T) &= u_T^{(1)}(x), \quad 0 \le x < s(T), \\ u_2(x,T) &= u_T^{(2)}(x), \quad s(T) < x \le l, \end{aligned} \tag{0.22}$$

where $s(T)$ is known, or along the boundary $x=\tilde{l}(t)$.

Coefficient inverse Stefan problems include determining $u_1(x,t)$, $u_2(x,t)$, $s(t)$ and also coefficients in the heat equation or Stefan condition, where we have additional information given above.

0.2.5 The inverse boundary Stefan problem

This class of problems, related to corrosion engineering, requires finding the temperature $u_1(x,t)$ and the moving boundary $s(t)$ satisfying the heat equation (0.1), initial condition (0.2), Dirichlet condition on the moving boundary (0.3), and the Neumann (0.5) and Dirichlet (0.6) boundary conditions on the fixed boundary, see [33, 58, 83]. The initial temperature condition (0.2) may also be unknown, [107].

In the next two types of inverse Stefan problems the position of the moving boundary $s(t)$ is known in advance and these problems are called moving boundary design problems, [96, 97].

0.2.6 The inverse Stefan problem

One of the inverse Stefan problems that is investigated in Chapter 6 of this thesis requires finding the temperature $u_1(x,t)$ satisfying the heat equation (0.1), initial condition (0.2), Dirichlet condition (0.3) and the Neumann condition on the moving boundary (0.4), or more generally equations (0.1), (0.2), and the more general Dirichlet and Neumann boundary conditions on the moving boundary (0.9) and (0.10), when the moving boundary $s(t)$ is known, [61, 93]. One can see that, in contrast to the inverse Cauchy–Stefan problem of Section 0.2.7, in the inverse Stefan

problem the initial condition (0.2) is specified, and we require finding the Neumann and Dirichlet boundary conditions (0.5) and (0.6) at the fixed boundary $x = 0$. This problem can immediately be reinterpreted in terms of the 'backward in space' inverse heat conduction problem with an 'initial' transient boundary, [89]. Consequently, the inverse Stefan problem, although it usually has a unique solution, [18], is still an ill-posed problem with respect to small perturbations in the input data (0.9) and (0.10), and regularization techniques are needed to restore some type of stability. In particular, conditional stability can be obtained if we impose the bound $|g(t)| \leq M$, $t \in (0, T]$ in (0.5), [18].

We shall consider the implementation of the MFS, a variant of the one developed in [58], where a different placement of source points in time was considered, for the inverse boundary Stefan problem (0.1)–(0.3), (0.5) and (0.6), and in [27] for the direct Stefan problem (0.1)–(0.5), combined with the Tikhonov regularization procedure for solving the inverse Stefan problem (0.1), (0.2), (0.9) and (0.10).

0.2.7 The inverse Cauchy-Stefan problem

This class of problems, which requires finding the temperature $u_1(x, t)$ satisfying equations (0.1), (0.9) and (0.10), when the moving boundary $s(t)$ is known, can be interpreted as a non-characteristic ill-posed Cauchy problem since both the initial condition (0.2) and the Neumann and Dirichlet boundary conditions (0.5) and (0.6) are unknown and need to be determined, [56].

A result about *a priori* (which assumes a solution exists) and *a posteriori* estimates, and hence uniqueness, can be found in [18], and is stated in the following theorem.

Theorem 0.2.1. *If* $u_1 \in C^2(D) \cap C^1(\overline{D})$ *is a solution of* (0.1), (0.9) *and* (0.10) *for* $s \in C^1([0, T])$ *and* $h_1, h_2 \in C^2([0, T])$, *then there exists a function* $\theta(x) > 0$ *for* $x \in (0, s(t))$, *and a constant* $C = C(M, \mu, \delta) > 0$ *such that*

$$|u_1(x, t)| \leq C(\|h_1\| + \|h_2\|)^{\theta(x)}, \quad (x, t) \in (0, s(t)) \times [\delta, T], \tag{0.23}$$

for any $\delta \in (0, T]$, *where* $M = \max\{\|u_1\|, \|u_{1x}\|\}, \mu = \min_{\tau \in [0,T]} s(\tau)$, *and* $\|\cdot\|$ *denotes the maximum continuity norm on a compact interval.*

For the non-characteristic Cauchy problem (0.1), (0.9) and (0.10) with analytic s, h_1 and h_2,

we can rearrange the Cauchy-Kowalewski series to formally obtain, see [56] and [15, Theorem 2.6.1] the solution

$$u_1(x,t) \;=\; \sum_{j=0}^{\infty} \frac{\partial^j}{\partial t^j} \left\{ h_1(t) \frac{(x-s(t))^{2j}}{(2j)!} + (s'(t)h_1(t) + h_2(t)) \frac{(x-s(t))^{2j+1}}{(2j+1)!} \right\}. \tag{0.24}$$

In the case of the Stefan problem with $h_1(t) = 0$ and $h_2(t) = -s'(t)$, (0.24) simplifies to

$$u_1(x,t) \;=\; \sum_{j=1}^{\infty} \frac{1}{(2j)!} \frac{\partial^j}{\partial t^j} \left\{ (x-s(t))^{2j} \right\}. \tag{0.25}$$

For $s(0) > 0$ the formal solution (0.24) is also a solution to the inverse Stefan problem (0.1), (0.2) (0.9) and (0.10) if the initial data $u_0(x)$ fits the limit, as t tends to zero, of (0.24). However, formula (0.24) is not useful numerically since it involves infinite differentiation of functions which are rarely smooth in practice.

0.2.8 An inverse two-phase nonlinear Stefan problem

In Chapter 8 we consider an inverse two-phase one-dimensional nonlinear Stefan problem. The problem is nonlinear due to the free surface being considered unknown. Compared to the one-phase case, the amount of literature on two-phase flows is much more scarce, see [2, 64, 96]. However, the inverse design problems in these papers were linear since the position of the moving boundary was known. When the position of the moving boundary $s(t)$ is unknown and temperature or heat flux data is missing from a part of the boundary then we have a nonlinear and ill-posed inverse Stefan problem, see [43]. Uniqueness of a solution to these problems is available in Hölder spaces, see [42, 44], however, the problem is still ill-posed since there is no continuous dependence of the solution on the input data.

This inverse nonlinear two-phase one-dimensional Stefan problem requires finding the triplet solution $(u_1, u_2, s) \in C^{2,1}(D_T^1) \times C^{2,1}(D_T^2) \times (C([0,T]) \cap C^1((0,T]))$, satisfying equations (0.11) ($s(0)$ is known), (0.12), (0.2), (0.13)–(0.14), (0.16) and both conditions in (0.15) and (0.17). Note that the fixed boundary $x = l$ is overspecified since both the Dirichlet and Neumann (i.e. Cauchy) data are prescribed in (0.15a) and (0.15b), whilst the fixed boundary $x = 0$ is underspecified. Of particular interest, and to be recovered, is the temperature (0.18a), the heat flux (0.18b), and the mass (0.18c) (and satisfy the compatibility conditions (0.19)).

A sketch of this two-phase inverse nonlinear Stefan problem is shown in Figure 3.

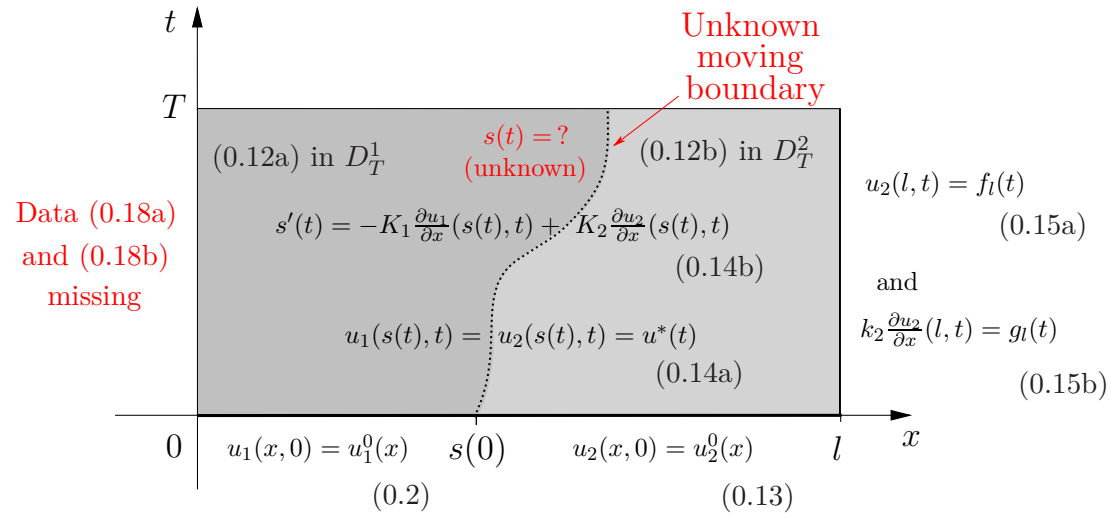

Figure 3: Sketch of the two-phase inverse nonlinear Stefan problem (0.11), (0.12), (0.2), (0.13)–(0.17), with unknown data annotated in red.

Sometimes real world measurements of both the boundary temperature (0.15a) and the heat flux (0.15b) can be difficult, and only one is available. As mentioned previously for a different class of two-phase inverse Stefan problem internal temperature measurements could instead be taken at the final time $t = T$, namely (0.22). However, it turns out, see [43], that this latter 'upper base data (0.22)' inverse problem is more ill-posed than the former 'Cauchy data (0.15a,b)' inverse problem because it can have more than one solution, i.e. the solution is not unique. However, as we shall see below, the former problem can have at most one solution, i.e. the solution is unique, though the problem is still ill-posed since small errors in the input data (0.15) can cause large errors in the output data (0.18). Therefore, in this thesis we only investigate the inverse problem of determining the free boundary and temperature data from the heat equations (0.12), initial data (0.2) and (0.13), Stefan interface conditions (0.14) and Dirichlet and Neumann boundary conditions at the fixed boundary (0.15) satisfying compatibility conditions (0.16) and (0.17). This problem may be considered as a continuation problem of the solution of the parabolic linear heat equation from the boundary $x = l$, where the Cauchy data (0.15a,b) is given, into the domain Q_T. This problem can also be reinterpreted as a 'backward in space' inverse heat conduction problem with an 'initial' transient boundary. However, in contrast to the non-characteristic Cauchy problem, there is the unknown phase transition moving boundary $x = s(t)$ in Q_T also to be determined and this essentially complicates

the task of analytic continuation.

An initial attempt would be to split the two-phase inverse Stefan problem (0.11), (0.12), (0.2), (0.13)–(0.17) into two problems. The first is the nonlinear inverse boundary determination problem for the pair (u_2, s) satisfying equations (0.11), (0.12b), (0.13), (0.15a), (0.15a), (0.17) and

$$u_2(s(t), t) = u^*(t), \quad t \in (0, T] \tag{0.26}$$

with

$$u_2^0(s(0)) = u^*(0). \tag{0.27}$$

The solution of this problem is unique, see [33], even if the initial condition (0.13) is not imposed, see [107]. Once the boundary $x = s(t)$ and the heat flux $\frac{\partial u_2}{\partial x}(s(t), t)$ have been determined, the second is the linear inverse problem for determining the temperature u_1 satisfying equations (0.12a), (0.2),

$$u_1(s(t), t) = u^*(t), \quad t \in (0, T] \tag{0.28}$$

with

$$u_1^0(s(0)) = u^*(0), \tag{0.29}$$

and

$$-K_1 \frac{\partial u_1}{\partial x}(s(t), t) = s'(t) - K_2 \frac{\partial u_2}{\partial x}(s(t), t), \quad t \in (0, T]. \tag{0.30}$$

The solution of this problem is unique, see [18] even if the initial condition (0.2) is not imposed, see [56] and [106]. However, both the above problems are ill-posed since once again their solutions do not depend continuously on the input data. In order to obtain stable solutions regularization is necessary and this involves choosing at least two regularization parameters, one for each problem. As well as depending on the amount of noise in the input data they depend on each other since the first nonlinear ill-posed problem has to be solved first to provide the input for the second linear ill-posed problem. So, this two-parameter choice becomes complicated.

Therefore, it appears more useful to solve the composite problem (0.11), (0.12), (0.2), (0.13)–(0.17) for simultaneously determining the solution (u_1, u_2, s).

0.3 A short history of fundamental solutions

In the last century, the study of the theory of distributions has been invaluable in the area of PDEs (the definition of a distribution can be found in the Appendix). Sergei Sobolev, in 1935, introduced the use of functionals as generalised functions, and Laurent Schwartz, in 1950, gave a broad definition of the concept of a fundamental solution. We present here a brief history of fundamental solutions, with details from [90, 105, 74].

Nils Zeilon, in 1911, gave a definition for locally integrable fundamental solutions [105]: F is a fundamental solution of the differential equation $f(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}) = 0$ if and only if

$$u = \iiint F(x-\lambda, y-\mu, z-\nu)\phi(\lambda,\mu,\nu)\ d\lambda d\mu d\nu$$

solves $f(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})u = \phi$, where ϕ is an arbitrary function.

Laurent Schwartz defined a fundamental solution, in his *Theorie des Distributions* (1950), as follows: F is a fundamental solution if and only if $P(\partial)F = \delta$, where $F : \mathbb{R}^n \to \mathbb{R}$, $P(\partial)$ is a differential operator, and δ is the Dirac delta function, representing a unit point mass. The solution to $P(\partial)u = f$, can then be obtained by taking the convolution of f and the fundamental solution F. We note that when boundary conditions are applied to fundamental solutions, they are often known as Green's functions [74].

Fundamental solutions, prior to any definition and the theory of distributions, had been used explicitly by several authors; probably with the first application of a 'non-trivial' fundamental solution given by Jean D'Alembert in 1747. D'Alembert worked on the one-dimensional wave equation, given by

$$\frac{\partial^2 u}{\partial t^2} - c^2 \frac{\partial^2 u}{\partial x^2} = f. \tag{0.31}$$

The solution of (0.31) is given by the convolution of f with the fundamental solution of (0.31), $F_W^{(1)}(x,t) = \frac{1}{2c} H\left(t - \frac{|x|}{c}\right)$, where H represents the Heaviside function, given by

$$H(x) = \begin{cases} 1, & \text{if } x > 0, \\ 0, & \text{if } x \leq 0. \end{cases}$$

Poisson, in 1818, worked with the fundamental solution of the wave equation in three-

dimensions, given by

$$\frac{\partial^2 u}{\partial t^2} \, - \, c^2 \Delta u \, = \, f, \tag{0.32}$$

where Δ is the Laplace operator, and found that convolutions with f of the fundamental solution

$$F_W^{(3)} \, = \, \frac{\delta(t - |\mathbf{x}|)}{4\pi|\mathbf{x}|},$$

produced solutions to (0.32).

The fundamental solution $F_W^{(n)}$, given in [74], of the n-dimensional wave equation,

$$\frac{\partial^2 F_W^{(n)}}{\partial t^2}(\mathbf{x}, t) \, - \, c^2 \nabla^2 F_W^{(n)}(\mathbf{x}, t) \, = \, \delta(\mathbf{x}, t),$$

where δ is the Dirac delta function, defined in the Appendix, for $n \geq 2$ is given by

$$F_W^{(n)}(\mathbf{x}, t) = \begin{cases} \frac{H(t)}{2\pi c} \left(\frac{1}{\pi c^2} \frac{d^2}{dt^2} \right)^{(n-3)/2} \delta(c^2 t^2 - |\mathbf{x}|^2), & \text{for } \; n \geq 3 \; \text{ odd}, \\ \frac{1}{2\pi c} \left(\frac{1}{\pi c^2} \frac{d^2}{dt^2} \right)^{(n-2)/2} \frac{H(ct - |\mathbf{x}|)}{\sqrt{c^2 t^2 - |\mathbf{x}|^2}}, & \text{for } \; n \geq 2 \; \text{ even}. \end{cases}$$

Later, in 1789, Pierre-Simon Laplace used the fundamental solution $F_L^{(3)}(\mathbf{x}) = \frac{1}{4\pi|\mathbf{x}|}$ of the Laplacian, generated by the partial differential operator

$$\Delta \, = \, \frac{\partial^2}{\partial x^2} \, + \, \frac{\partial^2}{\partial y^2} \, + \, \frac{\partial^2}{\partial z^2},$$

and discovered that $\Delta(F_L * f) = 0$ in the exterior of the compact support of f, and Siméon Poisson, in 1813, showed that $\Delta(F_L * f) = f$, see [105]. The fundamental solution of the Laplace equation in one-dimension is given by $F_L^{(1)} = -\frac{1}{2}|x|$, and in two-dimensions by $F_L^{(2)} = -\frac{1}{2\pi} \ln |\mathbf{x}|$.

Laplace, in 1809, gave the fundamental solution of the one-dimensional heat equation

$$F(x, t) \, = \, \frac{H(t)}{(4\pi t)^{\frac{1}{2}}} e^{\frac{-|x|^2}{4t}},$$

and Poisson, in 1818, gave the fundamental solution to the two-dimensional heat equation, [105]. Fourier contributed greatly to the theory of heat conduction, and also discussed point sources and terms which look like representations of the fundamental solution, see [37, p. 377]. The

fundamental solution of the heat equation in n-dimensions is given by

$$F(x,t) = \frac{H(t)}{(4\pi t)^{\frac{n}{2}}} e^{\frac{-|\mathbf{x}|^2}{4t}}. \tag{0.33}$$

The Malgrange-Ehrenprenis theorem, given below, states that for non-zero linear partial differential equations with constant coefficients, fundamental solutions exist.

Theorem 0.3.1. *(Malgrange/Ehrenpreis, 1953/1954)* [90]. *Let* $P(\boldsymbol{\xi}) = \sum_{|i|\leq M} \beta_i \boldsymbol{\xi}^i$ *be a not identically vanishing polynomial in* $\mathbb{R}^n$ *(where* $\beta_i \in \mathbb{C}$, $\boldsymbol{\xi}^i = \xi_1^{i_1} \dots \xi_n^{i_n}$ *and* $\beta_i \neq 0$ *for some* i*). Then there exists a fundamental solution of*

$$P(\partial) = \sum_{|i|\leq M} \beta_i \frac{\partial^{i_1}}{\partial x_1^{i_1}} \cdots \frac{\partial^{i_n}}{\partial x_n^{i_n}}.$$

0.4 The method of fundamental solutions

A variety of different numerical methods exist for solving partial differential equations. Three of the main methods are: the finite element method (FEM), the finite difference method (FDM), and the boundary element method (BEM). These methods have been shown to produce accurate approximations for various PDEs; however, there are certain limitations and difficulties when applying them. For example, the FEM and FDM are both domain methods, and require a mesh to be generated over the solution domain, which can be computationally expensive, and there are additional difficulties when the boundary of the domain changes in time. The BEM only requires a mesh to be formed over the boundary of the domain, and therefore, reduces the dimension of the problem compared to the FEM and FDM, however, we must integrate over this boundary. Figure 4 displays examples of possible meshes generated for the different methods discussed above.

In recent years, a new numerical method, the method of fundamental solutions (MFS), has been developed and applied when the fundamental solution of the governing PDE is known. The first authors and papers that discussed the ideas for the MFS were V.D. Kupradze and M.A. Aleksidze in [72, 73] (which was called the method of functional equations in the papers), however, it took over a decade before the method was implemented numerically by R. Mathon and R.L. Johnston in [85]. The MFS is a collocation method, i.e. the boundary conditions are

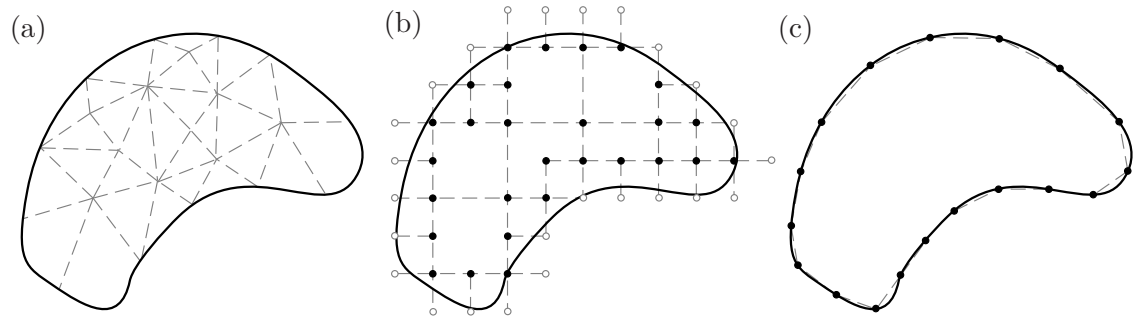

Figure 4: Examples of meshes used for a general two-dimensional domain for (a) the FEM using triangulation, (b) the FDM with adaptive mesh refinement, and (c) the BEM using straight line panels.

imposed only at a finite number of points, therefore, it shares advantages with the BEM over the FEM and FDM in that we do not need to generate a mesh over the interior domain. Unlike BEM the MFS does not require boundary element integrations, however, once the integrals have been calculated for the BEM we solve a linear system like the MFS, and the BEM has been successfully applied to a range of inverse heat conduction problems, see [54, 78, 80].

Prior to computers, and the numerical methods listed above, Walter Ritz developed a method, now called the Ritz method, based on the variational method and trial functions, which is considered a forerunner of the Finite Element Method [28]. Later, Erich Trefftz expanded on the Ritz method, and developed the boundary method now known as the Trefftz method, where trial functions must satisfy the governing equation, but not necessarily the boundary conditions, however integration is required over the solution boundary, for more details see [28]. The Trefftz method has been modified further, and, instead of integrating, we can choose a finite number of collocation points $\mathbf{x}_k$ over the boundary such that

$$\sum_{j=1}^{N} c_j \phi_j(\mathbf{x}_k) = g(\mathbf{x}_k),$$

where the ϕ_j are the trial functions, c_j are the constant coefficients we wish to determine, and $g(\mathbf{x}_k)$ is the boundary condition applied at $\mathbf{x}_k$. In the MFS we use fundamental solutions as trial functions, since they satisfy the governing equation

$$\mathcal{L}(F(\mathbf{x}, \mathbf{y})) = \delta(\mathbf{x} - \mathbf{y}),$$

where $F(\mathbf{x}, \mathbf{y})$ is the fundamental solution of $\mathcal{L}$, a linear partial differential operator with constant coefficients, with a source point placed at $\mathbf{y}$, and δ is the Dirac delta function. For details on distributions and the definition of the Dirac delta function see the Appendix.

If we have a solution domain $D \subset \mathbb{R}^n$, with boundary Γ, then an approximate solution can be taken to be

$$u(\mathbf{x}) \approx \sum_{j=1}^{N} c_j F(\mathbf{x}, \mathbf{y}_j) \quad \text{where} \quad \mathbf{x} \in D \quad \text{and} \quad \mathbf{y}_j \notin D, \tag{0.34}$$

where a finite number (N in this case) of source points $\mathbf{y}_j$ are placed outside the domain D (D bounded or unbounded).

We note that $u(\mathbf{x})$ satisfies the homogeneous differential equation in the domain D as a function of $\mathbf{x}$, and the collocation method is used (i.e. we impose boundary data at a finite number M of points), in the case of Dirichlet conditions, to obtain

$$u(\mathbf{x}_k) \approx \sum_{j=1}^{N} c_j F(\mathbf{x}_k, \mathbf{y}_j) = g(\mathbf{x}_k), \quad k = 1, \ldots, M, \quad \text{and} \quad \mathbf{x}_k \in \Gamma.$$

For a representation of a possible placement of collocation and source points for an example domain see Figure 5. The system of equations generated by imposing the boundary conditions

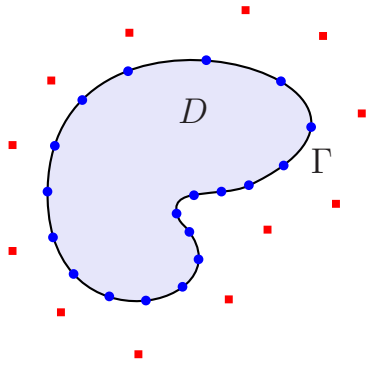

Figure 5: Representation of a possible placement of source (■) and collocation points (●).

can then be written as

$$A\mathbf{c} = \mathbf{g}, \tag{0.35}$$

where A is an $M \times N$ matrix taking values of the fundamental solution at the corresponding

collocation and source points, i.e.

$$A_{kj} = F(\mathbf{x}_k, \mathbf{y}_j), \quad \text{for } k = 1 \dots M,\ j = 1 \dots N,$$

$\mathbf{c}$ is the vector of unknown constants c_j we need to determine to form our approximation, and $\mathbf{g}$ is the vector of function values at the collocation points. Sometimes it will be possible to solve this system using Gaussian elimination, i.e. when we have a square matrix of full rank, however, systems generated from ill-posed problems and the MFS result in highly ill-conditioned matrices, therefore we will need to use regularization techniques, details of which can be found in the next section.

An important aspect of the MFS is the choice of an optimal placement of source points to achieve the best approximation. A method known as the dynamic MFS, used in [85], involves finding the placement of source points as well as the constant coefficients appearing in the approximation (0.34), which results in a nonlinear least squares minimisation problem, see [10, 70]. A description, and algorithm suggestion, of how this can accomplished can be found in [69]. Another approach, which has become more popular than the dynamic approach, called the static MFS, involves fixing the position of the source points, which leads to a linear problem, see [11, 62, 86, 92]. In this thesis we primarily focus on the static MFS, and find the position of the source points by trial and error, however, we test the dynamic MFS approach in Chapter 6.

There have been several approaches solving parabolic equations using the MFS. For instance, in [40], the Laplace transform was applied to the heat equation, and the resulting elliptic equation was solved using the MFS. The problem with this technique is that errors will accumulate after applying the MFS and the numerical inversion of the Laplace transform. A direct approach is to solve the heat equation without using transforms, and place source points on a surface in space and time. Mera, in [86], applied the MFS directly and placed source points on a surface below the solution domain (i.e. at a previous time point), see Figure 6(a). However, there are no theoretical properties (denseness) for this placement of source points. Based on theoretical (denseness) and numerical results presented in [72] and [62], we extend their work by also placing source points in time, i.e. if the domain of the problem is $D \times (0, T)$ then we place source points on a boundary $\Gamma_E \times (-T, T)$ external to the domain and in time, see Figure 6(b).

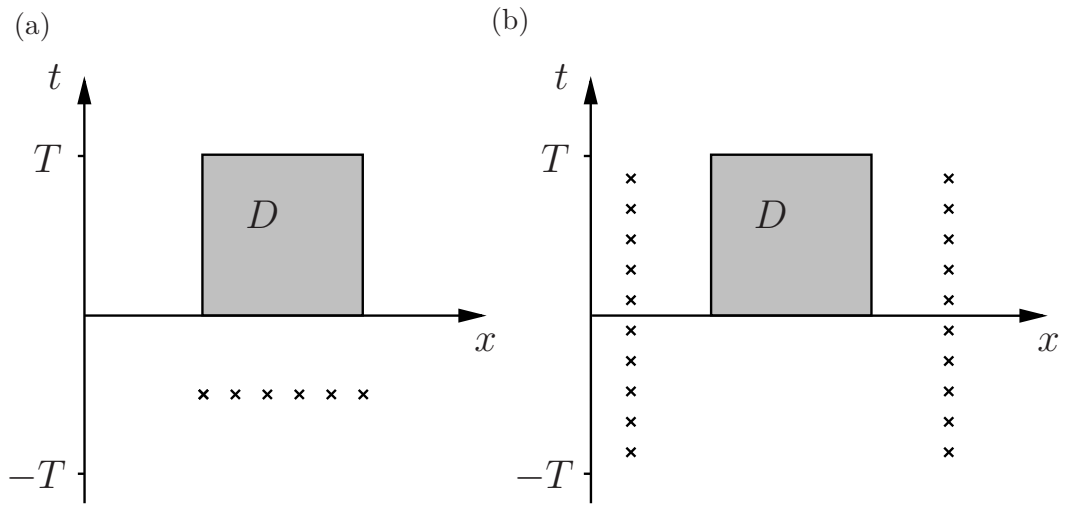

Figure 6: General representations of the placement of source points (×) for a one-dimensional domain used in (a) by Mera in [86] and (b) by Kupradze and Johansson et al. in [72] and [62], respectively.

0.5 Regularization

From the previous section we noted that employing the MFS will often result in a linear system of equations

$$A\mathbf{x} = \mathbf{b}, \tag{0.36}$$

where A is an $M \times N$ matrix with a high condition number. This type of system is usually referred to as a linear discrete ill-posed problem. Since the vector $\mathbf{b}$ is obtained through observation of some physical quantity, it contains measurement errors (and sometimes discretization errors). We denote the sum of these errors by $\mathbf{e}$, and the vector, which contains noise by $\mathbf{b_e}$, where

$$\mathbf{b_e} = \mathbf{b} + \mathbf{e} = A\mathbf{x} + \mathbf{e}. \tag{0.37}$$

A straightforward solution of (0.36) (using, for example, the generalised inverse) usually gives a meaningless solution, see [51] for details on the need for regularization of ill-posed problems.

The definition of the Moore-Penrose pseudoinverse $A^\dagger$ is given below, details of which can be found in [91].

Definition 0.5.1. *The Moore-Penrose pseudoinverse of a real (or complex) $M \times N$ matrix A*

over the field of real (or complex) numbers is the unique matrix $A^\dagger$ satisfying the four equations:

$$AA^\dagger A = A, \tag{0.38}$$

$$A^\dagger AA^\dagger = A^\dagger, \tag{0.39}$$

$$(AA^\dagger)^T = AA^\dagger, \tag{0.40}$$

$$(A^\dagger A)^T = A^\dagger A, \tag{0.41}$$

where A^T represents the conjugate transpose.

We note that the generalised inverse, see [9], is a matrix which satisfies conditions (0.38) and (0.39) of Definition 0.5.1. Also, if A is nonsingular then $A^{-1} = A^\dagger$. The following lemma gives the definitions of the left and right hand inverses of A.

Lemma 0.5.2. *If A is an $M \times N$ matrix, then:*

(i) *If $M \geq N$ and the rank of A is M, then $A^\dagger = (A^T A)^{-1} A^T$.*

(ii) *If $M \leq N$ and the rank of A is M, then $A^\dagger = A^T (AA^T)^{-1}$.*

It turns out that in (i) and (ii), $A^\dagger \mathbf{b}$ is the least squares solution of (0.36) of minimal norm. Returning to the point, to construct a meaningful approximation for (0.36) we apply regularization methods, which involve replacing the system of equations (0.36) by a system that is less sensitive to perturbations (of $\mathbf{b}$) and the solution to this system is then an approximation to $\mathbf{x}$. Popular regularization methods include Tikhonov regularization, truncated singular value decomposition (TSVD) and iterative stabilizing methods.

In this thesis we will use Tikhonov regularization. This method replaces (0.36) by the minimization problem

$$\min_{\mathbf{x} \in \mathbb{R}^n} \|A\mathbf{x} - \mathbf{b}\|^2 + \lambda \|L_k \mathbf{x}\|^2, \tag{0.42}$$

where $\lambda > 0$ is a regularization parameter.

The residual term $\|A\mathbf{x} - \mathbf{b}\|^2$ in (0.42) can be viewed as an indicator of how well the solution $\mathbf{x}$ approximates the system, therefore this term should be sufficiently small, i.e. λ should be small. However, one can show that to have a reasonably stable solution λ should be large. The appropriate choice of λ is a delicate problem. The norm $\|L_k \mathbf{x}\|^2$ in (0.42) controls the high frequency oscillatory behaviour of the solution caused by noise. The operators L_k are

constructed as approximations of derivatives and $k = 0$ imposes continuity (C^0) with L_0 equal to the identity matrix, $k = 1$ first order smoothness (C^1), $k = 2$ second order smoothness (C^2), etc., i.e.

$$L_0 = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \in \mathbb{R}^{N\times N}, \quad L_1 = \begin{pmatrix} 1 & -1 & 0 & \cdots & 0 \\ 0 & 1 & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & -1 \end{pmatrix} \in \mathbb{R}^{(N-1)\times N},$$

$$L_2 = \begin{pmatrix} 1 & -2 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & -2 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & -2 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & -2 & 1 \end{pmatrix} \in \mathbb{R}^{(N-2)\times N}.$$

The expression appearing in equation (0.42) can be rewritten

$$\begin{aligned} \|A\mathbf{x} - \mathbf{b}\|^2 + \lambda\|L_k\mathbf{x}\|^2 &= (A\mathbf{x} - \mathbf{b})^T(A\mathbf{x} - \mathbf{b}) + \lambda(L_k\mathbf{x})^T(L_k\mathbf{x}) \\ &= (\mathbf{x}^TA^T - \mathbf{b}^T)(A\mathbf{x} - \mathbf{b}) + \lambda\mathbf{x}^TL_k^TL_k\mathbf{x} \\ &= \mathbf{x}^TA^TA\mathbf{x} - \mathbf{b}^TA\mathbf{x} - \mathbf{x}^TA^T\mathbf{b} + \mathbf{b}^T\mathbf{b} + \lambda\mathbf{x}^TL_k^TL_k\mathbf{x} \\ &= \mathbf{x}^TA^TA\mathbf{x} - 2\mathbf{x}^TA^T\mathbf{b} + \mathbf{b}^T\mathbf{b} + \lambda\mathbf{x}^TL_k^TL_k\mathbf{x}. \end{aligned} \tag{0.43}$$

To determine the minimum (0.42) we set the gradient of (0.43) equal to zero, i.e.

$$2A^TA\mathbf{x} - 2A^T\mathbf{b} + 2\lambda L_k^TL_k\mathbf{x} = 0.$$

Therefore, after rearranging, the minimizer $\mathbf{x}_\lambda$ is the solution of

$$(A^TA + \lambda L_k^TL_k)\mathbf{x}_\lambda \;=\; A^T\mathbf{b}. \tag{0.44}$$

To justify the above methods we have the following theorems, which can be found in [102].

Theorem 0.5.3. *Let A be an $M \times N$ matrix. Then the linear system* (0.44) *has a unique*

solution ($\lambda > 0$).

Theorem 0.5.4. *Let A be an $M \times N$ matrix and let $\lambda > 0$. Then there is a unique element $\mathbf{x}_\lambda$ for which*

$$\min_{\mathbf{x}\in\mathbb{R}^n} \|A\mathbf{x} - \mathbf{b}\|^2 + \lambda\|L_k\mathbf{x}\|^2$$

is attained.

Note that

$$\lim_{\lambda\to 0}(A^TA + \lambda L_k^TL_k)^{-1}A^T\mathbf{b} = A^\dagger\mathbf{b}.$$

For a perturbed right hand side the parameter λ can be chosen using various methods, including the L-curve criterion, discrepancy principle, composite residual and smoothing operator (CRESO) method, zero-crossing method, etc. We note that the process of finding the regularization parameter can be automated but we will apply the L-curve criterion, which involves picking a value of λ which corresponds to the 'corner' of a plot of the residual 2-norm $\|A\mathbf{x}_\lambda - \mathbf{b_e}\|$ against the 2-norm $\|L_k\mathbf{x}_\lambda\|$, see Figure 7 for an example of a typical L-curve plot, where, in this case, we have used L_0, and we would choose $\lambda = 10^{-8}$ or $\lambda = 10^{-9}$. An informal justification for choosing the Tikhonov regularization parameter as the corner of the plot is that the solution will be under-smoothed for small λ (vertical line in Figure 7) or over-smoothed for large λ (horizontal line in Figure 7). For details on Tikhonov regularization and an analysis of the L-curve criterion see [50, 51]. However, the L-curve requires computing the solution to the original problem for a set of values for λ, and this can be time consuming. Therefore, it can be preferable to use more efficient methods to determine the optimal value, which corresponds to the point of maximum curvature (when dealing with more that one regularization parameter the generalised corner is the point of maximum Gaussian curvature of the L-hypersurface, see [8]). The discrepancy principle can be used to find λ such that

$$\|A\mathbf{x}_\lambda - \mathbf{b_e}\| = \tau\delta,$$

where $\tau > 1$ is a chosen constant, and δ is the noise level in the data ($\|\mathbf{b} - \mathbf{b_e}\| \leq \delta$).

In conclusion we note that $A^TA + \lambda L_k^TL_k$ forms a square matrix and can, in general, be inverted. For the system

$$(A^TA + \lambda L_k^TL_k)\mathbf{x} = A^T\mathbf{b_e}, \tag{0.45}$$

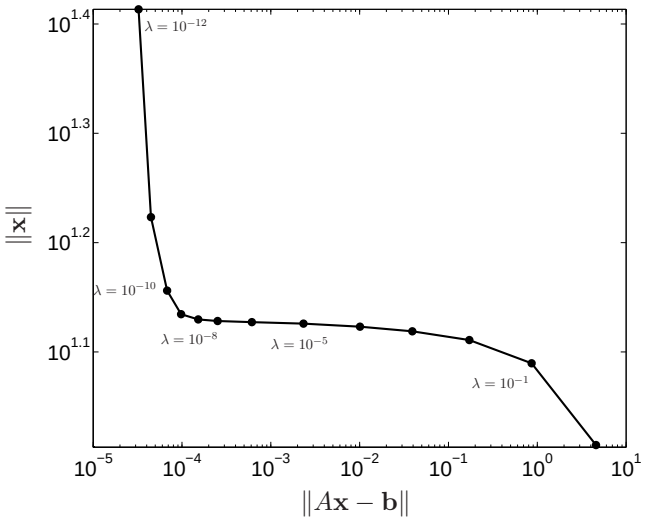

Figure 7: A typical log-log plot of the L-curve for Tikhonov regularization.

in this thesis we will use L_0, employ the backslash "\" command in MATLAB (which uses Gaussian elimination), and use the L-curve criterion [50] to determine a reasonable choice for the Tikhonov regularization parameter $\lambda > 0$.

0.6 Overview

We start in Chapter 1 by considering the two-dimensional heat conduction problem, extending the MFS for the one-dimensional heat equation investigated in [62]. Denseness results are given for both the lateral and base surfaces, justifying the use of this MFS. The MFS and the regularization technique are described, and numerical results will be produced for various domains, including squares and circles, and also various placements of source points.

In Chapter 2, we perform a comparative study of the MFS applied to the backward heat conduction problem (BHCP), an ill-posed problem, and extend the denseness results produced in Chapter 1, with results compared to [86], [59] and [82]. We extend the work from Chapters 1 and 2 to the two-dimensional Cauchy heat conduction problem in Chapter 3, where temperature and heat flux data are missing from a part of a boundary and must be recovered using measurements on the remaining part of the boundary.

In Chapter 4, we investigate the MFS applied to the radially symmetric BHCP. We prove results about the limits of the fundamental solution and its normal derivative, which are needed

in the proofs of denseness. Additionally, a denseness proof is given when we have Neumann boundary conditions. We extend this work to the radially symmetric inverse heat conduction problem (IHCP) in Chapter 5 and perform a comparison study to the results produced in [109]. In the radially symmetric IHCP Dirichlet and Neumann data needs to recovered on an internal boundary, using data overspecified on an outer boundary.

The inverse one-dimensional Stefan problem presented in Section 0.2.6 is investigated in Chapter 6, and involves finding Dirichlet and Neumann data on a fixed boundary, when data is given at an initial time point and overspecified on the free boundary. There will also be examples which consider the two-dimensional case and the determination, using a nonlinear least-squares solver, of a parameter which governs the position of the pseudo-boundary where the source points are placed. The inverse one-dimensional Cauchy-Stefan problem presented in Section 0.2.7 is considered in Chapter 7, which will be treated as an extension of the work presented in Chapter 6 for the inverse Stefan problem, since we do not impose the initial value of the solution.

Lastly, in Chapter 8 we consider the inverse two-phase one-dimensional nonlinear Stefan problem stated in Section 0.2.8, which, unlike Chapters 6 and 7, where the free boundary is known, requires the determination of both the temperature field and the free boundary. To solve this nonlinear problem we use the MATLAB toolbox *lsqnonlin*, in conjunction with Tikhonov regularization.

CHAPTER 1

THE DIRECT TWO-DIMENSIONAL HEAT CONDUCTION PROBLEM

1.1 Preliminaries and notation

We let $\mathbf{x} = (x_1, x_2)$ and $\mathbf{y} = (y_1, y_2)$ be points in $\mathbb{R}^2$ and $T > 0$ be a fixed real number. The conducting body D is a two-dimensional simply connected bounded domain in $\mathbb{R}^2$ with sufficiently smooth bounding surface $\Gamma = \partial D$, for example, C^2-smooth (or polygonal domains) is sufficient. The closure of the body D is $\overline{D} = D \cup \Gamma$. We define the following cylinders $D_T = D \times (0, T]$ and $\Gamma_T = \Gamma \times (0, T]$, respectively. The closures of D_T and Γ_T are given by $\overline{D}_T = \overline{D} \times [0, T]$ and $\overline{\Gamma}_T = \Gamma \times [0, T]$, respectively. Let, as usual, $\nabla = (\partial_{x_1}, \partial_{x_2})$ and $\Delta = \partial^2_{x_1} + \partial^2_{x_2}$.

We are interested in constructing the solution u to the heat equation in the domain D_T, supplied with initial and Dirichlet boundary conditions, that is, u solves

$$\frac{\partial u}{\partial t}(\mathbf{x}, t) - \Delta u(\mathbf{x}, t) = 0, \quad (\mathbf{x}, t) \in D_T, \tag{1.1}$$

$$u(\mathbf{x}, t) = h(\mathbf{x}, t), \quad (\mathbf{x}, t) \in \Gamma_T, \tag{1.2}$$

$$u(\mathbf{x}, 0) = u_0(\mathbf{x}), \quad \mathbf{x} \in D, \tag{1.3}$$

where $h(\mathbf{x}, t)$ and $u_0(\mathbf{x})$ are sufficiently smooth functions. We point out that, in principle, the MFS that we propose and investigate can be applied to other boundary conditions, such as Neumann and mixed boundary conditions.

To guarantee the existence and uniqueness of a solution to (1.1)–(1.3) we impose the following compatibility conditions:

$$u_0(\mathbf{x}) = h(\mathbf{x}, 0) \quad \text{and} \quad \frac{\partial h}{\partial t}(\mathbf{x}, 0) = \Delta u_0(\mathbf{x}), \ \mathbf{x} \in \Gamma. \tag{1.4}$$

With these conditions the following uniqueness theorem holds, see, for example, [38].

Theorem 1.1.1. *Let $u_0(\mathbf{x}) \in C^2(\overline{D})$ and $h(\mathbf{x},t) \in C^1(\overline{\Gamma}_T)$ satisfy the compatibility conditions* (1.4)*. Then there exists a unique solution $u \in C^{2,1}(\overline{D}_T)$, to the equations* (1.1)–(1.3)*, which depends continuously on the data.*

Theorem 1.1.1 tells us, in particular, that the problem given by the equations (1.1)–(1.3) is well-posed.

1.2 Denseness properties of linear combinations of fundamental solutions

The fundamental solution of (1.1) in two-dimensions is given by

$$F(\mathbf{x},t;\mathbf{y},\tau) \;=\; \frac{H(t-\tau)}{4\pi(t-\tau)} e^{-\frac{|\mathbf{x}-\mathbf{y}|^2}{4(t-\tau)}}, \tag{1.5}$$

where H is the Heaviside function, defined in the Appendix, which is necessary in order to emphasize that the fundamental solution is zero for $t \leq \tau$. We shall investigate some properties of linear combinations of such functions for various source points $\mathbf{y}$.

We begin by constructing a set of source points placed outside the region $\overline{D}$. Let D_E (E for enclosing) be an open domain, containing $\overline{D}$, with bounding surface Γ_E, where the distance between the points on the surfaces Γ and Γ_E is greater than zero. Let $\{\mathbf{y}_j, \tau_m\}_{j,m=1,2,\ldots}$ be a denumerable, everywhere dense set of points in $\Gamma_E \times [-T,T]$, ($\tau_m \neq 0$). Figure 1.1 shows how the source points may be placed around a domain D, either using a symmetric shape or shapes which take the general shape of Γ obtained by dilatation.

We now construct the following infinite series

$$u_\infty(\mathbf{x},t) \;=\; \sum_{j=1}^{\infty} \sum_{m=1}^{\infty} c_m^{(j)} F(\mathbf{x},t;\mathbf{y}_j,\tau_m), \tag{1.6}$$

where $c_m^{(j)}$ are set equal to zero except for a finite number of values. Note that, due to the Heaviside function in (1.5), we have $u_\infty(\mathbf{x},t) = 0$ for $t \leq \overline{\tau} = \min_{m,j:|c_m^{(j)}|\neq 0} \tau_m$. Also note that, by linearity, since F satisfies the heat equation (1.1) in D_T, so does u_∞.

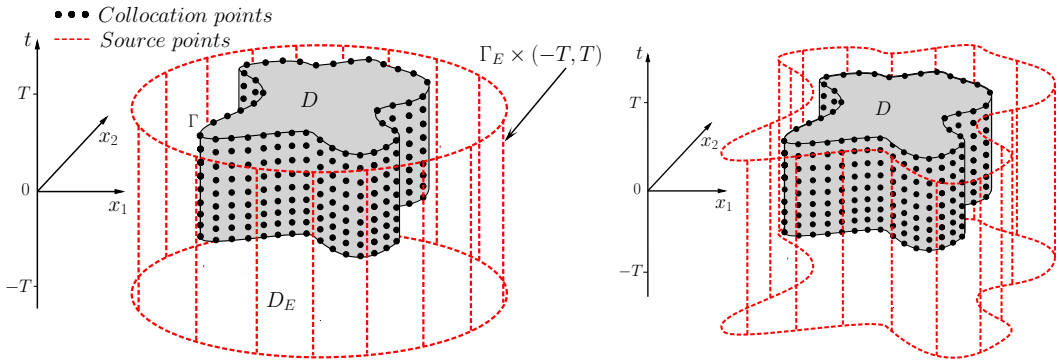

Figure 1.1: MFS for two-dimensional heat conduction, source points located outside of the spatial domain $\overline{D}$ and in time $[-T, T]$.

1.2.1 Denseness on the lateral surface

In this section we prove a denseness result on the lateral surface $\Gamma \times (-T, T)$. Before doing this we state a result from [95], which will be used to prove the linear independence part of Theorem 1.2.2.

Theorem 1.2.1. *Let Ω be a connected open set in $\mathbb{R}^n$ and $Q = \Omega \times (-T, T)$, $0 < T \leq \infty$. Suppose $u \in L^2(-T, T; H^2_{loc}(\Omega))$ satisfies the heat equation (1.1), and vanishes in the open set $\mathcal{O} \subset Q$ then u vanishes in the horizontal component of $\mathcal{O}$.*

We note that L^2 is the space of square integrable functions, $H^2_{\rm loc}$ is a local Sobolev space of order 2, $L^2(-T, T; H^2_{\rm loc}(\Omega))$ is a Bochner space, and is the space of all measurable functions $u : [0, T] \to H^2_{\rm loc}(\Omega)$ with

$$\|u\|_{L^2(-T,T;H^2_{\rm loc}(\Omega))} := \left(\int_{-T}^{T} \|u(\cdot, t)\|^2_{H^2_{\rm loc}(\Omega)} \, dt \right)^{\frac{1}{2}} < \infty,$$

see [35] for additional details. The horizontal component, defined in [95], of an open set $\mathcal{O}$ in $\mathcal{O}_\Omega$ is the union of all open sets (for t fixed) in $\mathcal{O}_\Omega$ which contain a point of $\mathcal{O}$.

We now state and prove the denseness result:

Theorem 1.2.2. *The set of functions $F(\mathbf{x}, t; \mathbf{y}_j, \tau_m)\}_{j,m=1}^{\infty}$ defined on $\Gamma \times (-T, T)$ form a linearly independent and dense set in $L^2(\Gamma \times (-T, T))$.*

Proof. A similar version of the proof of this theorem was given in one-dimension in [62] and in three-dimensions in [72], and we follow those ideas here in the two-dimensional case.

Linear independence: Assume that we do not have linear independence, then there exist positive integers N, $m_0, j_0 \in \{1, \ldots, N\}$, and a coefficient $c_{m_0}^{(j_0)} \neq 0$ such that

$$\sum_{j=1}^{N} \sum_{m=1}^{N} c_m^{(j)} F(\mathbf{x}, t; \mathbf{y}_j, \tau_m) \; = \; 0, \quad (\mathbf{x}, t) \in \Gamma \times (-T, T). \tag{1.7}$$

Define the function

$$U(\mathbf{x}, t) \; = \; \sum_{j=1}^{N} \sum_{m=1}^{N} c_m^{(j)} F(\mathbf{x}, t; \mathbf{y}_j, \tau_m), \quad (\mathbf{x}, t) \in D \times (-T, T). \tag{1.8}$$

Then U satisfies the following equations:

$$\frac{\partial U}{\partial t} - \Delta U \; = \; 0, \quad \text{in } \; D \times (-T, T), \tag{1.9}$$

$$U(\mathbf{x}, t) \; = \; 0, \quad (\mathbf{x}, t) \in \Gamma \times (-T, T), \tag{1.10}$$

$$U(\mathbf{x}, -T) \; = \; 0. \tag{1.11}$$

We have obtained the above equations by observing that the fundamental solution satisfies (1.9); the Heaviside function makes the fundamental solution equal to zero in equation (1.11), and (1.7) gives us (1.10). By the uniqueness Theorem 1.1.1, the only solution to the equations (1.9)–(1.11) is $U(\mathbf{x}, t) \equiv 0$ for $(\mathbf{x}, t) \in D \times (-T, T)$. Because the fundamental solution (1.5) is analytic in $D_E \times (-T, T)$, so is U, since linear combinations of analytic functions are analytic (F is analytic away from the source point $(\mathbf{y}_j, \tau_m)$ since the product and composition of analytic functions are analytic). We now use Theorem 1.2.1 (Theorem 1.1 of [95]). By definition, D is a connected set, $U \in L^2(-T, T; H^2_{\text{loc}}(D))$, since F is analytic, and F and the spatial derivatives of F are square integrable with respect to t, satisfies the heat equation and vanishes in $D \times (-T, T)$, therefore we also have $U(\mathbf{x}, t) = 0$ for $(\mathbf{x}, t) \in D_E \times (-T, T)$.

We now let the point $(\mathbf{x}, t)$ approach the point $(\mathbf{y}_{j_0}, \tau_{m_0}) \in \Gamma_E \times (-T, T)$ such that the ratio

$$\frac{|\mathbf{x} - \mathbf{y}_{j_0}|^2}{4(t - \tau_{m_0})} \tag{1.12}$$

remains bounded. Then the summand $c_{m_0}^{(j_0)} F(\mathbf{x}, t; \mathbf{y}_{j_0}, \tau_{m_0})$ in (1.8) may be made as large as we wish, while the other terms in the series (1.8) remain bounded; this gives us a contradiction and thus, we have linear independence for the set of functions $\{F(\mathbf{x}, t; \mathbf{y}_j, \tau_m)\}_{j,m=1}^{\infty}$ in $L^2(\Gamma \times (-T, T))$.

Denseness: We next prove that the sequence $\{F(\mathbf{x},t;\mathbf{y}_j,\tau_m)\}_{j,m=1}^{\infty}$ is a dense set in $L^2(\Gamma\times(-T,T))$. Assume on the contrary that it is not a dense set. Then there exists an element $f(\mathbf{x},t)$ in $L^2(\Gamma\times(-T,T))$, which we can assume is continuous, such that

$$\int_{-T}^{T}\int_{\Gamma} F(\mathbf{x},t;\mathbf{y}_j,\tau_m)f(\mathbf{x},t)\,d\mathbf{x}\,dt \;=\; 0,\quad j,m=1,2,\ldots \tag{1.13}$$

To show that $\{F(\mathbf{x},t;\mathbf{y}_j,\tau_m)\}$ is dense, we have to show that $f(\mathbf{x},t)\equiv 0$ in (1.13). Equation (1.13) can be rewritten as

$$\int_{\tau_m}^{T}\int_{\Gamma} F(\mathbf{x},t;\mathbf{y}_j,\tau_m)f(\mathbf{x},t)\,d\mathbf{x}\,dt \;=\; 0,\quad j,m=1,2,\ldots, \tag{1.14}$$

where the Heaviside function in (1.5) has been used to reduce the range of integration with respect to the time variable. We introduce an equivalent form of the classical single-layer heat potential given by

$$V(\mathbf{y},\tau) \;=\; \int_{\tau}^{T}\int_{\Gamma} F(\mathbf{x},t;\mathbf{y},\tau)f(\mathbf{x},t)\,d\mathbf{x}\,dt. \tag{1.15}$$

It is well-known that $V(\mathbf{y},\tau)$ is a smooth solution to the heat equation in the exterior of $\overline{D}\times(-T,T)$, see [38, p. 136], and it cannot vanish on any surface in this exterior region without being identically zero. Thus, by the continuity of F and (1.14), we find that $V(\mathbf{y},\tau)=0$ for $(\mathbf{y},\tau)\in\Gamma_E\times(-T,T)$, which is in the exterior of $\overline{D}\times(-T,T)$; we then conclude that $V=0$ in the exterior of $\overline{D}\times(-T,T)$. Moreover, since V is continuous across $\Gamma\times[-T,T]$ we also have $V(\mathbf{y},\tau)=0$ on $\Gamma\times[-T,T]$. This implies that $V=0$ also in $\overline{D}\times(-T,T)$ since V satisfies the heat equation in $D\times(-T,T)$. Finally, using the jump relations for the normal derivative of V on $\Gamma\times[-T,T]$, see [38, p. 133], we get

$$\frac{1}{2}f(\mathbf{x},t)\pm\frac{\partial V}{\partial\boldsymbol{\nu}}(\mathbf{x},t) \;=\; 0,\quad (\mathbf{x},t)\in\Gamma\times(-T,T), \tag{1.16}$$

where $\boldsymbol{\nu}$ represents the unit normal on the surface $\Gamma\times(-T,T)$. Thus, $f\equiv 0$ and therefore, $\{F(\mathbf{x},t;\mathbf{y}_j,\tau_m)\}_{j,m=1}^{\infty}$ is a dense set in $L^2(\Gamma\times(-T,T))$. □

1.2.2 Denseness on the base surface

We now show that we also have denseness on the "base" surface $D \times \{0\}$, where the initial condition is imposed in (1.1)–(1.3). We require the following result, taken from [66] to help prove the linear independence part of Theorem 1.2.3.

Corollary of Theorem 3 in [66]. *Let u be continuous in $\mathbb{R}^2 \times [T_0, T_1]$ and twice continuously differentiable in $\mathbb{R}^2 \times (T_0, T_1]$ such that u satisfies* (1.1) *and*

$$|u(\mathbf{x}, t)| \leq Be^{\beta \mathbf{x}^2}, \quad T_0 \leq t \leq T_1. \tag{1.17}$$

If $|u(\mathbf{x}, T_0)| \leq \epsilon$, then $|u(\mathbf{x}, t)| \leq \epsilon$ for $T_0 \leq t \leq T_1$.

We now state and prove the denseness result:

Theorem 1.2.3. *The set of functions $\{F(\mathbf{x}, 0; \mathbf{y}_j, \tau_m)\}_{j,m=1}^{\infty}$, with $\tau_m < 0$, forms a linearly independent and dense set in $L^2(D)$.*

Proof. The method of proof is similar to that used in [62] in one-dimension, and we give it here, for completeness, in higher dimensions.

Linear independence: Assume that we do not have linear independence, then there exist positive integers N, $m_0, j_0 \in \{1, \ldots, N\}$, and a coefficient $c_{m_0}^{(j_0)} \neq 0$ such that

$$\sum_{j=1}^{N} \sum_{m=1}^{N} c_m^{(j)} F(\mathbf{x}, 0; \mathbf{y}_j, \tau_m) \, = \, 0, \quad \mathbf{x} \in D. \tag{1.18}$$

From (1.8) and (1.18) we have that

$$U(\mathbf{x}, 0) \, = \, 0, \quad \mathbf{x} \in D. \tag{1.19}$$

We also have that U satisfies (1.9) and (1.11). Now, since $U(x_1, x_2, 0)$, where $\mathbf{x} = (x_1, x_2)$, is a real analytic function in each of the variables x_1 and x_2, we find that $U(\mathbf{x}, 0) = 0$ for every $\mathbf{x} \in \mathbb{R}^2$. Moreover, since each $\tau_m < 0$, U is continuous on $\mathbb{R}^2 \times [0, T]$ and is at least twice continuously differentiable in $\mathbb{R}^2 \times [0, T]$. Furthermore, U also satisfies the heat equation (1.1), and the following inequality holds

$$|U(\mathbf{x}, t)| \, \leq \, Be^{\beta |\mathbf{x}|^2}, \quad t \in [0, T], \tag{1.20}$$

for some positive constants B and β. We can see this if we set

$$B = \frac{N^2 c}{\bar{t}}, \quad \text{where } c = \max(|c_m^{(j)}|) \text{ and } \bar{t} = \min(|\tau_m|), \quad j, m = 1, \ldots, N,$$

and $\beta = 1$, then for $(\mathbf{x}, t) \in \mathbb{R}^2 \times [0, T]$ and $\tau_m < 0$,

$$|U(\mathbf{x}, t)| \leq \sum_{j=1}^{N} \sum_{m=1}^{N} |c_m^{(j)}| \frac{H(t - \tau_m)}{4\pi(t - \tau_m)} e^{-\frac{|\mathbf{x} - \mathbf{y}_j|^2}{4(t-\tau_m)}} \leq \frac{N^2 c}{\bar{t}} e^{|\mathbf{x}|^2} = B e^{\beta |\mathbf{x}|^2}.$$

Thus, from Theorem 3 in [66], and its corollary, we conclude that $U(\mathbf{x}, t) = 0$ in $\mathbb{R}^2 \times [0, T]$. In particular, from [95], we may extend U such that $U(\mathbf{x}, t) = 0$ also in $D_E \times [-T, T]$.

We now let the point $(\mathbf{x}, t)$ approach the point $(\mathbf{y}_{j_0}, \tau_{m_0}) \in \Gamma_E \times [-T, 0]$ such that the ratio (1.12) remains bounded. Now, the summand $c_{m_0}^{(j_0)} F(\mathbf{x}, t; \mathbf{y}_{j_0}, \tau_{m_0})$ may be made as large as we wish, while the other terms in the series (1.18) remain bounded; this gives a contradiction and we have linear independence for the set of functions $\{F(\mathbf{x}, 0; \mathbf{y}_j, \tau_m)\}$ in $L^2(D)$.

Denseness: We shall show that the set of functions $\{F(\mathbf{x}, 0; \mathbf{y}_j, \tau_m)\}$, where $\tau_m < 0$, is a dense set in $L^2(D)$. Assume that this is not a dense set, then there exists a function $f \in C^2(D)$ such that

$$\int_D F(\mathbf{x}, 0; \mathbf{y}_j, \tau_m) f(\mathbf{x})\, d\mathbf{x} \;=\; 0, \quad j, m = 1, 2, \ldots \tag{1.21}$$

We let w be a weak solution of the heat equation (1.1), see [35], with initial condition $w(\mathbf{x}, 0) = f(\mathbf{x})$ and boundary condition $w(\mathbf{x}, t) = 0$ for $(\mathbf{x}, t) \in \Gamma_T$. We may transform (1.21) using Green's identities, see [35, 72], into the following form

$$\int_0^T \int_\Gamma F(\mathbf{x}, t; \mathbf{y}_j, \tau_m) \frac{\partial w}{\partial \boldsymbol{\nu}}(\mathbf{x}, t)\, d\mathbf{x} dt \;=\; 0, \quad j, m = 1, 2, \ldots$$

where $\boldsymbol{\nu}$ is the outward pointing unit normal to Γ.

From Theorem 1.2.2 we know that $\{F(\mathbf{x}, t; \mathbf{y}_j, \tau_m)\}$ restricted on Γ_T is a dense set in $L^2(\Gamma_T)$, and we may conclude that the normal derivative of w is zero on Γ_T. Therefore, both w and $\frac{\partial w}{\partial \boldsymbol{\nu}}$ are zero on Γ_T. From [95, p. 132], we conclude that $w(\mathbf{x}, t) = 0$ for $(\mathbf{x}, t) \in \overline{D}_T$; hence $f \equiv 0$, and $\{F(\mathbf{x}, 0; \mathbf{y}_j, \tau_m)\}$, where $\tau_m < 0$, is a dense set in $L^2(D)$. □

1.3 The MFS for direct heat conduction in two-dimensions

The denseness results proved in the previous section, Theorems 1.2.2 and 1.2.3, which involved linear combinations of the fundamental solution (1.5) of the heat equation (1.1), enable us to describe a method for approximating the solution to the problem (1.1)–(1.3). We note that the MFS we propose may be applied to domains of general shape and size and the source points may also be placed arbitrarily, for example, placed symmetrically on a circular pseudo-boundary, or to match similarly the general shape of the domain D, see Figure 1.2 and [45]. The only restriction on the placement of the source points is that they are located on the boundary Γ_E outside the domain D, and placed relatively close to D such that u has no singularity in $D_E \times [0,T]$. Also, it might be more practical to take the sources on the interval $(-\epsilon, T)$, where $\epsilon > 0$ is small, instead of the full interval $(-T, T)$. However, some preliminary numerical investigations in [59] for the backward heat conduction problem showed that ϵ cannot be chosen too small if no loss in accuracy and stability is to be secured.

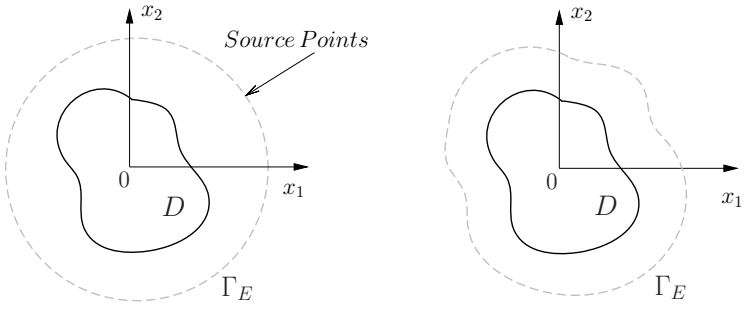

Figure 1.2: An arbitrary domain with the two panels showing schematically different source point locations relative to the domain boundaries.

We search for an approximation to the solution of equations (1.1)–(1.3) in the following form:

$$u_{M,N}(\mathbf{x},t) \;=\; \sum_{m=1}^{2M} \sum_{j=1}^{N} c_m^{(j)} F(\mathbf{x},t;\mathbf{y}_j,\tau_m), \quad (\mathbf{x},t) \in \overline{D}_T. \tag{1.22}$$

For simplicity, let us describe the MFS in the case of circular domains. We shall also consider square domains in the next section, see Examples 4 and 5.

We consider a two-dimensional circular domain D, with boundary Γ and radius $r_0 > 0$, centred at the origin, and let us place the source points $(\mathbf{y}_j)_{j=1,\ldots,N}$ on a circle $r_0 + h$, $h > 0$, also centred at the origin. We will choose a reasonable value for the parameter $h > 0$.

Take the time points $(\tau_m)_{m=1,\ldots,2M}$ (each in the interval $(-T,T)$) as given by

$$\tau_m \;=\; \frac{2(m-M)-1}{2M}T, \quad m=1,\ldots,2M, \tag{1.23}$$

and using polar coordinates, place the source points in space at

$$\mathbf{y}_j \;=\; (r_0+h,\theta_j) \;=\; \left(r_0+h,\frac{2\pi j}{N}\right), \quad j=1,\ldots,N.$$

In polar coordinates equation (1.22) is now represented by

$$u_{M,N}(r,\theta,t) \;=\; \sum_{m=1}^{2M}\sum_{j=1}^{N} c_m^{(j)} F(r,\theta,t;r_0+h,\theta_j,\tau_m). \tag{1.24}$$

We have located $2M\times N$ source points in total outside the domain D and in time; in order to obtain a unique solution we place the same number of collocation points in total on $\overline{\Gamma}_T\cup(D\times\{0\})$, the boundary in time and the domain at time $t=0$. Note that it is not necessary to place the same number of collocation and source points, see Section 0.5 of the Introduction. Of course, the location of source and collocation points may be chosen arbitrarily, here we choose points for ease of calculation. Let

$$t_i \;=\; \frac{i}{M}T, \quad i=0,\ldots,M,$$

and on Γ set

$$(r_0,\theta_k) \;=\; \left(r_0,\frac{2\pi k}{N}\right), \quad k=1,\ldots,N.$$

We have located $(M+1)\times N$ collocation points on the boundary. The remaining $(M-1)\times N$ points will be located on D when $t=0$. We consider $M-1$ circles of radius

$$r_l \;=\; \left(\frac{l}{M}\right)^{\frac{1}{2}} r_0, \quad l=1,\ldots,M-1,$$

where the square root has been introduced to spread the points out within the domain, and not to cluster them at the centre. We place N equally spaced points on each circle such that

$$(r_l,\theta_k) \;=\; \left(r_l,\frac{2\pi k}{N}\right), \quad k=1,\ldots,N,$$

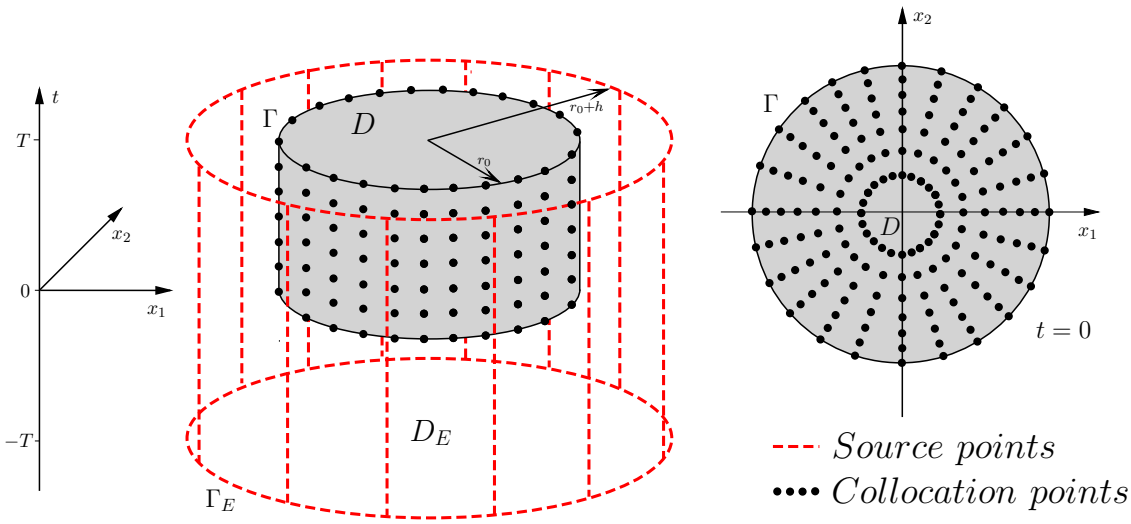

Figure 1.3: Location of source and collocation points when D is a circular domain.

see Figure 1.3 for a detailed graphical representation of the position of the various source and collocation points given above. Alternatively we could have equally distributed the collocation points on the initial base whilst making sure we have at most $(M-1)\times N$ points for a square or overdetermined system.

We now impose the boundary and initial conditions (1.2) and (1.3) so that we can determine the unknown coefficients $c_m^{(j)}$ in (1.24). In polar coordinates we obtain the equations

$$u_{M,N}(r_0,\theta_k,t_i) \;=\; h(r_0,\theta_k,t_i), \tag{1.25}$$

$$u_{M,N}(r_l,\theta_k,0) \;=\; u_0(r_l,\theta_k,0), \tag{1.26}$$

where $k=1,\ldots,N$, $i=0,\ldots,M$ and $l=1,\ldots,M-1$.

The system of equations (1.25) and (1.26) contains $(M-1)\times N+(M+1)\times N = 2MN$ equations and $2MN$ unknowns, therefore, we may obtain a unique solution. We can represent this system of equations as

$$A\mathbf{c} \;=\; \mathbf{g}, \tag{1.27}$$

where $\mathbf{c}$ is the vector of unknowns $c_m^{(j)}$, $\mathbf{g}$ is the vector representing the values of the functions u_0 and h at the respective collocation points, and A is the matrix corresponding to the value of the fundamental solution at the points outlined above. For certain distributions of boundary collocation and source points might be possible to use the properties of circulant matrices to develop a matrix decomposition algorithm [12] for the solution of the system of equations (1.27)

and thus, substantially reduce the computational cost.

In this Chapter we will apply Tikhonov regularization and solve (0.45) using L_0, see Section 0.5 of the Introduction for details.

1.4 Numerical results

In [62] it was shown that the direct MFS approximation applied to the one-dimensional heat equation with source points located outside the domain and in time appeared to be accurate. Below, we present numerical results for approximations in two-dimensional domains, such as circular and square domains. In order to assess the accuracy of the numerical MFS solutions we compare them with the available exact solutions for various benchmark test examples. Numerical results are presented for $N = 20$ and $M = 30$, which were found to be sufficiently large to ensure that any further increase in these numbers did not significantly improve the accuracy of the numerical solution.

1.4.1 Example 1

Let $D = \{\mathbf{x} : |\mathbf{x}|^2 < 1\}$, $D_T = \{(\mathbf{x}, t) : |\mathbf{x}|^2 < 1, t \in (0, 1]\}$, and $\Gamma_T = \{(\mathbf{x}, t) : |\mathbf{x}|^2 = 1, t \in (0, 1]\}$. We solve the following problem, using the direct MFS laid out in the previous section,

$$\frac{\partial u}{\partial t}(\mathbf{x}, t) - \Delta u(\mathbf{x}, t) = 0, \quad (\mathbf{x}, t) \in D_T, \tag{1.28}$$

$$u(\mathbf{x}, t) = 4t + 1, \quad (\mathbf{x}, t) \in \Gamma, \tag{1.29}$$

$$u(\mathbf{x}, 0) = |\mathbf{x}|^2, \quad \mathbf{x} \in D. \tag{1.30}$$

The exact solution of problem (1.28)–(1.30) is $u(\mathbf{x}, t) = 4t + |\mathbf{x}|^2$. The source points are placed on a circle with radius $1 + h$. The value of $h > 0$ will be chosen appropriately. However, the accuracy of the approximation appears to decrease when $h < 0.25$ or $h > 4$. In Figure 1.5 the exact solution and the MFS approximations are plotted in one-dimension, $\mathbf{x} = (x_1, 0)$, for times $t \in \{0.2, 0.8\}$ with $\lambda = 10^{-8}$ in the Tikhonov regularization. In Figure 1.4 plots have been produced when $\lambda = 0$, and we find that the approximation seems stable, however, we find in Figure 1.5 the approximation is more accurate for $\lambda = 10^{-8}$ (and for other small values of λ).

Figure 1.6 contains plots of the exact solution and the direct MFS approximations for $h \in \{0.5, 4\}$ and $\lambda = 10^{-8}$. From Figure 1.6(b) it can be seen that the numerical results obtained

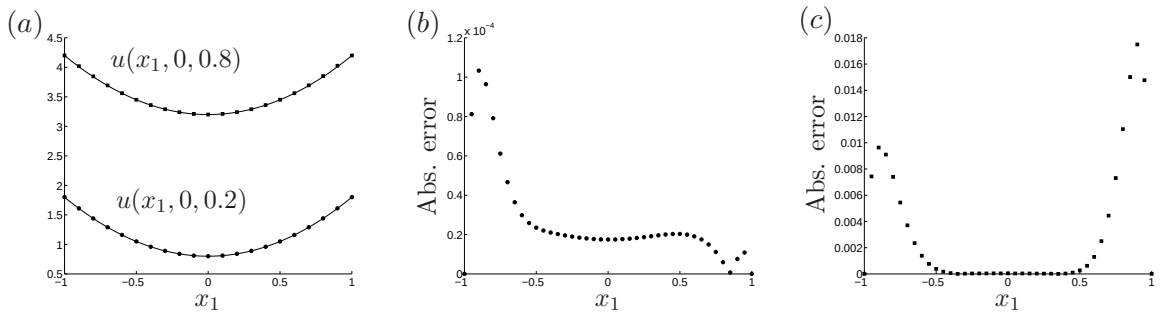

Figure 1.4: (a) The exact solution (—) and the approximate values for $u(x_1, 0, 0.2)$ (•) and $u(x_1, 0, 0.8)$ (▪), and (b) for $t = 0.2$ and (c) for $t = 0.8$, the corresponding absolute error plots. Both plots obtained for $h = 1$ and $\lambda = 0$, for Example 1.

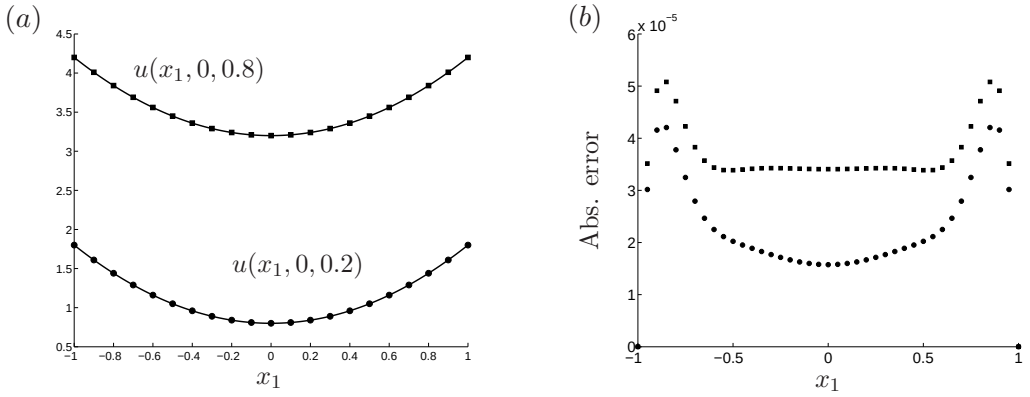

Figure 1.5: (a) The exact solution (—) and the approximate values for $u(x_1, 0, 0.2)$ (•) and $u(x_1, 0, 0.8)$ (▪), and (b) the corresponding absolute error plots. Both plots obtained for $h = 1$ and $\lambda = 10^{-8}$, for Example 1.

with $h = 0.5, 4$ are less accurate than those produced with $h = 1$ in Figure 1.5.

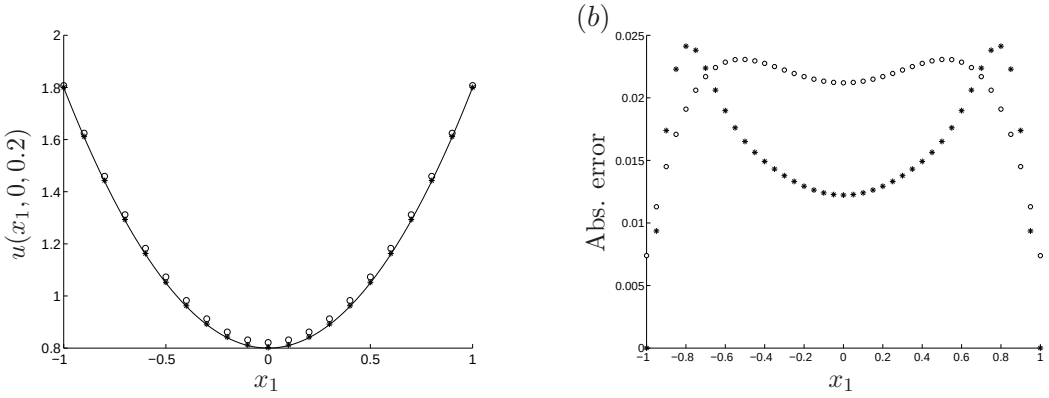

Figure 1.6: (a) The exact solution (—) and the approximate values for $u(x_1, 0, 0.2)$ and (b) the corresponding absolute error plots, obtained with $h = 0.5$ ($*$) and $h = 4$ ($\circ$), for Example 1.

Finally, we consider a three-dimensional plot of the exact solution $u(x_1, x_2, 0.8)$ in Figure 1.7(a), and the MFS approximation $u_{M,N}$ in Figure 1.7(b) obtained with $h = 1$. Figure 1.7(c)

shows the graph of the absolute error $|u(\mathbf{x}, 0.8) - u_{M,N}(\mathbf{x}, 0.8)|$, and we note that the approximation is very accurate with a maximum absolute error of $\mathcal{O}(10^{-5})$.

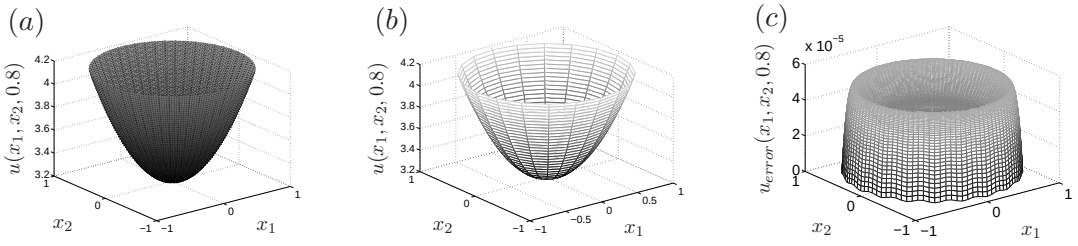

Figure 1.7: Three-dimensional plot of: (a) The exact solution $u(x_1, x_2, 0.8)$, (b) the approximate solution $u_{M,N}$ obtained with $\lambda = 10^{-8}$ and $h = 1$, and (c) the absolute error, for Example 1.

1.4.2 Example 2

In this example we choose the same D, D_T and Γ as in Example 1, but instead of $u(\mathbf{x}, t) = 4t + |\mathbf{x}|^2$, we consider the exact solution of the equations (1.1)–(1.3) given by

$$u(\mathbf{x}, t) \;=\; e^{x_1+x_2}\cos(x_1 + x_2 + 4t), \tag{1.31}$$

where the boundary and initial equations (1.2) and (1.3) have been obtained from (1.31). Note that this function is not constant on circles centred at the origin, and not symmetric, thus being different in character compared with the solution in Example 1.

Again, the source points will be placed on a circle with radius $1 + h$, with final time point $T = 1$. In Figure 1.8 we present a plot of the L-curve for Example 2, and in this example we take $\lambda = 10^{-8}$.

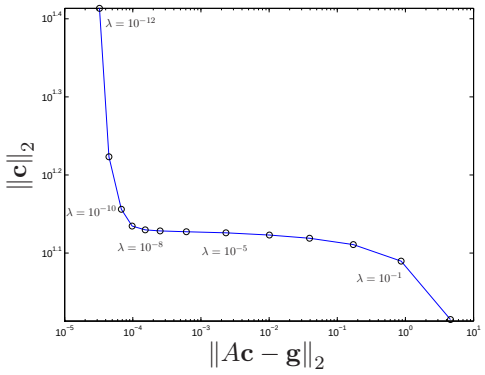

Figure 1.8: Plot of the L-curve, for Example 2.

In Figure 1.9 the exact solution and the direct MFS approximations for $\lambda = 10^{-8}$ are plotted in one-dimension, $\mathbf{x} = (x_1, 0)$, for times $t = \{0.2, 0.8\}$, with $h = 1$. From Figure 1.9 we see that the approximation appears accurate. Also, in this example varying λ between $[10^{-12}, 10^{-4}]$ only slightly changes the accuracy of the approximation, with the maximum and mean absolute errors remaining of the $\mathcal{O}(10^{-4})$ and $\mathcal{O}(10^{-5})$, respectively, for this range of values.

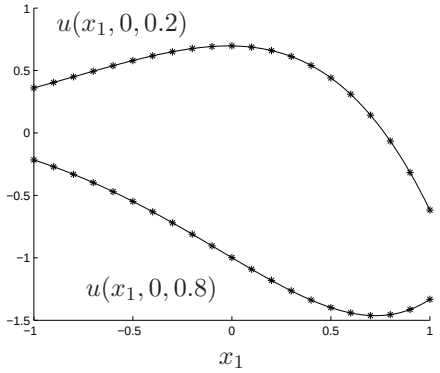

Figure 1.9: The exact (—) and the approximate values ($*$) for $u(x_1, 0, 0.2)$ and $u(x_1, 0, 0.8)$, for Example 2.

We now consider a three-dimensional plot of the exact solution $u(x_1, x_2, 0.8)$ in Figure 1.10(a), the MFS approximation $u_{M,N}$ in Figure 1.10(b), and the absolute error in Figure 1.10(c), obtained with $h = 1$.

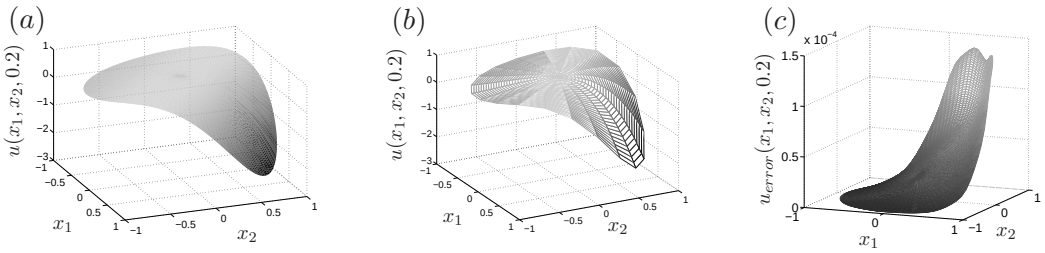

Figure 1.10: Three-dimensional plot of: (a) The exact solution $u(x_1, x_2, 0.2)$, (b) the approximate solution $u_{M,N}$ obtained with $\lambda = 10^{-8}$ and $h = 1$, and (c) the absolute error, for Example 2.

We finish this example by considering a different placement of source points, to see if there is justification for the current choice. The thought is that the placement of source points below $\tau = 0$ might affect the conditioning and hence accuracy. Instead of placing source points in time in the interval $[-T, T]$, which was defined in 1.23, we take the source points, in the interval

$[-\tilde{T}, T]$, where $\tilde{T} > 0$, as given by

$$\tilde{\tau}_m = \begin{cases} \frac{2(m-M)-1}{2M}\tilde{T}, & m = 1, \ldots, M, \\ \frac{2(m-M)-1}{2M}T, & m = M+1, \ldots, 2M. \end{cases}$$

In Figures 1.11(a) and 1.11(b) we take $\tilde{T} = 0.5$ and $\tilde{T} = 0.25$, respectively. We note that for $\tilde{\tau}_m$, as in the definition of τ_m, we have placed the same number of source points above and below time $\tau = 0$. Figure 1.11 shows that, for this example, changing the time interval does not improve the approximation, in fact from Figure 1.11(b) we can see that when we place source points in the interval $[-T/4, T] = [-0.25, 1]$ the error increases.

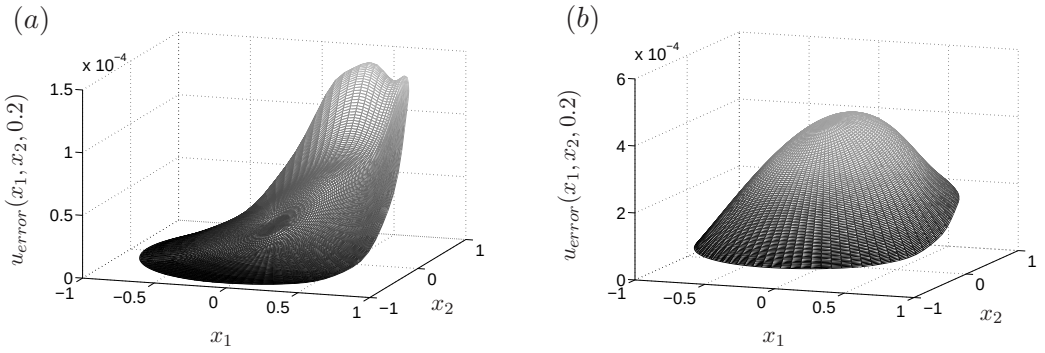

Figure 1.11: Three-dimensional plots of the absolute error $|u(x_1, x_2, 0.2) - u_{M,N}(x_1, x_2, 0.2)|$ when the MFS approximation $u_{M,N}$ has been obtained with $\lambda = 10^{-8}$, $h = 1$ and (a) $\tau_m \in [-T/2, T]$ and (b) $\tau_m \in [-T/4, T]$, for Example 2.

In Figures 1.12–1.15 we investigate further the placement of source points in time by producing error plots, as well as plots of the condition number, for more values of $\tilde{T}$ and at the time points $t = 0.2$ and $t = 0.8$. In Figures 1.12(a) and 1.12(b) we see that the approximation is worse for very small values of $\tilde{T}$, but stabilises after $\tilde{T} \approx 0.25$. The plot of the condition number of the corresponding matrices in Figure 1.12(c) show that there is no improvement, or correspondence between accuracy and conditioning in this example when it comes to this placement of source points in time.

In Figure 1.13 we find that for larger values of $\tilde{T} > 20$ the absolute value of the error gradually increases, and, similar to 1.12(c), the condition number is of the order 10^{25}, even for very large values of $\tilde{T}$.

In Figures 1.14 and 1.15 we produce error plots for $t = 0.8$, and obtain similar results to those found in Figures 1.12 and 1.13, although we do not have the gradual increase in the error

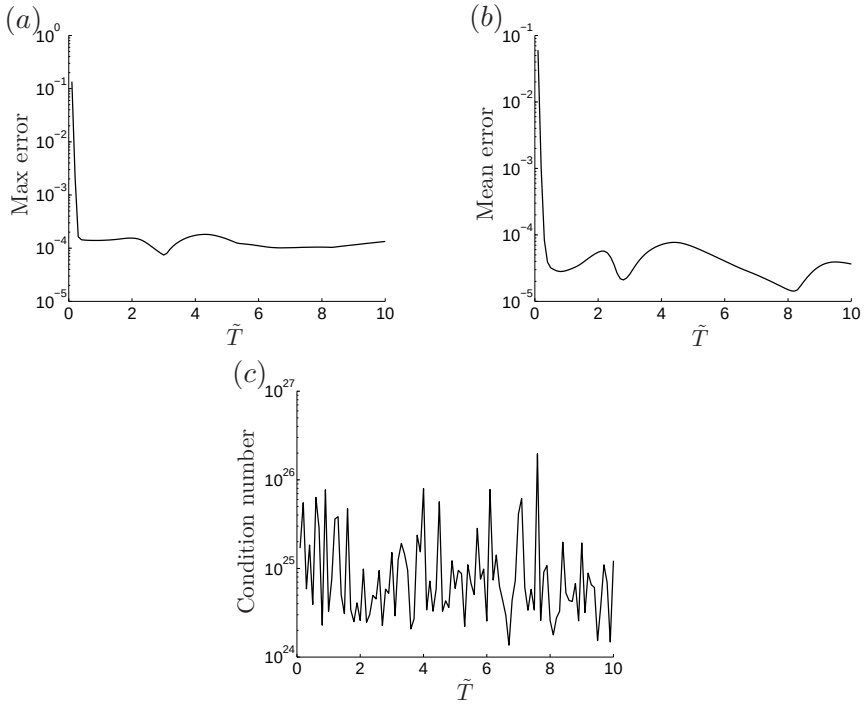

Figure 1.12: Plots of (a) the maximum absolute error, (b) the mean absolute error and (c) the condition number of the corresponding matrix at the time point $t = 0.2$ for $\tilde{T} \in (0, 10]$, when the MFS approximation has been obtained with $\lambda = 10^{-8}$ and $h = 1$, for Example 2.

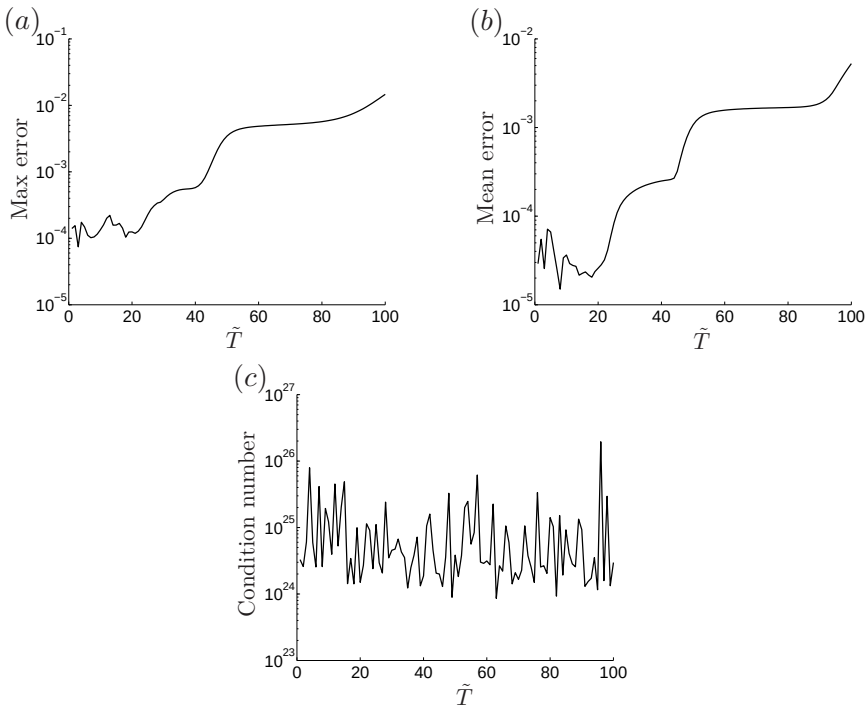

Figure 1.13: Plots of (a) the maximum absolute error, (b) the mean absolute error and (c) the condition number of the corresponding matrix at the time point $t = 0.2$ for $\tilde{T} \in (0, 100]$, when the MFS approximation has been obtained with $\lambda = 10^{-8}$ and $h = 1$, for Example 2.

as $\tilde{T}$ increases, and seems to stabilise for $\tilde{T} > 100$.

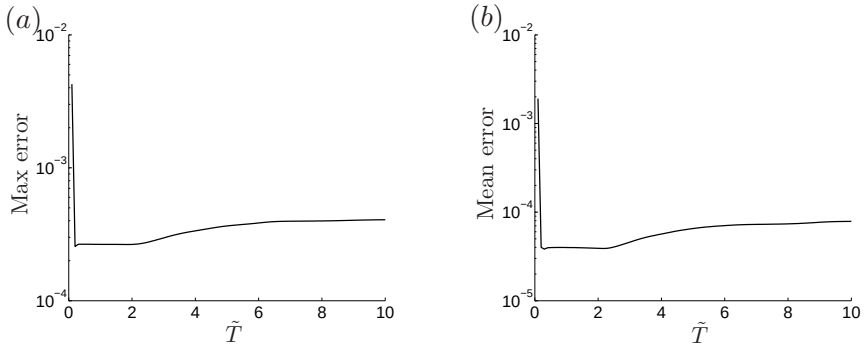

Figure 1.14: Plots of (a) the maximum absolute error, (b) and the mean absolute error, at the time point $t = 0.8$ for $\tilde{T} \in (0, 10]$, when the MFS approximation has been obtained with $\lambda = 10^{-8}$ and $h = 1$, for Example 2.

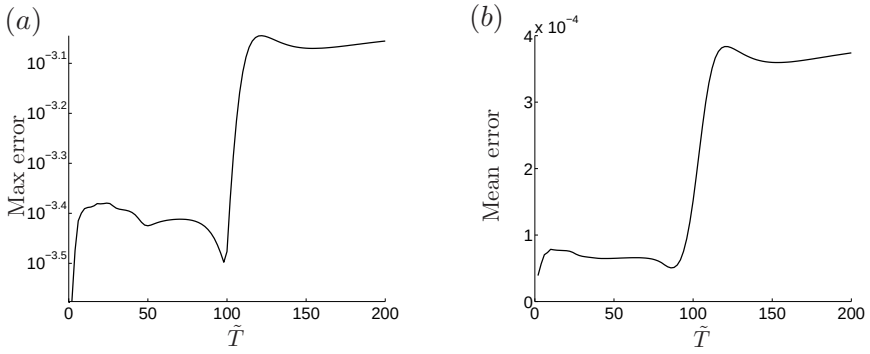

Figure 1.15: Plots of (a) the maximum absolute error, (b) and the mean absolute error, at the time point $t = 0.8$ for $\tilde{T} \in (0, 200]$, when the MFS approximation has been obtained with $\lambda = 10^{-8}$ and $h = 1$, for Example 2.

The plots obtained in Examples 1 and 2 show that the MFS approximation is accurate in circular domains, with errors usually in the interval $[10^{-4}, 10^{-2}]$, for a wide range of parameters h and λ.

1.4.3 Example 3

In this example we consider a domain D with boundary

$$\Gamma = \{(x_1, x_2) : (x_1, x_2) = \big(\cos(s) + 0.65\cos(2s) - 0.65, 1.5\sin(s)\big), \quad s \in [0, 2\pi)\}, \tag{1.32}$$

and $\Gamma_T = \Gamma \times (0, 1]$. The exact solution (1.31) from Example 2 will be used again, and we generate the initial and boundary conditions from this equation. We note that all source points from now on will be extended to $-T$ only.

Figures 1.16(a) and 1.16(c) show plots of the boundary (-●-) and the source point locations (-●-) when source points have been placed to take the shape of the boundary Γ by dilatation, so that $\Gamma_E = h\Gamma$, and on a circle of radius h, centred at $(-1/2, 0)$, respectively. Figures 1.16(b) and 1.16(d) are three-dimensional plots of the absolute error when source points have been placed as in Figures 1.16(a) and 1.16(c), respectively.

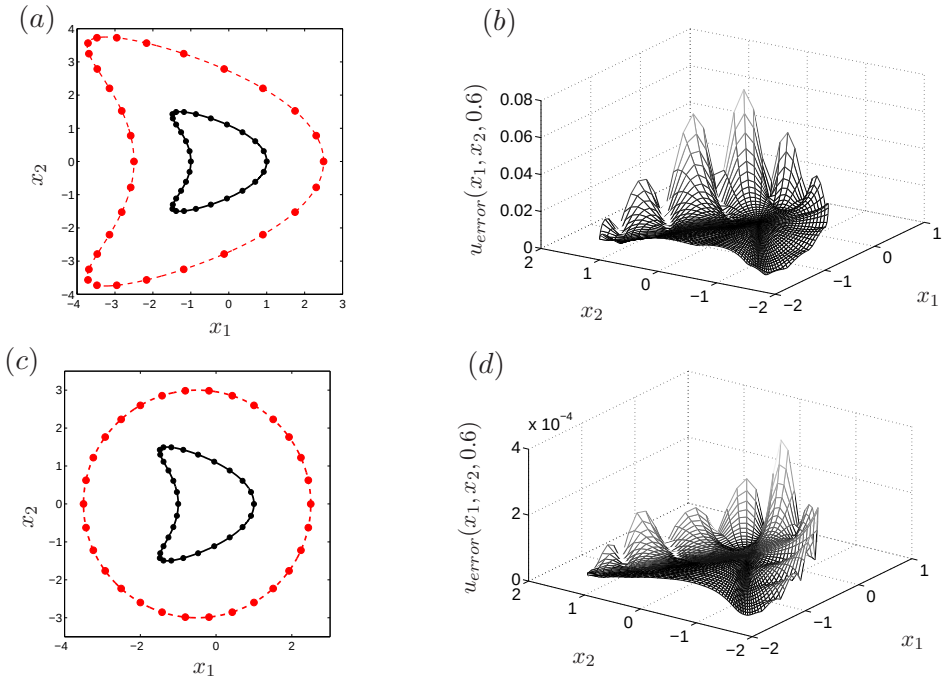

Figure 1.16: (a) Source points placed on $\Gamma_E \times (-T, T)$ and (b) the plot of the absolute error, $h = 2.5$, $t = 0.6$, $\lambda = 10^{-5}$, $N = 20$ and $M = 30$. (c) Source points placed on a circle of radius $h = 3$ centred at $(-1/2, 0)$ and (d) the plot of the absolute error, $t = 0.6$, $\lambda = 10^{-8}$, $N = 20$ and $M = 30$, for Example 3.

We have used an L-curve plot to obtain the regularization parameter λ. Figure 1.16 illustrates the accuracy of the MFS approximation when source points are placed on a circle with a maximum absolute error of $\mathcal{O}(10^{-4})$, however, results are not as good in Figure 1.16(b), where the maximum error is $\mathcal{O}(10^{-2})$. The oscillatory behaviour in Figure 1.16 at the boundary was examined further and we found that increasing N and changing the distribution of points on the boundary (and changing the distribution of source points on Γ_E in Figure 1.16(a)) gave better results. In particular we placed source points equally with respect θ, the angle, however we obtain a cluster of points in certain regions, see Figure 1.16(a) and (c).

1.4.4 Example 4

In the next two examples we consider square domains with edge length l. In Examples 4 and 5 source points will be placed on both squares, see Figure 1.17, as well as circles; we vary the shapes where we place the source points to highlight that the placement of the source points do not need to follow the shape of Γ. The numerical implementation is the same as in the previous examples, including the placement of the sources and boundary collocation points and their numbers $N = 20$, $M = 30$.

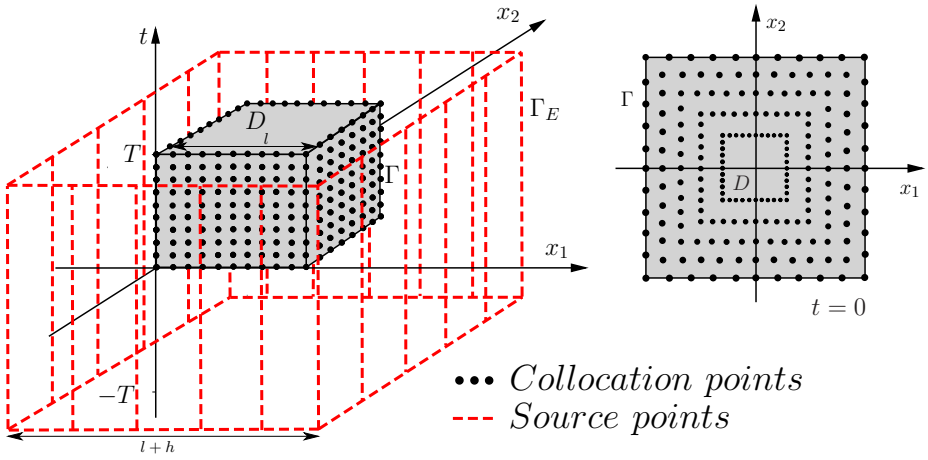

Figure 1.17: MFS in two-dimensions for square domains.

The following problem was considered in [41] and will be solved using Tikhonov regularization only. We take $D = (-0.2, 0.2) \times (-0.2, 0.2)$ to be a square of edge length $l = 0.4$, take $T = 0.9$, and solve

$$\frac{\partial u}{\partial t}(\mathbf{x}, t) - \alpha \Delta u(\mathbf{x}, t) = 0, \quad (\mathbf{x}, t) \in D_T = D \times (0, 0.9], \tag{1.33}$$

$$u(\mathbf{x}, t) = 0, \quad (\mathbf{x}, t) \in \Gamma_T = \partial D \times (0, 0.9], \tag{1.34}$$

$$u(\mathbf{x}, 0) = 1, \quad \mathbf{x} \in D. \tag{1.35}$$

Here, we have a very large value for the thermal diffusivity $\alpha = \frac{1000}{5.8}$, and the fundamental solution of equation (1.33) is now given by

$$F(\mathbf{x}, t; \mathbf{y}, \tau) = \frac{H(t-\tau)}{4\pi\alpha(t-\tau)} e^{-\frac{|\mathbf{x}-\mathbf{y}|^2}{4\alpha(t-\tau)}}. \tag{1.36}$$

We note that in this example the compatibility conditions (1.4) are violated. The exact solution

to the problem (1.33)–(1.35) is given by, see [25],

$$u(x_1,x_2,t)=\frac{16}{\pi^2}\left[\sum_{n=0}^{\infty}\frac{(-1)^n}{2n+1}e^{-\alpha(2n+1)^2t\pi^2/(4(l/2)^2)}\cos\left(\frac{(2n+1)\pi x_1}{2(l/2)}\right)\right]\times \left[\sum_{m=0}^{\infty}\frac{(-1)^m}{2m+1}e^{-\alpha(2m+1)^2t\pi^2/(4(l/2)^2)}\cos\left(\frac{(2m+1)\pi x_2}{2(l/2)}\right)\right]. \tag{1.37}$$

However, when we plot the exact (using 100 terms in the series expansion (1.37)) and the approximate solutions at the point $(x_1,x_2)=(0,0)$ for time $t\in[0,0.9]$, with source points placed on a square with edge length $0.4+h$, $h=5$, $\lambda=10^{-8}$, we get a large discrepancy, see Figure 1.18.

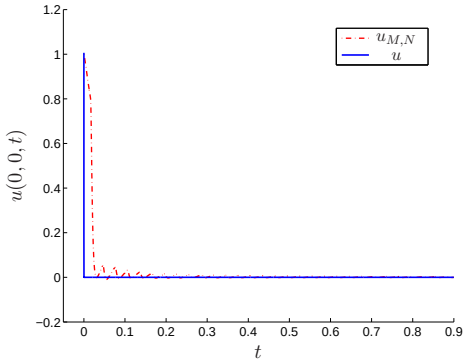

Figure 1.18: The exact solution $u(0,0,t)$ and the approximation $u_{M,N}$, as functions of time $t\in[0,0.9]$, for Example 4.

Changing the parameter λ in the Tikhonov regularization (0.45) did not seem to improve the approximation, however, we observe that the exact solution (1.37) decays very rapidly due to the exponential terms. This means that we should consider a much smaller time interval. Figure 1.19 shows the exact and the approximate solutions, with final time point $T=0.0006$, plotted over $t\in[0,0.0004]$, where $\lambda=10^{-6}$ and source points have now been placed on a circle of radius $h=0.84$.

These figures show that the choice of the final time T, in particular when considering rapidly decaying functions, is also important when implementing the MFS. Time-marching methods, [103], could then perhaps be used to extend our approximation to larger time intervals.

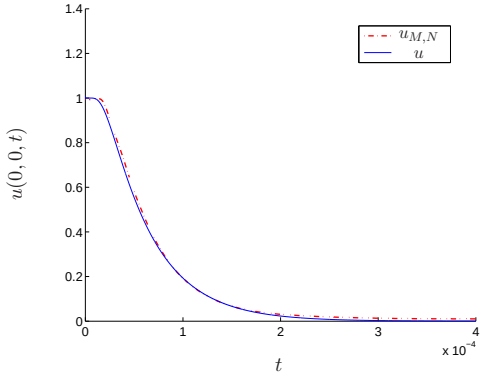

Figure 1.19: The exact solution $u(0,0,t)$ and the approximation $u_{M,N}$ with $T = 0.0006$, for Example 4.

1.4.5 Example 5 (inverse problem)

We consider $D = (0,1) \times (0,1)$ and $D_T = D \times (0,3]$, and we wish to solve the following problem:

$$\frac{\partial u}{\partial t}(\mathbf{x},t) - \Delta u(\mathbf{x},t) = 0, \quad (\mathbf{x},t) \in D_T, \tag{1.38}$$

$$u(x_1,0,t) = u(x_1,1,t) = \sqrt{2}e^{-\pi^2 t/4}\left[\cos\left(\frac{\pi x_1}{2} - \frac{\pi}{4}\right) + \frac{1}{\sqrt{2}}\right], \quad x_1 \in (0,1), t \in (0,3], \tag{1.39}$$

$$u(0,x_2,t) = \sqrt{2}e^{-\pi^2 t/4}\left[\cos\left(\frac{\pi x_2}{2} - \frac{\pi}{4}\right) + \frac{1}{\sqrt{2}}\right], \quad x_2 \in (0,1), t \in (0,3], \tag{1.40}$$

$$\frac{\partial u}{\partial x_1}(0,x_2,t) = \frac{\pi}{2}e^{-\pi^2 t/4}, \quad x_2 \in (0,1), t \in (0,3], \tag{1.41}$$

$$u(x_1,x_2,0) = \sqrt{2}\left[\cos\left(\frac{\pi x_1}{2} - \frac{\pi}{4}\right) + \cos\left(\frac{\pi x_2}{2} - \frac{\pi}{4}\right)\right], \quad (x_1,x_2) \in D. \tag{1.42}$$

This is an inverse problem with missing boundary data at $x_1 = 1$, which we wish to determine using the Cauchy boundary data over-specification at $x_1 = 0$. The exact solution of problem (1.38)–(1.42) is

$$u(x_1,x_2,t) = \sqrt{2}e^{-\pi^2 t/4}\left[\cos\left(\frac{\pi x_1}{2} - \frac{\pi}{4}\right) + \cos\left(\frac{\pi x_2}{2} - \frac{\pi}{4}\right)\right], \quad (\mathbf{x},t) \in D_T. \tag{1.43}$$

In this example, we shall show results when the source points are placed on squares, as well as when we place them on circles, to show that placement of the sources does not need to follow the shape of the solution domain. When the source points are placed on a square they will be located at $(-\frac{h}{2}, 1+\frac{h}{2}) \times (-\frac{h}{2}, 1+\frac{h}{2})$, whilst the source points placed on a circle will have radius h with centre at $(0.5, 0.5)$, see Figure 1.20.

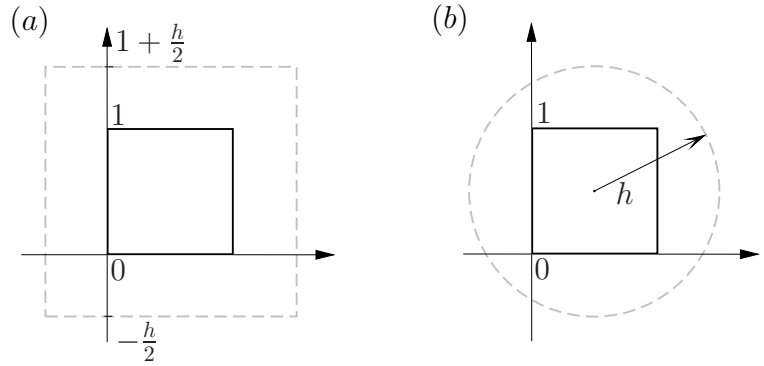

Figure 1.20: Examples of source point location for the unit square solution domain of Example 5.

Figures 1.21(a) and 1.21(b) show the exact solution $u(1, x_2, 1.5)$ and its normal derivative $\frac{\partial u}{\partial x_1}(1, x_2, 1.5)$, respectively, in comparison with the approximate solutions, obtained with $h = 3$ and $\lambda = 10^{-8}$.

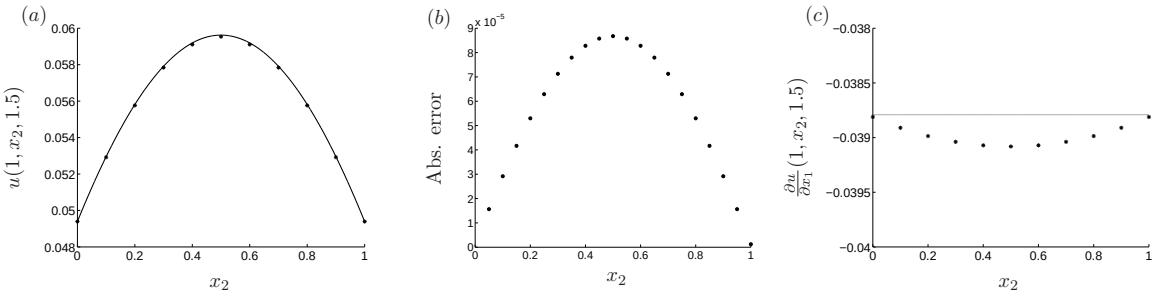

Figure 1.21: (a) The exact solution $u(1, x_2, 1.5)$ (—) and the MFS approximation $u_{M,N}$ $(*)$, and (b) the corresponding absolute error plot. (c) The exact normal derivative $\frac{\partial u}{\partial x_1}(1, x_2, 1.5)$ (—) and the MFS approximation $(\circ)$. All approximations generated with $h = 3$, for Example 5.

Figure 1.22 shows the exact solution $u(x_1, x_2, 0.5)$ and the MFS approximation $u_{M,N}$. Note that for the approximation $u_{M,N}$ in Figure 1.22(c) we have instead placed the source points on a circle and there is still good agreement with the exact solution. In Figure 1.23 we present plots of the absolute error at time $t = 2.5$ for two different values of h. From this figure it can be seen that the error increases when the source points have been placed too close to the boundary.

In Figure 1.24, random noise simulating measurement errors, have been added to the Dirichlet boundary data (1.40) as follows:

$$u^{\delta}(0, x_2, t) = u(0, x_2, t) + N(0, \sigma^2),$$

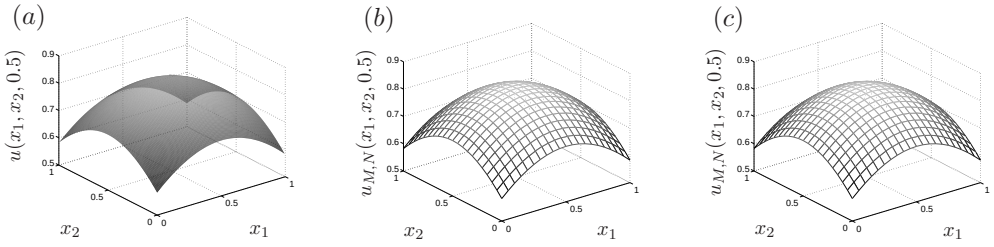

Figure 1.22: (a) The exact solution $u(x_1, x_2, 0.5)$, (b) the MFS approximation $u_{M,N}$ using source points placed on a square, $h = 3$, and (c) the MFS approximation $u_{M,N}$ using source points placed on a circle with radius $h = 3$ and centre $(0.5, 0.5)$, for Example 5.

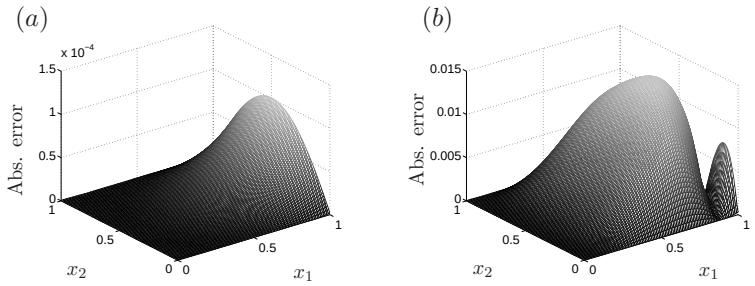

Figure 1.23: The absolute error $|u(x_1, x_2, 2.5) - u_{M,N}(x_1, x_2, 2.5)|$ when the MFS approximation $u_{M,N}$ has been generated using source points placed on a square with: (a) $h = 3$ and (b) $h = 1$, for Example 5.

where $N(0, \sigma^2)$ represents the normal distribution with mean zero and standard deviation

$$\sigma = \delta \times \max_{(x_2,t)\in(0,1)\times(0,3)} |u(0, x_2, t)|,$$

and δ is the relative noise level. A set of ten noisy random data functions $\{u_k^\delta(0, x_2, t)\}_{k=1,\ldots,10}$ was generated, and the source points in the MFS have been placed on a circle of radius h, centred at $(1/2, 1/2)$. Figure 1.24(a) presents a plot of the exact solution $u(1, x_2, 0.5)$, and the best $(*)$ and the least accurate $(\circ)$ MFS approximations from these ten data sets, obtained with $\delta = 3\% = 0.03$ noise, $h = 3$, $T = 3$ and $\lambda = 10^{-4}$. In figure 1.24(b) we present a three-dimensional plot of the absolute error when $\delta = 3\%$, $h = 3$, $\lambda = 10^{-4}$ and $t = 0.5$.

Adding more noise such as $\delta = 5\%$ did not significantly change the stability of the numerical results provided that regularization is applied appropriately (i.e. using the L-curve criterion or other scheme for choosing the regularization parameter). As expected, the accuracy decreases when the noise level increases and the regularizing parameter usually has to take a larger value. However, as time increases errors accumulate, since the values on the boundary decrease expo-

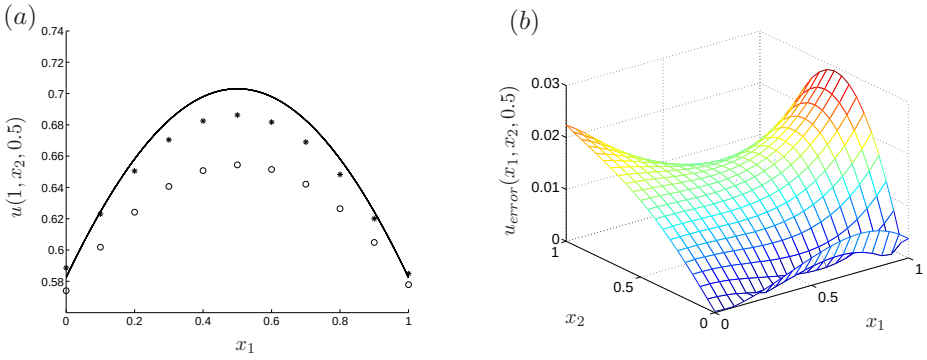

Figure 1.24: (a) The exact solution $u(1, x_2, 0.5)$ (—) and the best ($*$) and the least accurate ($\circ$) MFS approximations from ten different sets of noisy data with noise level $\delta = 3\%$ and (b) the absolute error $|u(x_1, x_2, 0.5) - u_{M,N}(x_1, x_2, 0.5)|$ when $\delta = 3\%$, for one of the noisy data sets, for Example 5.

nentially, hence the noise applied to the maximum boundary value at $t = 0$ will have a big effect on the boundary values for $t > 1$. Noise can also be added to the other data functions, such as the Neumann data (1.41), and the same stable and accurate numerical results are expected for $t \leq 1$. Thus, for this inverse problem, the regularized MFS is a stable approximation with respect to this noisy data for $t \leq 1$.

The noise we have used is rather severe; if we consider a different type of noise, such as multiplicative noise

$$u^{\delta}(0, x_2, t) = u(0, x_2, t)(1 + \delta\rho),$$

where δ is the percentage of noise and ρ is a random number taken between $[-1, 1]$, results are better for larger values of time t, since the error is applied as a percentage of the boundary data at each point.

1.5 Summary of Chapter 1

We proposed an MFS for the two-dimensional heat conduction problem and justified our approach by proving a denseness result for linear combinations of the fundamental solution to the heat equation.

Five examples were given exploring different domains, initial and boundary conditions, noise levels, the L-curve and source point locations. The results presented reinforce the justification of our placement of source points with what appear to be accurate results usually being obtained.

It should be noted that the results presented are not the best possible, rather it was the intention that the parameter h was chosen rather arbitrarily, with most of the examples having source points placed relatively close to the boundary on a circle of radius 2. In Examples 1 and 2, we chose $h = 1$, however, results obtained with $h = 2$ give better results.

We briefly recall the numerical results obtained for the 5 examples. In Example 1 accurate results using the MFS were obtained with a symmetrical solution to the heat equation in a circular domain with source points placed on a circle surrounding the domain, which was to be expected. We then moved on to a non-symmetric example, again obtaining good results, and in this example we plotted the L-curve to show a way in which engineers could choose the Tikhonov regularization parameter λ. Also, in Example 2, we tested to see if we could reduce the ill-conditioning in our matrix by placing the same number of source points going down to $T/2$ and $T/4$ instead of T (and tests were also carried out by extending the interval these points were placed over); results showed that this did not improve the accuracy, in fact results became worse. In Example 3 we used the exact solution from Example 2 to generate the initial and boundary conditions with a more complicated domain, and two different source point locations being tested. We found that placing source points on a circle gave better results, however, additional testing will need to be performed to verify optimal choices for the pseudo-boundary. Again we should note that in Example 3 we chose $N = 20$ and $M = 30$, however, when we tested $N = 40$ and $M = 20$ points we obtained more accurate results, with less oscillatory behaviour.

In Examples 4 and 5 we considered square domains, which could be readily extended to rectangular domains. Example 4 demonstrated a problem with our method, however, after some investigation, and a change of our final time point, we were able to obtain accurate results even for rapidly decaying solutions, which shows that the MFS can be manipulated with ease. In Example 5 we considered an ill-posed inverse problem with noise added to the Dirichlet boundary data and found that results in this section could be improved with a better choice of the locations of the source points around the domain and a better choice of λ.

The numerical results in this chapter have been computed using MATLAB on a computer with an Intel® Core™2 Duo Dual Core 3.00 GHz Processor, 3323 MB of memory, and all computations were completed in less than ten seconds. In testing the MFS we considered more than what has been presented, for example, whether it was necessary to use Tikhonov regularization in all examples, and it was found that better results were obtained when λ was

not set equal to zero. We also tried using other commands in MATLAB, in particular we tested the `pinv` command, used for determining the Moore-Penrose pseudoinverse of a matrix, and similar results were obtained compared to the backslash command (\), however, we chose not to use the `pinv` command as it requires more flops to compute.

CHAPTER 2

THE BACKWARD HEAT CONDUCTION PROBLEM

2.1 Introduction

In the previous chapter we considered heat conduction in two-dimensions, in particular, a direct initial-boundary value problem. In Example 5 of the previous chapter we tested the MFS for an inverse problem, and we continue by investigating another inverse problem in this chapter, the backward heat conduction problem (BHCP). Instead of an initial condition being given, we have a final condition given at time $t = T$, and we wish to recover the initial data at $t = 0$. The BHCP is a classical ill-posed problem that is notoriously hard to solve due to the way heat decays in time making it difficult to recover past information. To justify the use of the MFS for this problem, we generalise the denseness result for initial data in Theorem 1.2.3, and show that it holds for any $t \geq 0$. In particular, linear combinations of fundamental solutions are dense at $t = T$.

2.2 Preliminaries

We use the same notation introduced in the previous chapter. We wish to determine a solution u to the heat equation in the domain D_T, supplied with final and Dirichlet boundary conditions, that is u solves

$$\frac{\partial u}{\partial t}(\mathbf{x}, t) - \Delta u(\mathbf{x}, t) = 0, \quad (\mathbf{x}, t) \in D_T, \tag{2.1}$$

$$u(\mathbf{x}, t) = h(\mathbf{x}, t), \quad (\mathbf{x}, t) \in \Gamma_T, \tag{2.2}$$

$$u(\mathbf{x}, T) = u_T(\mathbf{x}), \quad \mathbf{x} \in D, \tag{2.3}$$

where $h(\mathbf{x}, t)$ and $u_T(\mathbf{x})$ are sufficiently smooth functions. We note that the Dirichlet boundary condition can be changed to a Neumann or mixed boundary condition, with little modification to the MFS implementation required.

The solution of the BHCP (2.1)–(2.3) is unique, however, it is ill-posed and no solution exists if u_T in (2.3) is not analytic, see [65, 88]. The following example illustrates the violation of the continuous dependence condition of Hadamard's definition of well-posedness. Consider the forward problem

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, \quad x \in (-\infty, \infty),\ t \in (0, T), \tag{2.4}$$

$$u(x, 0) = u_0(x) = n \sin(nx).$$

The solution of this problem is given by

$$u(x, t) = ne^{-n^2 t} \sin(nx),$$

which, as $n \to \infty$, decays exponentially to zero as $t \to T$, and $u \to \infty$ as $t \to 0$. Therefore the backward problem (2.4) and

$$u(x, T) = u_T(x) = ne^{-n^2 T} \sin(nx).$$

results in an ill-posed problem which does not continuously depend on the data.

To see why u_T must be analytic, consider the solution of the forward problem (2.4), and

$$u(x, 0) = u_0(x),$$

which can be given by

$$u(x, t) = \int_{-\infty}^{\infty} F(x, t; y, 0) u_0(y)\, dy,$$

where F is given by (2.5), and $n = 1$. Therefore, when $u(x, t)$ exists it is analytic for all $t > 0$.

Classes of functions to restore the well-posedness of the BHCP have been investigated in [88, 18], however, these conditions are rarely satisfied in practice, and therefore regularization methods of solutions appear to be more useful in order to obtain stable solutions [14, 49, 77, 81, 87, 67, 53, 55].

2.3 Denseness for any fixed time point

We recall here that the fundamental solution of the heat equation (2.1) in n-dimensions is given by

$$F(\mathbf{x},t;\mathbf{y},\tau) \;=\; \frac{H(t-\tau)}{(4\pi(t-\tau))^{\frac{n}{2}}} e^{-\frac{|\mathbf{x}-\mathbf{y}|^2}{4(t-\tau)}}. \tag{2.5}$$

In the previous chapter we proved denseness of linear combinations of fundamental solutions on the 'base' surface where $t = 0$, see Theorem 1.2.3; now we impose a condition at $t = T$. Therefore, to justify the use of the MFS for the BHCP, we prove denseness for any $T_0 \geq 0$. We note that the proof given below is similar to the proof of Theorem 1.2.3, with only a few changes.

Theorem 2.3.1. *The set of functions* $\{F(\mathbf{x},t;\mathbf{y}_j,\tau_m)\}_{j,m=1}^{\infty}$, *with* $\tau_m < T_0$, *where* $T_0 \geq 0$, *forms a linearly independent and dense set in* $L^2(D \times \{T_0\})$.

Proof. Linear independence: For a contradiction, suppose we do not have linear independence, therefore there exist positive integers N, and $m_0, j_0 \in \{1,\ldots,N\}$, and a coefficient $c_{m_0}^{(j_0)} \neq 0$ such that

$$\sum_{j=1}^{N}\sum_{m=1}^{N} c_m^{(j)} F(\mathbf{x},T_0;\mathbf{y}_j,\tau_m) \;=\; 0, \quad \mathbf{x} \in D. \tag{2.6}$$

We again use the corollary of Theorem 3 in [66], which was stated on page 37. From (1.8) and (2.6) we have that

$$U(\mathbf{x},T_0) \;=\; 0, \quad \mathbf{x} \in D, \tag{2.7}$$

and U satisfies (1.9) and (1.11). The function $U(x_1,x_2,T_0)$, where $\mathbf{x} = (x_1,x_2)$, is real analytic in each of the variables x_1 and x_2, and therefore $U(\mathbf{x},T_0) = 0$ for every $\mathbf{x} \in \mathbb{R}^2$. Since $\tau_m < T_0$, U is continuous on $\mathbb{R}^2 \times [T_0,T_1]$, for any $T_1 > T_0$, and is at least twice continuously differentiable in $\mathbb{R}^2 \times [T_0,T_1]$. Furthermore, U satisfies the heat equation (2.1), and

$$|U(\mathbf{x},t)| \;\leq\; Be^{\beta|\mathbf{x}|^2}, \quad t \in [T_0,T_1], \tag{2.8}$$

for some positive constants B and β, which was again shown in the proof of Theorem 1.2.3. Thus, from Theorem 3 of [66], and its corollary, $U(\mathbf{x},t) = 0$ in $\mathbb{R}^2 \times [T_0,T_1]$. In particular, from [95], we may extend U such that $U(\mathbf{x},t) = 0$ in $D_E \times [-T,T]$.

We make the point $(\mathbf{x},t)$ approach the point $(\mathbf{y}_{j_0},\tau_{m_0}) \in \Gamma_E \times [-T,T_0]$ such that the ratio

(1.12) remains bounded. Now, the summand $c_{m_0}^{(j_0)} F(\mathbf{x}, t; \mathbf{y}_{j_0}, \tau_{m_0})$ may be made as large as we wish, with other terms in (2.6) remaining bounded; giving us a contradiction.

Denseness: We shall show that the set of functions $\{F(\mathbf{x}, T_0; \mathbf{y}_j, \tau_m)\}$, where $\tau_m < T_0$, is a dense set in $L^2(D \times \{T_0\})$. Assume, for a contradiction, that this is not a dense set, then there exists a function $f \in C^2(D \times \{T_0\})$ such that

$$\int_D F(\mathbf{x}, T_0; \mathbf{y}_j, \tau_m) f(\mathbf{x})\, d\mathbf{x} \;=\; 0, \quad j, m = 1, 2, \ldots \tag{2.9}$$

We let w be a weak solution to the heat equation (2.1), see [35], with initial condition $w(\mathbf{x}, T_0) = f(\mathbf{x})$ and boundary condition $w(\mathbf{x}, t) = 0$ for $(\mathbf{x}, t) \in \Gamma \times [T_0, T_1]$, for any $T_1 > T_0$. We use Greens identities to transform equation (2.9), see [35, 72], into the following form

$$\int_{T_0}^{T_1} \int_\Gamma F(\mathbf{x}, t; \mathbf{y}_j, \tau_m) \frac{\partial w}{\partial \boldsymbol{\nu}}(\mathbf{x}, t)\, d\mathbf{x} dt \;=\; 0, \quad j, m = 1, 2, \ldots$$

where $\boldsymbol{\nu}$ is the outward pointing unit normal on Γ. From Theorem 1.2.2 we know $\{F(\mathbf{x}, t; \mathbf{y}_j, \tau_m)\}$ restricted on Γ_T is a dense set in $L^2(\Gamma \times [T_0, T_1])$, and therefore the normal derivative of w is zero on $\Gamma \times [T_0, T_1]$. Therefore, w and $\frac{\partial w}{\partial \boldsymbol{\nu}}$ are zero on $\Gamma \times [T_0, T_1]$, and according to [95], we conclude that $w(\mathbf{x}, t) = 0$ for $(\mathbf{x}, t) \in \bar{D} \times [T_0, T_1]$; hence $f \equiv 0$, and $\{F(\mathbf{x}, T_0; \mathbf{y}_j, \tau_m\}$, where $\tau_m < T_0$, is a dense set in $L^2(D \times \{T_0\})$. □

2.4 The MFS for the BHCP

In this section, we apply the MFS to both the one and the two-dimensional heat equation, with differences in notation being noted in the following subsections. The MFS is applied in the same way as in [59] and the previous chapter, with source points placed in time on an external boundary to the domain, but instead we have a final condition, at time $t = T$, on which we place collocation points. We consider three test examples, and compare the results to papers which have used these examples, but with different meshless methods.

2.4.1 The one-dimensional case

In one-dimension we wish to construct a solution u, in the domain $\overline{D}_T = [0,1] \times [0,T]$ (where $D = (0,1)$, $\Gamma = \{0,1\}$ is the boundary of D, $\Gamma_E = \{-h, 1+h\}$ is an external boundary with $h > 0$, and $T > 0$ is the final time point), which solves

$$\frac{\partial u}{\partial t}(x,t) - \frac{\partial^2 u}{\partial x^2}(x,t) = 0, \quad (x,t) \in D_T, \tag{2.10}$$

$$u(0,t) = h_1(t), \quad t \in (0,T], \tag{2.11}$$

$$u(1,t) = h_2(t), \quad t \in (0,T], \tag{2.12}$$

$$u(x,T) = u_T(x), \quad x \in D. \tag{2.13}$$

Based on the denseness results, Theorems 1.2.2 and 2.3.1, which are valid also in one-dimension, we construct a solution to (2.10)–(2.13) in the following form:

$$u(x,t) \approx u_N(x,t) = \sum_{n=1}^{2N_1} \sum_{j=1}^{2} c_n^{(j)} F(x,t;y_j,\tau_n), \quad (x,t) \in D_T, \tag{2.14}$$

where F is the fundamental solution of the heat equation given by equation (2.5) for $n = 1$.

In the previous chapter the same number of collocation and source points were used to obtain a square system of equations, which could be solved using Gaussian elimination. However, this does not give much flexibility, therefore, we remove this condition and use the least-squares method to solve non-square (over and underdetermined) systems of equations.

The time points that shall be used to generate the collocation points are given by

$$t_i = \frac{i}{M_1} T, \quad i = 1, \ldots, M_1.$$

When generating the source points over the interval $(-T, T)$, there are two different cases depending on whether $N_1 \geq M_1$ or $N_1 < M_1$. For simplicity, we assume that either $N_1 = kM_1$ or $M_1 = kN_1$ for some integer $k > 0$. Thus, we simply interlace points for time $t > 0$ and choose $(\tau_n)_{n=1,\ldots,2N_1}$ given by

$$\tau_n = \begin{cases} -T + \frac{n}{kM_1} T - \frac{T}{2kM_1}, & n = 1, \ldots, 2N_1, \text{ if } N_1 = kM_1 \geq M_1, \\ -T + \frac{kn}{M_1} T - \frac{T}{2M_1} & n = 1, \ldots, 2N_1, \text{ if } N_1 < M_1 = kN_1. \end{cases} \tag{2.15}$$

On the external boundary set

$$y_1(\tau_n) = -h \quad \text{and} \quad y_2(\tau_n) = 1 + h, \quad n = 1, \ldots, 2N_1,$$

and on $D \times \{T\}$ let

$$x_T^{(l)} = \frac{l}{K+1}, \quad l = 1, \ldots, K.$$

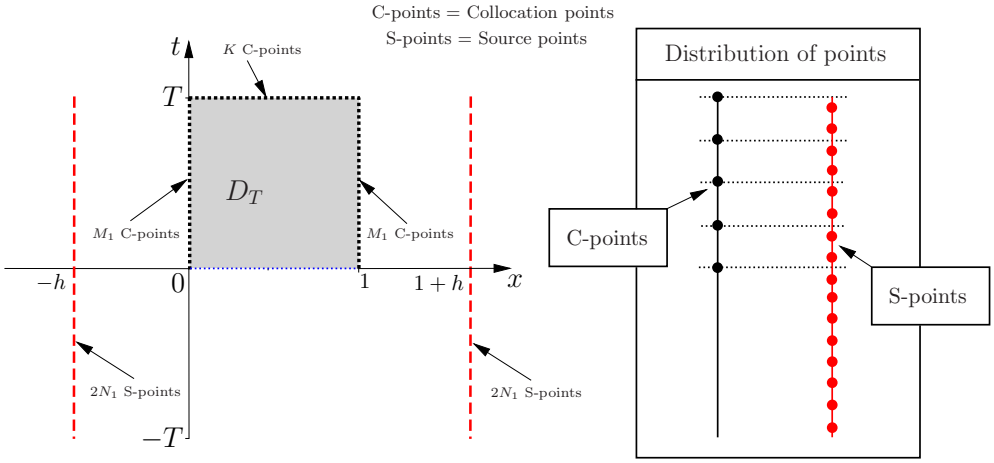

Figure 2.1: Representation of the one-dimensional domain in time, displaying the number of collocation (c) and source (s) points, as well as a possible distribution of these points on the boundaries.

Therefore, we obtain the following system of equations

$$u_N(x_T^{(l)}, 0) = u_T(x_T^{(l)}), \quad l = 1, \ldots, K, \tag{2.16}$$

$$u_N(0, t_i) = h_1(t_i), \quad u_N(1, t_i) = h_2(t_i), \quad i = 1, \ldots, M_1. \tag{2.17}$$

Equations (2.16) and (2.17), via (2.14), form a linear ill-conditioned system of $M = 2M_1 + K$ equations with $N = 4N_1$ unknowns $c_n^{(j)}$ for $j = 1, 2$ and $n = 1, \ldots, 2N_1$.

2.4.2 The two-dimensional case

The two-dimensional case is similar to the one-dimensional, except for slight differences in terminology, however, all details are given here for clarity. In two-dimensions, we wish to construct a solution u, in the square domain $D_T = D \times (0, T]$ (where $D = (0, 1) \times (0, 1)$, Γ is the boundary of D, Γ_E is an external boundary with $d(\Gamma, \Gamma_E) > 0$, and T is the final time point), which solves the BHCP (2.1)–(2.3).

In two-dimensions, again based on the denseness results in Theorems 1.2.2 and 2.3.1, we construct an MFS approximation u_{N_1,N_2} to (2.2)–(2.3) in the following form:

$$u(\mathbf{x},t) \approx u_{N_1,N_2}(\mathbf{x},t) = \sum_{n=1}^{2N_1}\sum_{j=1}^{4N_2} c_n^{(j)} F(\mathbf{x},t;\mathbf{y}_j,\tau_n), \quad (\mathbf{x},t) \in D_T, \tag{2.18}$$

where F is the fundamental solution of the heat equation given by equation (2.5) for $n = 2$.

The time points that shall be used to generate the collocation points are given by

$$t_i \;=\; \frac{i}{M_1}T, \quad i = 0,\ldots,M_1-1.$$

Using the same strategy as in the one-dimensional case, we assume that either $N_1 = kM_1$ or $M_1 = kN_1$ for some integer $k > 0$. We choose $(\tau_n)_{n=1,\ldots,2N_1}$ given by (2.15).

On Γ_E set

$$\mathbf{y}_j \;=\; \begin{cases} \left(-h + \frac{1+2h}{N_2}j, -h\right), & j = 0,\ldots,N_2-1, \\ \left(1+h, -h + \frac{1+2h}{N_2}(j-N_2)\right), & j = N_2,\ldots,2N_2-1, \\ \left(h - \frac{1+2h}{N_2}(j-2N_2), 1+h\right), & j = 2N_2,\ldots,3N_2-1, \\ \left(-h, h - \frac{1+2h}{N_2}(j-3N_2)\right), & j = 3N_2,\ldots,4N_2-1, \end{cases}$$

and on $D \times \{T\}$ set

$$\mathbf{x}_T^{(l_0,l_1)} \;=\; \left(\frac{l_0}{K-1}, \frac{l_1}{K-1}\right), \quad l_0, l_1 \;=\; 0,\ldots,K-1.$$

We denote by $\mathbf{x}_m$ the position of the collocation points on the square boundary, given by

$$\mathbf{x}_m \;=\; \begin{cases} \left(\frac{m}{M_2}, 0\right), & m = 0,\ldots,M_2-1, \\ \left(1, \frac{m-M_2}{M_2}\right), & m = M_2,\ldots,2M_2-1, \\ \left(1 - \frac{m-2M_2}{M_2}, 1\right), & m = 2M_2,\ldots,3M_2-1, \\ \left(0, 1 - \frac{m-3M_2}{M_2}\right), & m = 3M_2,\ldots,4M_2-1. \end{cases}$$

The linear system of $M = K^2 + 4M_1M_2$ equations with $N = 8N_1N_2$ unknowns $c_n^{(j)}$ for $n = 1,\ldots,2N_1$ and $j = 1,\ldots,4N_2$, obtained after applying the final and Dirichlet boundary

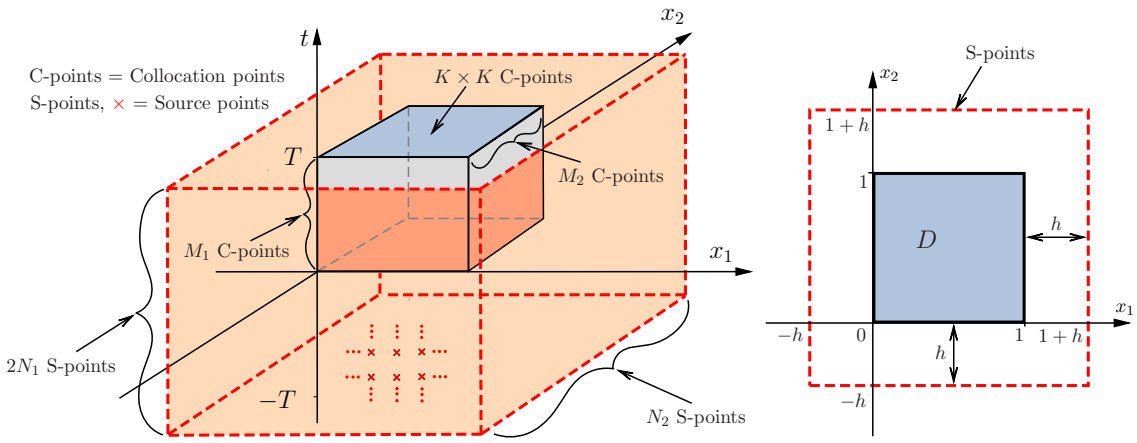

Figure 2.2: Representation of the domain in two-dimensions, displaying the number of collocation and source points, and a possible placement of source points on the external boundary.

conditions (2.2)–(2.3) at the above collocation points is given by

$$u_{N_1,N_2}(\mathbf{x}_T^{(l_0,l_1)}, T) = u_0(\mathbf{x}_T^{(l_0,l_1)}), \quad l_0, l_1 = 1, \ldots, K, \tag{2.19}$$

$$u_{N_1,N_2}(\mathbf{x}_m, t_i) = h(\mathbf{x}_m, t_i), \quad m = 0, \ldots, 4M_2 - 1, \ \ i = 0, \ldots, M_1 - 1. \tag{2.20}$$

For both the one and two-dimensional case we will employ the Tikhonov regularization and solve (0.45) using L_0, see Section 0.5 of the Introduction for details.

2.5 Numerical results

2.5.1 Example 1

Here, we examine a one-dimensional test example, which was considered in [86] and [59], both applying meshless methods, with the exact solution given by

$$u(x,t) = \sin(\pi x)\exp(-\pi^2 t), \quad (x,t) \in [0,1] \times [0,T]. \tag{2.21}$$

In [86], an MFS was applied by placing the source points on a domain at time point $t = -\epsilon \in (-\infty, 0)$, and accurate results were produced, however, to the authors knowledge there are no theoretical results justifying the use of the MFS in this way. In [59], the MFS was applied in a similar way as in this thesis, however, no theoretical results were given. We compare the results of both of these papers and examine the influence the various parameters have on the accuracy

of the approximation.

We use equation (2.21) to generate the necessary data; therefore we have the following final and boundary conditions:

$$u(0,t) \;=\; 0, \quad u(1,t) \;=\; 0, \quad t \in (0,T], \tag{2.22}$$

$$u(x,T) \;=\; \sin(\pi x)\exp(-\pi^2 T), \quad x \in (0,1). \tag{2.23}$$

We note that in this example the final condition is the only non-zero data provided, therefore accurate approximations are difficult to generate for large values of the final time point T.

Random additive noise is added to the final data (2.23) as follows:

$$u_T^{\delta}(x) \;=\; \sin(\pi x)\exp(-\pi^2 T) + N(0,\sigma^2), \tag{2.24}$$

where $N(0,\sigma^2)$ represents the normal distribution with mean zero and standard deviation

$$\sigma \;=\; \delta \times \max_{x\in D} |u_T| \;=\; \delta \exp(-\pi^2 T), \tag{2.25}$$

where δ is the relative (percentage) noise level.

The aim in this example is to numerically recover the initial data at time $t = 0$, given by

$$u(x,0) \;=\; \sin(\pi x), \quad x \in (0,1). \tag{2.26}$$

Figure 2.3 is a plot of the L-curve, using the same parameters as in Figures 2.4(a) and 2.4(b), and by considering the vertex of the "L" in the plot, see [50], and the introduction of this thesis. We note that different values of the regularization parameter have been chosen depending on the noise level. In some cases an educated guess is required when choosing the regularization parameter, since there might be no clear vertex in the graph, as displayed in Figure 2.3.

In Figures 2.4(a) and 2.4(b), we plot the MFS approximation of the initial condition using the same parameters (same number of collocation points, source points, noise levels and final time point) as in [86], except that the source points are placed in different positions and the Tikhonov regularization parameter has been chosen using the L-curve criterion. This corresponds to the overdetermined case mentioned in Section 0.5 in the Introduction, note that $M = N$ works as well, see Figure 2.6 ($N_1 = 30$). From these figures it can be seen that the numerical results are

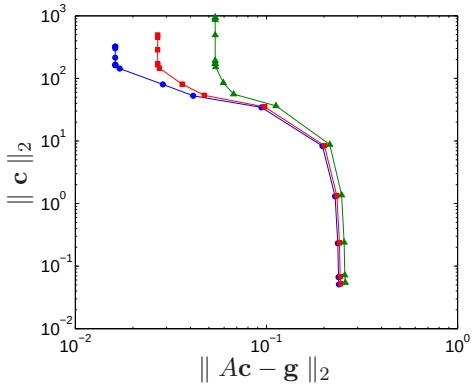

Figure 2.3: L-curve plots for $\delta = 3\%$ (●●), $\delta = 5\%$ (■■) and $\delta = 10\%$ (▲▲) when $K = 20$, $M_1 = 20$, $N_1 = 5$ (i.e. $M = 60$ collocation points and $N = 20$ source points), $T = 0.25$, and $h = 1$, for Example 1.

similar, and as stable and accurate as the results of [86]. Furthermore, errors are of the same order as the noise level applied to the final data, as expected; however, the final time point is small, with $T = 0.25$.

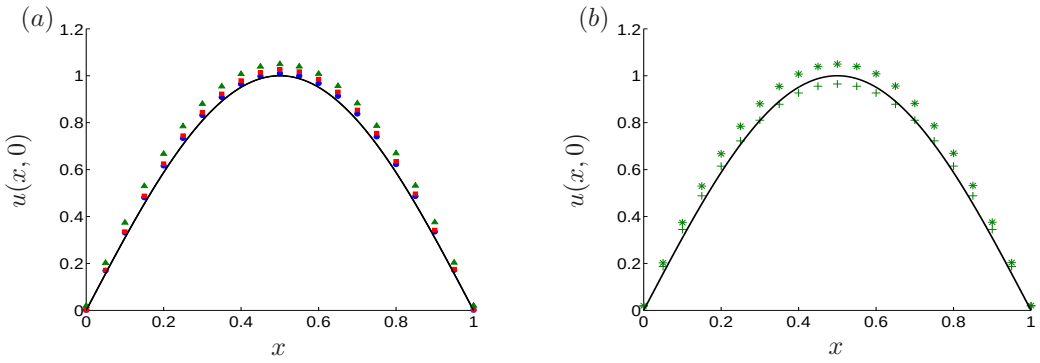

Figure 2.4: The exact solution $u(x, 0)$ (——) in comparison with **(a)** the MFS solution obtained for noise levels $\delta = 3\%$ with $\lambda = 10^{-10}$ (●), $\delta = 5\%$ with $\lambda = 10^{-10}$ (■), and $\delta = 10\%$ with $\lambda = 10^{-8}$ (▲), and **(b)** best (∗) and least (+) accurate MFS approximations from ten different sets of noisy data with noise level $\delta = 10\%$ with $\lambda = 10^{-8}$. Both plots obtained with $K = 20$, $M_1 = 20$, $N_1 = 5$ (i.e. $M = 60$ collocation points and $N = 20$ source points), $T = 0.25$, $h = 1$, for Example 1.

In Figure 2.5(a) the final time point is increased from $T = 0.5$ to $T = 0.7$, and with a similar choice of parameters as in [86]. From this figure it can be seen that the accuracy decreases quite drastically. However, in Figure 2.5(b) we increase the number of collocation and source points and find that the final time point can be extended further with accurate results still being produced, which was not seen in [86]. We attempted to reproduce the results that were obtained in [86], however, our code was only able to obtain similar results with a smaller final time point.

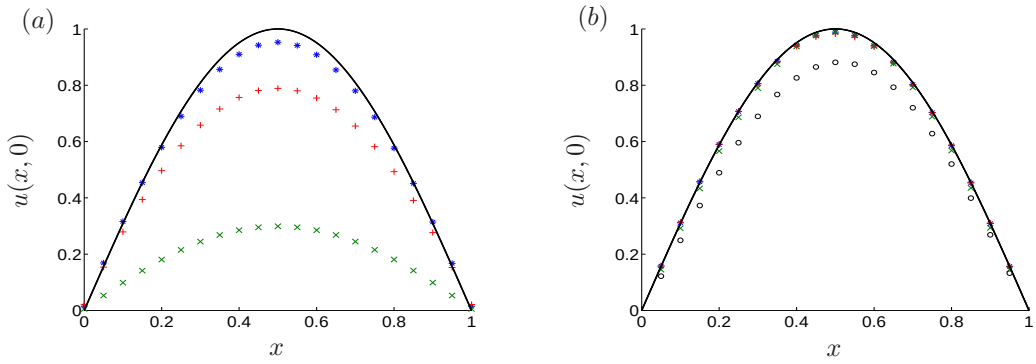

Figure 2.5: The exact solution $u(x,0)$ (——) and the MFS approximation obtained with $h = 1$ and noise level $\delta = 5\%$, for the final time points **(a)** $T = 0.5$ with $\lambda = 10^{-10}$ ($*$), $T = 0.6$ with $\lambda = 10^{-13}$ ($+$), $T = 0.7$ with $\lambda = 10^{-14}$ ($\times$), obtained with $K = 20$, $M_1 = 20$, $N_1 = 5$ (i.e. $M = 60$ collocation points and $N = 20$ source points), and **(b)** $T = 0.5$ with $\lambda = 10^{-10}$ ($*$), $T = 0.6$ with $\lambda = 10^{-12}$ ($+$), $T = 0.7$ with $\lambda = 10^{-14}$ ($\times$), and $T = 0.8$ with $\lambda = 10^{-14}$ ($\circ$), obtained with $K = 40$, $M_1 = 40$, $N_1 = 10$ (i.e. $M = 120$ collocation points and $N = 40$ source points), for Example 1.

In Figure 2.6, plots of the MFS approximation are produced for various numbers of source points, with the accuracy increasing as N increases for $M \geq N$, corresponding to a square or overdetermined system, and good results are also obtained when $M < N$ (underdetermined system). A similar comparison was performed in [59], however, the final time point was $T = 0.5$, and here it is $T = 1$. We note that increasing the number of collocation and source points too much results in the system becoming more ill-conditioned, affecting the accuracy of the approximation. In Figures 2.7(a) and 2.7(b) the same parameters are used that were given in [86] and [59], respectively, and also different values of h are tested. We find that in Figure 2.7(a), for $T = 0.5$, we are free to choose values of h from 0.5 to 1.5, however, in Figure 2.7(b), with $T = 1$, there is only a small region for h where we get accurate results with $\lambda = 10^{-14}$. This restrictive choice for h is due to the difficulty of numerically recovering the initial data when $T = 1$ (with reconstructions for $T > 1.5$ not possible). We note that results can improve by increasing the number of collocation and source points, despite the fact that this increases the conditioning. For example, when $h = 2$ and $T = 1$, the condition numbers when $(K = 20, N = 20, M = 10)$, $(K = 40, N = 40, M = 20)$ and $(K = 80, N = 80, M = 40)$ are $O(10^{17})$, $O(10^{35})$ and $O(10^{70})$, respectively.

The remaining figures in this example display plots of the maximum errors produced by the MFS for $h \in (0, 5]$ and $T \in (0, 2]$. We find that accurate results can be obtained for certain choices of parameters, however, for the BHCP, and the severe test example (2.21), we are more

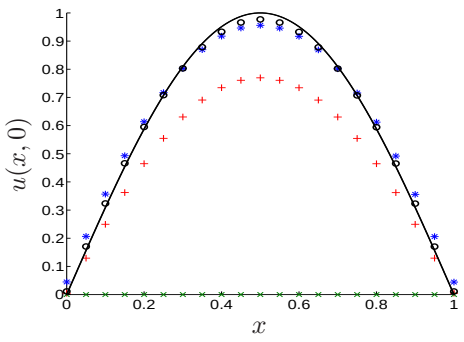

Figure 2.6: The exact solution $u(x,0)$ (——) and the MFS approximation obtained with $N_1 = 40$ with $\lambda = 10^{-12}$ ($*$), $N_1 = 30$ with $\lambda = 10^{-12}$ ($\circ$), $N_1 = 20$ with $\lambda = 10^{-12}$ ($+$), and $N_1 = 10$ with $\lambda = 10^{-5}$ ($\times$). All plots obtained with $K = 40$, $M_1 = 40$ ($M = 120$), $h = 1$, $\delta = 5\%$, $T = 1$, for Example 1.

selective than we were in the previous chapter, where we were working with a direct well-posed problem.

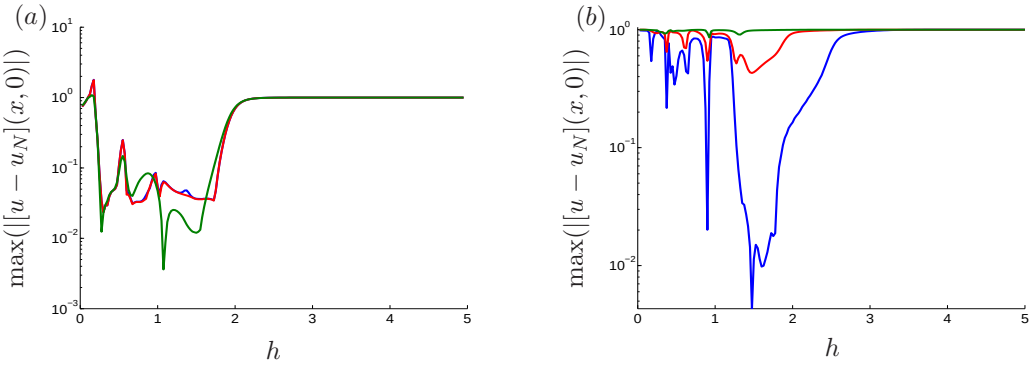

Figure 2.7: Plots of the maximum absolute error, $\max_{x\in[0,1]} |u(x,0) - u_N(x,0)|$, over $h \in [0,5]$, for $\lambda = 10^{-14}$ (——), $\lambda = 10^{-12}$ (——), and $\lambda = 10^{-10}$ (——), obtained with $K = 20$, $M_1 = 20$ ($M = 60$), and **(a)** $T = 0.5$, $N_1 = 5$, $\delta = 5\%$ and **(b)** $T = 1$, $N_1 = 10$, $\delta = 3\%$, for Example 1.

In Figures 2.8(a) and 2.8(b), we fix $h = 1.5$ and vary $T \in (0,2]$, using the same number of collocation points given in both [86] and [59]. In Figure 2.8(a) the different plots are obtained by varying λ and again note that there are regions in which the MFS is accurate, with the approximation not as accurate for small T (< 0.25) or large T (> 1). In Figure 2.8(b), the number of source points is varied, and we find that for $N > M$, the underdetermined case, the region in which the MFS is accurate is greater than when $M > N$, the overdetermined case. Note that results involving smaller values of T might be improved by decreasing λ.

Figure 2.9 in this example displays the accuracy of the approximation when we increase the number of collocation and source points, as we did in Figures 2.5(b) and 2.6, and again find

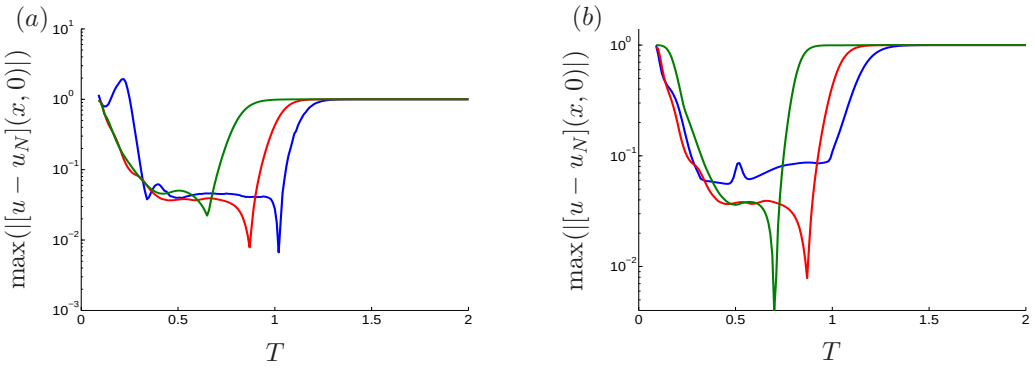

Figure 2.8: Plots of the maximum absolute error, $\max_{x\in[0,1]}|u(x,0)-u_N(x,0)|$, over $T\in[0,2]$, for **(a)** $\lambda=10^{-14}$ (——), $\lambda=10^{-12}$ (——), and $\lambda=10^{-10}$ (——), obtained with $N_1=10$, and **(b)** $N_1=20$ (——), $N_1=10$ (——), and $N_1=5$ (——), obtained with $\lambda=10^{-12}$. Both plots also obtained with $h=1.5$, $K=20$, $M_1=20$ ($M=60$), and noise level $\delta=5\%$, for Example 1.

that the accuracy improves, however, the region remains roughly the same as in Figure 2.8(a).

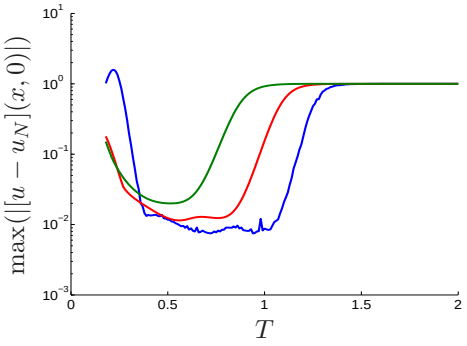

Figure 2.9: Plots of the maximum absolute error, $\max_{x\in[0,1]}|u(x,0)-u_N(x,0)|$, over $T\in[0,2]$, for $\lambda=10^{-14}$ (——), $\lambda=10^{-12}$ (——), and $\lambda=10^{-10}$ (——), obtained with $h=1.5$, $K=40$, $M_1=40$, $N_1=20$ ($M=120, N=80$), and $\delta=5\%$, for Example 1.

2.5.2 Example 2

In this example, we wish to find an approximation for the following one-dimensional model problem, considered in [60]:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, \quad (x,t)\in(0,\pi)\times(0,T), \tag{2.27}$$

$$u_x(0,t) = u_x(\pi,t) = 0, \quad t\in(0,T] \tag{2.28}$$

$$u(x,T) = e^{-T}\cos(x), \quad x\in[0,\pi], \tag{2.29}$$

with the exact solution given by $u(x,t)=e^{-t}\cos(x)$. Note that we have Neumann boundary

conditions instead of Dirichlet boundary conditions (2.17). Also note that the proposed MFS from Section 2.4.1 can easily be adjusted to this case by using the derivative of the fundamental solution (2.5), when $n = 1$, given by

$$\frac{\partial F}{\partial x}(x,t;y,\tau) \;=\; -\frac{H(t-\tau)}{4\sqrt{\pi}(t-\tau)^{\frac{3}{2}}}(x-y)e^{-\frac{(x-y)^2}{4(t-\tau)}}, \tag{2.30}$$

to determine the derivative of u_N.

In [60], the final time considered was very small, with $T = 0.005$; here we try to extend the final time point further. The aim of this example is to numerically recover the data at the initial base $t = 0$ given by:

$$u(x,0) \;=\; \cos(x), \quad x \in [0,\pi]. \tag{2.31}$$

Random additive noise simulating measurement errors have been added to the final data (2.29) and included as:

$$u_T^{\delta}(x) \;=\; e^{-T}\cos(x) + N(0,\sigma^2), \tag{2.32}$$

where, according to (2.25), $\sigma = e^{-T}$.

In Figure 2.10(a), MFS approximations have been produced for different values of h, which determine the distance that the source points are from the boundary, and found that better results are obtained for larger h. Figure 2.10(b) contains error plots for different values of the final time point T, and we find that more accurate results are produced for larger T, in part due to the matrix A having a larger condition number for smaller T (overdetermined case).

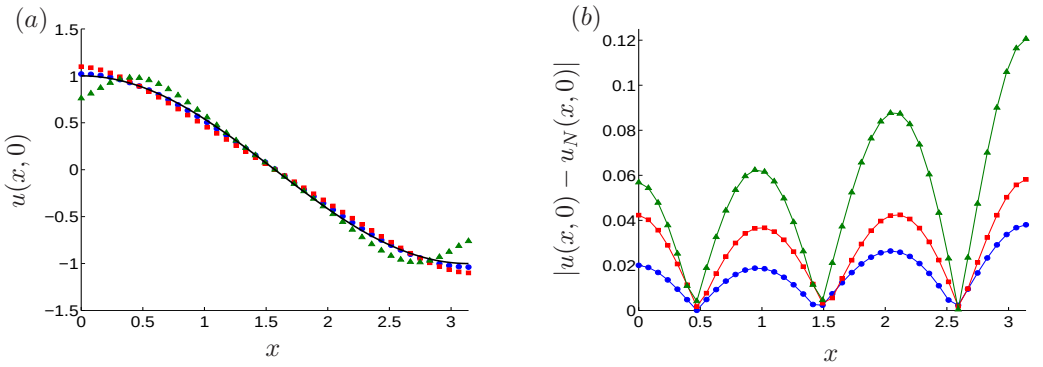

Figure 2.10: **(a)** The exact solution $u(x,0)$ (——) and the MFS approximation with $T = 2$, $h = 2$ with $\lambda = 10^{-8}$ (●), $h = 1$ with $\lambda = 10^{-5}$ (■), and $h = 0.5$ with $\lambda = 10^{-7}$ (▲) and **(b)** plots of the absolute error $|u(x,0) - u_N(x,0)|$, with $h = 2$, $T = 2$ with $\lambda = 10^{-8}$ (●), $T = 1.5$ with $\lambda = 10^{-7}$ (■) and $T = 1$ with $\lambda = 10^{-8}$ (▲). Both plots obtained with $K = 40$, $M_1 = 40$, $N_1 = 20$ (i.e. $M = 120$ collocation points and $N = 80$ source points), and noise level $\delta = 1\%$, for Example 2.

In Figure 2.11(a), the MFS approximation has been plotted for different values of the final time point T, and we note that for $T = 1$, $T = 0.9$ and $T = 0.8$, the matrix A produced has condition number $\mathcal{O}(10^{35})$, $\mathcal{O}(10^{39})$ and $\mathcal{O}(10^{44})$, respectively. A plot for $T = 0.7$ was not possible, however, if the number of collocation and source points was decreased, or the positions of the source points changed, then a lower final time would be possible. This is in contrast with the stability of the BHCP which decreases as T increases. Lastly, in Figure 2.11(b), we produce error plots for different N, the total number of source points, and find that, in this particular example and this set of parameters, fewer source points produce better results.

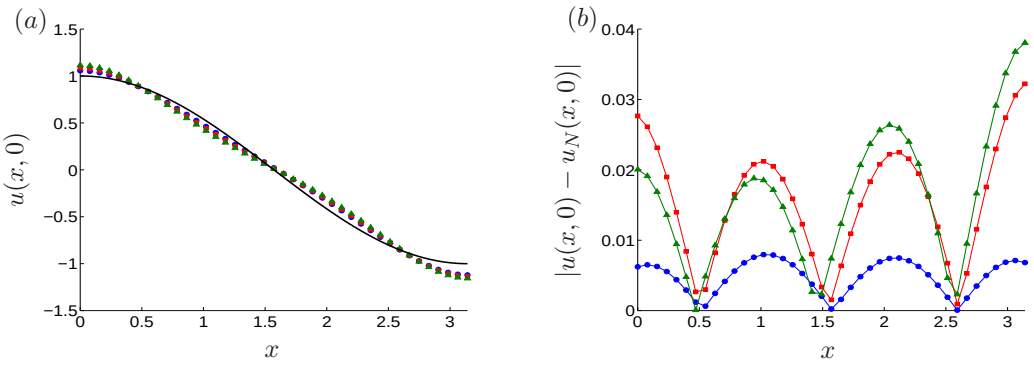

Figure 2.11: **(a)** The exact solution $u(x, 0)$ (——) and the MFS approximation with $N_1 = 20$, $T = 1$ with $\lambda = 10^{-8}$ (●), $T = 0.9$ with $\lambda = 10^{-8}$ (■), and $T = 0.8$ with $\lambda = 10^{-8}$ (▲) and **(b)** plots of the absolute error $|u(x, 0) - u_N(x, 0)|$, for $T = 2$, with $N_1 = 5$ with $\lambda = 10^{-7}$ (●), $N_1 = 10$ with $\lambda = 10^{-8}$ (■) and $N_1 = 20$ with $\lambda = 10^{-8}$ (▲). Both plots obtained with $h = 2$, $K = 40$, $M_1 = 40$, and noise level $\delta = 1\%$, for Example 2.

2.5.3 Example 3 (two-dimensional case)

The following two-dimensional example was considered in similar forms in [86], [82] and [59], with all of them considering a square domain with edge length one. In the first part of this example, we break down into cases all of the parameters considered in [86], [82] and [59], and attempt to compare results.

Case (i): Mera [86] applied the MFS by placing source points on $t = -\epsilon \in (-\infty, 0)$ below the solution domain $D = [0, 1] \times [0, 1]$, ($D_T = D \times [0, T]$). The exact solution given by

$$u(\mathbf{x}, t) \;=\; \sin(\pi(x_1 + x_2 - 1)) \exp(-2\pi^2 t), \quad (\mathbf{x}, t) \in D \times [0, T], \tag{2.33}$$

was then used to generate the data (2.2) and (2.3).

Furthermore, random additive noise was added to the final data as follows:

$$u_T^{\delta}(\mathbf{x},t) \;=\; \sin(\pi(x_1+x_2-1))\exp(-2\pi^2 T) + N(0,\sigma^2), \tag{2.34}$$

where σ is given by (2.25).

For the collocation and source points we take $M_1 = 5$, $M_2 = 5$, $K = 5$, $N_1 = 5$ and $N_2 = 2$ (i.e. $M = 125$ collocation points and $N = 80$ source points, overdetermined case). We note that the number of collocation points is the same as [86], however, there are fewer source points because of how we wish to place source points in time.

Case (ii): Very recently, Li et al. [82] used radial basis functions (RBFs) to solve the nonhomogeneous BHCP. The method involved reducing the nonhomogeneous heat equation into a series of elliptic PDEs using the method of lines, they are then converted into Poisson equations which are solved using RBFs. We consider the same domain as in case (i), with the exact solution given by

$$u(\mathbf{x},t) \;=\; \sin(\pi(x_1+x_2))\exp(-2\pi^2 t), \quad (\mathbf{x},t)\in D\times[0,T], \tag{2.35}$$

which is used to generate the data (2.2) and (2.3).

The random additive noise is implemented in the same way as in case (i), however, we add noise to both the final and boundary data, as in [82]. We note that the noise added in [82] was given by

$$\tilde{b} = b(1+\text{randn}\times\delta),$$

where randn is a Gaussian random variable with mean 0 and variance 1.

We point out that the method used in [82] does not use source points, therefore, to compare results, we use the same number of source points stated in case (i). Thus, we take $M_1 = 2$, $M_2 = 5$, $K = 9$, $N_1 = 2$ and $N_2 = 5$ (i.e. $M = 121$ collocation points and $N = 80$ source points, overdetermined case).

Finally, we remark that in references [86, 82] the factor 2 is missing in the time exponentials in equations (2.33) and (2.35) and therefore, their analytical solutions are incorrect since they do not satisfy the heat equation (2.1).

Case (iii): Hon and Li [59] applied the MFS by placing source points as in this thesis. They

considered the domain $D = [-0.5, 0.5] \times [-0.5, 0.5]$, and a slightly different exact solution, given by

$$u(\mathbf{x}, t) \;=\; \sin\left(\frac{1}{\sqrt{2}}\pi(x_1 + x_2)\right)\exp(-\pi^2 t), \quad (\mathbf{x}, t) \in D \times [0, T], \tag{2.36}$$

which is used to generate the data (2.2) and (2.3).

Random additive noise is applied to both the final and boundary data, as in case (ii). We take $M_1 = 7$, $M_2 = 7$, $K = 8$, $N_1 = 7$ and $N_2 = 7$ (i.e. $M = 260$ collocation points and $N = 392$ source points, underdetermined case).

In Figure 2.12(a), we have plotted the MFS approximation, and in Figure 2.12(b) the absolute error, using parameters taken from case (i), and we see similar results to [86], although it is difficult to do a direct comparison since there is no error plot in [86]. However, when we produced plots using the same placement of source points as in [86], the maximum error was approximately double the error obtained in Figure 2.12(b). In Figure 2.12, we have used the value $h = 0.1$, since the only paper that uses a similar placement of source points to ours appears to be [59], and for the purposes of comparison we use a similar value of h that they used. We also note that for this problem it seems that a smaller value of h is preferable, see later Figure 2.15.

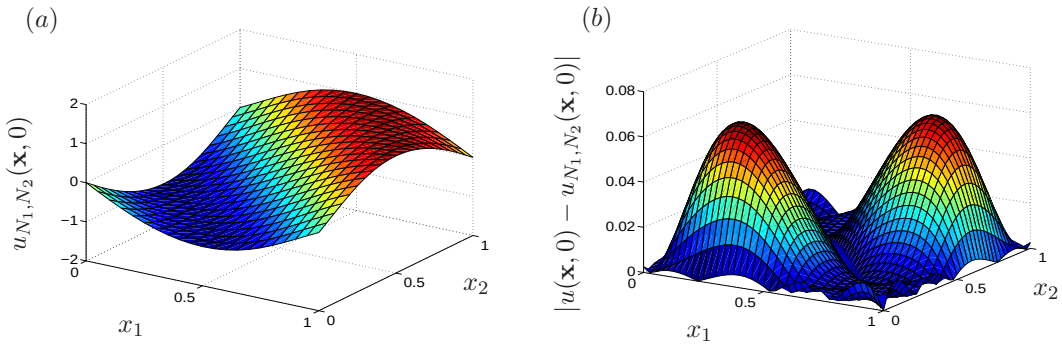

Figure 2.12: **(a)** The MFS approximation $u_{N_1,N_2}(\mathbf{x}, 0)$ plotted over $\mathbf{x} \in D$ and **(b)** the absolute error for $(\mathbf{x}, 0) \in D_T$, both obtained with $\delta = 5\%$, $h = 0.1$, $T = 0.25$, $\lambda = 10^{-5}$, and parameters from case (i), for Example 3.

In Figures 2.13(a) and 2.13(b), we have plotted the MFS approximation and absolute error, respectively, using parameters from case (ii). Comparing these to Figures 2.12(a) and 2.12(b) and [82], we find that results are not quite as good, however, we note that in case (i) we place more points on the boundary than at the final time point.

In Figures 2.14(a) and 2.14(b) the MFS approximation and absolute error, respectively, have

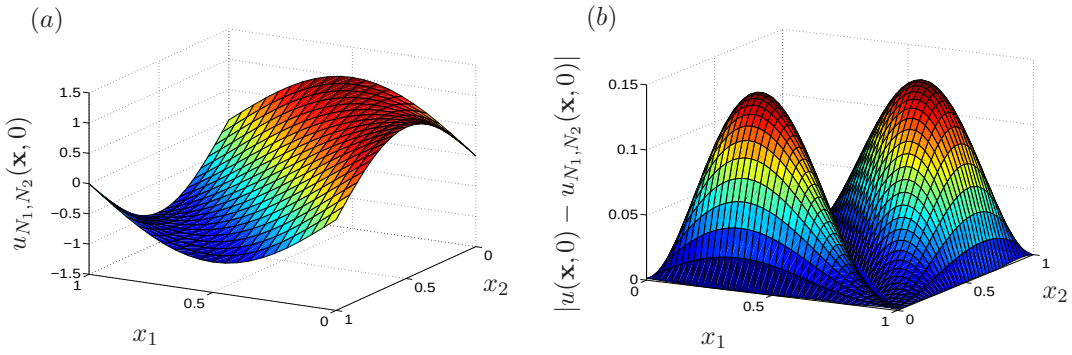

Figure 2.13: **(a)** The MFS approximation $u_{N_1,N_2}(\mathbf{x},0)$ plotted over $\mathbf{x}\in D$ and **(b)** the absolute error for $(\mathbf{x},0)\in D_T$, both obtained with $\delta = 0.1\%$, $h = 0.1$, $T = 0.5$, $\lambda = 10^{-8}$, and parameters from case (ii), for Example 3.

been plotted using parameters from case (iii), and this plot is the most accurate compared to the previous figures. The reason for this is that we have used more collocation and source points, however, the condition number has increased, therefore, we have had to use a larger value for λ.

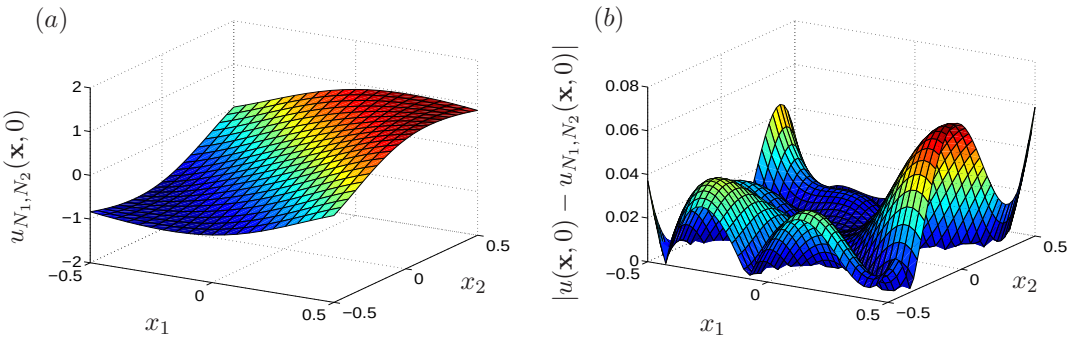

Figure 2.14: **(a)** The MFS approximation $u_{N_1,N_2}(\mathbf{x},0)$ plotted over $\mathbf{x}\in D$ and **(b)** the absolute error for $(\mathbf{x},0)\in D_T$, both obtained with $\delta = 5\%$, $h = 0.11$, $T = 1$, $\lambda = 10^{-1}$, and parameters from case (iii), for Example 3.

Figures 2.15(a), (b) and (c) display the maximum absolute errors plotted over $h = [0,2]$, and we find that results are better for smaller h, for all cases (i), (ii) and (iii). Lastly, Figures 2.16(a), (b) and (c) contain the maximum absolute errors plotted as a function of T showing that for larger T we obtain better results, and this result shows that for this example the data on the boundary is more important than the data on the final condition for larger values of T. In fact, in Figures 2.17(a) and 2.17(b) we have set $K = 0$ (i.e. no collocation points placed on the final base), and we see that for large T (= 0.5 for case (ii) and 1 for case (iii)) there is no significant difference in accuracy compared to Figures 2.13(b) and 2.14(b), therefore, collocation points placed on the boundary are more important. However, for smaller values of T data on

the final base becomes more important, which is to be expected.

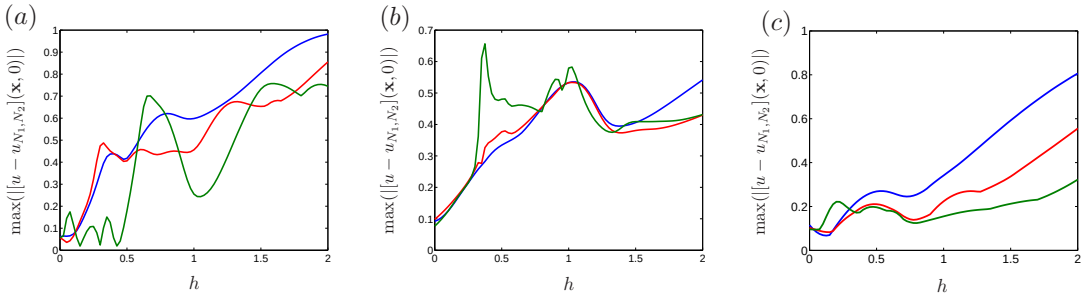

Figure 2.15: Plots of the maximum absolute error, $\max_{x\in D}|u(\mathbf{x},0)-u_{N_1,N_2}(\mathbf{x},0)|$, over $h\in[0,2]$, and **(a)** $\lambda=10^{-3}$ (—), $\lambda=10^{-5}$ (—), $\lambda=10^{-7}$ (—), for $T=0.25$, and parameters taken from case (i), with $\delta=5\%$, **(b)** $\lambda=10^{-6}$ (—), $\lambda=10^{-8}$ (—), $\lambda=10^{-10}$ (—), for $T=0.5$, and parameters taken from case (ii), with $\delta=0.1\%$, **(c)** $\lambda=10^{-1}$ (—), $\lambda=10^{-2}$ (—), $\lambda=10^{-3}$ (—), for $T=1$, and parameters taken from case (iii), with $\delta=5\%$, for Example 3.

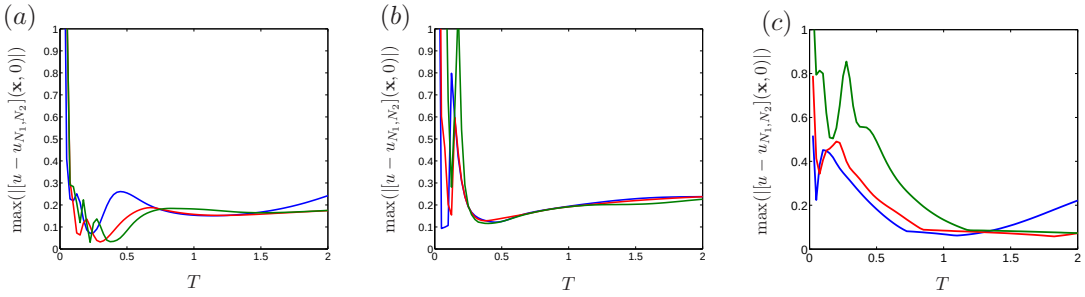

Figure 2.16: Plots of the maximum absolute error, $\max_{x\in D}|u(\mathbf{x},0)-u_{N_1,N_2}(\mathbf{x},0)|$, as a function of T, and **(a)** $\lambda=10^{-3}$ (—), $\lambda=10^{-5}$ (—), $\lambda=10^{-7}$ (—), for $h=0.1$, and parameters taken from case (i), with $\delta=5\%$, **(b)** $\lambda=10^{-6}$ (—), $\lambda=10^{-8}$ (—), $\lambda=10^{-10}$ (—), for $h=0.1$, and parameters taken from case (ii), with $\delta=0.1\%$, **(c)** $\lambda=10^{-1}$ (—), $\lambda=10^{-2}$ (—), $\lambda=10^{-3}$ (—), for $h=0.11$, and parameters taken from case (iii), with $\delta=5\%$, for Example 3.

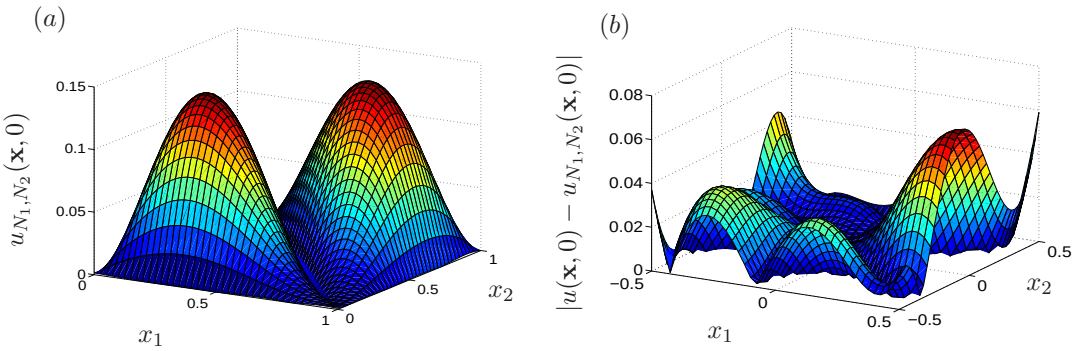

Figure 2.17: Plots of the the absolute error over $(\mathbf{x},0)\in D_T$ for **(a)** $\delta=0.1\%$, $h=0.1$, $T=0.5$, $\lambda=10^{-8}$, and parameters from case (ii), and **(b)** $\delta=5\%$, $h=0.11$, $T=1$, $\lambda=10^{-1}$, and parameters from case (iii). Both plots obtained with $K=0$ (no collocation points placed on the final base), for Example 3.

2.6 Summary of Chapter 2

In this chapter, an MFS was proposed and investigated for the inverse and ill-posed problem of one and two-dimensional heat conduction backwards in time. This extends the work of [62] and the work in the previous chapter, which applied the MFS to the direct time-dependent heat equation in one and two-dimensions, respectively.

A denseness result was proved for sets of linear combinations of fundamental solutions in the L^2-sense for any fixed time. This enables collocation points to be placed on any flat surface $T_0 > 0$. The MFS was then implemented and generalized, by allowing for varying numbers of collocation and source points, resulting in over and underdetermined systems.

Three numerical examples were investigated and compared with results obtained in the papers [86, 82, 59] with different meshless methods. In the first example compared to results obtained in [86, 59] and found that we were able to extend the final time T further than [86], where the MFS was also used but with a different placement of source points. In the second example we changed the boundary condition from Dirichlet to Neumann, and the MFS still produced accurate and stable results, and when comparing to the results obtained in [60], we were able to substantially extend the final time point. In the final example comparisons were made to the papers [86, 82, 59], with results again comparable. With error plots we investigated different placements of source points, final time points and varying the Tikhonov regularization parameter and discovered regions where better results were achievable. This work could possibly be used as a guide for future work or real world applications.

CHAPTER 3

THE TWO-DIMENSIONAL CAUCHY HEAT CONDUCTION PROBLEM

3.1 Introduction

The Cauchy problem is a classical inverse problem, where temperature and normal heat flux data is missing from a part of the boundary, and is recovered from data overspecified on the remaining part of the boundary. For this Cauchy problem we assume that the boundary shape is known (including the part of the boundary with missing data), and thermal diffusivities, conductivities etc. are also known. The problem is ill-posed, see [48], since even if a solution exists it will not depend continuously on the given data.

In [84] it was demonstrated that the MFS worked well and was easy to adjust to various solution domains. Thus it is natural to extend [84] to the time-dependent setting, and to the best of the authors' knowledge, there are considerably fewer results than in the stationary case.

3.2 Mathematical formulation of the Cauchy problem

We use the same notation as in the previous two chapters, with domain D and boundary Γ, extended in time to $D_T = D \times (0, T]$ and $\Gamma_T = \Gamma \times (0, T]$. Let $\Gamma^{(1)}$ be an open arc of Γ and set $\Gamma^{(2)} = \Gamma \backslash \overline{\Gamma^{(1)}}$. Define $\Gamma_T^{(i)} = \Gamma^{(i)} \times (0, T]$, $i = 1, 2$. We assume that Cauchy data is given on $\Gamma_T^{(1)}$.

We wish to construct an approximation for the solution of the heat equation u, endowed with Dirichlet and Neumann conditions on the boundary $\Gamma_T^{(1)}$, i.e. to find an approximation to

$$\frac{\partial u}{\partial t}(\mathbf{x}, t) = \Delta u(\mathbf{x}, t), \quad (\mathbf{x}, t) \in D_T, \tag{3.1}$$

$$u(\mathbf{x},t) = g_1(\mathbf{x},t), \quad (\mathbf{x},t) \in \Gamma_T^{(1)}, \tag{3.2}$$

$$\frac{\partial u}{\partial \boldsymbol{\nu}}(\mathbf{x},t) = g_2(\mathbf{x},t), \quad (\mathbf{x},t) \in \Gamma_T^{(1)}, \tag{3.3}$$

where $\boldsymbol{\nu}$ is the outward unit normal to the boundary Γ, $\frac{\partial u}{\partial \boldsymbol{\nu}} = \nabla u \cdot \boldsymbol{\nu}$, and g_1 and g_2 are sufficiently smooth functions. Note that no initial condition is prescribed. The uniqueness of a solution is still guaranteed as is well known and explained below. Although it is known that initial data is not required for Cauchy problems for the heat equation, almost all numerical examples in the literature do have initial data imposed. It is noteworthy that with the MFS it is easy to work with or without this data.

For smooth Cauchy data given on a non-characteristic smooth curve (or surface in three-dimensions) uniqueness of a solution to the heat equation with this data is a consequence of Holmgren's theorem [57]. More general results, for example for non-characteristic Cauchy problems and also for the closely connected problem of unique continuation, have been presented in various function classes and solution domains. We do not aim to give an overview of these results and refer instead to two more recent papers which contain overviews of some of these [101, 104]. In our situation with the smoothness imposed on the data and on the solution domain we can thus be certain that there can be at most one solution to the Cauchy problem. We shall assume that data is such that a solution exists. Note though that this solution will not depend continuously on the data due to the ill-posedness of the Cauchy problem.

3.3 The MFS for the two-dimensional heat conduction Cauchy problem

In this section, we construct an approximate solution to (3.1)–(3.3) using the MFS. To do this we need to make use of the fundamental solution of the two-dimensional heat equation (3.1), given by (1.5) having partial derivative with respect to x_1 and x_2, given by

$$\frac{\partial F}{\partial x_j}(\mathbf{x},t;\mathbf{y},\tau) \;=\; -\frac{H(t-\tau)}{8\pi(t-\tau)^2}(x_j - y_j)\exp\left(-\frac{|\mathbf{x}-\mathbf{y}|^2}{4(t-\tau)}\right), \quad j = 1,2. \tag{3.4}$$

The approximation to (3.1) takes the form

$$u_M(\mathbf{x},t) \;=\; \sum_{m=1}^{2M}\sum_{j=1}^{N} c_m^{(j)} F(\mathbf{x},t;\mathbf{y}_j,\tau_m), \quad (\mathbf{x},t) \in \overline{D_T}. \tag{3.5}$$

As usual, the source points will be placed at the time points

$$\tau_m \;=\; \frac{2(m-M)-1}{2M}T, \quad m=1,\ldots,2M, \tag{3.6}$$

and at such time points N points will be placed in space on a pseudo-boundary generated by expanding the boundary Γ of the domain using dilatation, and using, again, h as an expansion parameter. The collocation points will be placed at the time points

$$t_i \;=\; \frac{i}{M_1}T, \quad i=1,\ldots,M_1, \tag{3.7}$$

and N_1 points will be placed on $\Gamma^{(1)}$ for a fixed t_i. See Figure 3.1 for an example of a solution domain, and a possible placement of source points in space.

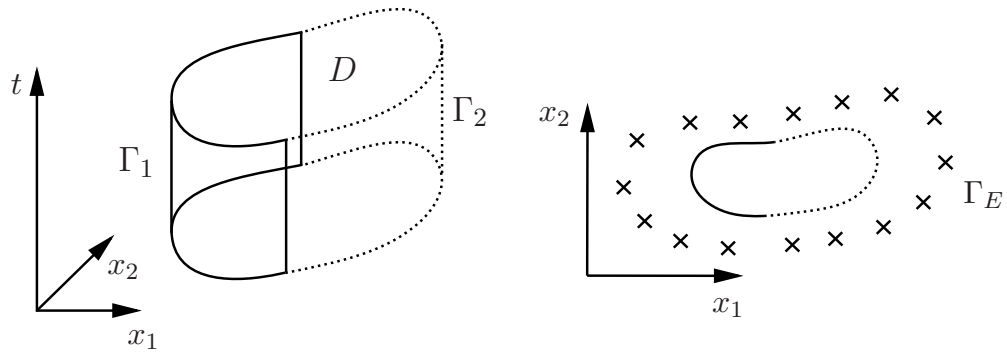

Figure 3.1: Representation of a solution domain and the location of Cauchy data (——) on $\Gamma_T^{(1)}$, unknown boundary data (▪▪▪) on $\Gamma_T^{(2)}$, and the source points ($\times$) on Γ_E.

In this chapter we consider the static MFS, and finding the position of the source points is accomplished by trial and error.

Using (3.5), and the collocation and source points defined above, we obtain the following linear system of equations:

$$u_M(\mathbf{x}_l,t_i) \;=\; g_1(\mathbf{x}_l,t_i), \quad l=1,\ldots,N_1,\ i=1,\ldots,M_1, \tag{3.8}$$

$$\frac{\partial u_M}{\partial \boldsymbol{\nu}}(\mathbf{x}_l,t_i) \;=\; g_2(\mathbf{x}_l,t_i), \quad l=1,\ldots,N_1,\ i=1,\ldots,M_1. \tag{3.9}$$

We employ Tikhonov regularization, see Section 0.5 of the Introduction for details.

In the next section, we investigate the accuracy of MFS approximations for the Cauchy

problem (3.1)–(3.3) for four examples with different shapes of the solution domain and for various boundary data, with the domains being a rectangle and an epitrochoid in Examples 1 and 2, respectively, and in Examples 3 and 4 data is given on an internal teardrop shaped boundary. However, in Example 4 the inner Neumann boundary data is generated after solving a forward problem.

3.4 Numerical results

In the numerical experiments contained in this section we use (unless otherwise stated) $h = 3$, $M = 10$, $N = 40$, $M_1 = 10$ and, $N_1 = 40$ (which results in 800 equations and 800 unknowns). Random noise is added to the functions g_1 and g_2 in equations (3.8) and (3.9), respectively, as follows:

$$g_n^p = (1 + p_n(2 \times \text{rand} - 1))g_n, \quad n = 1, 2, \tag{3.10}$$

where p_n is the percentage of random noise we add to the boundary and rand generates a uniformly distributed random number in the range $(0, 1)$. To compare the accuracy of the MFS approximations with the exact data we use the root mean square error (RMSE) and the relative root mean square error (RRMSE), defined as:

$$\text{RMSE}(\mathbf{u}, \tilde{\mathbf{u}}) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(u_i - \tilde{u}_i)^2} \tag{3.11}$$

and

$$\text{RRMSE}(\mathbf{u}, \tilde{\mathbf{u}}) = \frac{\sqrt{\sum_{i=1}^{N}(u_i - \tilde{u}_i)^2}}{\sqrt{\sum_{i=1}^{N} u_i^2}}, \tag{3.12}$$

where N is the size of the vectors $\mathbf{u}$ and $\tilde{\mathbf{u}}$.

3.4.1 Example 1

In this first example we consider a rectangular domain $D = (-1, 1)\times(-0.5, 0.5)$, final time $T = 1$, $D_T = D \times (0, 1]$, and place Cauchy data along the boundary part $\Gamma_T^{(1)} = \Gamma^{(1)} \times (0, 1]$, where $\Gamma^{(1)} = (-1, 1)\times\{-0.5\}$, and reconstruct data along the boundary part $\tilde{\Gamma}_T^{(2)} = \tilde{\Gamma}^{(2)} \times (0, 1]$, where $\tilde{\Gamma}^{(2)} = (-1, 1) \times \{0.5\}$ (a line segment of $\Gamma^{(2)}$). The N source points will be equally distributed on the pseudo-boundary of the rectangular domain $(-1 - h, 1 + h) \times (-0.5 - h, 0.5 + h)$, where

$h = 3$. See Figure 3.2 for a representation of the rectangular domain and possible placement of collocation and source points.

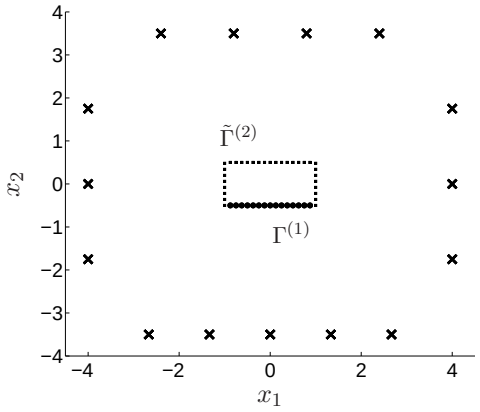

Figure 3.2: Representation of the rectangular domain, Cauchy data points (•) on $\Gamma^{(1)}$, the unknown boundary data (▪▪▪) on $\Gamma^{(2)}$, and the source points (×).

On $\Gamma_T^{(1)}$ the outward unit normal is $\boldsymbol{\nu} = (0, -1)$, and on $\tilde{\Gamma}_T^{(2)}$ we have $\boldsymbol{\nu} = (0, 1)$. We consider the analytic solution given by $u(\mathbf{x}, t) = 4t + |\mathbf{x}|^2$, i.e. we solve the problem

$$\frac{\partial u}{\partial t}(\mathbf{x}, t) = \Delta u(\mathbf{x}, t), \quad (\mathbf{x}, t) \in D_T, \tag{3.13}$$

$$u(\mathbf{x}, t) = g_1(\mathbf{x}, t) = 4t + |\mathbf{x}|^2, \quad (\mathbf{x}, t) \in \Gamma_T^{(1)}, \tag{3.14}$$

$$\frac{\partial u}{\partial \boldsymbol{\nu}}(\mathbf{x}, t) = g_2(\mathbf{x}, t) = -2x_2, \quad (\mathbf{x}, t) \in \Gamma_T^{(1)}. \tag{3.15}$$

Noise is added to equations (3.14) and (3.15) as in (3.10), with $p_1 = 1\% = 0.01$ and $p_2 = 5\% = 0.05$ (with $p_2 > p_1$ due to heat flux errors usually being larger than temperature errors). The unknown data (to be determined) on $\tilde{\Gamma}_T^{(2)}$ is given by

$$u(\mathbf{x}, t) = 4t + |\mathbf{x}|^2, \quad (\mathbf{x}, t) \in \tilde{\Gamma}_T^{(2)}, \tag{3.16}$$

$$\frac{\partial u}{\partial \boldsymbol{\nu}}(\mathbf{x}, t) = 2x_2, \quad (\mathbf{x}, t) \in \tilde{\Gamma}_T^{(2)}. \tag{3.17}$$

Figure 3.3(b) presents a plot of the MFS approximation of the temperature along the boundary $x_2 = 0.5$ when no noise is added to the boundary data, and results appear accurate and stable when compared with the exact solution in Figure 3.3(a), and confirmed in the absolute error plot in Figure 3.3(c). The error increases for the initial and final time points, which is to be expected. Plots are also produced for the approximation of the outward heat flux data on the boundary $x_2 = 0.5$, which is often difficult to recover, with results similar to those in Figure

3.3 being produced, see Figure 3.4. We note that for this example the MATLAB code took 3.07 seconds to run, and the condition number of the matrix was $\mathcal{O}(10^{59})$.

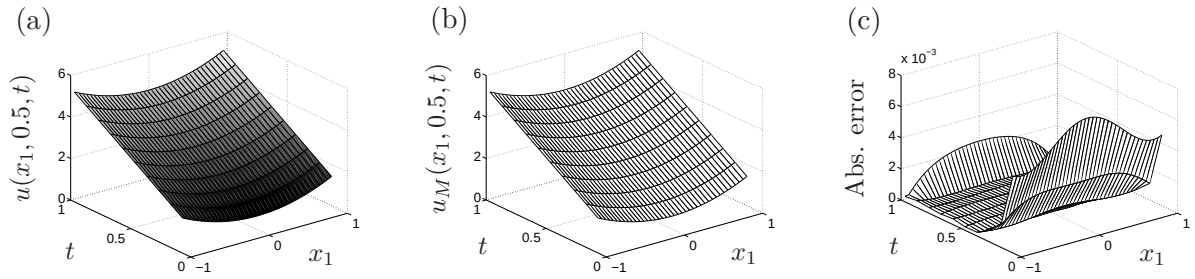

Figure 3.3: (a) The temperature, (b) the MFS approximation and (c) the absolute error for $x_2 = 0.5$, $(x_1, t) = (-1, 1) \times (0, 1]$, obtained with $\lambda = 10^{-14}$, no noise $p_1 = p_2 = 0$, when Cauchy data is given along $x_2 = -0.5$, $(x_1, t) = (-1, 1) \times (0, 1]$, for Example 1.

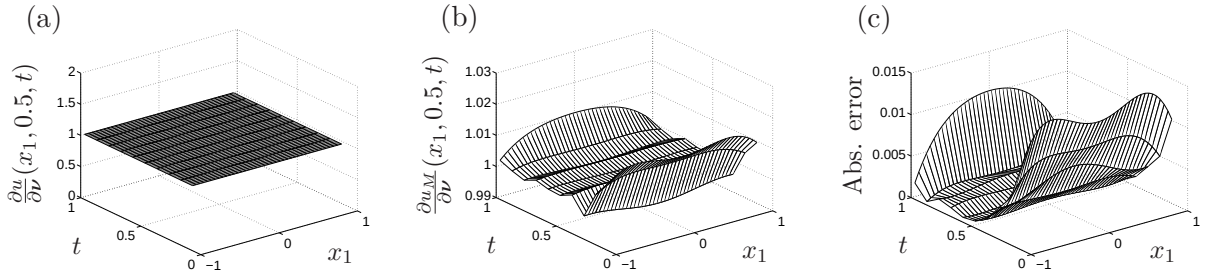

Figure 3.4: (a) The outward heat flux, (b) the MFS approximation and (c) the absolute error for $x_2 = 0.5$, $(x_1, t) = (-1, 1) \times (0, 1]$, obtained with $\lambda = 10^{-14}$, no noise $p_1 = p_2 = 0$, when Cauchy data is given along $x_2 = -0.5$, $(x_1, t) = (-1, 1) \times (0, 1]$, for Example 1.

In Figure 3.5 we present plots of the MFS approximation for the temperature and the outward heat flux when noise is added to the boundary data. Figure 3.5(a) displays a stable approximation for the temperature, but in Figure 3.5(c) it is clear that noise has caused instability in the heat flux data, however, when we consider the RMSE and RRMSE values presented in Table 3.1, for Figure 3.5, we see that the error is of the same order as the noise added to the boundary data, and varying the regularization parameter does not improve the results much further. The RMSE and RRMSE values for Figures 3.3 and 3.4 are also given in Table 3.1, and show that the method is satisfactorily accurate.

Table 3.1: The RMSE and RRMSE values, see (3.11) and (3.12), for the approximations presented in Figures 3.3–3.5 (noise free data in 3.3(b) and 3.4(b)).

Function	u_M 3.3(b)	$\frac{\partial u_M}{\partial \boldsymbol{\nu}}$ 3.4(b)	u_M 3.5(a)	$\frac{\partial u_M}{\partial \boldsymbol{\nu}}$ 3.5(c)
RMSE	0.00207893	0.00587242	0.0324379	0.0665422
RRMSE	0.000690793	0.00587242	0.0107786	0.0665422

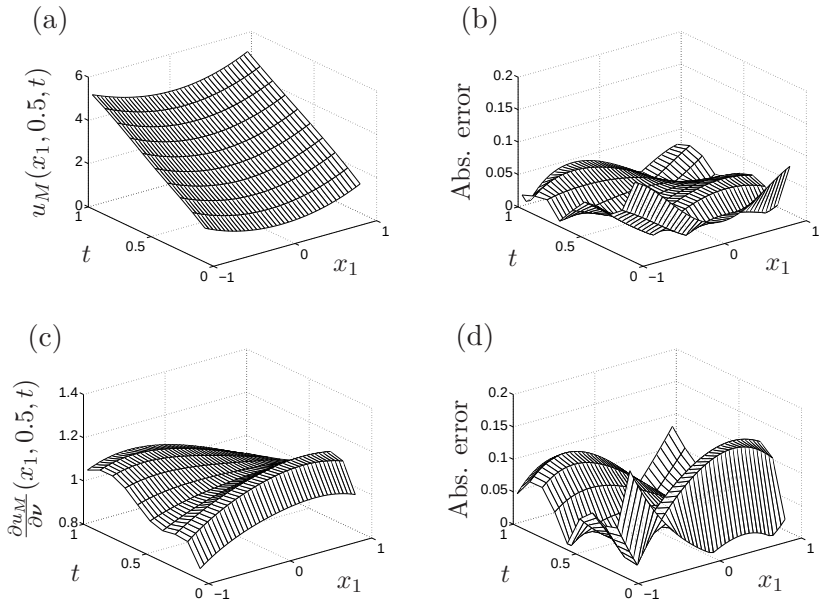

Figure 3.5: (a) The MFS approximation of the temperature, and (c) the MFS approximation of the outward heat flux, with (b) and (d) the absolute error of the approximations in (a) and (c), respectively, for $x_2 = 0.5$, $(x_1, t) = (-1, 1) \times (0, 1]$, obtained with $\lambda = 10^{-6}$, $p_1 = 1\%$ and $p_2 = 5\%$, when Cauchy data is given along $x_2 = -0.5$, $(x_1, t) = (-1, 1) \times (0, 1]$, for Example 1.

In Figures 3.6 and 3.7 we increase the difficulty of recovering the boundary data on $x_2 = 0.5$ by assuming that Cauchy data is given on a smaller portion of the line segment $\tilde{\Gamma}^{(2)} = (-1, 1) \times \{-0.5\}$. In Figures 3.6 and 3.7 we produce absolute error plots, when no noise is added to the boundary data, and Cauchy data is given along the half-segment

$$\text{(HS)} \ : \ x_2 = -0.5, \ (x_1, t) = (-0.5, 0.5) \times (0, 1]$$

and the quarter-segment

$$\text{(QS)} \ : \ x_2 = -0.5, \ (x_1, t) = (-0.25, 0.25) \times (0, 1],$$

respectively. Note that we have only changed the portion where we place collocation points and not the actual number of collocation points, as this has an impact on the approximation. We notice that approximations become marginally poorer the smaller the portion is where Cauchy data is given, but are still very good approximations. However, there is a greater impact to the approximations when the portion where Cauchy data is given is decreased, especially for the heat flux data, when we apply noise to the Cauchy data; the results for this are given in Table

3.3. The corresponding reconstructions in the noisy case are similar to those in Figure 3.5 and therefore left out.

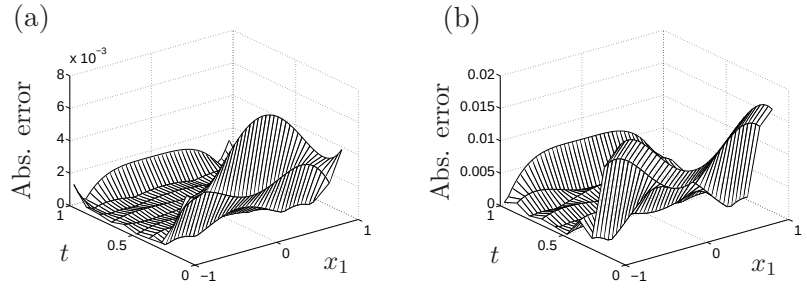

Figure 3.6: The absolute error of (a) the MFS approximation of the temperature, and (b) the MFS approximation of the outward heat flux, for $x_2 = 0.5$, $(x_1, t) = (-1, 1) \times (0, 1]$, obtained with $\lambda = 10^{-14}$, no noise $p_1 = p_2 = 0$, when Cauchy data is given along $x_2 = -0.5$, $(x_1, t) = (-0.5, 0.5) \times (0, 1]$, for Example 1.

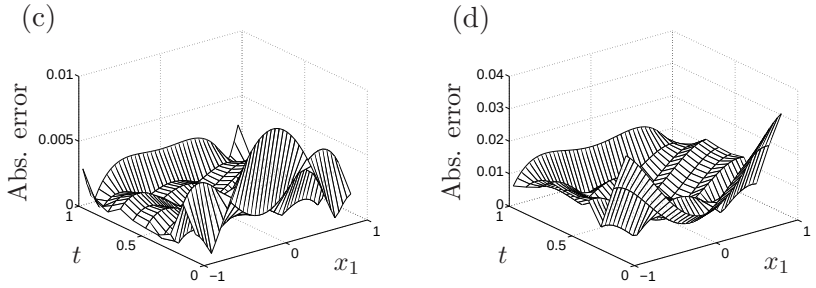

Figure 3.7: The absolute error of (a) the MFS approximation of the temperature, and (b) the MFS approximation of the outward heat flux, for $x_2 = 0.5$, $(x_1, t) = (-1, 1) \times (0, 1]$, obtained with $\lambda = 10^{-14}$, no noise $p_1 = p_2 = 0$, when Cauchy data is given along $x_2 = -0.5$, $(x_1, t) = (-0.25, 0.25) \times (0, 1]$, for Example 1.

Table 3.2: The RMSE and RRMSE values, see (3.11) and (3.12), for the approximations presented in Figures 3.6 and 3.7 (noise free data).

Function	u_M (3.6(a), HS)	$\frac{\partial u_M}{\partial \boldsymbol{\nu}}$ (3.6(b), HS)	u_M (3.7(a), QS)	$\frac{\partial u_M}{\partial \boldsymbol{\nu}}$ (3.7(b), QS)
RMSE	0.00200764	0.00624982	0.00225143	0.00980177
RRMSE	0.000667104	0.00624982	0.000748114	0.00980177

Table 3.3: The RMSE and RRMSE values for, see (3.11) and (3.12), MFS approximations of the temperature u_M and the outward heat flux $\partial u_M/\partial \boldsymbol{\nu}$ generated for $x_2 = 0.5$, $(x_1, t) = (-1, 1) \times (0, 1]$, obtained with $\lambda = 10^{-6}$, $p_1 = 1\%$ and $p_2 = 5\%$, when Cauchy data is given along the HS: $x_2 = -0.5$, $(x_1, t) = (-0.5, 0.5) \times (0, 1]$ and the QS: $x_2 = -0.5$, $(x_1, t) = (-0.25, 0.25) \times (0, 1]$, for Example 1.

Function	u_M (HS)	$\frac{\partial u_M}{\partial \boldsymbol{\nu}}$ (HS)	u_M (QS)	$\frac{\partial u_M}{\partial \boldsymbol{\nu}}$ (QS)
RMSE	0.0675812	0.153444	0.0903007	0.253353
RRMSE	0.0224561	0.153444	0.0300054	0.253353

3.4.2 Example 2

In this example the domain is an epitrochoid, which has boundary defined by

$$(x_1(\phi), x_2(\phi)) = \left((a+b)\cos(\phi) - d\cos\left(\frac{a+b}{b}\phi\right), (a+b)\sin(\phi) - d\sin\left(\frac{a+b}{b}\phi\right)\right), \quad (3.18)$$
$$\phi \in [0, 2\pi).$$

We consider the case when $a = 1$, $b = 0.25$ and $d = 0.125$ (note that ϕ is not the polar angle). The source points will be placed on a pseudo-boundary generated by extending radially the boundary points given in (3.18), from r to $r + h = r + 3$. In Figure 3.8 is a representation of the epitrochoid domain and possible placement of collocation and source points. Calculation of the outward unit normal for the directional derivative can be achieved, for example, by differentiating (3.18), converting to polar coordinates, and subtracting $\pi/2$ from the polar angle. We use the following solution of the heat equation

$$u(x_1, x_2, t) = e^{x_1+x_2}\cos(x_1 + x_2 + 4t),$$

to generate the Dirichlet and Neumann data, (3.2) and (3.3), respectively.

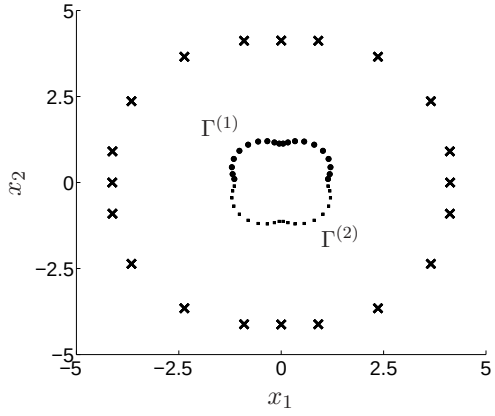

Figure 3.8: Representation of the epitrochoid domain, Cauchy data points (•) on $\Gamma_T^{(1)}$, the unknown boundary data (▪▪▪) on $\Gamma_T^{(2)}$, and the source points (×).

In this example we take Cauchy data over the interval described by $\phi \in (0, \pi)$, with the data to be recovered on the interval $(\pi, 2\pi)$. In Figures 3.9 and 3.10, we recover the temperature and heat flux when no noise is added to the boundary data. From the absolute error plots 3.9(b)

and 3.10(b) we see that the largest errors occur at $\phi = 3\pi/2$, which makes sense since the point on the boundary corresponding to this value is the furthest away from the Cauchy data. Figures 3.11(b) and 3.11(d) show reasonable approximations also when noise is applied to the boundary data, for what appears to be a rather difficult example.

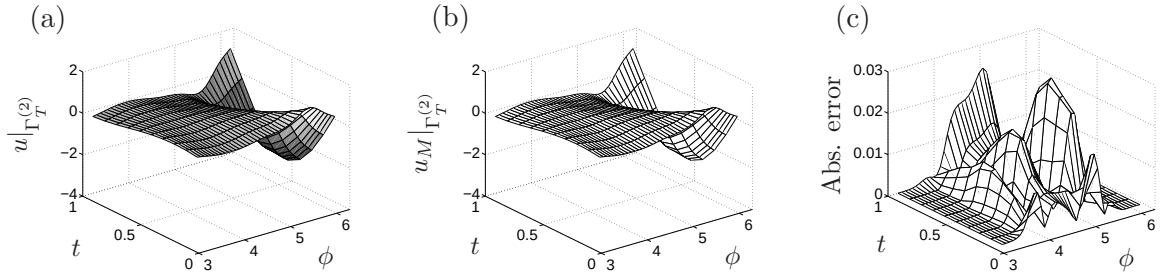

Figure 3.9: (a) The temperature, (b) the MFS approximation and (c) the absolute error for $\phi \in (\pi, 2\pi)$, obtained with $\lambda = 10^{-14}$, no noise $p_1 = p_2 = 0$, when Cauchy data is given for $\phi \in (0, \pi)$, for Example 2.

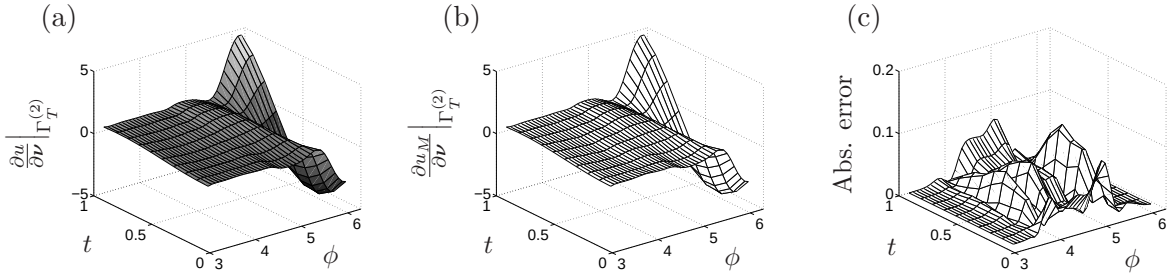

Figure 3.10: (a) The outward heat flux, (b) the MFS approximation and (c) the absolute error for $\phi \in (\pi, 2\pi)$, obtained with $\lambda = 10^{-14}$, no noise $p_1 = p_2 = 0$, when Cauchy data is given for $\phi \in (0, \pi)$, for Example 2.

Table 3.4: The RMSE and RRMSE values, see (3.11) and (3.12), for the approximations presented in Figures 3.9–3.11, (noise free data in 3.9(b) and 3.10(b)).

Function	u_M 3.9(b)	$\frac{\partial u_M}{\partial \boldsymbol{\nu}}$ 3.10(b)	u_M 3.11(a)	$\frac{\partial u_M}{\partial \boldsymbol{\nu}}$ 3.11(c)
RMSE	0.00777251	0.0321907	0.10967	0.238725
RRMSE	0.0104075	0.0259728	0.146849	0.192614

As in Example 1 and with similar conclusions, we reduce the interval over which we place the Cauchy data, and in Table 3.5 we present these results. The corresponding figures do not show any additional features and are therefore left out.

We point out that in the two examples considered no initial data has been used. This has apparently not been the case in previous studies in the literature. We note that if this data were supplied the reconstructions should be slightly more accurate. In fact, an attractive additional

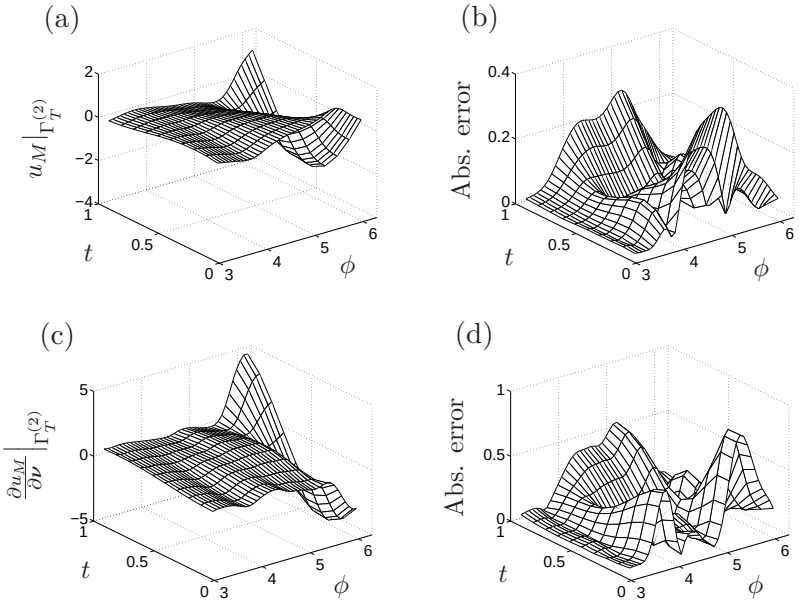

Figure 3.11: (a) The MFS approximation of the temperature, and (c) the MFS approximation of the outward heat flux, with (b) and (d) the absolute error of the approximations in (a) and (c), respectively, for $\phi \in (\pi, 2\pi)$, obtained with $\lambda = 10^{-6}$, $p_1 = 1\%$ and $p_2 = 5\%$, when Cauchy data is given for $\phi \in (0, \pi)$, for Example 2.

Table 3.5: The RMSE and RRMSE values for the temperature and the outward heat flux for the boundaries (a) $\phi \in (\frac{3\pi}{4}, \frac{9\pi}{4})$ (when Cauchy data is given along $\phi \in (\frac{\pi}{4}, \frac{3\pi}{4})$) and (b) $\phi = (\frac{5\pi}{8}, \frac{19\pi}{8})$ (when Cauchy data is given along $\phi \in (\frac{3\pi}{8}, \frac{5\pi}{8})$), obtained with $\lambda = 10^{-14}$, no noise, for Example 2.

Function	u_M (a)	$\frac{\partial u_M}{\partial \boldsymbol{\nu}}$ (a)	u_M (b)	$\frac{\partial u_M}{\partial \boldsymbol{\nu}}$ (b)
RMSE	0.0295178	0.0933453	0.0649556	0.155235
RRMSE	0.0193955	0.0357425	0.0340448	0.0444876

feature of the proposed MFS is that the initial data can also be reconstructed. To illustrate this, in Figure 3.12 we recover the initial data at time $t = 0$, given in Figure 3.12(a), when Cauchy data is given for $\phi \in (0, \pi)$.

3.4.3 Example 3

In this example we consider an inner teardrop shaped boundary, where Cauchy data is given, defined by

$$(x_1(\phi), x_2(\phi)) = (0.5\cos\phi, 0.4\sin\phi - 0.2\sin(2\phi)), \quad \phi \in [0, 2\pi) \tag{3.19}$$

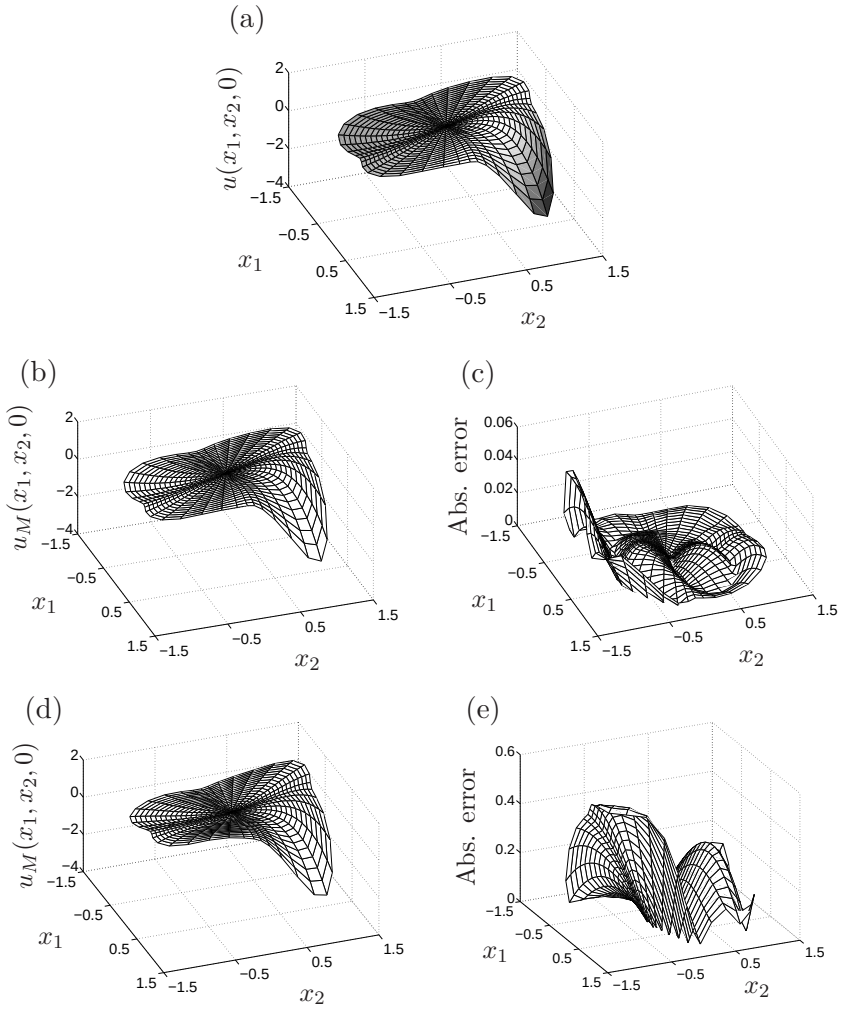

Figure 3.12: (a) The temperature, (b) and (d) The MFS approximation, and (c) and (e) the absolute error of the approximations in (b) and (d), respectively, for the initial time $t = 0$ and $(x_1, x_2) \in D$, obtained with $\lambda = 10^{-14}$, $p_1 = 0$ and $p_2 = 0$ for (b) and (c), and $\lambda = 10^{-6}$, $p_1 = 1\%$ and $p_2 = 5\%$ for (d) and (e), when Cauchy data is given for $\phi \in (0, \pi)$ on $\Gamma^{(1)}$, for Example 2.

and recover data on an outer elliptical boundary, defined by

$$(x_1(\phi), x_2(\phi)) = (1.5\cos\phi, \sin\phi), \quad \phi \in [0, 2\pi). \tag{3.20}$$

The domain D is the region enclosed by the inner (3.19) and outer (3.20) boundaries. Note that in this example, as in the last example, ϕ is not the polar angle. The source points will be placed on inner and outer pseudo-boundaries generated by decreasing or increasing the radius of the boundary points. For the inner pseudo-boundary we replace r, in (3.19) and (3.20), by $r - h_I = r - 0.2$ and $r + h_O = r + 3$, respectively, i.e. the constants $h_I = 0.2$ and $h_O = 3$

represents the distance the source points are from the inner and outer boundaries. For this example we set $M = 10$, $N = 40$, $M_1 = 10$ and, $N_1 = 40$, however, for a fixed time, $N/2$ source points will be placed on the outer pseudo-boundary and $N/2$ source points placed on the inner pseudo-boundary (and there are still 800 equations and 800 unknowns). In Figure 3.13 is a representation of the shape of the inner and outer boundary and a possible placement of collocation and source points. For this example the analytic solution has a singularity located

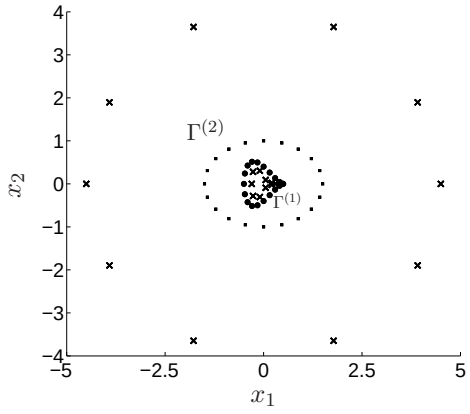

Figure 3.13: Representation of the domain for Example 3, Cauchy data points (•) on $\Gamma_T^{(1)}$, the unknown boundary data (▪▪▪) on $\Gamma_T^{(2)}$, and the source points (×) (placed both inside and outside the domain).

at $t = 0$. The analytic solution is given by the fundamental solution of the two-dimensional heat equation (1.5), i.e.

$$u(x_1, x_2, t) = F(x_1, x_2, t; 0, 0, 0),$$

and we use this, along with the derivative of the fundamental solution (3.4), to generate the Dirichlet and Neumann data.

In Figures 3.14, 3.15 and 3.16 we produce similar figures to those found in Example 2. No additional features can be reported, reasonable approximations were obtained also with noisy data as in the previous examples, showing that annular solution domains can be handled as well. Note that the roles of $\Gamma_T^{(1)}$ and $\Gamma_T^{(2)}$ can also be interchanged in this example. The RMSE and RRMSE results given in Table 3.6 are promising.

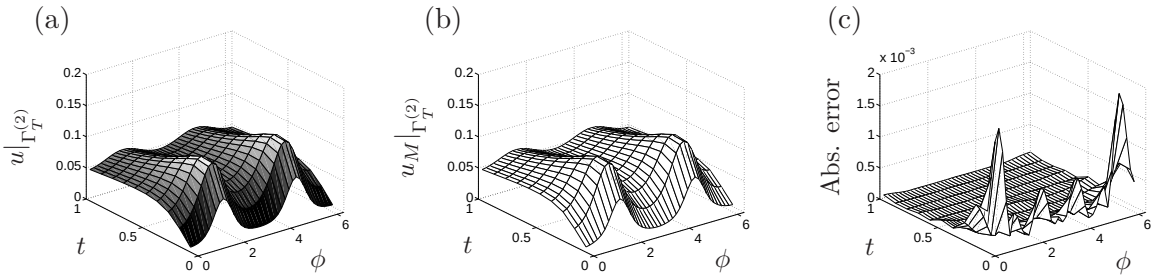

Figure 3.14: (a) The temperature, (b) the MFS approximation and (c) the absolute error for $\phi \in [0, 2\pi)$ on $\Gamma^{(2)}$, obtained with $\lambda = 10^{-12}$, no noise $p_1 = p_2 = 0$, when Cauchy data is given for $\phi \in [0, 2\pi)$ on $\Gamma^{(1)}$, for Example 3.

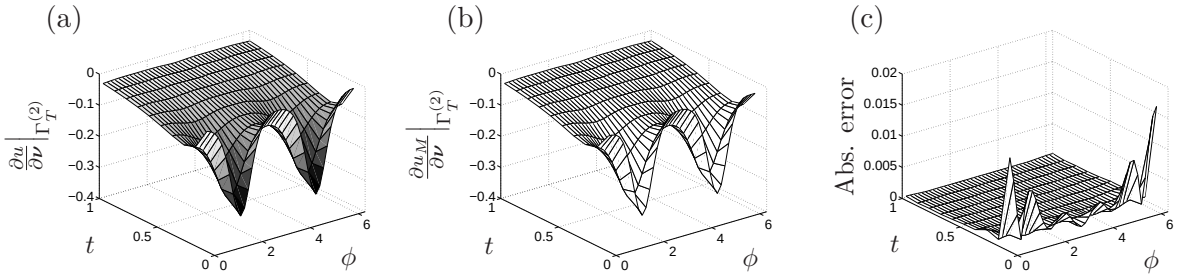

Figure 3.15: (a) The outward heat flux, (b) the MFS approximation and (c) the absolute error for $\phi \in [0, 2\pi)$ on $\Gamma^{(2)}$, obtained with $\lambda = 10^{-12}$, no noise $p_1 = p_2 = 0$, when Cauchy data is given for $\phi \in [0, 2\pi)$ on $\Gamma^{(1)}$, for Example 3.

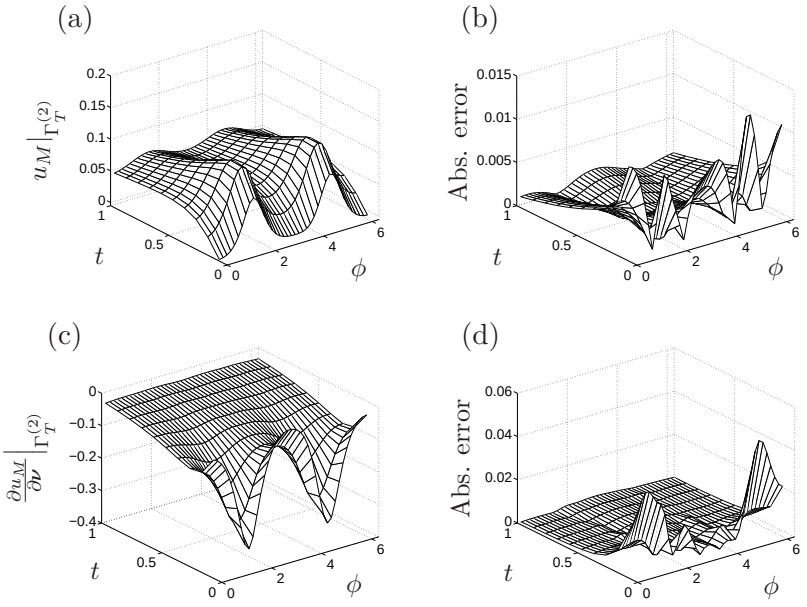

Figure 3.16: (a) The MFS approximation of the temperature, and (c) the MFS approximation of the outward heat flux, with (b) and (d) the absolute error of the approximations in (a) and (c), respectively, for $\phi \in [0, 2\pi)$ on $\Gamma_T^{(2)}$, obtained with $\lambda = 10^{-2}$, $p_1 = 1\%$ and $p_2 = 5\%$, when Cauchy data is given for $\phi \in [0, 2\pi)$ on $\Gamma_T^{(1)}$, for Example 3.

Table 3.6: The RMSE and RRMSE values, see (3.11) and (3.12), for the approximations presented in Figures 3.14–3.16 (noise free data in 3.14(b) and 3.15(b)).

Function	u_M 3.14(b)	$\frac{\partial u_M}{\partial \boldsymbol{\nu}}$ 3.15(b)	u_M 3.16(a)	$\frac{\partial u_M}{\partial \boldsymbol{\nu}}$ 3.16(c)
RMSE	0.000268	0.001971	0.002814	0.007670
RRMSE	0.004127	0.018569	0.043328	0.072235

3.4.4 Example 4

In this final example we consider the same boundaries, domain, and placement of source points as used in Example 3. This time, however, we do not have an analytic expression for the solution and therefore a forward problem is solved to obtain the Cauchy data on the inner boundary. The forward problem requires solving (3.1) and

$$u(\mathbf{x}, 0) = 0, \quad \mathbf{x} \in D, \tag{3.21}$$

$$u(\mathbf{x}, t) = -t^2 e^{-t} \cos(x_1), \quad (\mathbf{x}, t) \in \Gamma_T^{(1)}, \tag{3.22}$$

$$\frac{\partial u}{\partial \boldsymbol{\nu}}(\mathbf{x}, t) = 0, \quad (\mathbf{x}, t) \in \Gamma_T^{(2)}. \tag{3.23}$$

For this forward problem we set $h_I = 0.2$, $h_O = 2$, $\lambda = 10^{-5}$ and $M = 15$, $N = 60$, $M_1 = 15$ and, $N_1 = 60$ (which results in 1800 equations and 1800 unknowns), where, for a fixed time, $N_1 = 60$ is the number of collocation points on both the inner and outer boundaries. Note that we are not committing an inverse crime, since there are more collocation and source points being used for the forward problem (we also take different values for h_I, h_O and λ as well). After solving the forward problem we generate the missing Neumann data (3.9) on the inner boundary $\Gamma_T^{(1)}$, which will be used for the Cauchy data along with (3.22), in the inverse problem.

For the inverse problem we set $h_I = 0.22$, $h_O = 3$, $\lambda = 10^{-4}$ (used for both the noise and noiseless cases) and $M = 10$, $N = 40$, $M_1 = 10$ and, $N_1 = 40$ (which results in 800 equations and 800 unknowns). In Figure 3.17 are plots of the MFS approximations of the temperature and outward heat flux on $\Gamma_T^{(2)}$. The approximations required a large value for the Tikhonov regularization parameter, and moderately unstable results have been produced for the outward heat flux data 3.17(b) and 3.17(d), which should be equal to 0. However, the temperature data 3.17(a) and 3.17(c) appears rather stable, even when noise is applied to the Cauchy data.

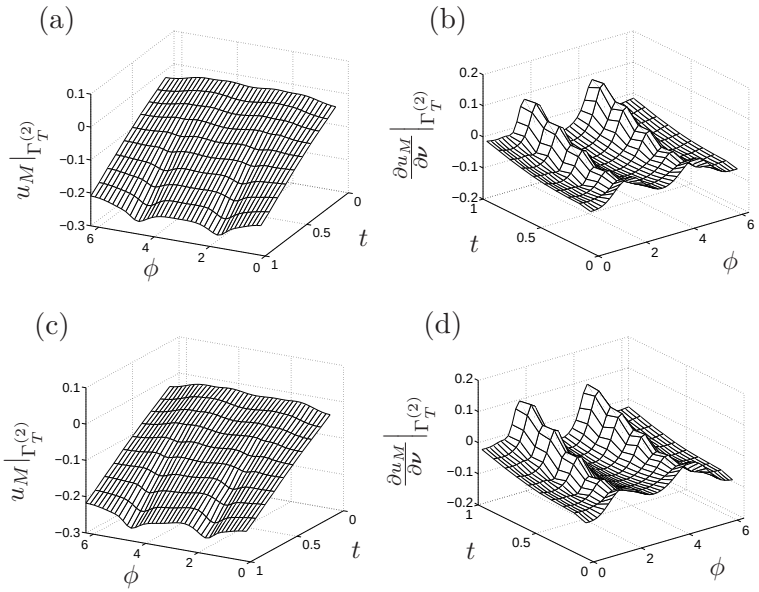

Figure 3.17: (a) and (c) The MFS approximation of the temperature and (b) and (d) the MFS approximation of the outward heat flux for $\phi \in [0, 2\pi)$ on $\Gamma^{(2)}$, obtained with $\lambda = 10^{-4}$, for (a) and (b) no noise $p_1 = p_2 = 0$, and for (c) and (d) $p_1 = 1\%$ and $p_2 = 5\%$, when Cauchy data is given for $\phi \in [0, 2\pi)$ on $\Gamma^{(1)}$, for Example 4.

3.5 Summary of Chapter 3

In this chapter, we have investigated the MFS for the two-dimensional Cauchy heat conduction problem, an inverse and ill-posed problem. The MFS [62] was adjusted to this Cauchy problem and numerical examples demonstrated the flexibility of the method in terms of solution domains (including a non-simply connected domain) and data (including examples with singular functions and without analytic expressions).

Four examples were considered in the numerics and it was highlighted in the first example, when considering a rectangular domain, that decreasing the size of the boundary part where data was imposed made the approximation less accurate. For example, the RMSE values when noise is applied to the boundary data (see Tables 3.1 and 3.2) increased from 0.0324379, to 0.0675812 when collocation points are placed on the half space and 0.0903007 when collocation points are placed on the quarter space. In the second example an epitrochoidal domain was considered, which highlighted an additional advantage for our method in that knowledge of the initial condition is not required. In Example 3 we had an inner teardrop shaped boundary and an outer elliptic boundary with an analytic solution given by the fundamental solution of the heat equation, and the recovered temperature and heat flux on the outer boundary appeared

accurate. Lastly, in Example 4, an analytic solution was missing and a forward problem was solved to generate the data for the inverse problem (making sure not to commit an inverse crime). Results appeared stable for the temperature when noise was applied in Example 4, however, the accuracy of the heat flux was not as good, and improving this could be investigated in a future work.

The numerical results show promise, being accurate and relatively stable with respect to noise added to the boundary data, for extending the method to other problems presented in subsequent chapters.

In the next chapter we move on to applying the MFS to radially symmetric heat conduction problems.

CHAPTER 4

THE RADIALLY SYMMETRIC BACKWARD HEAT CONDUCTION PROBLEM

4.1 Preliminaries

In this chapter we investigate the radially symmetric heat conduction problem governed by

$$\frac{1}{\alpha}\frac{\partial u}{\partial t}(r,t) = \frac{1}{r^{n-1}}\frac{\partial}{\partial r}\left(r^{n-1}\frac{\partial u}{\partial r}(r,t)\right) = \frac{\partial^2 u}{\partial r^2}(r,t) + \frac{n-1}{r}\frac{\partial u}{\partial r}(r,t), \qquad r_0 < r < R,\ 0 < t < T, \tag{4.1}$$

where α is the constant thermal diffusivity, r is the polar radius, with $r_0 = 0$ representing a solid body, and $r_0 > 0$ a hollow body, $T > 0$ is the final time point, and the problem is independent of the polar angle θ. We will consider the case when $n = 2$, i.e. heat conduction in a annulus, where

$$\frac{1}{\alpha}\frac{\partial u}{\partial t}(r,t) = \frac{\partial^2 u}{\partial r^2}(r,t) + \frac{1}{r}\frac{\partial u}{\partial r}(r,t), \quad r_0 < r < R,\ 0 < t < T. \tag{4.2}$$

The fundamental solution of equation (4.2) is given by, see [7, p. 202],

$$F_1(r,t;\rho,\tau) \;=\; \frac{H(t-\tau)}{4\pi\alpha(t-\tau)} e^{-\frac{r^2+\rho^2}{4\alpha(t-\tau)}} I_0\left(\frac{r\rho}{2\alpha(t-\tau)}\right), \tag{4.3}$$

where H is the Heaviside function, and I_0 is the modified Bessel function of the first kind of order zero. We note that $I_0(0) = 1$ and $\lim_{x\to\infty} I_0(x) = \infty$.

The normal (radial) derivative of the fundamental solution (4.3) is

$$\frac{\partial F_1}{\partial r}(r,t;\rho,\tau) = -\frac{H(t-\tau)}{8\pi\alpha^2(t-\tau)} e^{-\frac{r^2+\rho^2}{4\alpha(t-\tau)}} \left\{ r I_0\left(\frac{r\rho}{2\alpha(t-\tau)}\right) + \rho I_1\left(\frac{r\rho}{2\alpha(t-\tau)}\right)\right\}, \tag{4.4}$$

where I_1 is the modified Bessel function of the first kind of order one.

If the model also depends on z, i.e. has azimuthal symmetry, and the medium is isotropic, then, in cylindrical coordinates, we get axisymmetric transient heat conduction governed by

$$\frac{1}{\alpha}\frac{\partial u}{\partial t}(r,z,t) = \frac{\partial^2 u}{\partial r^2}(r,z,t) + \frac{1}{r}\frac{\partial u}{\partial r}(r,z,t) + \frac{\partial^2 u}{\partial z^2}(r,z,t), \qquad (4.5)$$
$$r_0 < r < R,\ 0 < z < Z,\ 0 < t < T.$$

The fundamental solution of equation (4.5) is given by, see [108],

$$F_2(r,z,t;\rho,\xi,\tau) \;=\; \frac{H(t-\tau)}{[4\pi\alpha(t-\tau)]^{3/2}} e^{-\frac{r^2+\rho^2+(z-\xi)^2}{4\alpha(t-\tau)}} I_0\left(\frac{r\rho}{2\alpha(t-\tau)}\right). \qquad (4.6)$$

When $n = 3$, equation (4.1) represents radial heat conduction in a solid ($r_0 = 0$) or hollow sphere ($r_0 > 0$), and the fundamental solution is given by, see [7, p. 255],

$$F_3(r,t;\rho,\tau) \;=\; \frac{H(t-\tau)}{8\pi r\rho\sqrt{\pi\alpha(t-\tau)}}\left[e^{-\frac{(r-\rho)^2}{4\alpha(t-\tau)}} - e^{-\frac{(r+\rho)^2}{4\alpha(t-\tau)}})\right]. \qquad (4.7)$$

In this chapter we will prove linear independence and denseness results of linear combinations of fundamental solutions (4.3) and its derivative (4.4), and produce numerical results for the radially symmetric heat equation (4.2) in an annulus and a disk and for the axisymmetric heat equation (4.5) in a hollow cylinder.

4.2 Boundary conditions

If $r_0 = 0$ then we wish to find a solution u to equation (4.2) along with the boundary conditions given below. At $r = 0$ we require the symmetry boundary condition given by

$$\frac{\partial u}{\partial r}(0,t) = 0, \quad 0 < t < T, \qquad (4.8)$$

and a boundary condition at $r = R$ given by

$$k\frac{\partial u}{\partial r}(R,t) + h_R(t)u(R,t) = f(t), \quad 0 < t < T, \qquad (4.9)$$

which models surface convection, where k is the thermal conductivity, $h_R(t)$ is the heat transfer coefficient and $f(t)$ is usually equal to $h_R(t)u_\infty$ with u_∞ the ambient temperature, but $f(t)$ can also include a prescribed heat flux.

Finally, we impose the temperature at time $t = T$

$$u(r, T) = u_T(r), \quad 0 < r < R. \tag{4.10}$$

We wish to find a solution u to equations (4.2), (4.8), (4.9) and (4.10), and, in particular, find the temperature at time $t = 0$, given by

$$u(r, 0) = u_0(r) \text{ (to be determined)}, \quad 0 < r < R. \tag{4.11}$$

When $r_0 > 0$, we replace the symmetry boundary condition (4.8) by

$$-k\frac{\partial u}{\partial r}(r_0, t) + h_0(t)u(r_0, t) = f_0(t), \quad 0 < t < T, \tag{4.12}$$

which again models surface convection, where $h_0(t)$ is the heat transfer coefficient, and $f_0(t)$ is usually equal to $h_0(t)u_\infty$, but $f_0(t)$ can also include a prescribed heat flux. In this case we wish to find a solution u to equations (4.2), (4.12), (4.9) and (4.10), and, in particular, determine the initial temperature (4.11).

4.3 Denseness properties of linear combinations of fundamental solutions

The denseness properties for linear combinations of the fundamental solution (4.3) to the radial symmetric heat equation (4.2) can be proved using the same arguments in the proofs of Theorems 1.2.2 and 1.2.3 in Chapter 1. An additional result is required, however, and is provided in Lemma 4.3.1. To this end we place source points on a boundary external to the interval (r_0, R), which we denote by D, and let D_E denote an open interval, which contains $\overline{D}$. We let $\Gamma = \{r_0, R\}$ denote the boundary and Γ_E denote the end points of D_E, and, for $r \in \Gamma$, $\rho \in \Gamma_E$, $d(r, \rho) > 0$. The source points are given by $\{\rho_j, \tau_n\}_{n=1,2,\ldots}$, $j = 1, 2$, and form a denumerable, everywhere dense set of points in $\Gamma_E \times [-T, T]$, $(\tau_n \neq 0)$.

To approximate the solution to equations (4.2), (4.12), (4.9) and (4.10) ((4.2), (4.8), (4.9) and (4.10), when $r_0 = 0$), we construct the following infinite series

$$u_\infty(r,t) \;=\; \sum_{j=1}^{2}\sum_{n=1}^{\infty} c_n^{(j)} F_1(r,t;\rho_j,\tau_n). \tag{4.13}$$

Before proceeding, we prove a lemma which will be useful in the proof of linear independence.

Lemma 4.3.1. *If $t > \tau$ and $r = \rho + \sqrt{t-\tau}$, then $F_1(r,t;\rho,\tau)$, defined by (4.3), tends to ∞ as (r,t) tends to (ρ,τ).*

Proof. We have to show that

$$\frac{H(t-\tau)}{4\pi\alpha(t-\tau)} e^{-\frac{r^2+\rho^2}{4\alpha(t-\tau)}} I_0\left(\frac{r\rho}{2\alpha(t-\tau)}\right) \to \infty, \;\text{ as }\; (r,t)\to(\rho,\tau),$$

for fixed (ρ,τ), where r approaches ρ along the curve $r = \rho + \sqrt{t-\tau}$ and $t > \tau$.

Consider first $\rho \neq 0$, then

$$I_0\left(\frac{r\rho}{2\alpha(t-\tau)}\right) \;=\; I_0\left(\frac{(\rho+\sqrt{t-\tau})\rho}{2\alpha(t-\tau)}\right) \;\geq\; \frac{e^{\frac{(\rho+\sqrt{t-\tau})\rho}{2\alpha(t-\tau)}}}{\sqrt{(\rho+\sqrt{t-\tau})\rho\pi}}\sqrt{\alpha(t-\tau)}, \tag{4.14}$$

for $t-\tau$ sufficiently small, since the argument in I_0 tends to ∞ as $(t-\tau)\to 0$, and $I_0(x) \geq \frac{e^x}{\sqrt{2\pi x}}$ for sufficiently large x, see [39].

Now, multiplying with the factor $e^{(r^2+\rho^2)/(4\alpha(t-\tau))}$, we obtain

$$e^{-\frac{(\rho+\sqrt{t-\tau})^2+\rho^2}{4\alpha(t-\tau)}} e^{\frac{(\rho+\sqrt{t-\tau})\rho}{2\alpha(t-\tau)}} \;=\; e^{\frac{2\rho^2+2\rho\sqrt{t-\tau}-2\rho^2-2\rho\sqrt{t-\tau}-(t-\tau)}{4\alpha(t-\tau)}} \;=\; e^{-\frac{1}{4\alpha}} \;<\; \infty, \tag{4.15}$$

therefore,

$$F_1(r,t;\rho,\tau) \;\geq\; \frac{H(t-\tau)}{4\pi\alpha(t-\tau)} \frac{e^{-\frac{1}{4\alpha}}\sqrt{\alpha(t-\tau)}}{\sqrt{(\rho+\sqrt{t-\tau})\rho\pi}} \;=\; \frac{H(t-\tau)}{4\pi\sqrt{\alpha(t-\tau)}} \frac{e^{-\frac{1}{4\alpha}}}{\sqrt{(\rho+\sqrt{t-\tau})\rho\pi}},$$

and clearly then

$$\lim_{t\to\tau^+} F_1(r,t;\rho,\tau) = \infty.$$

For $\rho = 0$, the result is obviously true by setting $r = \sqrt{t-\tau}$, and noting that $I_0(0) = 1$ and $e^{-\frac{r^2}{4\alpha(t-\tau)}} = e^{-\frac{1}{4\alpha}}$. □

Lemma 4.3.2. *If $t > \tau$ and $r = \rho + \sqrt{t-\tau}$, then $\left|\frac{\partial F_1}{\partial r}(r,t;\rho,\tau)\right|$, with the normal (radial) derivative of the fundamental solution defined by (4.4), tends to ∞ as (r,t) tends to (ρ,τ).*

Proof. Following a similar argument to the proof of Lemma 4.3.1, we have to show that

$$\left|\frac{H(t-\tau)}{8\pi\alpha^2(t-\tau)}e^{-\frac{r^2+\rho^2}{4\alpha(t-\tau)}}\left\{rI_0\left(\frac{r\rho}{2\alpha(t-\tau)}\right)+\rho I_1\left(\frac{r\rho}{2\alpha(t-\tau)}\right)\right\}\right| \to \infty, \text{ as } (r,t)\to(\rho,\tau),$$

for fixed (ρ,τ), where r approaches ρ along the curve $r = \rho + \sqrt{t-\tau}$ and $t > \tau$.

Consider $\rho \neq 0$, then

$$I_0\left(\frac{r\rho}{2\alpha(t-\tau)}\right) = I_0\left(\frac{(\rho+\sqrt{t-\tau})\rho}{2\alpha(t-\tau)}\right) \geq \frac{e^{\frac{(\rho+\sqrt{t-\tau})\rho}{2\alpha(t-\tau)}}}{\sqrt{(\rho+\sqrt{t-\tau})\rho\pi}}\sqrt{\alpha(t-\tau)}, \tag{4.16}$$

and

$$I_1\left(\frac{r\rho}{2\alpha(t-\tau)}\right) = I_1\left(\frac{(\rho+\sqrt{t-\tau})\rho}{2\alpha(t-\tau)}\right) \geq \frac{e^{\frac{(\rho+\sqrt{t-\tau})\rho}{2\alpha(t-\tau)}}}{\sqrt{(\rho+\sqrt{t-\tau})\rho\pi}}\sqrt{\alpha(t-\tau)}e^{-\frac{\alpha(t-\tau)}{(\rho+\sqrt{t-\tau})\rho}} \tag{4.17}$$

for $t-\tau$ sufficiently small, since the arguments in I_0 and I_1 tend to ∞ as $(t-\tau)\to 0$, and $I_n(x) \geq \frac{e^x}{\sqrt{2\pi x}}e^{-n^2/2x}$ for sufficiently large x, see [39]. Now for $t-\tau$ sufficiently small $(\rho+\sqrt{t-\tau})\rho > 0$ and

$$e^{-\frac{\alpha(t-\tau)}{(\rho+\sqrt{t-\tau})\rho}} \to 1 \quad \text{monotonically as} \quad (r,t)\to(\rho,\tau),$$

therefore, there exists $c \in (0,1)$ such that

$$c \leq e^{-\frac{\alpha(t-\tau)}{(\rho+\sqrt{t-\tau})\rho}} \quad \text{as} \quad (r,t)\to(\rho,\tau).$$

Hence, for $t-\tau$ sufficiently small from equation (4.17) we see that

$$I_1\left(\frac{r\rho}{2\alpha(t-\tau)}\right) \geq c\frac{e^{\frac{(\rho+\sqrt{t-\tau})\rho}{2\alpha(t-\tau)}}}{\sqrt{(\rho+\sqrt{t-\tau})\rho\pi}}\sqrt{\alpha(t-\tau)}$$

and using equation (4.15) we find that

$$\begin{aligned}
|F_1(r,t;\rho,\tau)| &\geq \left| \frac{H(t-\tau)}{8\pi\alpha^2(t-\tau)} e^{-\frac{r^2+\rho^2}{4\alpha(t-\tau)}} (r+c\rho) \sqrt{\frac{\alpha(t-\tau)}{(\rho+\sqrt{t-\tau})\rho\pi}} e^{\frac{(\rho+\sqrt{t-\tau})\rho}{2\alpha(t-\tau)}} \right| \\
&\geq \left| \frac{H(t-\tau)}{8\pi\alpha^2(t-\tau)} \sqrt{\frac{\alpha(t-\tau)}{(\rho+\sqrt{t-\tau})\rho\pi}} (r+c\rho) e^{-\frac{1}{4\alpha}} \right| \\
&= \left| \frac{H(t-\tau)}{8\pi\alpha^{\frac{3}{2}}\sqrt{t-\tau}} \frac{1}{\sqrt{(\rho+\sqrt{t-\tau})\rho\pi}} (r+c\rho) e^{-\frac{1}{4\alpha}} \right|
\end{aligned}$$

and clearly along the curve $r = \rho + \sqrt{t-\tau}$

$$\lim_{t\to\tau^+} \left| \frac{\partial F_1}{\partial r}(r,t;\rho,\tau) \right| = \infty.$$

For $\rho = 0$, we have

$$\frac{\partial F_1}{\partial r}(r,t;\rho,\tau) = -\frac{H(t-\tau)}{8\pi\alpha^2(t-\tau)} e^{-\frac{r^2}{4\alpha(t-\tau)}} r,$$

which, by setting $r = \sqrt{t-\tau}$, and noting that $I_0(0) = 1$ and $e^{-\frac{r^2}{4\alpha(t-\tau)}} = e^{-\frac{1}{4\alpha}}$, and $\frac{r}{t-\tau} = \frac{1}{\sqrt{t-\tau}} \to \infty$ as $t \to \tau^+$. □

4.3.1 Denseness on the lateral surface

We now state denseness theorems for Dirichlet and Neumann boundary conditions, as well as denseness for any fixed time point.

Theorem 4.3.3. *Let $F_1(r,t;\rho,\tau)$ be defined by (4.3). The set of functions $\{F_1(r,t;\rho_j,\tau_n)\}_{n=1}^{\infty}$, $j = 1,2$, restricted on $\Gamma\times(-T,T)$ forms a linearly independent and dense set in $L^2(\Gamma\times(-T,T))$.*

Theorem 4.3.4. *Let $F_1(r,t;\rho,\tau)$ be defined by (4.3). The set of functions $\{F_1(r,T_0;\rho_j,\tau_n)\}_{n=1}^{\infty}$, $j = 1,2$, with $\tau_n < T_0$, where $T_0 \geq 0$, forms a linearly independent and dense set in $L^2(D\times\{T_0\})$.*

To prove Theorems 4.3.3 and 4.3.4 use the same arguments that were used in the proofs of Theorems 1.2.2 and Theorem 2.3.1, respectively. The only difference is that in the parts proving linear independence when we use the 'ratio' argument, we let (r,t) approach the point $(\rho_{j_0},\tau_{n_0}) \in \Gamma_E \times (-T,T)$, (with $t > \tau$), along the curve $r = \rho + \sqrt{t-\tau}$ and apply Lemma 4.3.1 to show that the summand can be made as large as we wish. The arguments involving analytic continuation, single-layer potentials, jump conditions, the corollary of Theorem 3 in [66], and

Greens identities all still hold in the radial case.

We now state a denseness theorem (and the main details of the proof) when we have Neumann boundary conditions.

Theorem 4.3.5. *Let $F_{1r}(r,t;\rho,\tau)$ be defined by (4.4). The set of functions $\{F_{1r}(r,t;\rho,\tau)\}_{n=1}^{\infty}$, $j=1,2$, restricted on $\Gamma\times(-T,T)$ forms a linearly independent and dense set in $L^2(\Gamma\times(-T,T))$.*

Proof. The proof uses similar arguments to those presented in the proof of Theorem 1.2.2.

Linear independence: For a contradiction we assume there exist positive integers $N, n_0 \in \{1,\dots,N\}$, $j_0 \in \{1,2\}$, with $c_{n_0}^{(j_0)} \neq 0$, and

$$\sum_{j=1}^{2}\sum_{n=1}^{N} c_n^{(j)} F_{1r}(r,t;\rho_j,\tau_n) \;=\; 0, \quad (r,t)\in\Gamma\times(-T,T), \tag{4.18}$$

and define

$$U(r,t) \;=\; \sum_{j=1}^{2}\sum_{n=1}^{N} c_n^{(j)} F_{1r}(r,t;\rho_j,\tau_n), \quad (r,t)\in D\times(-T,T). \tag{4.19}$$

Since U satisfies the heat equation, assumption (4.18), $F_{1r}(r,-T;\rho_j,\tau_n)=0$ we have

$$\frac{\partial^2 U}{\partial r^2}(r,t) + \frac{1}{r}\frac{\partial U}{\partial r}(r,t) \;=\; \frac{\partial U}{\partial t}(r,t), \quad \text{in } D\times(-T,T), \tag{4.20}$$

$$U(r,t) \;=\; 0, \quad (r,t)\in\Gamma\times(-T,T), \tag{4.21}$$

$$U(r,-T) \;=\; 0. \tag{4.22}$$

Therefore, by uniqueness, the only solution to equations (4.20)–(4.22) is $U(r,t) = 0$ for $(r,t)\in D\times(-T,T)$, and since U is analytic, $U\equiv 0$ for $(r,t)\in D_E\times(-T,T)$ by unique continuation, see [95].

We then let (r,t) approach the point $(\rho_{j_0},\tau_{n_0})\in\Gamma_E\times(-T,T)$, such that $t>\tau$ and $r=\rho+\sqrt{t-\tau}$, and by applying Lemma 4.3.2, the summand $c_{n_0}^{(j_0)}F_{1r}(r,t;\rho_{j_0},\tau_{n_0})$ in (4.18) may be made as large as we wish, while the other terms in the series (4.18) remain bounded; giving a contradiction, we have linear independence for the set of functions $\{F_{1r}(r,t;\rho_j,\tau_n)\}_{n=1}^{\infty}$, $j=1,2$, in $L^2(\Gamma\times(-T,T))$.

Denseness: Next we wish to prove that the sequence $\{F_{1r}(r,t;\rho_j,\tau_n)\}_{n=1}^{\infty}$, $j=1,2$, is a dense set in $L^2(\Gamma\times(-T,T))$. Therefore, for a contradiction, we assume it is not a dense set and there

exists a continuous function $f(r,t)$ in $L^2(\Gamma \times (-T,T))$, such that

$$\int_{-T}^{T}\int_{\Gamma} F_{1r}(r,t;\rho_j,\tau_n) f(r,t)\, dr\, dt \;=\; 0, \quad j=1,2,\; n=1,2,\ldots \tag{4.23}$$

We have to show that $f(r,t)\equiv 0$ in (4.23) to prove that $\{F(r,t;\rho_j,\tau_n)\}$ is a dense set. Due to the Heaviside function, equation (4.23) is equivalent to

$$\int_{\tau_n}^{T}\int_{\Gamma} F_{1r}(r,t;\rho_j,\tau_n) f(r,t)\, dr\, dt \;=\; 0, \quad j=1,2,\; ,m=1,2,\ldots, \tag{4.24}$$

and, therefore, we introduce an equivalent form of the classical double-layer heat potential given by

$$K(\rho,\tau) \;=\; \int_{\tau}^{T}\int_{\Gamma} F_{1r}(r,t;\rho,\tau) f(r,t)\, dr\, dt. \tag{4.25}$$

Here we apply properties of the double-layer heat potential, which can be found in Chapter 3 of [76]. By continuity and (4.24) we find that $K(\rho,\tau)=0$ for $(\rho,\tau)\in \Gamma_E\times(-T,T)$, thus we have $K=0$ in the exterior of $\overline{D}\times(-T,T)$, also, since K is continuous across $\Gamma\times[-T,T]$ we have $K(\rho,\tau)=0$ on $\Gamma\times[-T,T]$. Therefore, $K=0$ in $\overline{D}\times(-T,T)$ since K satisfies the heat equation in $D\times(-T,T)$. We note $K=0$ in $\overline{D}\times(-T,T)$ when $r_0=0$, since K is continuous for all points $r>0$. Finally, applying the jump relation for the double-layer heat potential,

$$\frac{1}{2} f(r,t) \pm \frac{\partial K}{\partial r}(r,t) \;=\; 0, \quad (r,t)\in \Gamma\times(-T,T), \tag{4.26}$$

therefore, $f\equiv 0$ and $\{F_{1r}(r,t;\rho_j,\tau_n)\}_{m=1}^{\infty}$, $j=1,2$, is a dense set in $L^2(\Gamma\times(-T,T))$. □

4.4 The MFS for the radially symmetric BHCP

We generate an approximation to the radial heat equation (4.2) as a linear combination of fundamental solutions given by

$$u(r,t)\approx u_N(r,t)=\sum_{n=1}^{2N_1} c_n^{(0)} F_1(r,t;r_0-h_1,\tau_n)+\sum_{n=1}^{2N_1} c_n^{(1)} F_1(r,t;R+h_2,\tau_n), \quad r_0\le r\le R,\; 0\le t\le T, \tag{4.27}$$

where $N = 4N_1$, $h_1, h_2 > 0$, and the source point locations in time are given by

$$\tau_n = -T + \frac{T(2kn-1)}{2M_1} \quad \text{for } n = 1, \ldots, 2N_1, \text{ where } M_1 = kN_1,$$

for $k > 0$, and the $4N_1$ constant coefficients $c_n^{(0)}$ and $c_n^{(1)}$ for $n = 1, \ldots, 2N_1$ are determined by imposing the final and boundary conditions (4.9), (4.10) and (4.8) (in the case $r_0 = 0$), or (4.12) (in the case $r_0 > 0$). Let

$$t_i = iT/M_1 \quad \text{for } i = 1, \ldots, M_1,$$

and

$$r_l = r_0 + (R - r_0)l/(L+1), \quad \text{for } l = 1, \ldots, L,$$

then we collocate at the points described above to give

$$\frac{\partial u_N}{\partial r}(r_0, t_i) = 0, \quad i = 1, \ldots, M_1 \text{ (in case } r_0 = 0), \tag{4.28}$$

or

$$-k\frac{\partial u_N}{\partial r}(r_0, t_i) + h_0(t_i)u_N(r_0, t_i) = f_0(t_i), \quad i = 1, \ldots, M_1 \text{ (in case } r_0 > 0), \tag{4.29}$$

and

$$k\frac{\partial u_N}{\partial r}(R, t_i) + h(t_i)u_N(R, t_i) = f(t_i), \quad i = 1, \ldots, M_1, \tag{4.30}$$

$$u_N(r_l, T) = u_T(r_l), \quad l = 1, \ldots, L. \tag{4.31}$$

For a graphical representation of the domain and placement of collocation and source points given above, see Figure 4.1.

Expressions (4.30), (4.31) and (4.28) or (4.29) form a system,

$$A\mathbf{c} = \mathbf{g}, \tag{4.32}$$

of $M = (2M_1 + L)$ linear equations in $N = 4N_1$ unknowns. In order to obtain a unique solution we require $M \geq N$. Further, since the MFS system of equations (4.32) is ill-conditioned and the BHCP is ill-posed, we use the Tikhonov regularization method and solve (0.45) using L_0, see Section 0.5 of the Introduction.

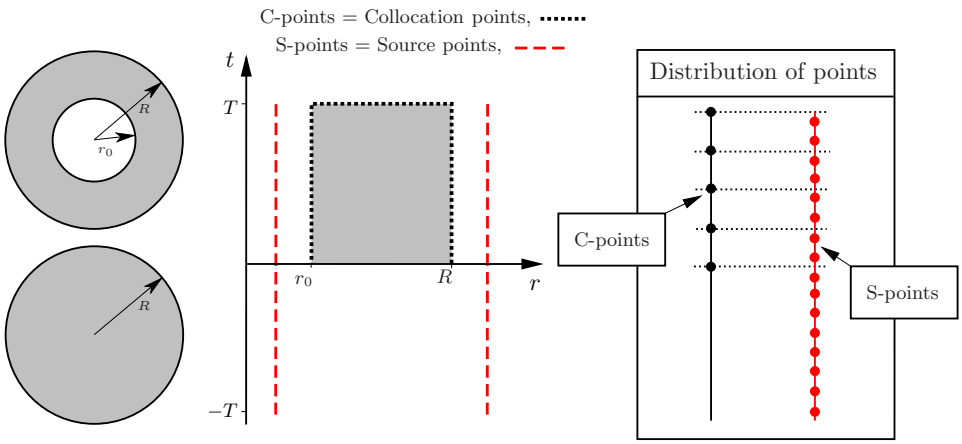

Figure 4.1: Representations of the cylindrical problems being considered and the one-dimensional domain in time together with location of collocation and source points.

4.4.1 The two-dimensional axisymmetric case

We can extend the MFS to heat conduction processes which depend also on z, see Figure 4.2 for a graphical representation of the domains and possible placement of source points in this case.

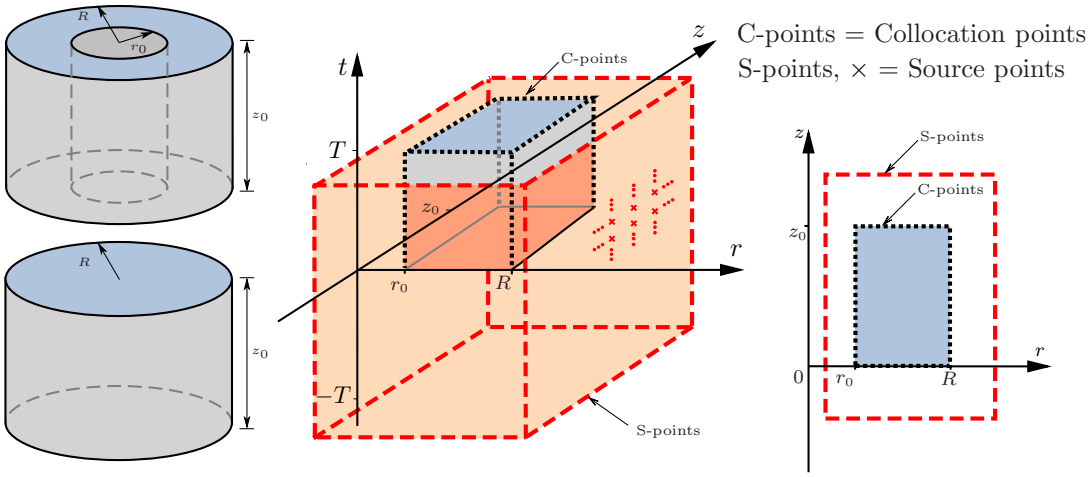

Figure 4.2: Representation of the three-dimensional domain in time, displaying a possible distribution of these points on the boundaries.

The denseness results produced in this chapter, Theorems 4.3.3, 4.3.4 and 4.3.5, can be extended to the axisymmetric transient heat conduction equation (4.5).

4.5 Numerical results

4.5.1 Example 1 (annulus)

We consider the radial heat conduction (4.2) in a hollow cylinder $0.5 = r_0 < r < R = 1$, with the analytic solution given by

$$u(r,t) = r^2 + 4t, \quad 0.5 \leq r \leq 1,\, 0 \leq t \leq 1. \tag{4.33}$$

Clearly, the function (4.33) satisfies equation (4.2) with $\alpha = 1$.

The equations (4.9) and (4.12) with $h_R = h_0 = 0$, $f(t) = 2$, $f_0(t) = -1$ when $k = R = T = 1$, $r_0 = 0.5$, satisfied by (4.33), are given by

$$\frac{\partial u}{\partial r}(1,t) = 2, \quad -\frac{\partial u}{\partial r}(0.5,t) = -1, \quad 0 < t < 1. \tag{4.34}$$

Also, equation (4.10) when $T = 1$ and $u_T(r) = r^2 + 4$ is given by

$$u(r,1) = r^2 + 4, \quad 0 < r < 1. \tag{4.35}$$

The initial temperature (4.11) to be retrieved is given by

$$u(r,0) = u_0(r) = r^2, \quad 0 < r < 1. \tag{4.36}$$

In Figure 4.3(a) we plot the MFS approximation for varying values of h_1 and h_2, with the approximation improving for larger values of h. In Figure 4.3(b) the best, average and least accurate MFS approximations after ten sets of noisy data are plotted, with results accurate and stable. We note that in both examples the L-curve criterion has been used to determine a reasonable choice for the Tikhonov regularization parameter $\lambda > 0$, and the choice of collocation and source points have generated a square system.

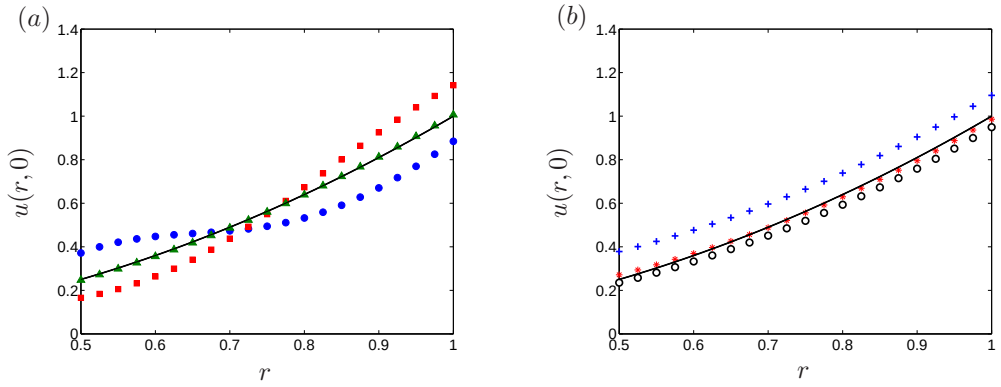

Figure 4.3: **(a)** The exact solution $u(r,0)$ (——) and the MFS approximation for $h_1 = h_2 = 1$ with $\lambda = 10^{-11}$ (●), $h_1 = h_2 = 1.5$ with $\lambda = 10^{-14}$ (■), and $h_1 = h_2 = 2$ with $\lambda = 10^{-15}$ (▲). **(b)** The exact solution $u(r,0)$ (——) and the best (∗), average (○), and least (+) accurate MFS approximations from ten different sets of noisy data with noise level $\delta = 5\%$ with $\lambda = 10^{-5}$ and $h_1 = h_2 = 3$. Both plots obtained with $r_0 = 0.5$, $R = 1$, $T = 1$, $L = 40$, $M_1 = 20$, $N_1 = 20$ (i.e. 80 collocation points and 80 source points), for Example 1.

4.5.2 Example 2 (disk)

We consider the radial heat conduction in a solid cylinder $0 = r_0 < r < R = 1$, with the analytic solution given by

$$u(r,t) = e^{-t} J_0(r), \quad 0 \le r \le 1, \ 0 \le t \le 1. \tag{4.37}$$

A similar example has been considered in [8] where highly unstable numerical solutions were reported. Clearly, the function (4.37) satisfies equation (4.2) with $\alpha = 1$ and the symmetry condition (4.8) since $J_1(0) = 0$. We also replace equation (4.9) with the Dirichlet boundary condition

$$u(1,t) = e^{-t} J_0(1), \quad 0 < t < 1. \tag{4.38}$$

Note that in this case, equation (4.30) should be replaced by

$$u_N(1,t_i) = e^{-t_i} J_0(1), \quad i = 1, \ldots, M_1. \tag{4.39}$$

The upper-base final temperature condition (4.10) is satisfied by $u_T(r) = e^{-T} J_0(r)$, i.e. for $T = 1$,

$$u(r,1) = e^{-1} J_0(r), \quad 0 < r < 1. \tag{4.40}$$

In Figure 4.4(a) we plot the MFS approximation for varying noise levels, with the approxi-

mation being reasonably accurate when $\delta = 0$, however, results are unstable for $\delta = 5\%$, with only very specific values of the Tikhonov regularization parameter ensuring accurate results. We note that the corner of the L-curve occurs at $\lambda = 10^{-6}$, which in this case does not generate the best results. We try to improve the results by considering a hollow cylinder and impose the Neumann boundary (4.12), instead of the symmetry condition (4.8), at $r_0 = 0.1$, given by

$$u_N(0.1, t_i) = e^{-t} J_1(r), \quad i = 1, \ldots, M_1. \tag{4.41}$$

In Figure 4.4(b) we note that non-zero Neumann data at r_0, for all noise levels, produces more accurate and stable results compared to the plots produced in Figure 4.4(a).

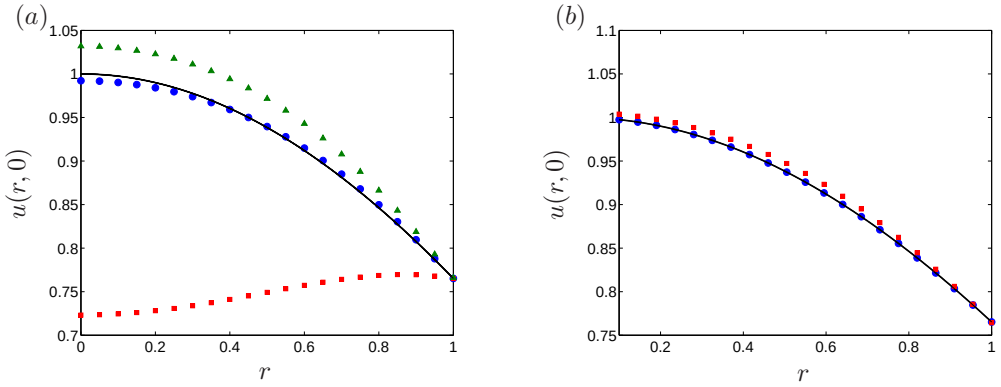

Figure 4.4: The exact solution $u(r, 0)$ (——) and the MFS approximation for **(a)** $r_0 = 0$ and $\delta = 0$ with $\lambda = 10^{-16}$ (●), $\delta = 5\%$ with $\lambda = 10^{-8}$ (■), and $\delta = 5\%$ with $\lambda = 10^{-9}$ (▲), and **(b)** $r_0 = 0.1$ and $\delta = 0$ with $\lambda = 10^{-16}$ (●), and $\delta = 5\%$ with $\lambda = 10^{-8}$ (■). Both plots obtained with $h_1 = h_2 = 3$, $R = 1$, $T = 1$, $L = 40$, $M_1 = 20$, $N_1 = 20$ (i.e. 80 collocation points and 80 source points), for Example 2.

4.5.3 Example 3 (axisymmetric example)

In this final example we investigate the axisymmetric backward heat conduction problem, with governing equation (4.5), in a hollow cylinder with $0.5 = r_0 < r < R = 1$ and $0 < z < 1$, and fundamental solution given by (4.6). See Figure 4.2 for a visual representation of the domain and placement of collocation and source points. The exact solution, which was considered in

[4], is given by

$$u(r,z,t) = -1 + z + (r^2+z^2)^{-1/2} \sin\left(\frac{\pi}{4}\sqrt{r^2+z^2}\right) e^{-\frac{\pi^2 t}{16}}, \tag{4.42}$$
$$0.5 \le r \le 1,\ 0 \le z \le 1,\ 0 \le t \le 1,$$

and using this we generate the final data and Dirichlet boundary conditions.

In Figures 4.4(a) and 4.4(b) we plot the exact solution and absolute error, respectively, at time $t = 0$. Results are accurate and the regularization parameter was determined using the L-curve criterion. Plots were also produced for varying numbers and positions of collocation and source points, as well as applying Neumann and mixed boundary conditions, with accurate results produced in all cases after an appropriate choice of parameters. As in Chapter 2, we find that the contribution from the final data is small compared to the boundary data, especially for larger T.

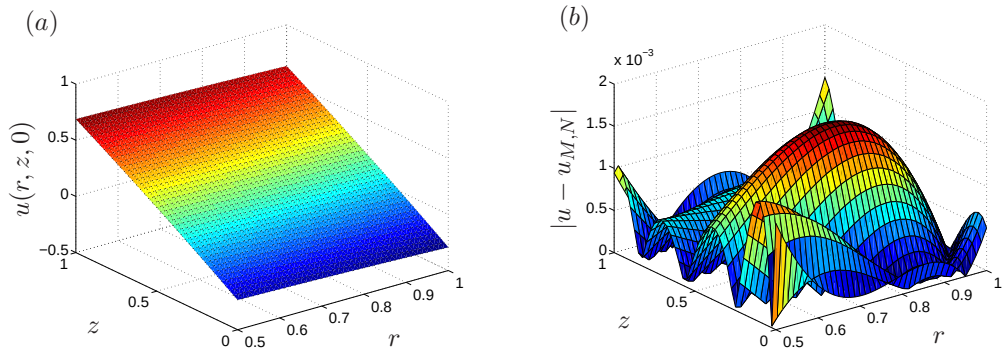

Figure 4.5: (a) The exact solution $u(r,z,0)$ and (b) the absolute error $|u(r,z,0) - u_{M,N}(r,z,0)|$ (using 400 collocation points and 300 source points), with source points placed on the boundary of the domain $(-1.5,3) \times (-2,3)$, final time $T = 1$, and noise level $\delta = 5\%$ applied to the final data, for Example 3.

4.6 Summary of Chapter 4

In this chapter we have investigated the radially symmetric backward heat conduction problem. Governing equations were given for cylinders, spheres (both hollow and solid), and three-dimensional cylinders, when the model also depends on z. Boundary conditions were stated for hollow and solid cylinders, and denseness results were stated and (where necessary) proved.

Two examples were investigated for radial heat conduction and good results were obtained, however, for Example 1 we found that results improved as h increased, which can be seen in Figure 4.3(a). To be more precise, in Figure 4.3(a) the mean absolute error was 0.0971 for $h = 1$, 0.0806 for $h = 1.5$, 0.0031 for $h = 2$, 4.0914×10^{-4} for $h = 2$, and $h = 4$ for 3.2244×10^{-4}. However, for $h \geq 4$ the accuracy deteriorated and MATLAB produced a 'matrix is singular' warning for $h = 7$. In Example 2, results were not good for a disk when $r_0 = 0$, and the best results were not produced with the Tikhonov regularization parameter corresponding to the point of maximum Gaussian curvature. Results improving when we used a small value of $r_0 > 0$, but further investigation will need to be performed for the case when $r_0 = 0$. Lastly, in Example 3, we considered the axisymmetric backward heat conduction problem, and accurate results were obtained for both Dirichlet and Neumann boundary conditions, for suitably chosen parameters h_1, h_2 and λ.

In the next chapter we investigate the MFS applied to the radially symmetric inverse heat conduction problem (IHCP), where data on an inner fixed boundary is determined from Cauchy data given on an outer boundary.

CHAPTER 5

THE RADIALLY SYMMETRIC INVERSE HEAT CONDUCTION PROBLEM

5.1 Introduction

In this chapter, we consider the radially symmetric inverse heat conduction problem (IHCP), which requires finding the temperature at a fixed inner boundary when Cauchy data is given at a fixed outer boundary [78]. The radially symmetric IHCP has at most one solution, but, like most of the work in this thesis, is ill-posed and difficult to solve due to instability with respect to small errors in the measured input data. Unlike the Cartesian counterpart, the radially symmetric IHCP has had rather less treatment, with two recent papers investigating regularization methods for this problem including [29] and [109]. However, the method presented in [109] is rather involved and the results obtained appear relatively inaccurate with approximations for smooth data being rather oscillatory. We shall show that the MFS can be used for the radially symmetric IHCP to obtain stable reconstructions with accuracy comparable to [109] or better. Moreover, the proposed method is flexible and can be readily adjusted to incorporate other data, like initial or final time data, and also higher dimensional solution domains.

5.2 Mathematical formulation

In the radially symmetric IHCP we wish to determine data along an inner boundary r_0, where Cauchy data (Dirichlet and Neumann) is given on an outer boundary $r = R$. The one-dimensional radially symmetric heat equation in a cylinder is given by (4.2), and the Cauchy boundary conditions are given by

$$u(R,t) = f(t), \quad 0 < t \leq T, \tag{5.1}$$

$$\frac{\partial u}{\partial r}(R,t) = g(t), \quad 0 < t \leq T. \tag{5.2}$$

Note that no initial condition at $t = 0$ needs to be imposed to ensure a unique solution. The problem (4.2), (5.1) and (5.2) is ill-posed, and as with most heat conduction problems involving Cauchy data is difficult to solve numerically and regularization methods are required to obtain stable results.

5.3 The MFS for the radially symmetric IHCP

We approximate the solution of the radially symmetric inverse heat conduction problem (4.2), (5.1) and (5.2) using

$$u_N(r,t) = \sum_{n=1}^{2N} c_n^{(0)} F_1(r,t;r_0-h_1,\tau_n) + \sum_{n=1}^{2N} c_n^{(1)} F_1(r,t;R+h_2,\tau_n), \quad r_0 \leq r \leq R,\ 0 \leq t \leq T, \tag{5.3}$$

where the fundamental solution F_1 is given by (4.3) and the normal (radial) derivative of the fundamental solution is given by (4.4).

As in the previous chapter, the source points will be placed in space at $r_0 - h_1$ and $R + h_2$, and along the time points

$$\tau_n = -T + \frac{T(2kn-1)}{2M} \quad \text{for } n = 1, \ldots, 2N, \text{ where } M = kN,$$

where k is a constant. We set

$$t_i = \frac{iT}{M}, \quad i = 1, \ldots, M,$$

and collocate the boundary conditions (5.1) and (5.2) as

$$u(R,t_i) = f(t_i), \quad i = 1, \ldots, M, \tag{5.4}$$

$$\frac{\partial u}{\partial r}(R,t_i) = g(t_i), \quad i = 1, \ldots, M. \tag{5.5}$$

For a representation of the domain and placement of collocation and source points, see Figure 5.1.

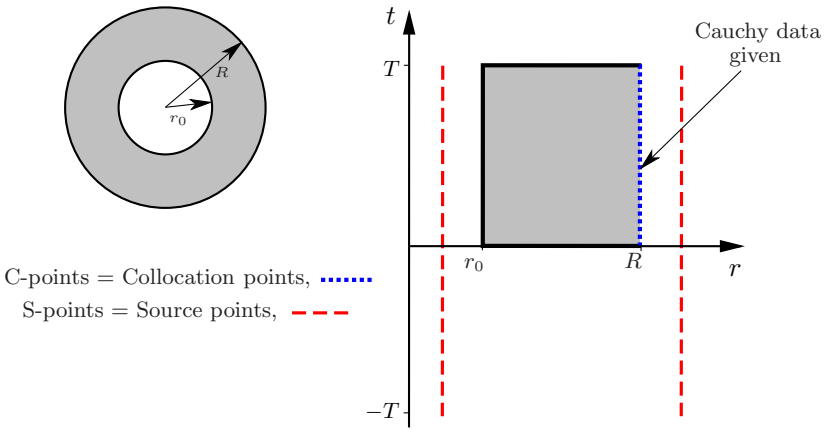

Figure 5.1: Representation of the problem considered and the one-dimensional domain in time.

Expressions (5.4) and (5.5) form a linear system of the form

$$A\mathbf{c} = \mathbf{g}, \tag{5.6}$$

and Tikhonov regularization ((0.45) using L_0) and the L-curve criterion are used again, see Section 0.5 of the Introduction.

5.4 Numerical results

In this section we consider two examples of radially symmetric inverse heat conduction problems. Both examples are taken from, and will be compared to [109], which considered a quasi-reversibility regularization method with results appearing a bit unstable.

5.4.1 Example 1

In this first example we solve the problem (4.2), (5.1) and (5.2), with the Cauchy boundary data (5.1) and (5.2) given by

$$u(1,t) = f(t) = \frac{1}{4\pi t}\exp\left(-\frac{1}{4t}\right), \quad 0 < t \leq 1 = T, \tag{5.7}$$

$$\frac{\partial u}{\partial r}(1,t) = g(t) = -\frac{1}{8\pi t^2}\exp\left(-\frac{1}{4t}\right), \quad 0 < t \leq 1 = T. \tag{5.8}$$

The analytic solution to (4.2), (5.7) and (5.8) is given by

$$\begin{aligned} u(r,t) =& \frac{1}{4\pi t}\exp\left(-\frac{r^2}{4t}\right), \\ & r_0 = 0.5 < r < 1 = R,\ 0 < t \leq 1. \end{aligned} \tag{5.9}$$

In [109], noise was uniformly distributed in the following way

$$\begin{aligned} f^{\delta}(t_i) &= f(t_i) + \delta_1(2 \times \texttt{rand}(\text{size}(f(t_i))) - 1), \\ g^{\delta}(t_i) &= g(t_i) + \delta_2(2 \times \texttt{rand}(\text{size}(g(t_i))) - 1), \quad i = 1, \ldots, M, \end{aligned} \tag{5.10}$$

where δ_1 and δ_2 are noise levels and `rand` is a uniformly distributed pseudo-random number generated in MATLAB on the interval $(0, 1)$. For all of the results we apply the noise given in (5.10), however, as it happens, for Example 1 the noise used will perturb the boundary data too much, whilst for Example 2 too little. Therefore, for selected figures, we apply noise in the following way

$$u^{\delta_1}(1,t) = f^{\delta_1}(t) = f(t) + N(0, \sigma_1^2), \tag{5.11}$$

where $N(0, \sigma^2)$ represents the normal distribution with mean zero and standard deviation

$$\sigma_1 = \delta_1 \times \max_{t\in[0,1]} |f(t)|, \tag{5.12}$$

and

$$\frac{\partial u^{\delta_2}}{\partial r}(1,t) = g^{\delta_2}(t) = g(t) + N(0, \sigma_2^2), \tag{5.13}$$

where

$$\sigma_2 = \delta_2 \times \max_{t\in[0,1]} |g(t)|. \tag{5.14}$$

To compare results we use the root mean square error (RMSE) and the relative root mean square error (RRMSE), which were defined in (3.11) and (3.12), on page 81.

In Figures 5.2(a) and 5.2(b), we have plotted the exact solution and exact derivative (of (5.9)) for every $(r, t) \in [0.5, 1] \times [0, 1]$. We can see from these figures that for small times there is a rapid change in the data, which might prove difficult to recover.

In Figures 5.3(a) and 5.3(b) the MFS approximation of the exact solution u in (5.9) (shown in Figure 5.2(a)), when noise has not been applied to the boundary data, has been generated

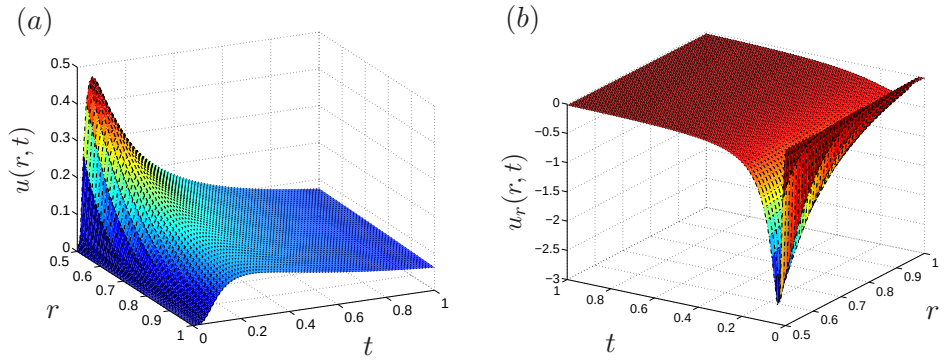

Figure 5.2: (a) The exact solution $u(r,t)$ and (b) the exact derivative $u_r(r,t)$ for all $(r,t) \in [0.5,1] \times [0,1]$, for Example 1.

for different values of $M = 2N$ and N. We find that a higher level of accuracy is attainable for larger values of these parameters, which can be seen from the error plot in Figure 5.3(c).

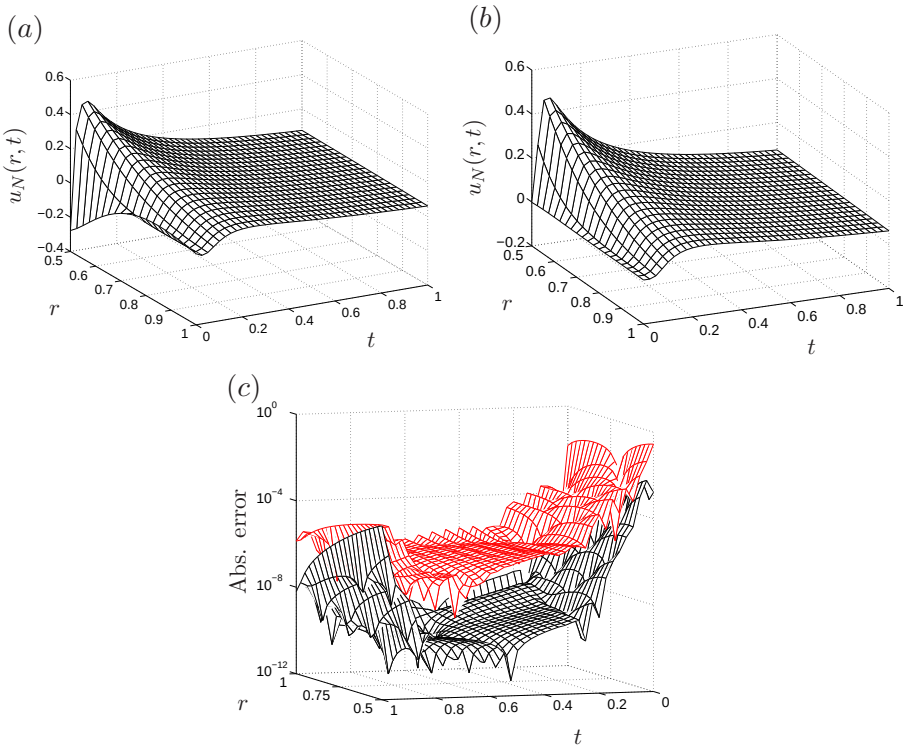

Figure 5.3: (a) The MFS approximation $u_N(r,t)$ with $M = 40$ and $N = 20$, (b) $u_N(r,t)$ with $M = 160$ and $N = 80$, and (c) the absolute error of the two approximations generated in (a) (——) and (b) (——), obtained with $h_1 = h_2 = h = 0.5$, $\lambda = 10^{-8}$, $\delta_1 = \delta_2 = 0$, for Example 1.

In Figures 5.4(a)–5.4(c) noise has been added to the Dirichlet and Neumann boundary data. We note that in Figure 5.4(a) the results appear slightly unstable, however, for the amount of noise added, and compared to the results published in [109] they are relatively accurate. We also note that the value of the Tikhonov regularization parameter we have chosen using the

L-curve is rather large with $\lambda = 10^{-1}$. Also, for Figure 5.4(c) we have the RMSE = 0.0381 and RRMSE = 0.1615, compared with 0.0836 and 0.3983, respectively, in [109].

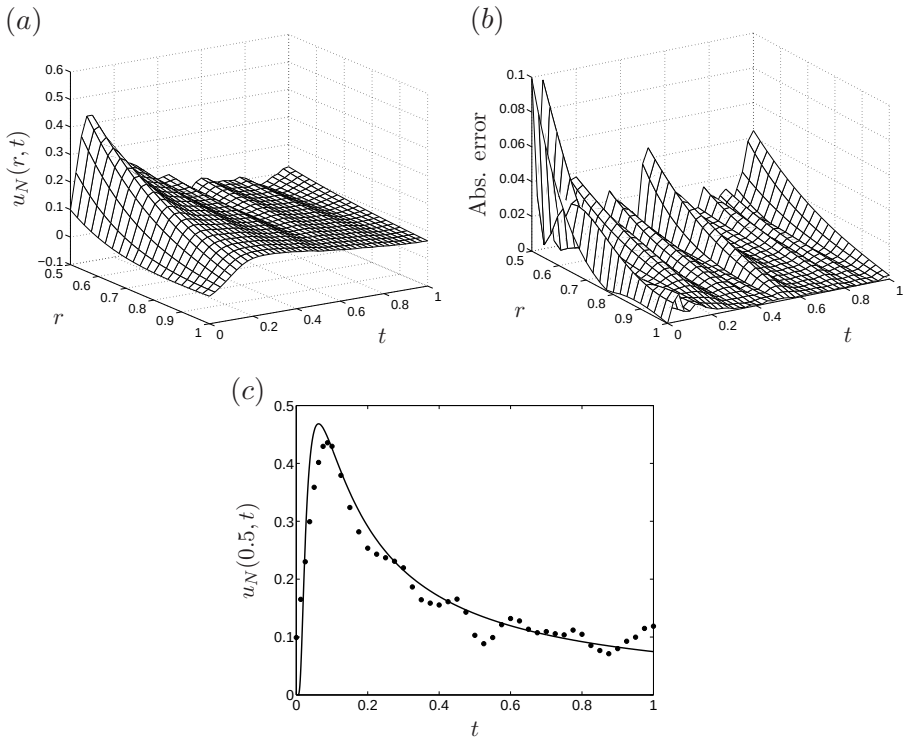

Figure 5.4: (a) The MFS approximation $u_N(r,t)$, (b) the absolute error, and (c) the exact solution $u(0.5,t)$ (——) and the MFS approximation $u_N(0.5,t)$ (•), obtained with $h_1 = h_2 = h = 0.5$, $M = 80$, $N = 40$, $\lambda = 10^{-1}$, $\delta_1 = 0.1\%$, $\delta_2 = 10\%$, for Example 1.

Figure 5.5(a) contains a plot of the MFS approximation, without noise, of the normal derivative, and Figure 5.5(b) is the absolute error, and we again see that results are very accurate.

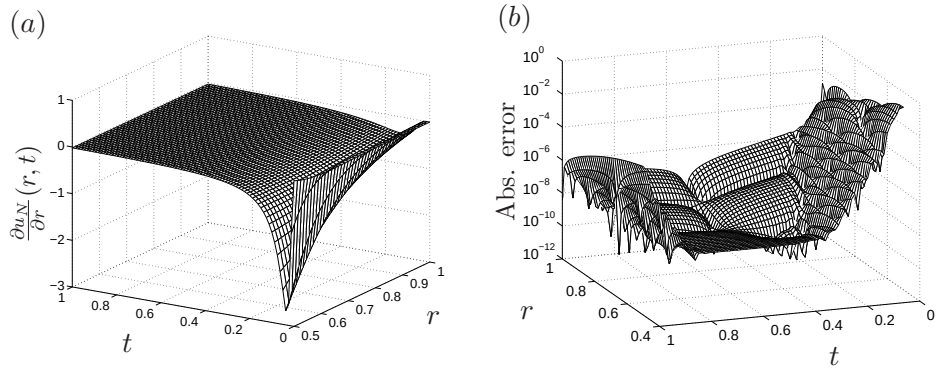

Figure 5.5: (a) The MFS approximation of the derivative $\frac{\partial u_N}{\partial r}(r,t)$ and (b) the absolute error obtained with $M = 160$, $N = 80$, $h_1 = h_2 = h = 0.5$, $\lambda = 10^{-8}$, $\delta_1 = \delta_2 = 0$, for Example 1.

In Figures 5.6(a) and 5.6(b) the MFS approximation and absolute error have been generated for the normal derivative of (5.9) (shown in Figure 5.2(b)) when noise has been added. Figure

5.6(c) is a plot of the exact normal derivative and the MFS approximation for $r = 0.5$, and again we find that the obtained RMSE = 0.2691 and RRMSE = 0.3228 are better than those given in [109].

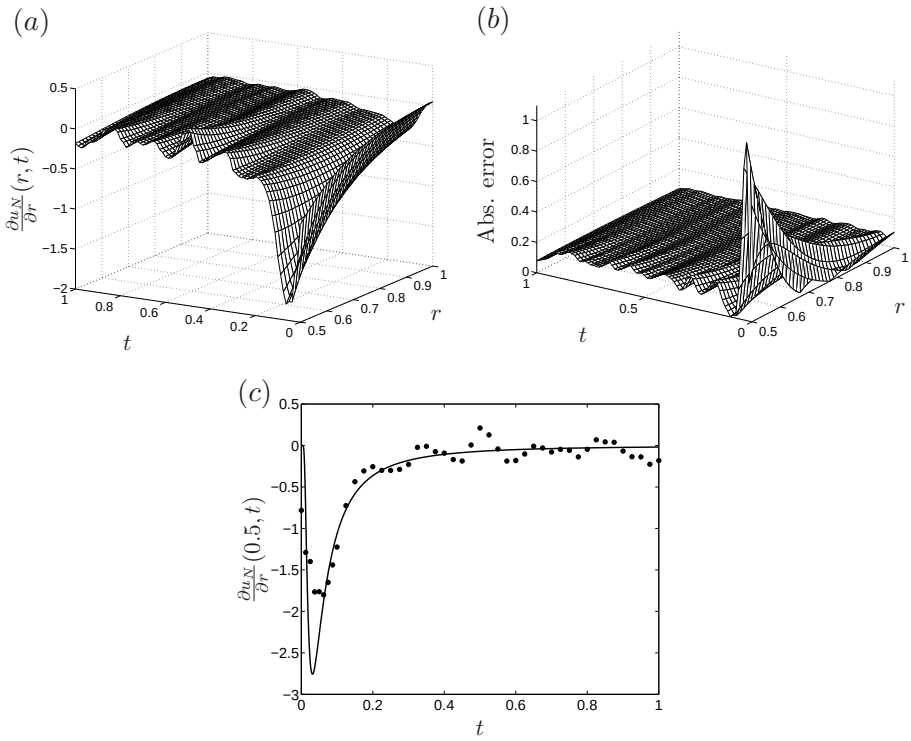

Figure 5.6: (a) The MFS approximation of the derivative $\frac{\partial u_N}{\partial r}(r,t)$, (b) the absolute error, and (c) the exact solution $u_r(0.5,t)$ (——) and the MFS approximation $\frac{\partial u_N}{\partial r}(0.5,t)$ (•), obtained with $h_1 = h_2 = h = 0.5$, $M = 80$, $N = 40$, $\lambda = 10^{-1}$, $\delta_1 = 0.1\%$, $\delta_2 = 10\%$, for Example 1.

In Table 5.1 we have applied ten different sets of noisy data and given the RMSE and RRMSE for the best, average and least accurate approximations. In [109] plots and error results were produced for $r_0 = 0.8$, with results improving over the case $r_0 = 0.5$, which is to be expected, however, we only produce the RMSE and RRMSE results in Table 5.2 for comparison purposes. Stability is seen in Tables 5.1 and 5.2 when, for the ten trials, we observe that the RMSE and RRMSE values are relatively close.

Table 5.1: The best, average and least accurate MFS approximations for $u_N(0.5,t)$ and $\frac{\partial u_N}{\partial r}(0.5,t)$ for $t \in [0,1]$, after ten different sets of noisy data with noise levels $\delta_1 = 0.1\%$ and $\delta_2 = 10\%$, for $M = 80$, $N = 40$, $h_1 = h_2 = h = 0.5$, $\lambda = 10^{-1}$, for Example 1.

	RMSE(u_N)	RRMSE(u_N)	RMSE($\frac{\partial u_N}{\partial r}$)	RRMSE($\frac{\partial u_N}{\partial r}$)
Best	0.0381	0.1615	0.2668	0.3200
Average	0.0419	0.1776	0.2967	0.3559
Least	0.0514	0.2180	0.3584	0.4298

Table 5.2: The best, average and least accurate MFS approximations for $u_N(0.8,t)$ and $\frac{\partial u_N}{\partial r}(0.8,t)$ for $t \in [0,1]$, after ten different sets of noisy data with noise levels $\delta_1 = 0.1\%$ and $\delta_2 = 10\%$, for $M = 80$, $N = 40$, $h_1 = h_2 = h = 0.5$, $\lambda = 10^{-1}$, for Example 1.

	RMSE(u_N)	RRMSE(u_N)	RMSE($\frac{\partial u_N}{\partial r}$)	RRMSE($\frac{\partial u_N}{\partial r}$)
Best	0.0072	0.0609	0.0493	0.1694
Average	0.0088	0.0747	0.0566	0.1942
Least	0.0125	0.1064	0.0748	0.2567

In Figures 5.7 and 5.8 we produce numerical results where the noise has been added as described in (5.11)–(5.14). We feel that this noise is fairer and can model errors in real applications, since it scales with the maximum value of the boundary data, compared to the noise (5.10) that was used in [109]. The results in Figures 5.7 and 5.8 appear more accurate and stable than those obtained for Figures 5.4 and 5.6, which is to be expected. We obtained RMSE = 0.0229 and RRMSE = 0.0973, and RMSE = 0.1736 and RRMSE = 0.2083 in Figures 5.7(b) and 5.8(b), respectively.

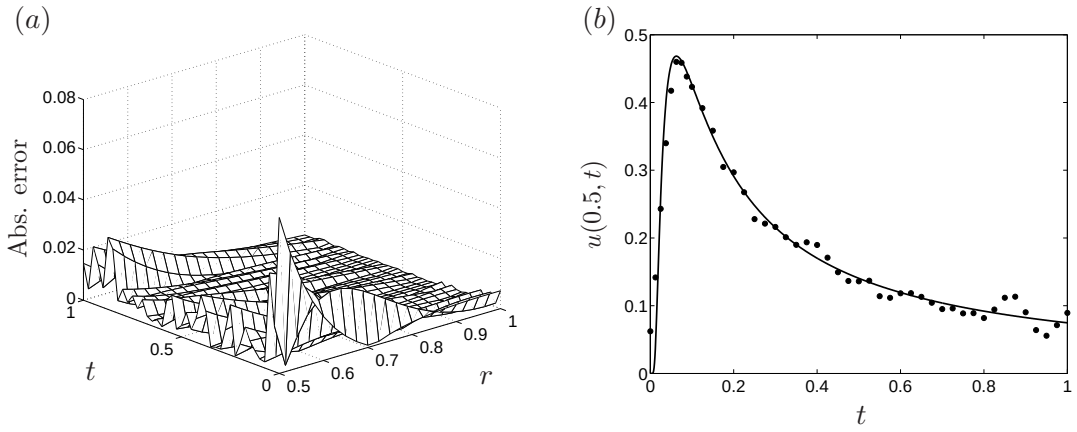

Figure 5.7: (a) The absolute error and (b) the exact solution $u(0.5,t)$ (——) and the MFS approximation $u_N(0.5,t)$ (•), obtained with $h_1 = h_2 = h = 0.5$, $M = 160$, $N = 80$, $\lambda = 10^{-2}$, $\delta_1 = 0.1\%$, $\delta_2 = 10\%$, for Example 1.

Lastly, in Figures 5.9(a)–5.9(c) we have produced plots of the RMSE, for no noise, for different values of the parameters h, N and λ. We find that there are regions where the approximation is the best ($h \approx 0.5$ and $N = 80$). Additional observations: the approximation does not appear to depend on h_2 for reasonable choices of this parameter, but is rather sensitive to changes in the parameter h_1 (this is probably because h_1 moves the source towards the centre of the inner cylinder, and $h_1 = 0.5$ is the furthest point away from the inner boundary), and choices for N and λ are more flexible than $h = h_1 = h_2$.

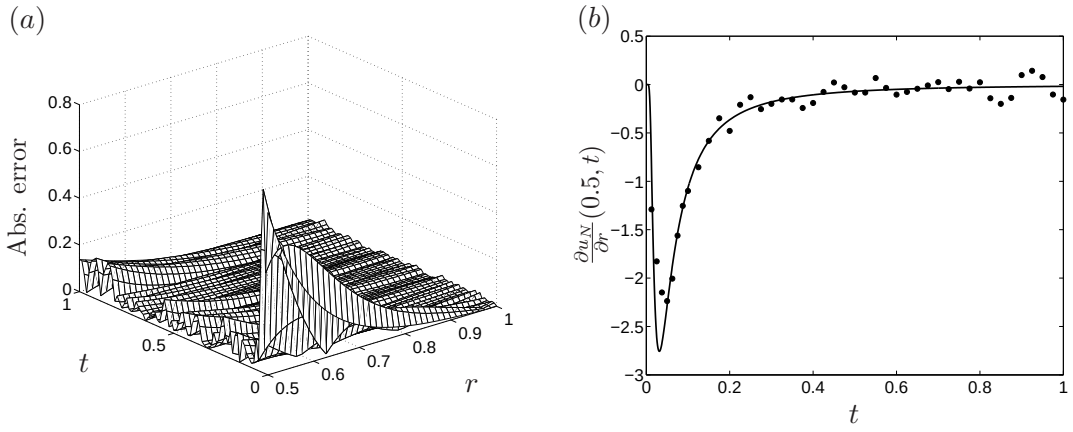

Figure 5.8: (a) The absolute error and (b) the exact derivative $u_r(0.5,t)$ (——) and the MFS approximation $\frac{\partial u_N}{\partial r}(0.5,t)$, obtained with $h_1 = h_2 = h = 0.5$, $M = 160$, $N = 80$, $\lambda = 10^{-2}$ $\delta_1 = 0.1\%$, $\delta_2 = 10\%$, for Example 1.

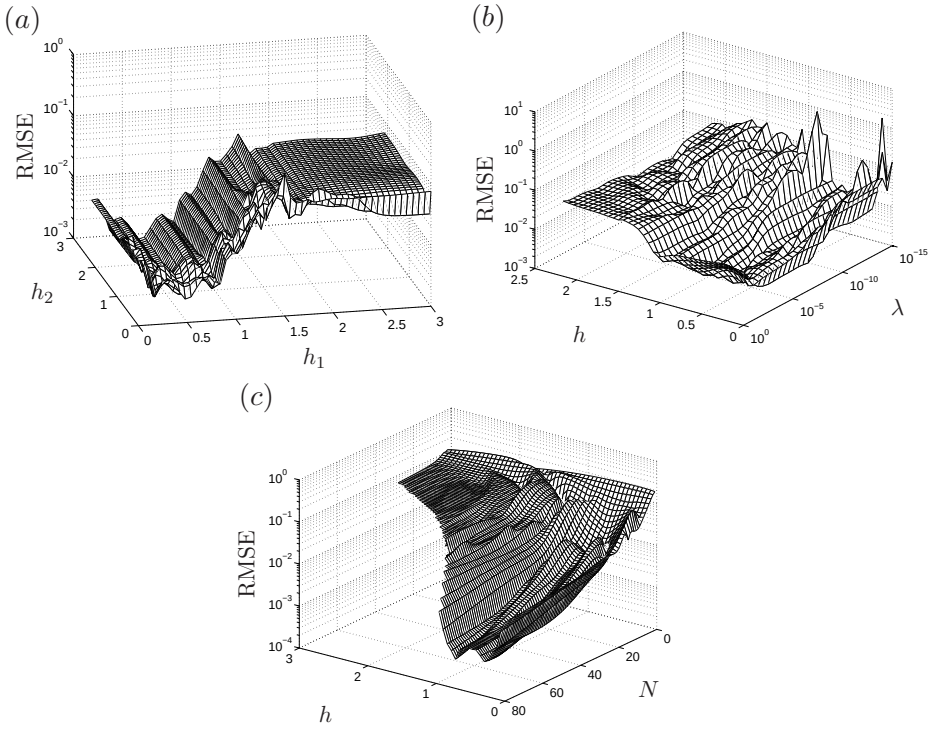

Figure 5.9: The RMSE for: (a) $(h_1, h_2) \in [0,3] \times [0,3]$, obtained with $M = 80$, $N = 40$, (160 collocation points and 160 source points), (b) $(h, \lambda) \in [0,3] \times [10^{-15}, 1]$ $(h = h_1 = h_2)$ obtained with $M = 80$, $N = 40$, and (c) $(h, N) \in [0,3] \times [1,80]$ $(4N$ $(2N = M)$ collocation and source points), obtained with $\lambda = 10^{-5}$, $\delta_1 = \delta_2 = 0$, for Example 1.

5.4.2 Example 2

We now consider the other example in [109], however, in [109] only tables of the RMSE and RRMSE results were given for $r_0 = 0.8$ (we will produce plots and tables for $r_0 = 0.5$ and tables

for $r_0 = 0.8$). We solve the problem (4.2), (5.1) and (5.2), with the Cauchy boundary data (5.1) and (5.2) given by

$$u(1,t) = f(t) = 1 + 4t, \quad 0 < t \leq 1, \tag{5.15}$$

$$\frac{\partial u}{\partial r}(1,t) = g(t) = 2, \quad 0 < t \leq 1, \tag{5.16}$$

and the analytic solution is given by

$$u(r,t) = r^2 + 4t, \quad 0.5 < r < 1,\ 0 < t \leq 1. \tag{5.17}$$

We again add noise to the boundary data in a similar way to Example 1, i.e. we apply (5.10) when noise is the same as in [109], and in Figure 5.14 we add noise in a similar way to (5.11)–(5.14).

In Figure 5.10(b) is the MFS approximation of the exact solution (5.17) given in Figure 5.10(a). The absolute error plot in Figure 5.11(a) displays the accuracy and stability of the approximate solution. We note that in this example we have set $h_1 = h_2 = h = 2$, i.e. the source points have been placed further from the boundary than in the previous example, where $h_1 = h_2 = h = 0.5$. In the first example, due to the shape and rapid change of the solution at the inner boundary r_0, the source points need to be placed closer to recover this behaviour. In Figure 5.11(b) we have plotted the exact solution and the MFS approximation along the boundary $r_0 = 0.5$, as a function of time $t \in [0,1]$, and the agreement is good, with the RMSE = 0.0107 and the RRMSE = 0.0042. Figures 5.12 and 5.13 contain plots of the approximation of the derivative, and we find that the results are accurate, even for the derivative in this example. We obtained RMSE = 0.0398 and RRMSE = 0.0199 in Figure 5.13(b).

Tables 5.3 and 5.4 for Example 2 are analogous to Tables 5.1 and 5.2 in Example 1.

Table 5.3: The best, average and least accurate MFS approximations for $u_N(1/2,t)$ and $\partial u_N/\partial r(1/2,t)$ for $t \in [0,1]$, after ten different sets of noisy data with noise levels $\delta_1 = 0.1\%$ and $\delta_2 = 10\%$, for $M = 80$, $N = 40$, $h_1 = h_2 = h = 2$, $\lambda = 10^{-4}$, for Example 2.

	RMSE(u_N)	RRMSE(u_N)	RRMSE($\frac{\partial u_N}{\partial r}$)
Best	0.0010	0.0004	0.0071
Average	0.0044	0.0017	0.0123
Least	0.0107	0.0042	0.0316

In Figures 5.14(a) and 5.14(b) the absolute errors between the numerical MFS solutions

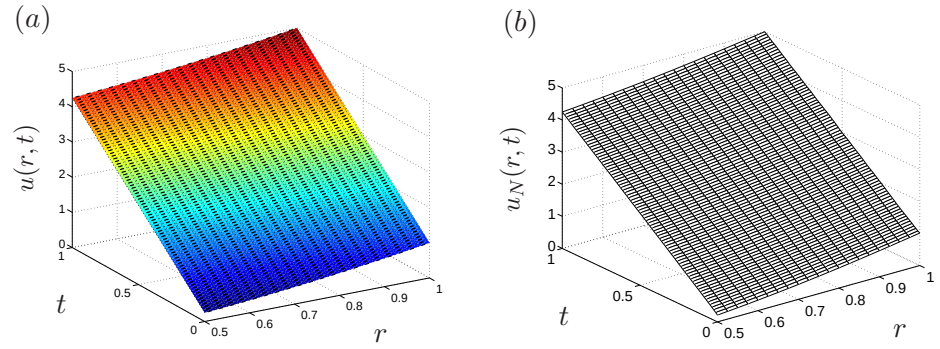

Figure 5.10: (a) The exact solution $u(r,t)$ and (b) the MFS approximation $u_N(r,t)$, obtained with $h_1 = h_2 = h = 2$, $M = 80$, $N = 40$, $\lambda = 10^{-4}$, $\delta_1 = 0.1\%$, $\delta_2 = 10\%$, for Example 2.

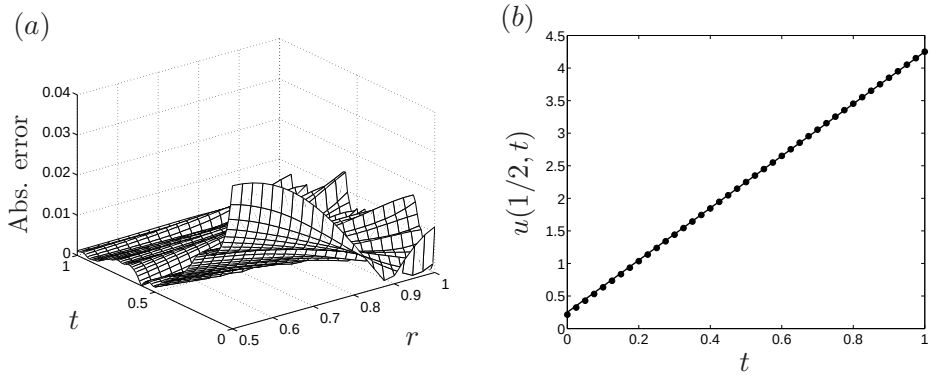

Figure 5.11: (a) The absolute error and (b) the exact solution $u(1/2,t)$ (——) and the MFS approximation $u_N(1/2,t)$ (•), obtained with $h_1 = h_2 = h = 2$, $M = 80$, $N = 40$, $\lambda = 10^{-4}$, $\delta_1 = 0.1\%$, $\delta_2 = 10\%$, for Example 2.

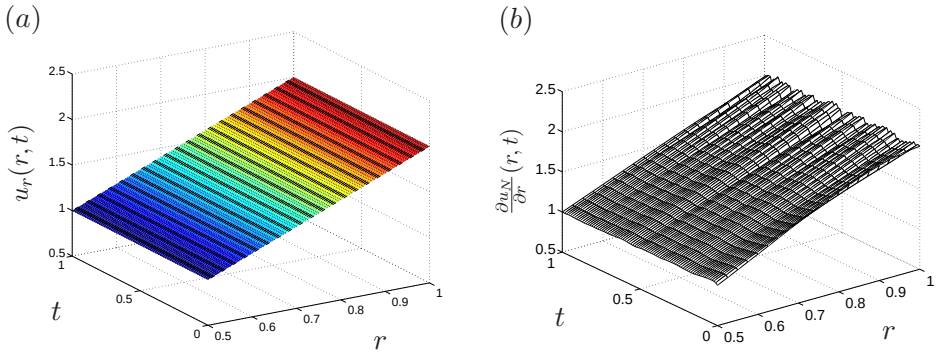

Figure 5.12: (a) The exact derivative $u_r(r,t)$ and (b) the MFS approximation $\frac{\partial u_N}{\partial r}(r,t)$, obtained with $h_1 = h_2 = h = 2$, $M = 80$, $N = 40$, $\lambda = 10^{-4}$, $\delta_1 = 0.1\%$, $\delta_2 = 10\%$, for Example 2.

and the exact solutions for $u(r,t)$ and $\partial u/\partial r(r,t)$, respectively, have been plotted when noise has been added as in (5.11)–(5.14). We obtained RMSE = 0.0123 and RRMSE = 0.0043, and RMSE = 0.0636 and RRMSE = 0.0416 in Figures 5.14(a) and 5.14(b), respectively.

Lastly, as in Example 1 we produce RMSE plots for different parameter values in Figure

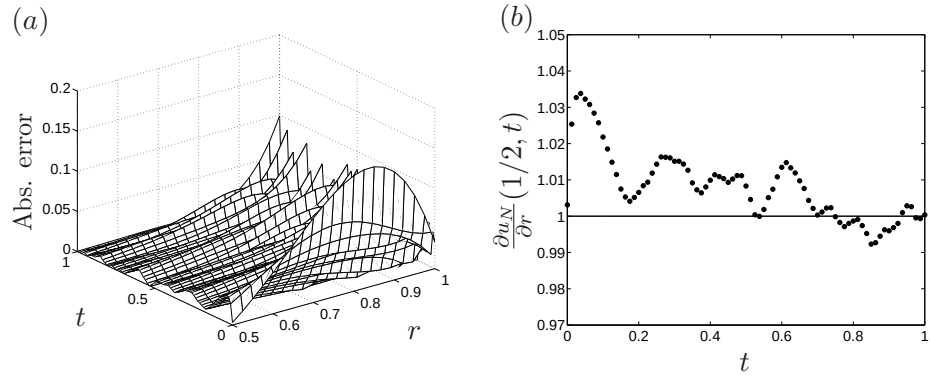

Figure 5.13: (a) The absolute error, and (b) the exact derivative $u_r(1/2,t)$ (——) and the MFS approximation $\frac{\partial u_N}{\partial r}(1/2,t)$ (•), obtained with $h_1 = h_2 = h = 2$, $M = 80$, $N = 40$, $\lambda = 10^{-4}$, $\delta_1 = 0.1\%$, $\delta_2 = 10\%$, for Example 2.

Table 5.4: The best, average and least accurate MFS approximations for $u_N(0.8,t)$ and $\partial u_N/\partial r(0.8,t)$ for $t \in [0,1]$, after ten different sets of noisy data with noise level $\delta_1 = 0.1\%$ and $\delta_2 = 10\%$, for $M = 80$, $N = 40$, $h_1 = h_2 = h = 2$, $\lambda = 10^{-4}$, for Example 2.

	RMSE(u_N)	RRMSE(u_N)	RMSE($\frac{\partial u_N}{\partial r}$)	RRMSE($\frac{\partial u_N}{\partial r}$)
Best	0.0039	0.0013	0.0475	0.0296
Average	0.0050	0.0017	0.0733	0.0459
Least	0.0063	0.0022	0.0894	0.0559

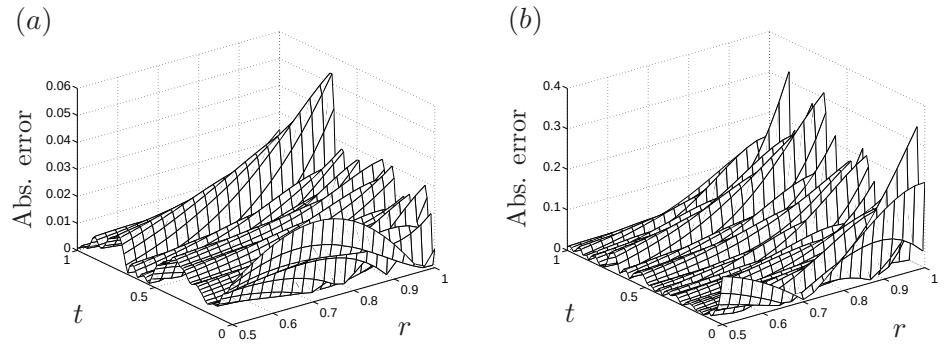

Figure 5.14: The absolute errors (a) $|u(r,t) - u_N(r,t)|$, and (b) $|\partial u/\partial r(r,t) - \partial u_N/\partial r(r,t)|$, obtained with $h_1 = h_2 = h = 2$, $M = 80$, $N = 40$, $\lambda = 10^{-4}$, $\delta_1 = 0.1\%$, $\delta_2 = 10\%$, for Example 2.

5.15, and we find that larger h produces better results, as stated earlier in this example.

5.5 Summary of Chapter 5

In this chapter, we have investigated MFS applied to the radially symmetric inverse heat conduction problem. Two examples were investigated and compared to [109] where a different method

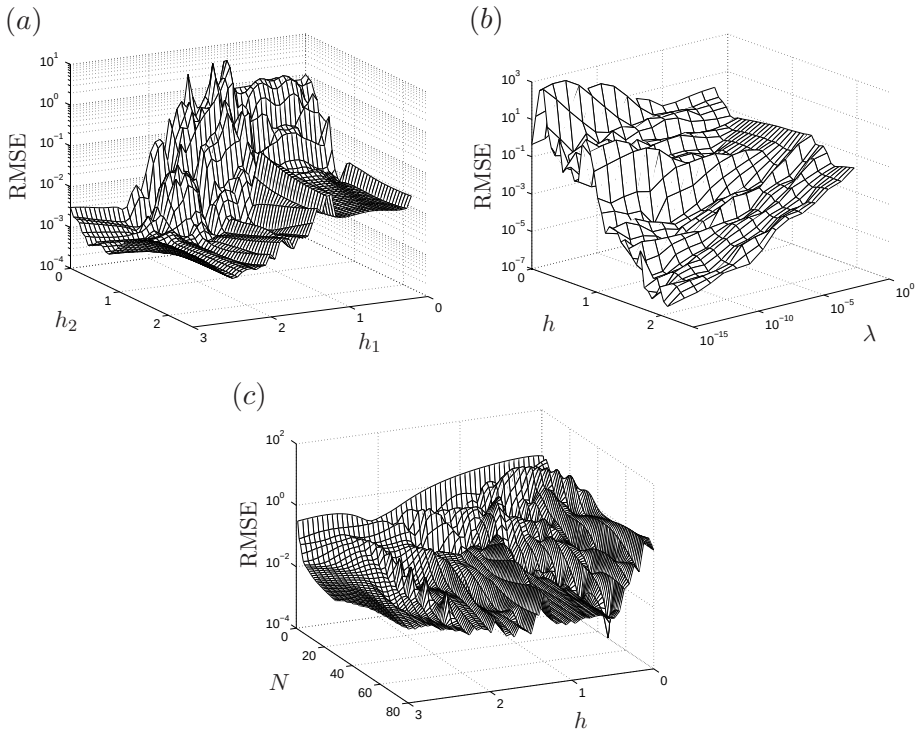

Figure 5.15: The RMSE for: (a) $(h_1, h_2) \in [0,3] \times [0,3]$, obtained with $M = 80$, $N = 40$, (160 collocation points and 160 source points), (b) $(h, \lambda) \in [0,3] \times [10^{-15}, 1]$ ($h = h_1 = h_2$) obtained with $M = 80$, $N = 40$, and (c) $(h, N) \in [0,3] \times [1, 80]$ ($4N$ ($2N = M$) collocation and source points), obtained with $\lambda = 10^{-5}$, $\delta_1 = \delta_2 = 0$, for Example 2.

was employed, and the results were found to be at least the same or better. Furthermore, stable results were obtained when noise was applied to the boundary data.

In the following chapters we investigate inverse Stefan problems, which involve a moving phase boundary.

CHAPTER 6

THE CLASSICAL ONE-DIMENSIONAL INVERSE STEFAN PROBLEM

6.1 Preliminaries

In the remaining chapters we investigate inverse Stefan problems, which feature a moving boundary, details of which can be found in Section 0.2 of the Introduction, and in this chapter we consider the classical case. In this problem we again wish to solve the heat equation equipped with an initial condition and overspecified boundary data given on the free boundary (which is known), and we want to determine the temperature and flux data at a fixed inner boundary.

Let the free boundary (sufficiently smooth) be denoted by Γ_S, and given by $x = s(t)$, and denote the temperature solution $u(x,t)$ in the heat conduction domain $D = (0, s(t)) \times (0, T]$, where $T > 0$ is a given arbitrary final time of interest. We have a fixed boundary at $x = 0$, which we denote by Γ_U. We denote the union of the boundaries by $\Gamma = \Gamma_U \cup \Gamma_S$ and the closure of the domain D by $\overline{D} = [0, s(t)] \times [0, T]$. For the inverse one-phase Stefan problem we seek a solution $u(x,t)$, which satisfies the one-dimensional heat equation

$$\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} = 0, \quad (x,t) \in D, \tag{6.1}$$

subject to the initial condition

$$u(x,0) = u_0(x), \quad x \in [0, s(0)], s(0) > 0, \tag{6.2}$$

the Dirichlet and Neumann boundary conditions on the moving boundary $x = s(t)$

$$u(s(t),t) = 0, \quad t \in (0,T], \tag{6.3}$$

$$\frac{\partial u}{\partial x}(s(t),t) = -s'(t), \quad t \in (0,T]. \tag{6.4}$$

Conditions (6.3) and (6.4) can be replaced by the more general boundary conditions (0.9) and (0.10). In (6.2), $u_0 \in C^1([0,s(0)])$, in (6.4), $s \in C^1([0,T])$, in (0.9) and (0.10) $h_1, h_2 \in C^1([0,T])$ and satisfy the compatibility conditions $h_1(0) = u_0(s(0))$, $h_2(0) = u_0'(s(0))$.

For details on existence and uniqueness see Section 0.2 of the Introduction.

6.2 Denseness properties of linear combinations of fundamental solutions

We recall here the fundamental solution for the one-dimensional heat equation (0.1):

$$F(x,t;y,\tau) \;=\; \frac{H(t-\tau)}{(4\pi(t-\tau))^{\frac{1}{2}}} e^{-\frac{(x-y)^2}{4(t-\tau)}}, \tag{6.5}$$

where H is the Heaviside function.

We construct a set of source points placed external to the domain $\overline{D}$. We denote by D_E the domain, which contains the domain $\overline{D}$, with bounding surface Γ_E, with both D_E and Γ_E extended for all time $\tau \in (-T,T)$, see Figure 6.1 for a representation of the domain, boundary and placement of source points. The boundary Γ_E will be split into two boundaries $\Gamma_E^{(1)}$ (for $y < 0$ with points on this boundary denoted by $y_1(t)$) and $\Gamma_E^{(2)}$ (for $y > 0$ with points on this boundary denoted by $y_2(t)$). We make the restriction that the distance, with respect to x, between the points on Γ_E for $\tau < 0$ and $S_0 = [0,s(0)] \times [-T,0]$ should be greater than zero.

Linear independence and denseness results were proved in [27] for a certain placement of source points; in the next section we extend these results to a more general placement of source points, hence justifying the MFS representation (6.6). The proof is very similar to that of the proof of linear independence and denseness presented in Section 1.2 of Chapter 1 (Theorems 1.2.2 and 1.2.3). Therefore, only the salient details will be given below.

We construct a denumerable, everywhere dense set of source points on the external boundary Γ_E, and denote this set by $\{y_j(\tau_m), \tau_m\}_{m=1,2,\ldots}$ and $j = 1,2$. We construct an MFS approxima-

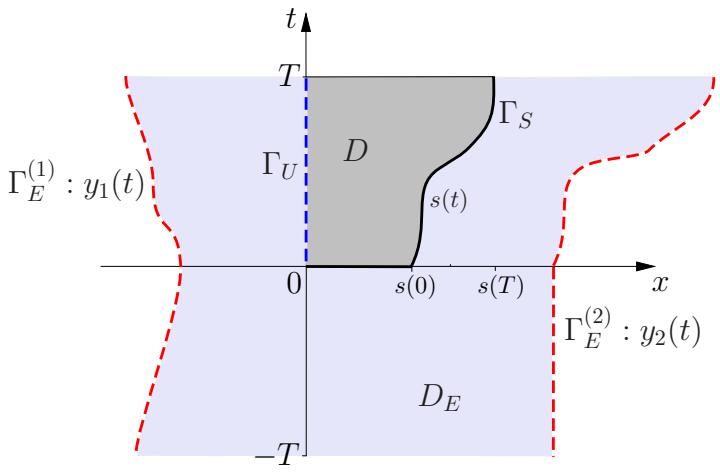

Figure 6.1: General representation of the domain D and boundary $\Gamma = \Gamma_U \cup \Gamma_S$, with unspecified boundary condition (---) on Γ_U, and source points (---) placed on $\Gamma_E = \Gamma_E^{(1)} \cup \Gamma_E^{(2)}$ external to the domain D.

tion to (0.1)–(0.4) as a linear combination of these fundamental solutions given by

$$u_\infty(x,t) = \sum_{j=1}^{2} \sum_{m=1}^{\infty} c_m^{(j)} F(x,t;y_j(\tau_m),\tau_m), \quad (x,t) \in \overline{D}. \tag{6.6}$$

We set the constants $c_m^{(j)}$ equal to zero except for a finite number of values.

Before stating the following theorem we define the function $\tilde{s}(t)$ to describe any boundary such that $s(0) = \tilde{s}(0)$, $\tilde{s}(t) > 0$ and $\tilde{s}(t) < y_2(t)$ for all $t \in (-T, 0)$ and let

$$S(t) = \begin{cases} s(t) & \text{if } t \in [0,T), \\ \tilde{s}(t) & \text{if } t \in (-T,0). \end{cases}$$

The boundary $S(t)$ has been introduced for use with proving the following theorem and will not be needed for the numerical implementation, since we only have conditions on the boundary $s(t)$. We also note that source points can be placed down to any time point $\tau < 0$, and the choice of $-T$ is based on the numerical results produced in Example 2 in Chapter 1, which showed that using different initial points did not improve the accuracy. The theorems will prove denseness for both Dirichlet and Neumann boundary conditions, which allows us to use mixed boundary conditions, justifying the use of the MFS for the inverse Stefan problem.

6.2.1 Denseness on the lateral surfaces

Theorem 6.2.1. *The set of restrictions* $(\{F(0,t;y_j(\tau_m),\tau_m)\},\{F(S(t),t;y_j(\tau_m),\tau_m)\})_{m=1}^{\infty}$, $j=1,2$, *form a linearly independent and dense set in* $L^2(-T,T)\times L^2(-T,T)$.

Proof. Linear independence: Following the structure of the proof of Theorem 1.2.2, we assume we do not have linear independence, for a contradiction, then there exist integers N, $j_0\in\{1,2\}$, $m_0\in\{1,\dots,N\}$, and a non-zero constant $c_{m_0}^{(j_0)}$, such that

$$\sum_{m=1}^{N}\sum_{j=1}^{2} c_m^{(j)}(F(0,t;y_j(\tau_m),\tau_m),F(S(t),t;y_j(\tau_m),\tau_m)) \;=\; (0,0). \tag{6.7}$$

We define a function

$$U(x,t) \;=\; \sum_{m=1}^{N}\sum_{j=1}^{2} c_m^{(j)}F(x,t;y_j(\tau_m),\tau_m),\quad (x,t)\in(0,S(t))\times(-T,T), \tag{6.8}$$

which satisfies

$$\frac{\partial U}{\partial t}-\Delta U \;=\; 0,\quad \text{in } (0,S(t))\times(-T,T), \tag{6.9}$$

$$U(0,t) \;=\; U(S(t),t)=0,\;\; t\in(-T,T), \tag{6.10}$$

$$U(x,-T) \;=\; 0,\;\; x\in(0,S(-T)). \tag{6.11}$$

By uniqueness $U=0$ in $(0,S(t))\times(-T,T)$, and since U is analytic, $U\equiv 0$ in D_E. Hence, letting the point (x,t) approach the point $(y_{j_0}(\tau_{m_0}),\tau_{m_0})\in\Gamma_E$ such that the ratio

$$\frac{(x-y_{j_0}(\tau_{m_0}))^2}{4(t-\tau_{m_0})} \tag{6.12}$$

remains bounded, we can make the term $c_{m_0}^{(j_0)}F(x,t;y_{j_0}(\tau_{m_0}),\tau_{m_0})$ in (6.8) as large as we wish, with the other terms in (6.8) remaining bounded; which gives us a contradiction.

Denseness: To prove that $(\{F(0,t;y_j(\tau_m),\tau_m)\},\{F(S(t),t;y_j(\tau_m),\tau_m)\})_{m=1}^{\infty}$, $j=1,2$, form a dense set in $L^2(-T,T)\times L^2(-T,T)$ assume, for a contradiction, that we do not have denseness; hence there exists a continuous function $f(t)=(f_1(t),f_2(t))$ in $L^2(-T,T)\times L^2(-T,T)$ such that

$$\int_{-T}^{T} F(0,t;y_j(\tau_m),\tau_m)f_1(t)+F(S(t),t;y_j(\tau_m),\tau_m)f_2(t)\,dt \;=\; 0,\;\; j=1,2,\; m\geq 1. \tag{6.13}$$

We define an equivalent form of the single-layer potential given by

$$V(y,\tau) \;=\; \int_{\tau}^{T} F(0,t;y,\tau)f_1(t) + F(S(t),t;y,\tau)f_2(t)\,dt, \;\; (y,\tau)\in D_E. \tag{6.14}$$

It can be shown that $(f_1, f_2) \equiv (0,0)$ using the same jump condition argument presented in the proof of denseness for Theorem 1.2.2, which gives us a contradiction. □

We now state and give the main details for the proof of a denseness theorem when we have Neumann boundary conditions. The spatial derivative of the fundamental solution (6.5) is given by

$$F_x(x,t;y,\tau) = -\frac{(x-y)H(t-\tau)}{4\pi^{\frac{1}{2}}(t-\tau)^{\frac{3}{2}}}e^{-\frac{(x-y)^2}{4(t-\tau)}}. \tag{6.15}$$

The proof will have the same structure as the proofs of Theorems 1.2.2 and 6.2.1.

Theorem 6.2.2. *The restrictions* $(\{F_x(0,t;y_j(\tau_m),\tau_m)\},\{F_x(S(t),t;y_j(\tau_m),\tau_m)\})_{m=1}^{\infty}$, $j=1,2$, *form a linearly independent and dense set in* $L^2(-T,T)\times L^2(-T,T)$.

Proof. Linear independence: We again assume, for a contradiction, we do not have linear independence, then there exist integers N, $j_0\in\{1,2\}$, $m_0\in\{1,\dots,N\}$, and a non-zero constant $c_{m_0}^{(j_0)}$, such that

$$\sum_{m=1}^{N}\sum_{j=1}^{2} c_m^{(j)}(F_x(0,t;y_j(\tau_m),\tau_m), F_x(S(t),t;y_j(\tau_m),\tau_m)) \;=\; (0,0). \tag{6.16}$$

We define a function

$$U(x,t) \;=\; \sum_{m=1}^{N}\sum_{j=1}^{2} c_m^{(j)}F_x(x,t;y_j(\tau_m),\tau_m), \;\; (x,t)\in(0,S(t))\times(-T,T), \tag{6.17}$$

which again satisfies equations (6.9)–(6.11), since derivatives of functions which satisfy the heat equation also satisfy the heat equation, (6.7), and due to the presence of the Heaviside function.

By uniqueness $U=0$ in $(0,S(t))\times(-T,T)$, and since U is analytic (since F_x is also analytic) $U\equiv 0$ in D_E. Letting the point (x,t) approach the point $(y_{j_0}(\tau_{m_0}),\tau_{m_0})\in\Gamma_E$ along the curve $x = y_{j_0}(\tau_{m_0}) + \sqrt{t-\tau_{m_0}}$ we can guarantee that the ratio

$$\frac{(x-y_{j_0}(\tau_{m_0}))^2}{4(t-\tau_{m_0})} \tag{6.18}$$

remains bounded, and

$$\frac{|x - y_{j_0}(\tau_{m_0})|}{(t - \tau_{m_0})^{\frac{3}{2}}} \to \infty.$$

Therefore, we can make the term $c_{m_0}^{(j_0)} F_x(x, t; y_{j_0}(\tau_{m_0}), \tau_{m_0})$ in (6.17) as large as we wish, with the other terms in (6.8) remaining bounded; which gives us a contradiction.

Denseness: To prove that $(\{F_x(0, t; y_j(\tau_m), \tau_m)\}, \{F_x(S(t), t; y_j(\tau_m), \tau_m)\})_{m=1}^{\infty}$, $j = 1, 2$, form a dense set in $L^2(-T, T) \times L^2(-T, T)$ assume, for a contradiction, that we do not have denseness; hence there exists a continuous function $f(t) = (f_1(t), f_2(t))$ in $L^2(-T, T) \times L^2(-T, T)$ such that

$$\int_{-T}^{T} F_x(0, t; y_j(\tau_m), \tau_m) f_1(t) + F_x(S(t), t; y_j(\tau_m), \tau_m) f_2(t)\, dt \;=\; 0, \;\; j = 1, 2, \; m \geq 1. \tag{6.19}$$

We define an equivalent form of the double-layer potential given by

$$K(y, \tau) \;=\; \int_{\tau}^{T} F_x(0, t; y, \tau) f_1(t) + F_x(S(t), t; y, \tau) f_2(t)\, dt, \;\; (y, \tau) \in D_E. \tag{6.20}$$

Since the double-layer potential is a smooth solution of the heat equation in the exterior of $[0, S(t)] \times (-T, T)$, is continuous across $(\{0\} \cup \{S(t)\}) \times [-T, T]$, satisfies the heat equation in $(0, s(t)) \times (-T, T)$, and satisfies the jump relation, see [38] and Chapter 3 of [76], we can show using a similar argument presented in the proof of denseness for Theorem 1.2.2 that $(f_1, f_2) \equiv (0, 0)$, which gives us a contradiction. Hence $(\{F_x(0, t; y_j(\tau_m), \tau_m)\}, \{F_x(S(t), t; y_j(\tau_m), \tau_m)\})_{m=1}^{\infty}$, $j = 1, 2$ is a dense set in $L^2(-T, T) \times L^2(-T, T)$. □

6.2.2 Denseness on the base surface

We now state and prove the following theorem, which states that linear combinations of the fundamental solution $\{F(x, 0; y_j(\tau_m), \tau_m)\}_{m=1}^{\infty}$ are dense, for functions in L^2, on the initial base.

Theorem 6.2.3. *The set of restrictions denoted* $\{F(x, 0; y_j(\tau_m), \tau_m)\}_{m=1}^{\infty}$, $j = 1, 2$, *where* $F(x, 0; y_j(\tau_m), \tau_m)$, *is given by* (6.5), *with* $\tau_m < 0$, *form a linearly independent and dense set in* $L^2(0, s(0))$.

Proof. Linear independence: The proof follows the same arguments used in the proof of Theorem 1.2.3, i.e. using proof by contradiction, corollary of Theorem 3 given in [66], and the 'ratio/limit'

argument.

Denseness: We show that the sequence of functions $\{F(x,0;y_j(\tau_m),\tau_m\}_{m=1}^{\infty}$, $j=1,2$, where $\tau_m<0$, is a dense set in $L^2(0,s(0))$. We assume that we do not have denseness, therefore there exists a function $f\in C^2(0,s(0))$ such that

$$\int_0^{s(0)} F(x,0;y_j(\tau_m),\tau_m)f(x)\,dx \;=\; 0. \tag{6.21}$$

We let w be a weak solution to the heat equation with initial condition $w(x,0)=f(x)$, and boundary conditions $w(0,t)=0$ and $w(s(t),t)=0$ for $t\in(0,T)$. We use Green's identities, see [35, 72], to transform the integral (6.21) into the equations

$$\int_0^T F(0,t;y_j(\tau_m),\tau_m)\frac{\partial w}{\partial x}(0,t)\,dt \;=\; 0, \tag{6.22}$$

$$\int_0^T F(s(t),t;y_j(\tau_m),\tau_m)\frac{\partial w}{\partial x}(s(t),t)\,dt \;=\; 0. \tag{6.23}$$

We know, by Theorem 6.2.1, that $\{F(0,t;y_j(\tau_m),\tau_m)\}$ and $\{F(s(t),t;y_j(\tau_m),\tau_m)\}$ constitute dense sets, which means $\frac{\partial w}{\partial x}(0,t)=\frac{\partial w}{\partial x}(s(t),t)=0$. Therefore $w(0,t)=\frac{\partial w}{\partial x}(0,t)=0$ and by [95] we have $w(x,t)=0$ for all $(x,t)\in[0,\min_{t\in[0,T]}s(t)]\times[0,T]$, and by unique continuation we have $w(x,t)=0$ for all $(x,t)\in D$. Hence, $f(x)\equiv 0$ and the set of functions $\{F(x,0;y_j(\tau_m),\tau_m\}_{m=1}^{\infty}$, $j=1,2$, where $\tau_m<0$, is a dense set in $L^2(0,s(0))$. □

6.3 The MFS for the inverse one-phase one-dimensional Stefan problem

To approximate equations (0.1), (0.2), (0.9) and (0.10) we seek a solution by truncating the infinite series in (6.6) and consider the finite sum

$$u_M(x,t) \;=\; \sum_{j=1}^{2}\sum_{m=1}^{2M} c_m^{(j)}F(x,t;y_j(\tau_m),\tau_m),\quad (x,t)\in\overline{D}. \tag{6.24}$$

We have a one-dimensional domain, with a fixed boundary at $x=0$ and a moving boundary given by $x=s(t)$, and source points placed at the coordinates $(-h,\tau)$ for $\tau\in(-T,T)$, $(h+s(\tau),\tau)$ for $\tau\in(0,T)$, and $(h+s(-\tau),\tau)$ for $\tau\in(-T,0)$. Source points have been placed symmetrically with respect to τ via a reflection through $t=0$, see Figure 6.2. In Section 6.4 we will test other

source point locations to see if a different placement produces better results.

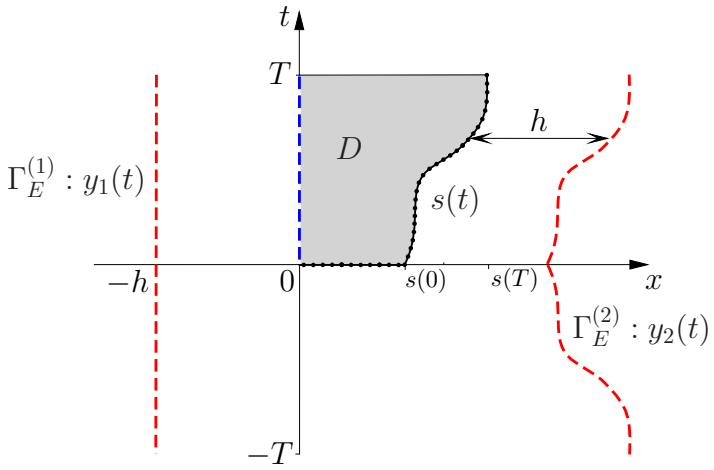

Figure 6.2: General representation of the domain D and boundary Γ, with unspecified boundary condition (---) on Γ_U, collocation points (-●-) on $S_0 \cup \Gamma_S$, and source points (---) placed on Γ_E external to the domain D and symmetrically to $t = 0$.

The source points will be placed at time points $(\tau_m)_{m=1,\dots,2M} \in (-T, T)$ given by

$$\tau_m = \frac{2(m-M)-1}{2M}T, \quad m = 1, \dots, 2M, \tag{6.25}$$

and on Γ_E set

$$y_1(\tau_m) = -h \;\text{ and }\; y_2(\tau_m) = h + s(|\tau_m|), \quad m = 1, \dots, 2M. \tag{6.26}$$

We have in total $4M$ source points on the external boundary Γ_E, and we place the same number of collocation points on the lateral and base surfaces $S_0 \cup \Gamma_S$. Letting

$$t_i = \frac{i}{M}T, \;\; x_1^{(i)} = s(t_i), \quad i = 0, \dots, M, \qquad x_0^{(k)} = \frac{ks(0)}{K+1}, \;\; k = 1, \dots, K. \tag{6.27}$$

We obtain the following system of equations

$$u_M(x_0^{(k)}, 0) = u_0(x_0^{(k)}), \quad k = 1, \dots, K, \tag{6.28}$$

$$u_M(x_1^{(i)}, t_i) = 0, \tag{6.29}$$

$$\frac{\partial u_M}{\partial x}(x_1^{(i)}, t_i) = -s'(t_i), \quad i = 0, \dots, M. \tag{6.30}$$

The system of equations (6.28)–(6.30) contains $K + 2(M + 1)$ equations and $4M$ unknowns. Therefore, in order to obtain a unique solution, we require $K \geq 2M - 2$. We will apply Tikhonov regularization, with the L-curve criterion.

6.4 Numerical results

Two benchmark test examples previously investigated in [89] for different Stefan-type problems, will be presented in this section; the first will have $s(t)$ being a linear function of t, whilst in the second example $s(t)$ will be a non-linear function of t. In both examples different noise levels will be applied to the Neumann boundary condition (0.4) (or (0.10)).

6.4.1 Example 1

The first example has a moving boundary given by the linear function

$$s(t) = \sqrt{2} - 1 + \frac{t}{\sqrt{2}}, \;\; t \in [0, T = 1]. \tag{6.31}$$

We shall place source points on the external boundaries $(-h, \tau)$, $\tau \in (-1, 1)$, $(s(\tau) + h, \tau)$, $\tau \in (0, 1)$ and $(s(-\tau) + h, \tau)$, $\tau \in (-1, 0)$, for an illustration see Figure 6.3. We take the exact solution given by

$$u(x,t) \;=\; -1 + \exp\Big(1 - \frac{1}{\sqrt{2}} + \frac{t}{2} - \frac{x}{\sqrt{2}}\Big), \;\; (x,t) \in [0, s(t)] \times [0, 1]. \tag{6.32}$$

Therefore, this example has the following initial and boundary conditions:

$$u(x,0) \;=\; -1 + \exp\Big(1 - \frac{1}{\sqrt{2}} - \frac{x}{\sqrt{2}}\Big), \;\; x \in [0, s(0)], \;\; s(0) = \sqrt{2} - 1, \tag{6.33}$$

$$u(s(t), t) \;=\; 0, \;\; t \in (0, 1], \tag{6.34}$$

$$\frac{\partial u}{\partial x}(s(t), t) \;=\; -s'(t) \;=\; -\frac{1}{\sqrt{2}}, \;\; t \in (0, 1]. \tag{6.35}$$

Random additive noise simulating measurement errors to the Neumann data (6.35) has been included in this example and is generated as follows:

$$u_x^{\delta}(s(t), t) = -\frac{1}{\sqrt{2}} + N(0, \sigma^2), \tag{6.36}$$

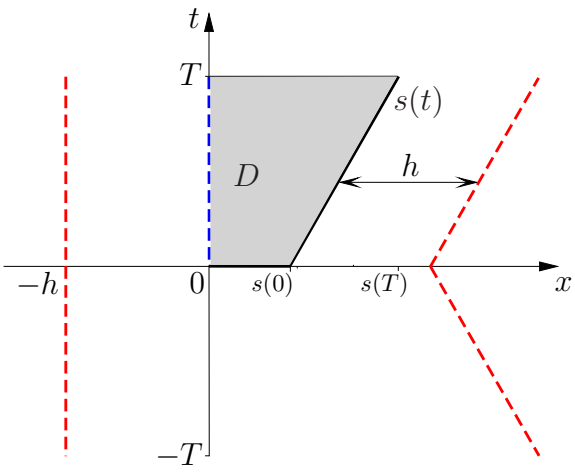

Figure 6.3: Particularisation of Figure 6.2 for $s(t)$ given by equation (6.31) in Example 1.

where $N(0, \sigma^2)$ represents the normal distribution with mean zero and standard deviation

$$\sigma = \delta \times \max_{(s(t),t),\, t\in(0,1]} |u_x(s(t), t)| \;=\; \delta/\sqrt{2}, \tag{6.37}$$

where δ is the relative (percentage) noise level. Noise could also be added in some other quantity related to the position of the moving boundary $s(t)$, [3], but this case is not considered here.

Figure 6.4 shows plots of the L-curve for varying levels of random noise, where source points have been placed with $h = 2.5$ and final time point $T = 1$. We choose $\lambda = 10^{-6}$, which corresponds to the corner of the "L" in Figure 6.4, for all three noise levels.

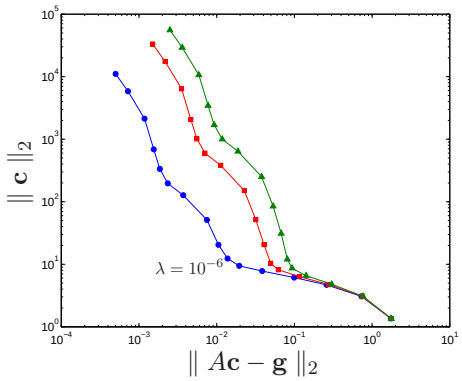

Figure 6.4: L-curve plots for $\delta = 1\%$ (●–●), $\delta = 3\%$ (■–■), $\delta = 5\%$ (▲–▲) when $K = 30$ and $M = 16$, for Example 1.

In this example we wish to recover the Dirichlet and Neumann boundary conditions along

the fixed boundary $x = 0$ given by

$$\begin{aligned} u(0,t) &= -1 + \exp\left(1 - \frac{1}{\sqrt{2}} + \frac{t}{2}\right), \qquad t \in [0,1]. \\ \frac{\partial u}{\partial x}(0,t) &= -\frac{1}{\sqrt{2}} \exp\left(1 - \frac{1}{\sqrt{2}} + \frac{t}{2}\right), \qquad t \in [0,1]. \end{aligned} \tag{6.38}$$

In Figures 6.5(a) and 6.5(b) the MFS approximations for $u(0,t)$ and $u_x(0,t)$, respectively, are plotted for three different noise levels $\delta \in \{1,3,5\}\%$ and, when compared, they match well with the exact solution, however, the accuracy deteriorates as time increases, which is to be expected, because of the noise and as t increases so does the distance to the data points, [6, 34]. From these figures it can also be seen that, as expected, the heat flux is more difficult to estimate accurately than the boundary temperature; however, both plots show that the numerical solutions are stable and they become more accurate as the amount of noise δ decreases.

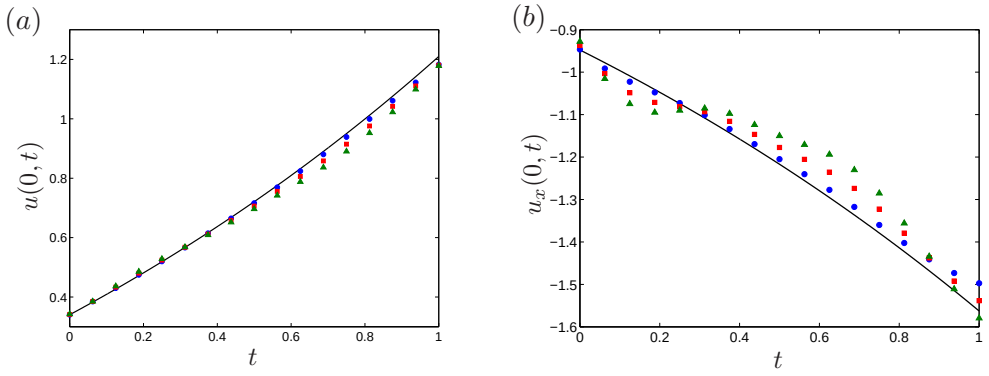

Figure 6.5: (a) The exact solution $u(0,t)$ (——) and the MFS approximation, and (b) the exact normal derivative $u_x(0,t)$ (——) and the MFS approximation. Both plots for $\delta = 1\%$ (●), $\delta = 3\%$ (■) and $\delta = 5\%$ (▲), and obtained with $h = 2.5$, $\lambda = 10^{-6}$, $K = 30$ and $M = 16$, for Example 1.

Figure 6.6(a) is a plot of the exact solution and the best and least accurate MFS approximations after ten trials for $\delta = 5\%$. Figure 6.6(b) is a three-dimensional plot of the absolute error for all $(x,t) \in D$ and noise level $\delta = 5\%$. The maximum error appears to be of the same order as the error we have introduced in the boundary data and occurs for large t close to $x = 0$ (which is to be expected).

Next, we increase the number of collocation and source points to see if we can improve the results obtained in Figures 6.5 and 6.6. We set $K = 60$ and $M = 31$ and note that these results have been generated using a new list of random variables for use with the random noise applied to the Neumann boundary condition (6.36). In Figure 6.7 we again plot the L-curve and find

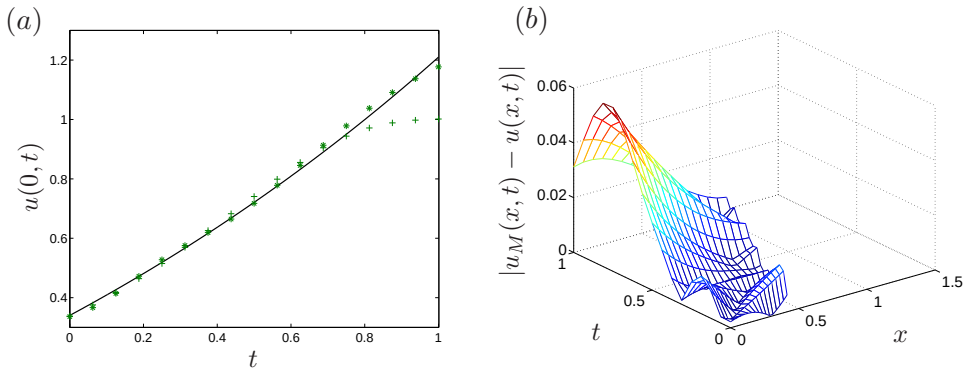

Figure 6.6: (a) The exact solution $u(0,t)$ (——) and the best ($*$) and least ($+$) accurate MFS approximations from ten different sets of noisy data with noise level $\delta = 5\%$, and (b) the absolute error for all $(x,t) \in D$ for noise level $\delta = 5\%$. Both plots obtained with $h = 2.5$, $\lambda = 10^{-6}$, $K = 30$ and $M = 16$, for Example 1.

that $\lambda = 10^{-5}$ is a reasonable choice for the regularization parameter. Figures 6.8 and 6.9 show that increasing the number of points improves the accuracy marginally. However, we feel that the results do not justify the extra computations required.

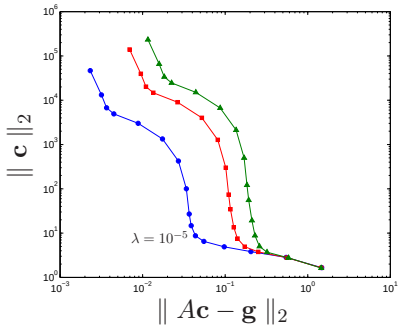

Figure 6.7: L-curve plots for $\delta = 1\%$ (—●—●—), $\delta = 3\%$ (—■—■—) and $\delta = 5\%$ (—▲—▲—) when $K = 60$ and $M = 31$, for Example 1.

In this example we have obtained accurate results with absolute errors of $\mathcal{O}(10^{-2})$, which was to be expected since the error on the Neumann boundary condition was of $\mathcal{O}(10^{-2})$.

Our numerical results compare similarly well with the results previously obtained in [89] using the mollification method. However, these latter results (presented in Figures 5 and 6 of [89]) were found quite oscillatory for small t, whilst Figures 6.5, 6.6, 6.8 and 6.9 show that our method is free of such oscillations. Our method also performs better than the unregularized decomposition method employed in [47] which is valid only for exact data and small times.

In this example, the moving boundary function $s(t)$ was a linear function (6.31); in the next example we will consider a non-linear boundary function.

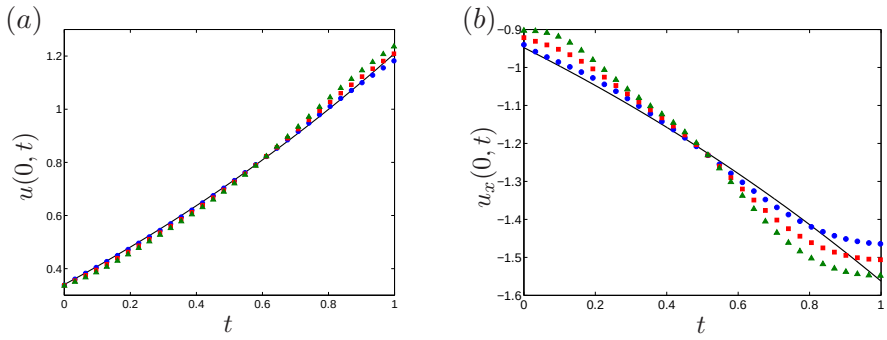

Figure 6.8: (a) The exact solution $u(0,t)$ (——) and the MFS approximation, and (b) the exact normal derivative $u_x(0,t)$ (——) and the MFS approximation. Both plots for $\delta = 1\%$ (●), $\delta = 3\%$ (■) and $\delta = 5\%$ (▲), and obtained with $h = 2.5$, $\lambda = 10^{-5}$, $K = 60$ and $M = 31$, for Example 1.

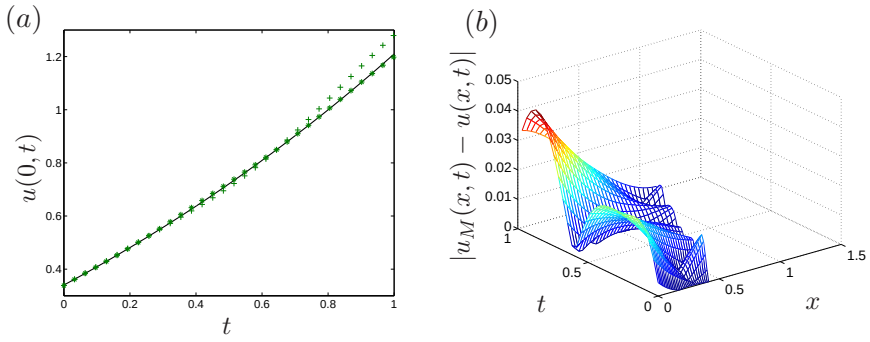

Figure 6.9: (a) The exact solution $u(0,t)$ (——) and the best (∗) and least (+) accurate MFS approximations from ten different sets of noisy data with noise level $\delta = 5\%$, and (b) the absolute error for all $(x,t) \in D$ for noise level $\delta = 5\%$. Both plots obtained with $h = 2.5$, $\lambda = 10^{-5}$, $K = 60$ and $M = 31$, for Example 1.

6.4.2 Example 2

In this example the moving boundary is given by the nonlinear function

$$s(t) = 2 - \sqrt{3 - 2t}, \quad t \in [0, T = 1]. \tag{6.39}$$

Again, we place source points symmetrically on the external boundaries $(-h, \tau)$, $\tau \in (-1, 1)$, $(s(\tau) + h, \tau)$, $\tau \in (0, 1)$ and $(s(-\tau) + h, \tau)$, $\tau \in (-1, 0)$, for an illustration see Figure 6.10. Although not illustrated, we note that similar results have been obtained if we change the reflected fictitious curve $(s(-\tau) + h, \tau)$ by the more natural curve $(s(0) + h, \tau)$ for $\tau \in (-1, 0)$.

We take the exact solution given by

$$u(x,t) \;=\; -\frac{x^2}{2} + 2x - \frac{1}{2} - t, \quad (x,t) \in [0, s(t)] \times [0, 1]. \tag{6.40}$$

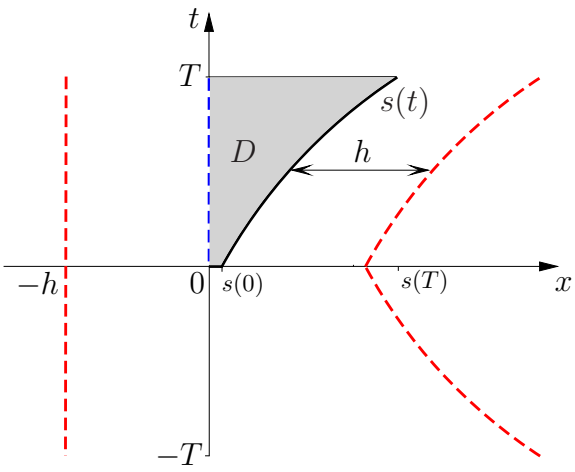

Figure 6.10: Particularisation of Figure 6.2 for $s(t)$ given by the equation (6.39) in Example 2.

From this, we generate the following initial and boundary conditions:

$$u(x,0) = -\frac{x^2}{2} + 2x - \frac{1}{2}, \quad x \in [0, s(0)], \quad s(0) = 2 - \sqrt{3}, \tag{6.41}$$

$$u(s(t),t) = 0, \quad t \in (0,1], \tag{6.42}$$

$$\frac{\partial u}{\partial x}(s(t),t) = \sqrt{3-2t}, \quad t \in (0,1]. \tag{6.43}$$

Note that the Neumann boundary condition (6.43) is not given by the classical Stefan boundary condition (0.4), but rather by the more general boundary condition (0.10). In this case equation (6.30) is changed to

$$\frac{\partial u_M}{\partial x}(x_1^{(i)}, t_i) = \sqrt{3-2t_i}, \quad i = 0, \dots M.$$

Random additive noise simulating measurement errors to the Neumann data (6.43) has been included as:

$$u_x^\delta(s(t),t) = \sqrt{3-2t} + N(0,\sigma^2), \tag{6.44}$$

where, according to (6.37), $\sigma = \sqrt{3}\delta$.

Figure 6.11 shows plots of the L-curve for varying levels of random noise, where source points have been placed with $h = 2.5$ and final time point $T = 1$. We choose $\lambda = 10^{-5}$, which corresponds to the corner of the "L" in Figure 6.11, for all three noise levels.

In this example we wish to recover the Dirichlet and Neumann boundary conditions along

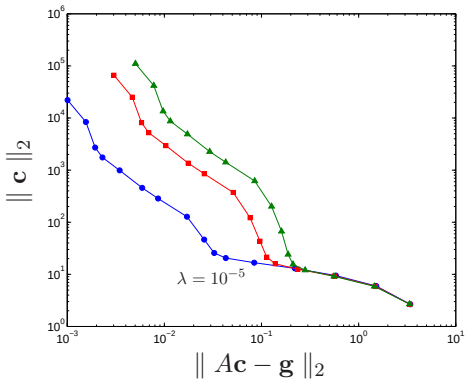

Figure 6.11: L-curve plots for $\delta = 1\%$ (●●), $\delta = 3\%$ (■■) and $\delta = 5\%$ (▲▲) when $K = 30$ and $M = 16$, for Example 2.

the fixed boundary $x = 0$ given by

$$u(0,t) = -\frac{1}{2} - t, \quad \frac{\partial u}{\partial x}(0,t) = 2, \quad t \in [0,1]. \tag{6.45}$$

Figure 6.12(a) is a plot of the exact solution and the MFS approximation for $u(0,t)$ for three different noise levels. Similar to Example 1, the approximation appears to get less accurate as t increases. In Figure 6.12(b) the MFS approximation of the normal derivative $u_x(0,t)$ shows an oscillatory behaviour; this could be attributed to the value of λ and the random noise levels.

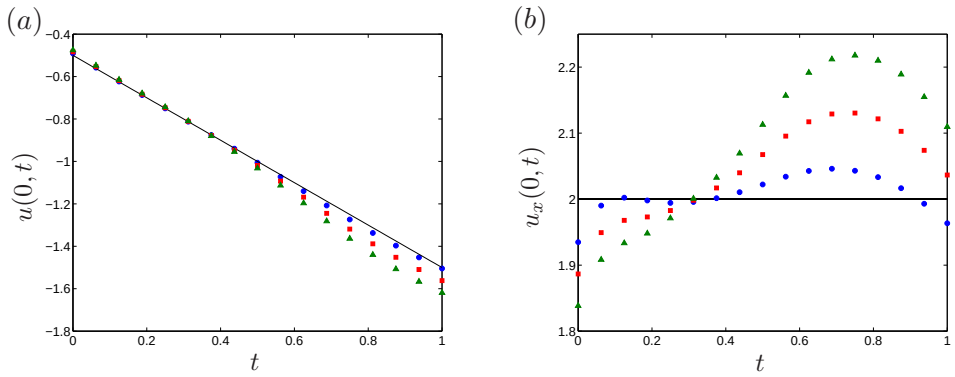

Figure 6.12: (a) The exact solution $u(0,t)$ (——) and the MFS approximation, and (b) the exact normal derivative $u_x(0,t)$ (——) and the MFS approximation. Both plots from noise values $\delta = 1\%$ (●), $\delta = 3\%$ (■) and $\delta = 5\%$ (▲), and obtained with $h = 2.5$, $\lambda = 10^{-5}$, $K = 30$ and $M = 16$, for Example 2.

Figure 6.13(a) is a plot of the exact solution and the best and least accurate MFS approximations after ten trials, and Figure 6.13(b) is a three-dimensional plot of the absolute error, and similar conclusions to those drawn for Figure 6.6 can be observed.

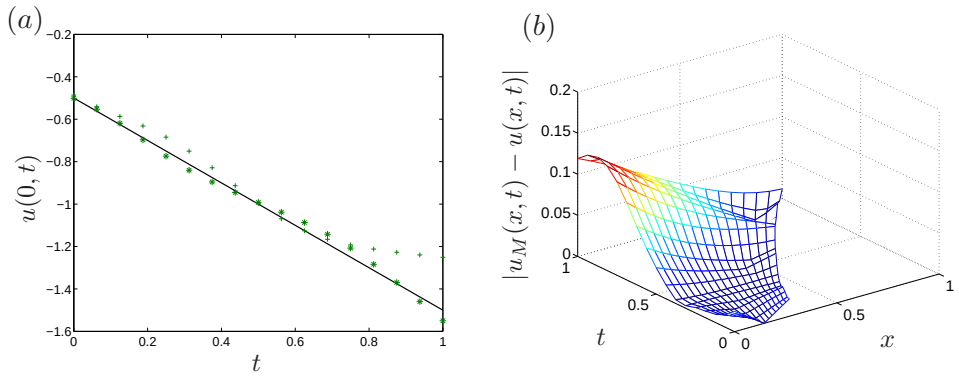

Figure 6.13: (a) The exact solution $u(0,t)$ (——) and the best ($*$) and least ($+$) accurate MFS approximations from ten different sets of noisy data with noise level $\delta = 5\%$, and (b) the absolute error for all $(x,t) \in D$ for noise level $\delta = 5\%$. Both plots obtained with $h = 2.5$, $\lambda = 10^{-5}$, $K = 30$ and $M = 16$, for Example 2.

6.4.3 Example 3 (two-dimensional case)

In this final example we investigate the inverse two-dimensional Stefan problem, with domain

$$D = \{(x,y,t) \in \mathbb{R}^3 |\ y \in (0,1), t \in (0,T], 0 < x < s(y,t)\}$$

where the free surface $s(y,t)$ is given, and

$$\Sigma = \{(x,y,t) \in \mathbb{R}^3 |\ y \in (0,1), t \in (0,T], x = s(y,t)\}.$$

The governing equation (1.1) is given by

$$\frac{\partial u}{\partial t}(x,y,t) \ = \ \Delta u(x,y,t), \quad (x,y,t) \in D.$$

This example is taken from [58], which applied the MFS to an inverse free boundary problem. We also take, for simplicity, dimensional physical quantities equal to unity. This inverse two-dimensional Stefan problem requires determining the unknown data displayed in Figure 6.14. The collocation and source points will be placed as shown in Figure 6.15.

The input data for this example is given by

$$s(y,t) = 1 - \frac{t}{10} - \frac{(y+1)^2}{20},$$

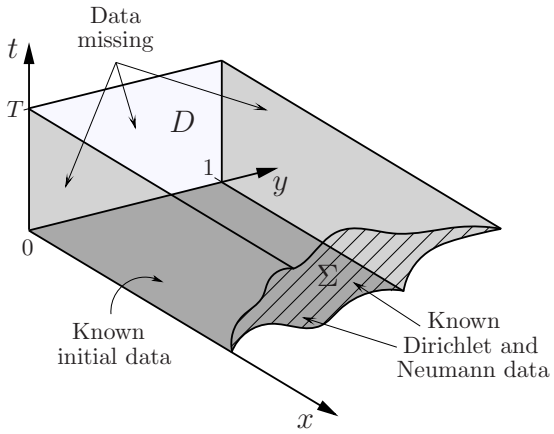

Figure 6.14: Representation of the two-dimensional inverse Stefan problem, with locations of the initial and boundary conditions.

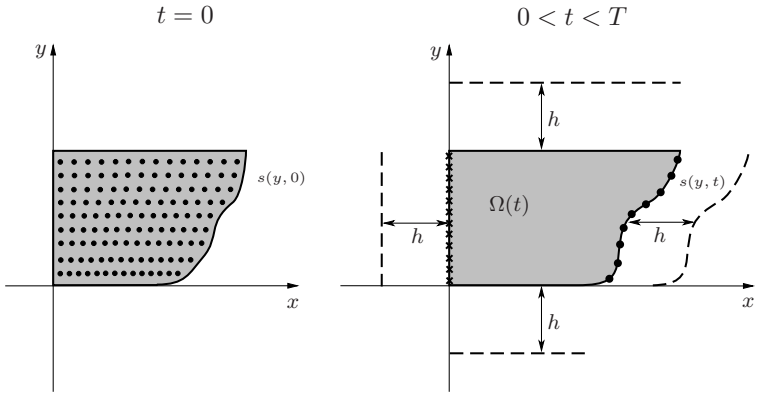

Figure 6.15: Representation of the placement of the collocation (•) and source points (---), along with the unknown boundary data (×).

$$u = g_1 = 1, \quad \frac{\partial u}{\partial \boldsymbol{\nu}} = g_2 = \sqrt{\left(\frac{y+1}{10}\right)^2 + 1}$$
$$\text{on } \overline{\Sigma} = \{(x, y, t) \in \mathbb{R}^3 |\ y \in [0, 1], t \in [0, 1], x = s(y, t)\}$$

$$u(x, y, 0) = f(x, y) = x + \frac{(y+1)^2}{20},$$
$$(x, y) \in \{(x, y) \in \mathbb{R}^2 |\ y \in (0, 1), x \in (0, s(y, 0))\}$$

where the inward unit normal to the free boundary Σ is given by

$$\boldsymbol{\nu} = \frac{1}{\sqrt{\left(\frac{y+1}{10}\right)^2 + 1}} \left(1, \frac{y+1}{10}\right).$$

The analytic solution of the problem above is given by

$$u(x,y,t) = \frac{(y+1)^2}{20} + x + \frac{t}{10},$$
$$(x,y,t) \in \{(x,y,t) \in \mathbb{R}^3 |\ y \in [0,1], t \in [0,1], 0 \leq x \leq s(y,t)\}.$$

The unknown data which is sought is given by

$$u(0,y,t) = \frac{(y+1)^2}{20} + \frac{t}{10}, \quad \frac{\partial u}{\partial x}(0,y,t) = 1, \quad (y,t) \in (0,1) \times (0,1),$$

$$u(x,0,t) = \frac{1}{20} + x + \frac{t}{10}, \quad t \in (0,1), 0 < x < s(0,t) = \frac{19}{20} - \frac{t}{10},$$
$$u(x,1,t) = \frac{1}{5} + x + \frac{t}{10}, \quad t \in (0,1), 0 < x < s(1,t) = \frac{4}{5} - \frac{t}{10}.$$

Noise is added to the flux data as

$$\frac{\partial u}{\partial \boldsymbol{\nu}} = g_2(1 + p\rho), \tag{6.46}$$

where p is the percentage of noise and ρ are random variables taken from a uniform distribution in $[-1, 1]$. In the numerics we have $h = 3$, 800 collocation points and 800 source points, and the Tikhonov regularization parameter is again given by λ.

In Figures 6.16 and 6.17 we have produced plots of the exact solution and exact normal derivative, the MFS approximation and absolute error on the fixed boundary $x = 0$. In Figures 6.16(b) and 6.17(b) we can see that the MFS has produced accurate and stable results, even for the normal derivative, which is usually difficult to recover. This accuracy can be seen in the plots of the absolute error provided in Figures 6.16(c) and 6.17(c).

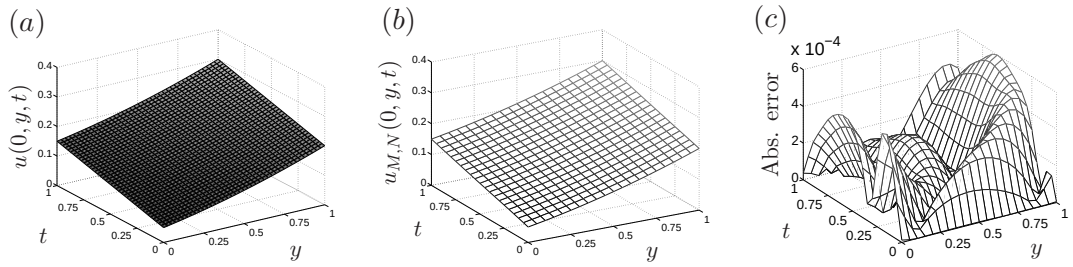

Figure 6.16: (a) The exact solution, (b) the MFS approximation, and (c) the absolute error for $(y,t) \in [0,1] \times [0,1]$, on the fixed boundary $x = 0$, obtained with $\lambda = 10^{-14}$ and $p = 0$, for Example 1.

Additionally, in Figures 6.18 and 6.19, we have attempted to reconstruct the data along the

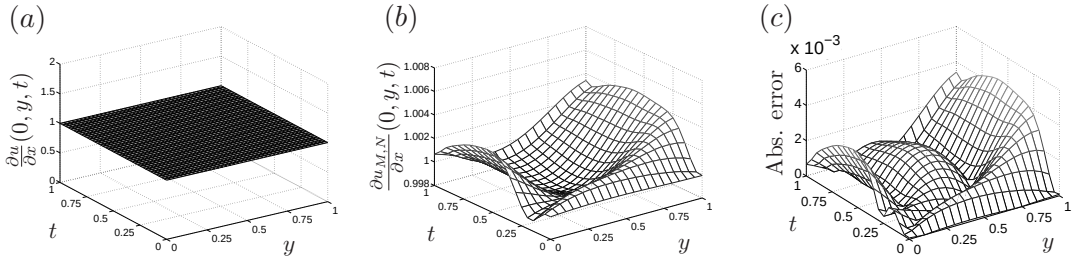

Figure 6.17: (a) The exact normal derivative, (b) the MFS approximation, and (c) the absolute error for $(y,t) \in [0,1] \times [0,1]$, on the fixed boundary $x = 0$, obtained with $\lambda = 10^{-14}$ and $p = 0$, for Example 1.

boundaries $y = 0$ and $y = 1$, and again the absolute error plots in Figures 6.18(c) and 6.19(c) show the very good accuracy of the proposed method.

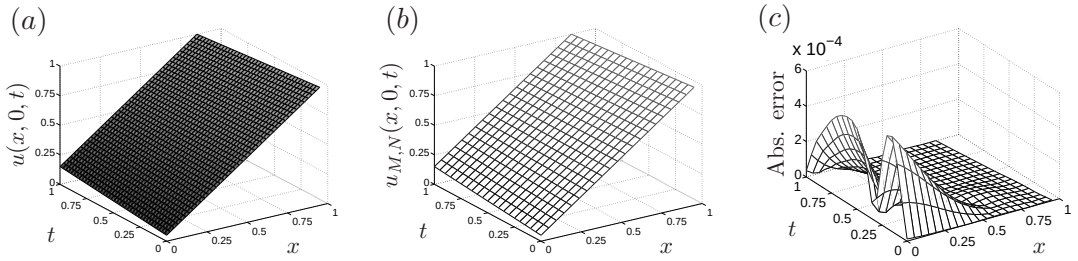

Figure 6.18: (a) The exact solution, (b) the MFS approximation, and (c) the absolute error for $(x,t) \in [0, s(0,t)] \times [0,1]$, on $y = 0$, obtained with $\lambda = 10^{-14}$ and $p = 0$, for Example 1.

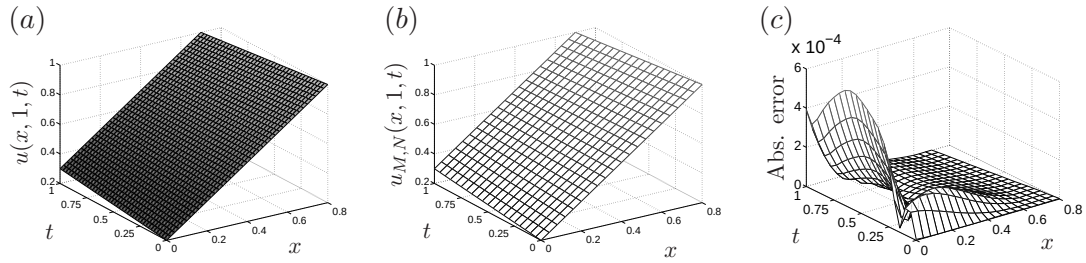

Figure 6.19: (a) The exact solution, (b) the MFS approximation, and (c) the absolute error for $(x,t) \in [0, s(1,t)] \times [0,1]$, on $y = 1$, obtained with $\lambda = 10^{-14}$ and $p = 0$, for Example 1.

Figures 6.20(a)-6.20(d) contain plots of the absolute error when we have added $p = 5\%$ noise to the flux data in (6.46). Reassuringly, these plots show that the error is of the same order as the amount of noise.

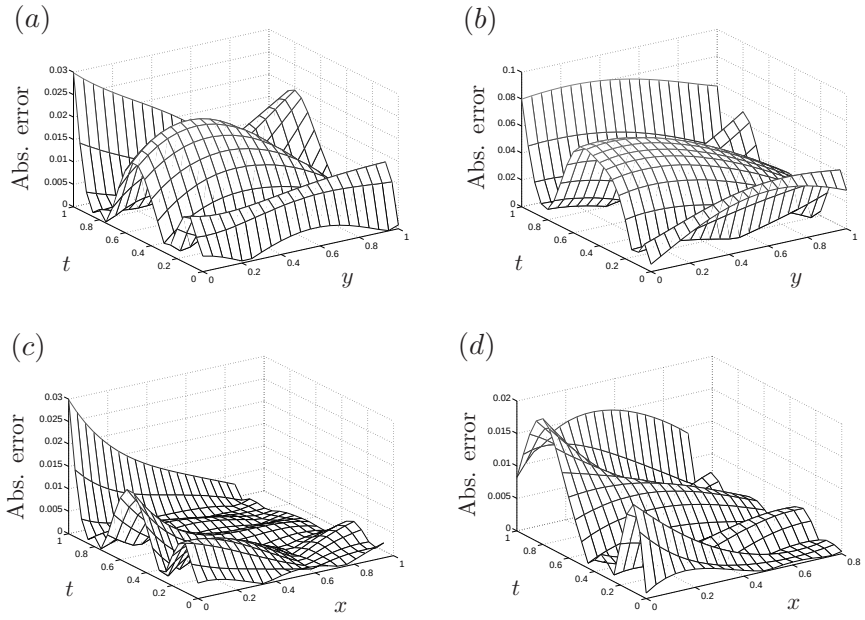

Figure 6.20: Plots of the absolute error of the exact solution and the MFS approximation on the boundaries (a) $x = 0$, (c) $y = 0$, (d) $y = 1$, and (b) the absolute error of the normal derivative and the MFS approximation on $x = 0$. Plots were obtained with $\lambda = 10^{-6}$ and $p = 5\%$, for Example 1.

6.5 Numerical investigation of a nonlinear algorithm for determining the optimal position of the source points

In this section we numerically reinvestigate Examples 1 and 2 from the previous section, with the focus being the determination of the optimal position of the pseudo-boundary. This method, which was described in Section 0.4 of the Introduction, is known as the dynamic MFS.

We again wish to find an MFS approximation which satisfies (6.1)–(6.4), but now the parameter h in (6.26) is unknown, as well as the constant coefficients in (6.24). We use the placement of points as described in (6.25)–(6.27) and, as before, collocate to obtain the system of equations (6.28)–(6.30).

The system we obtain is linear in $\mathbf{c}$, but nonlinear in h. We apply the nonlinear Tikhonov

regularization method based on minimizing the nonlinear regularized least squares functional

$$\begin{aligned}\mathcal{F}(\mathbf{c},h) = &\sum_{k=1}^{K}\left(\sum_{j=1}^{2}\sum_{m=1}^{2M} c_m^{(j)} F(x_0^{(k)},0;y_j(\tau_m),\tau_m) - u_0(x_0^{(k)})\right)^2 \\ &+ \sum_{i=0}^{M}\left(\sum_{j=1}^{2}\sum_{m=1}^{2M} c_m^{(j)} F(x_1^{(i)},t_i;y_j(\tau_m),\tau_m)\right)^2 \\ &+ \sum_{i=0}^{M}\left(\sum_{j=1}^{2}\sum_{m=1}^{2M} c_m^{(j)} \frac{\partial F}{\partial x}(x_1^{(i)},t_i;y_j(\tau_m),\tau_m) + s'(t_i)\right)^2 \\ &+ \lambda|\mathbf{c}|^2, \end{aligned} \tag{6.47}$$

where $\lambda > 0$ is the Tikhonov regularization parameter. The squared terms in the functional (6.47) represent the discrepancy between the approximation and the boundary data, in particular, the first line is for the initial data (6.28), the second and third lines are for the Stefan Dirichlet (6.29) and Neumann (6.30) boundary conditions on the moving boundary. The final term in the functional is the regularization term.

To accomplish the minimization of functional (6.47) we use the MATLAB toolbox *lsqnonlin*, which is designed to minimize a sum of squares of arbitrary differentiable functions. In MATLAB we use the command

```
Options=
   optimset('Display','iter','MaxFunEvals',MFE,'MaxIter',MI,'TolFun',TF,'TolX',TX)
[ch,resnorm,residual,exitflag,output]=
   lsqnonlin(@(ch) functionalcs(ch,parameters),ch0,LBch,UBch,Options)
```

where `ch` is the solution vector consisting of the vectors $\mathbf{c}$ and h (since only one vector can be an input for *lsqnonlin*). The initial guesses for h and $\mathbf{c}$, denoted by h_0 and $\mathbf{c}_0$, respectively, are arbitrary and contained in `ch0` above, and we shall test the convergence of different starting values. We note that we can place simple bounds on the variables, for example the bound $h > 0$ is allowed, therefore, we set the entry corresponding to h in `LBch` (vector representing the lower bound of `ch`) to be 0. Note that there are no upper bound restrictions (which would be imposed using `UBch`) and no bound is enforced for the solution vector $\mathbf{c}$. The options for *lsqnonlin* are set using the `optiset` function, and the default algorithm used is the trust-region-reflective algorithm based on the interior-reflective Newton method [30, 31]. In `optiset` we use `'Display', 'iter'` to display the output at each iteration, and the following termination options:

- 'MaxFunEvals' sets the maximum number of function evaluations to be MFE,
- 'MaxIter' sets the maximum number of iterations to be MI,
- 'TolFun' sets the termination tolerance on the function value to be TF (i.e. change in residual is less than TF),
- 'TolX' sets the termination tolerance on cs to be TX.

Unless otherwise stated we set MFE=50000, MI=500, and TF and TX to be set equal to (1.0e-08 $= 10^{-8}$).

6.5.1 Example 1

Example 1 in the previous section had a free boundary given by (6.31), an exact solution (6.32), boundary conditions (6.33)–(6.35), noise applied via (6.36), and the unknown data to be recovered at $x = 0$ given by (6.38). We analyse the behaviour of the program for different parameters, including the initial values for h and $\mathbf{c}$, and their impact on the RMSE and RRMSE (defined on page 81), the time taken, the number of iterations, the squared 2-norm residual of the functional, and the final 'optimized' value of h, which we denote by h_F.

In Table 6.1 we present the results (using $K = 30$ and $M = 16$ as in the previous section) when we have $h_0 = 0$ and $\mathbf{c}_0 = \mathbf{0}$, with poor results obtained (the value of h has not moved significantly from its initial designation). The near termination tolerance value of the squared 2-norm residual suggests that the approximation matches well with the boundary data, however, it appears that $h_F = 0.051243$ needs to be larger to achieve a better approximation.

Table 6.1

Initial parameters for Example 1					
h_0	$\mathbf{c}_0$	K	M	λ	δ
0	**0**	30	16	10^{-14}	0

Number of iterations	77
Time taken	24.9 seconds
(2-norm residual)^2	1.04653e-007
h_F	0.051243

Function	$u_M(0,t)$, $t \in [0,1]$	$\frac{\partial u_M}{\partial x}(0,t)$, $t \in [0,1]$	$u_M(x,t)$, $(x,t) \in \overline{D}$
RMSE	0.337567	0.502683	0.217088
RRMSE	0.428668	0.403599	0.506534

In Table 6.2 we try and improve the results in Table 6.1 by setting h_0 much larger, with $h_0 = 10$, and find that there is convergence towards a more accurate solution, with $h_F (= 2.4246)$

very close to the value used in the previous section ($h = 2.5$). It is possible that the code was unable to achieve a low squared 2-norm residual with such a large value of h, which would make sense since the system would be extremely ill-conditioned.

Table 6.2

Initial parameters for Example 1					
h_0	$\mathbf{c}_0$	K	M	λ	δ
10	**0**	30	16	10^{-14}	0

Number of iterations	28
Time taken	12.2 seconds
(2-norm residual)^2	5.59777e-007
h_F	2.4246

Function	$u_M(0,t),\ t \in [0,1]$	$\frac{\partial u_M}{\partial x}(0,t),\ t \in [0,1]$	$u_M(x,t),\ (x,t) \in \overline{D}$
RMSE	0.00462033	0.0181802	0.00170708
RRMSE	0.00586725	0.0145967	0.00398315

In Table 6.3 we again set $h_0 = 0$, but now set $\mathbf{c}_0 = \mathbf{100}$ (a vector with all values equal to 100), and we see that h converges to a reasonable value, but results are not quite as good as in Table 6.2. Additional trials were performed (not given here) and we found that convergence is not possible when h_0 is too large ($h_0 > 12$), and when both $h_0 > 5$ and $\mathbf{c}_0 > \mathbf{5}$, therefore care should be taken when choosing these initial values. An initial conclusion is that choosing $h_0 \in (1, 12)$ and $\mathbf{c}_0 = \mathbf{0}$, for this example, will result in reasonable approximations using *lsqnonlin*.

Table 6.3

Initial parameters for Example 1					
h_0	$\mathbf{c}_0$	K	M	λ	δ
0	**100**	30	16	10^{-14}	0

Number of iterations	59
Time taken	20.3 seconds
(2-norm residual)^2	5.4778e-006
h_F	3.64383

Function	$u_M(0,t),\ t \in [0,1]$	$\frac{\partial u_M}{\partial x}(0,t),\ t \in [0,1]$	$u_M(x,t),\ (x,t) \in \overline{D}$
RMSE	0.03756	0.150953	0.0137101
RRMSE	0.0476966	0.121198	0.0319901

In Figure 6.21 we present plots of the Dirichlet and Neumann approximations at $x = 0$, as well as the absolute error, using the input data given in Table 6.2. We now compare results using *lsqnonlin* to those produced in the previous section, when noise is applied to the boundary data.

In Table 6.4 we attempt the same experiment that was carried out for Table 6.1 (with $h_0 = 0$ and $\mathbf{c}_0 = \mathbf{0}$), and the same results are produced, as well as a large number of iterations and time required for convergence. Therefore, in Table 6.5 we set $h_0 = 10$ again (with what appears to be a reasonable choice for h_F) and better results are produced, suggesting a pattern, that will

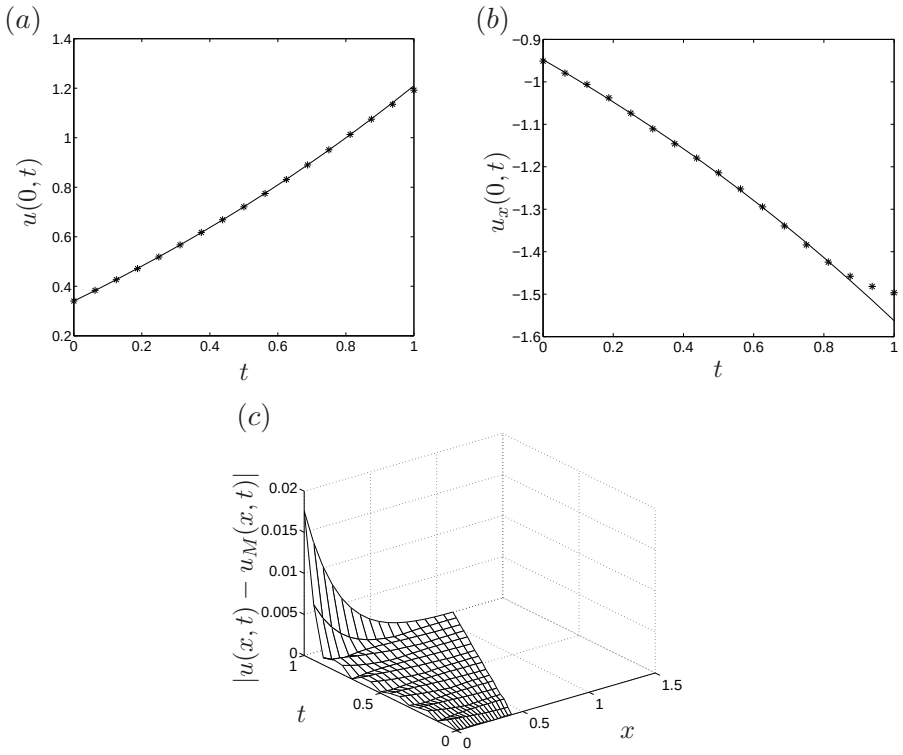

Figure 6.21: (a) The exact solution $u(0,t)$ (——) and the MFS approximation, (b) the exact normal derivative $u_x(0,t)$ (——) and the MFS approximation, and the absolute error over the entire domain $(x,t) \in D$. Obtained with the data presented in Table 6.2, for Example 2.

be tested for Example 2 as well.

Table 6.4

Initial parameters for Example 1					
h_0	$\mathbf{c}_0$	K	M	λ	δ
0	**0**	30	16	10^{-6}	0.05

Number of iterations	258
Time taken	81.2 seconds
(2-norm residual)^2	2.15398e-005
h_F	0.12069

Function	$u_M(0,t),\ t \in [0,1]$	$\frac{\partial u_M}{\partial x}(0,t),\ t \in [0,1]$	$u_M(x,t),\ (x,t) \in \overline{D}$
RMSE	0.453958	0.681334	0.228165
RRMSE	0.57647	0.547035	0.532381

Table 6.5

Initial parameters for Example 1					
h_0	$\mathbf{c}_0$	K	M	λ	δ
10	**0**	30	16	10^{-6}	0.05

Number of iterations	75
Time taken	24.2 seconds
(2-norm residual)^2	0.000270466
h_F	1.45264

Function	$u_M(0,t),\ t \in [0,1]$	$\frac{\partial u_M}{\partial x}(0,t),\ t \in [0,1]$	$u_M(x,t),\ (x,t) \in \overline{D}$
RMSE	0.0449813	0.147323	0.0220566
RRMSE	0.0571206	0.118284	0.0514651

In Figure 6.21 we present plots for numerous noise levels, and we see that results are stable. In fact, the h that *lsqnonlin* converged to was similar (including the time taken (26-28 seconds) and the number of iterations (79-86 iterations)) in all the different levels of noise, which is reassuring.

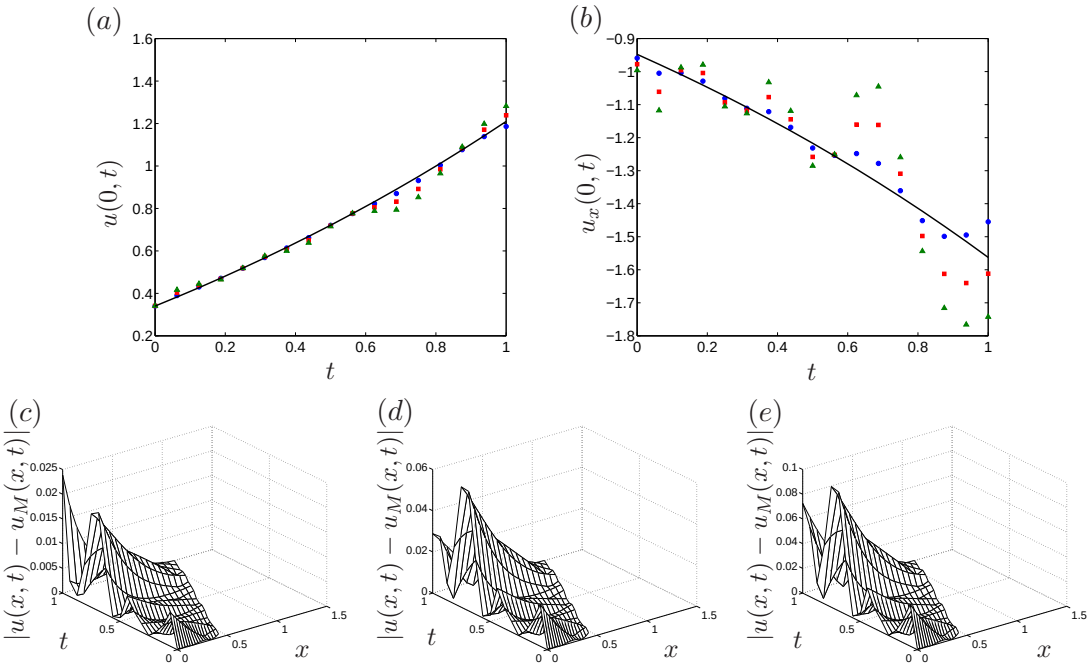

Figure 6.22: (a) The exact solution $u(0,t)$ (——) and the MFS approximation, (b) the exact normal derivative $u_x(0,t)$ (——) and the MFS approximation, and (c), (d) and (e) are absolute error plots over the entire domain $(x,t) \in D$. Plots have been obtained with the data presented in Table 6.5 except for $\delta = 0.01$ (c,●) when $h_F = 1.44227$, $\delta = 0.03$ (d,■) when $h_F = 1.45381$, and $\delta = 0.05$ (e,▲) when $h_F = 1.45264$, for Example 1.

We now increase the number of collocation and source points, and compare results with those in Figures 6.8. Similar conclusions on the choice of initial parameters are drawn in this case to when we had $K = 30$ and $M = 16$, and the approximation when no noise is applied appear accurate. However, results are not as good when noise is applied to the boundary data when $K = 60$ and $M = 31$, see Table 6.6 and Figure 6.23, compared with Figure 6.8, and it seems the final value of h found using *lsqnonlin* is lower than the value of h chosen in the previous section.

6.5.2 Example 2

For Example 2 we had a free boundary given by (6.39), an exact solution (6.40), boundary conditions (6.41)–(6.43), noise applied via (6.44), and the unknown data to be recovered at

Table 6.6

Initial parameters for Example 1					
h_0	$\mathbf{c}_0$	K	M	λ	δ
10	**0**	60	31	10^{-5}	0.05

Number of iterations	89
Time taken	137.6 seconds
(2-norm residual)^2	0.00106695
h_F	1.04589

Function	$u_M(0,t),\ t \in [0,1]$	$\frac{\partial u_M}{\partial x}(0,t),\ t \in [0,1]$	$u_M(x,t),\ (x,t) \in \overline{D}$
RMSE	0.0863473	0.288282	0.035601
RRMSE	0.10965	0.231458	0.0830685

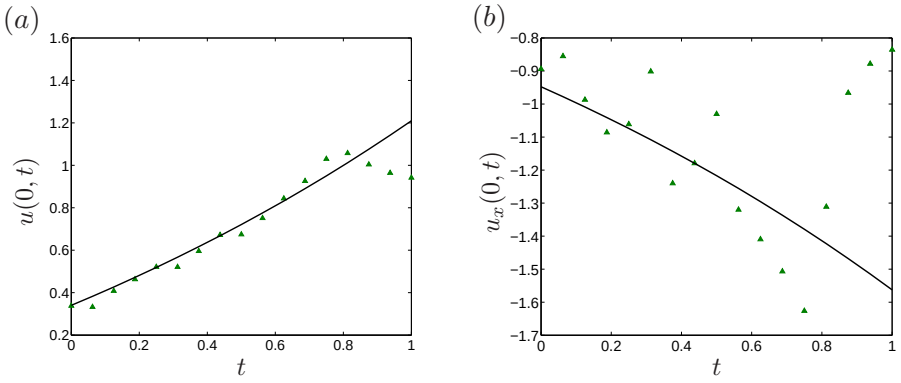

Figure 6.23: (a) The exact solution $u(0,t)$ (——) and the MFS approximation and (b) the exact normal derivative $u_x(0,t)$ (——) and the MFS approximation. Plots have been obtained with the data presented in Table 6.6, for Example 1.

$x = 0$ given by (6.45). Testing revealed the same behaviour as Example 1, therefore, in Table 6.7 and Figure 6.24, we only present results when $h_0 = 10$ and $\mathbf{c} = \mathbf{0}$, and are almost as good as those in Figures 6.12 and 6.13, except the Neumann data in Figure 6.24(b) seems a little more unstable.

Table 6.7

Initial parameters for Example 2					
h_0	$\mathbf{c}_0$	K	M	λ	δ
10	**0**	30	16	10^{-6}	0.05

Number of iterations	57
Time taken	19.2 seconds
(2-norm residual)^2	0.0126814
h_F	1.96353

Function	$u_M(0,t),\ t \in [0,1]$	$\frac{\partial u_M}{\partial x}(0,t),\ t \in [0,1]$	$u_M(x,t),\ (x,t) \in \overline{D}$
RMSE	0.0764337	0.255653	0.0398766
RRMSE	0.0730846	0.127827	0.0684787

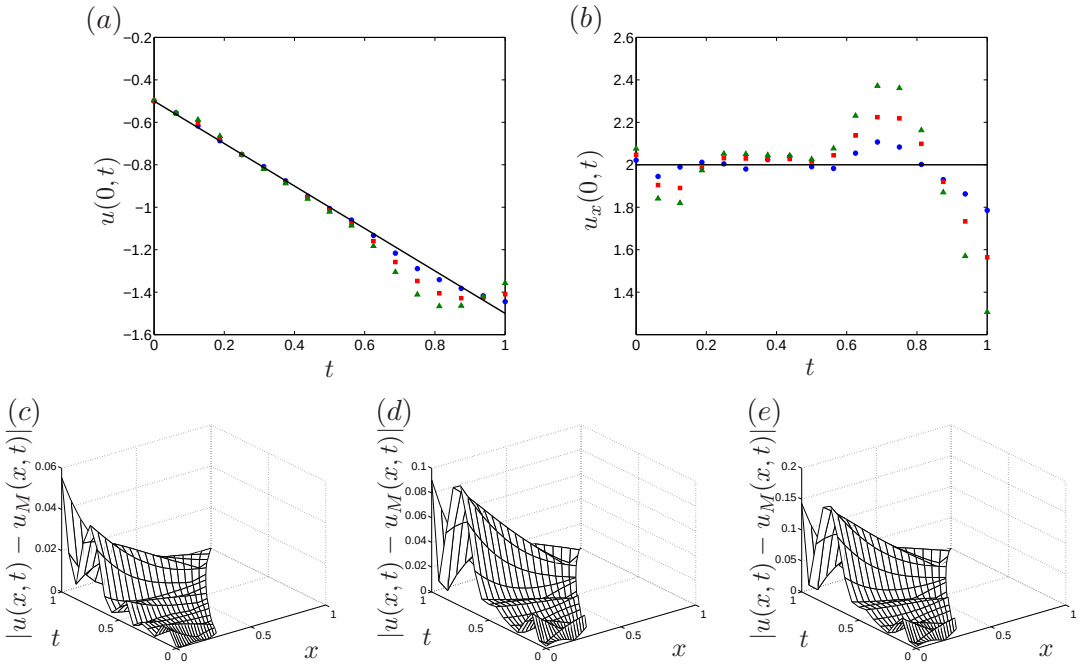

Figure 6.24: (a) The exact solution $u(0,t)$ (——) and the MFS approximation, (b) the exact normal derivative $u_x(0,t)$ (——) and the MFS approximation, and (c), (d) and (e) are absolute error plots over the entire domain $(x,t) \in D$. Plots have been obtained with the data presented in Table 6.5 except for $\delta = 0.01$ (c,●) when $h_F = 1.44024$, $\delta = 0.03$ (d,■) when $h_F = 1.95649$, and $\delta = 0.05$ (e,▲) when $h_F = 1.96353$, for Example 2.

6.6 Summary of Chapter 6

In this chapter an MFS was applied to the one and two-dimensional inverse Stefan problems and linear independence and denseness results were proved for linear combinations of the fundamental solution to the heat equation, based on work produced in [72] and [27]. Two examples were also presented, exploring the effect on the approximations obtained when the number of collocation and source points are increased, along with noise being applied to the Neumann boundary condition.

The regularized MFS approximations obtained in the three examples show good accuracy in comparison with the available exact solutions, and are stable even for relatively high noise levels being applied to the boundary input data. In Examples 1 and 2 changing the noise level did not seem to change the shape of the approximation, except for t close to the final time T and for the approximation of the flux along the fixed boundary, with errors usually of the same order as those applied in the input data. In the first part of Example 1 we chose $K = 30$ and $M = 16$ (corresponding to 64 collocation and source points), and in the second part increased

these to $K = 60$ and $M = 31$ (corresponding to 124 collocation and source points), and noted there was a slight increase in accuracy, however, it was not enough to warrant the increase in computations.

In Section 6.5 we applied the dynamic MFS to Examples 1 and 2 in Section 6.4, which involved trying to find an optimal position for the source points. In our study we only changed the parameter h (a scalar) in (6.26), although it is possible to treat all the source point positions as unknowns, and h then becomes a vector of unknowns, however, the computation time increases. We found that results were not as good as when we found h by trial and error, but results were mostly stable, and this method is a viable alternative.

At the end of this thesis we provide the MATLAB code (for the one-dimensional case) that was used to generate the computations in this chapter.

In the next chapter, the MFS is applied to the inverse Cauchy-Stefan problem in one-dimension, which will require determining both the Dirichlet and Neumann boundary conditions on the fixed boundary as well as the data on the initial base.

CHAPTER 7

THE ONE-DIMENSIONAL INVERSE CAUCHY-STEFAN PROBLEM

7.1 Introduction

In this chapter we extend the MFS to the inverse Cauchy-Stefan problem, where the initial condition as well as the Dirichlet and Neumann boundary conditions are missing, and have to be determined simultaneously. As was pointed out in [17, 52], the knowledge of the initial condition is in theory not needed for the recovery of the boundary data in the inverse Stefan problem. The inverse problem of simultaneously finding the initial data and boundary data is termed the inverse Cauchy-Stefan problem. We compare the numerical reconstructions with those obtained when the initial condition was specified.

For the inverse Cauchy-Stefan problem we wish to determine a solution that satisfies the heat equation (6.1), and the boundary conditions (6.3) and (6.4) (or the more general boundary conditions (0.9) and (0.10)). The aim is then to reconstruct the initial value of the solution $t = 0$, as well as its value and the normal derivative at the fixed stationary boundary $x = 0$, i.e. to find $u_0(x)$, $f(t)$ and $g(t)$ with

$$u(x,0) = u_0(x), \quad x \in [0, s(0)], \tag{7.1}$$

$$u(0,t) = f(t), \quad t \in (0,T], \tag{7.2}$$

$$\frac{\partial u}{\partial x}(0,t) = g(t), \quad t \in (0,T]. \tag{7.3}$$

If $s(0) = 0$ then the initial condition (7.1) is not involved. For existence, uniqueness and a formal solution see section 0.2.7 of the Introduction.

For a visual representation of a general domain and possible location of collocation and

source points, see Figure 7.1.

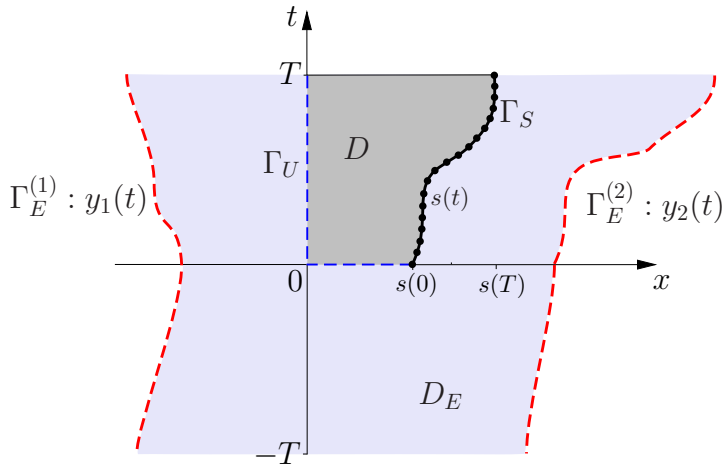

Figure 7.1: General representation of the domain D and boundary $\Gamma = \Gamma_U \cup \Gamma_S$, with unspecified initial and boundary conditions (---) on Γ_U, collocation points (-●-) on Γ_S, and source points (---) placed on $\Gamma_E = \Gamma_E^{(1)} \cup \Gamma_E^{(2)}$ external to the domain D.

7.2 The MFS for the inverse one-phase one-dimensional Cauchy-Stefan problem

We approximate problem (6.1), (6.3) and (6.4) using

$$u_M(x,t) \;=\; \sum_{j=1}^{2} \sum_{m=1}^{M+1} c_m^{(j)} F(x,t;y_j(\tau_m),\tau_m), \quad (x,t) \in \overline{D}, \tag{7.4}$$

where $M+1$ collocation points are placed on the lateral surface Γ_S (the moving interface) with the time points given by

$$t_i \;=\; iT/M, \quad x_i = s(t_i), \quad i = 0,\ldots,M.$$

There are then $M+1$ collocation points on Γ_S, where, according to (6.3) and (6.4), both the Dirichlet and Neumann conditions will be applied. Therefore, we place $2M+2$ source points external to the domain and equally distributed on the boundary Γ_E (see Figure 7.1). To specify

these source points let $(\tau_m)_{m=1,\ldots,M+1} \in (-T,T)$ be given by

$$\tau_m \;=\; -T + \frac{2m-5/2}{M}T, \quad m = 1,\ldots,M+1,$$

and the source points on Γ_E are then specified by defining $y_1(\tau_m)$ and $y_2(\tau_m)$ as

$$y_1(\tau_m) = -h \;\text{ and }\; y_2(\tau_m) = h + s(\tau_m H(\tau_m)), \quad m = 1,\ldots,M+1, \tag{7.5}$$

where $h > 0$ is a parameter to be prescribed. The location and number of collocation and source points are different from those in the previous chapter to adjust to the fact that the initial condition is not specified in the inverse Cauchy-Stefan problem.

We impose the Dirichlet and Neumann boundary conditions (6.3) and (6.4) to determine the constant coefficients $c_m^{(j)}$ in (7.4), and obtain the following system of linear algebraic equations

$$u_M(x_i,t_i) = h_1(t_i), \quad \frac{\partial u_M}{\partial x}(x_i,t_i) = h_2(t_i), \qquad i = 0,\ldots,M. \tag{7.6}$$

Again, we apply Tikhonov regularization, along with the L-curve criterion.

7.3 Numerical results

7.3.1 Example 1

We investigate Example 1 from Chapter 6 first; this time with the initial data not given. From Chapter 6 and [89] the free boundary is given by (6.31), the exact solution given by (6.32) and the boundary conditions (6.34) and (6.35). We place source points according to (7.5) on the external boundaries $(-h,\tau)$, $\tau \in (-1,1)$, $(s(\tau)+h,\tau)$, $\tau \in [0,1)$, and $(s(0)+h,\tau)$, $\tau \in (-1,0)$; for an illustration see Figure 7.2. Random additive noise is added to the Neumann data as in (6.36) and (6.37).

We wish to determine the initial condition at $t = 0$, as well as the Dirichlet and Neumann conditions along $x = 0$, given, respectively, by

$$u(x,0) \;=\; -1 + \exp\left(1 - \frac{1}{\sqrt{2}} - \frac{x}{\sqrt{2}}\right), \qquad x \in [0, s(0) = \sqrt{2}-1], \tag{7.7}$$

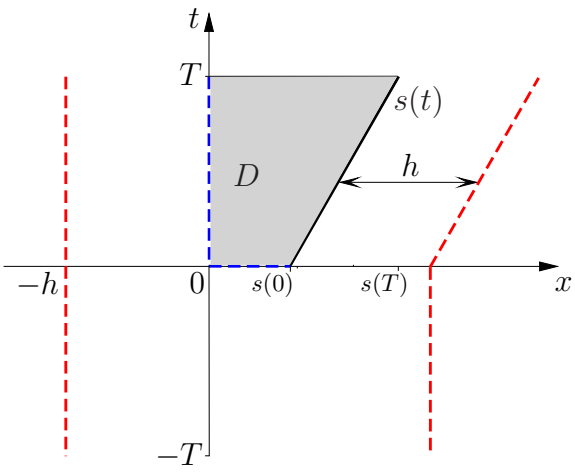

Figure 7.2: Particularisation of Figure 7.1 for $s(t)$ given by equation (6.31) in Example 1.

and (6.38).

Figure 7.3 contains plots of the L-curve for different noise levels, we choose $\lambda = 10^{-6}$ for $\delta = 1\%$, and $\lambda = 10^{-5}$ for $\delta = 3\%$ and 5%. Different values for the regularization parameter have been chosen for different noise levels to give the best possible results to reduce the error in the vector $\mathbf{c}$ and the oscillatory behaviour of the calculated approximation.

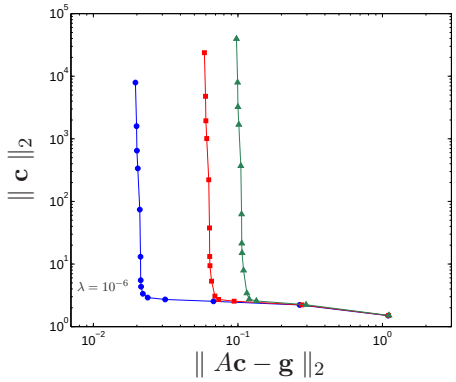

Figure 7.3: L-curve plots for $\delta = 1\%$ (—●—●—), $\delta = 3\%$ (—■—■—) and $\delta = 5\%$ (—▲—▲—) when $M = 30$ in (7.4), for Example 1.

Figures 7.4(a), 7.4(b), and 7.4(c) contain plots of the MFS approximations for the reconstructed data $u(0,t)$, $u_x(0,t)$, and $u(x,0)$, respectively, for noise levels $\delta \in \{1,3,5\}\%$. First, from Figure (7.4)(c) it can be seen that the numerical results for the initial data are in good agreement with the exact solution (7.7) and they are relatively insensitive to noise. On the other hand, and rather surprising, but in agreement with the theoretical results of [17], Figures

7.4(a) and 7.4(b) suggest that we have even better results considering the inverse Cauchy-Stefan problem compared to the inverse Stefan-problem in Chapter 6. There may be a few reasons for this, in particular, a different value of h and other values of the regularization parameter have been chosen, and different sets of random noise have been generated for this example compared to Example 1 of the previous chapter. Also, more source points have been placed in time, i.e. $M = 30$ in this example instead of $M = 16$ in Chapter 6. We note that it is difficult to perform a direct comparison with Example 1 of Chapter 6, because of the removal of the initial condition and, hence, a change in the number of collocation points, i.e. if $M = 16$ then the size of the matrix A in Chapter 6 is 64×64 and whilst in this chapter it is 34×34. However, at the end of this example we compare results when using similar parameters to those in Example 1 of Chapter 6.

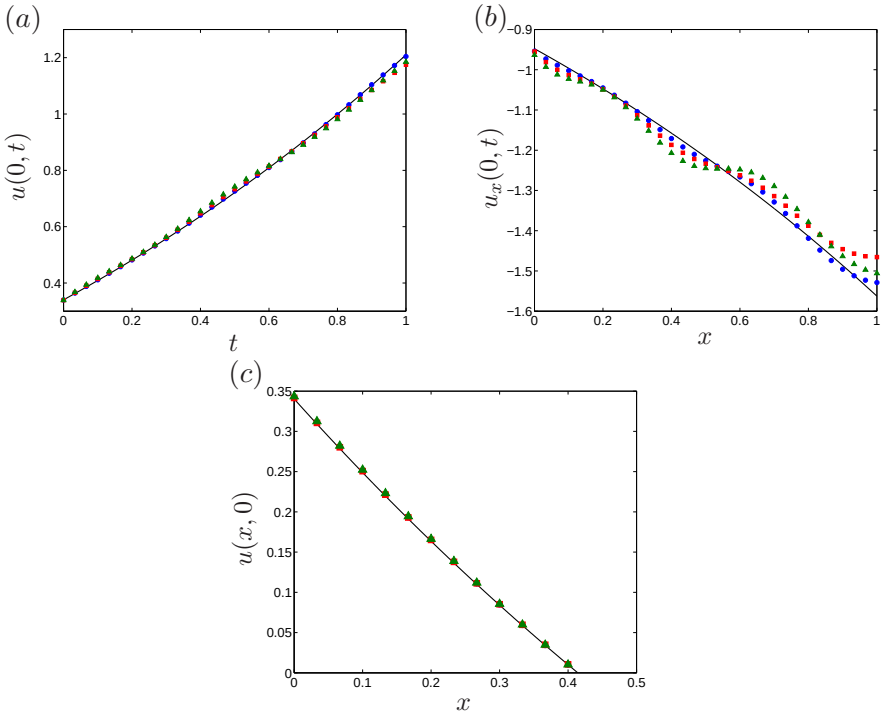

Figure 7.4: The exact solutions (——) and MFS approximations for: (a) $u(0,t)$, (b) $u_x(0,t)$, and (c) $u(x,0)$. All MFS approximations have been generated for noise levels $\delta = 1\%$ (●) with $\lambda = 10^{-6}$, $\delta = 3\%$ (■) with $\lambda = 10^{-5}$, and $\delta = 5\%$ (▲) with $\lambda = 10^{-5}$, and obtained with $h = 2$, and $M = 30$, for Example 1.

Figures 7.5(a) and 7.5(b) contain plots of the best and least accurate approximations after ten trials for the MFS approximation of $u(0,t)$ and $u(x,0)$, respectively. We see similar results to Chapter 6, with errors increasing as t increases. This is to be expected since if we invert the

Cauchy data (6.3) and (6.4) at $x = s(t)$ collected over the interval $[0, T]$, we cannot determine accurately the boundary temperature (7.2) and heat flux (7.3) at $x = 0$ for $t = T$, see [34].

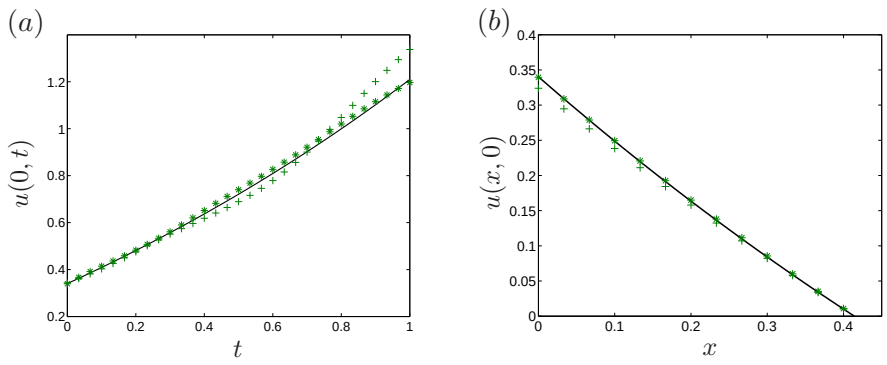

Figure 7.5: Plots of the exact solution (——) and the best ($*$) and least ($+$) accurate MFS approximations from ten different sets of noisy data with noise level $\delta = 5\%$ for (a) $u(0, t)$ and (b) $u(x, 0)$. Both plots obtained with $h = 2$, $M = 30$ and $\lambda = 10^{-5}$, for Example 1.

Figure 7.6(a) contains a plot of the exact solution, and 7.6(b) is a plot of the absolute error for the noise level $\delta = 5\%$. Even though the numerical solution is slightly oscillatory, the maximum error is relatively low, and of the same order as the noise level.

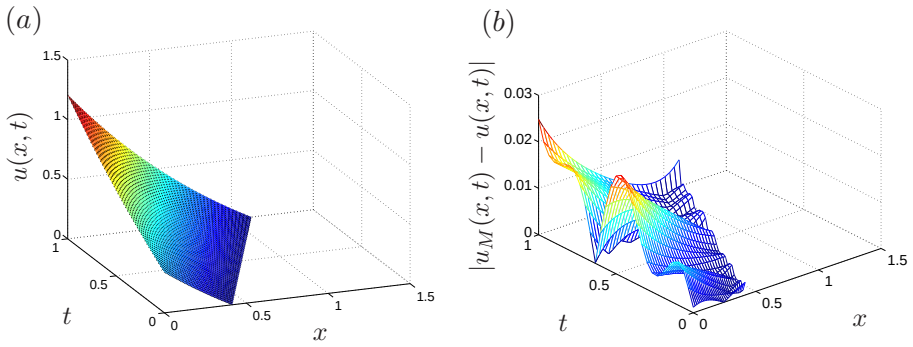

Figure 7.6: (a) The exact solution $u(x, t)$ for all $(x, t) \in D$ and (b) the absolute error for all $(x, t) \in D$ for noise level $\delta = 5\%$, obtained with $h = 2$, $\lambda = 10^{-5}$, $M = 30$, for Example 1.

The final two figures in this example are used for comparison with the results obtained in Example 1 of Chapter 6 (where the initial data (7.7) was given). Figures 7.7(a) and 7.7(b) differ only by the number of source and collocation points that have been placed in time (note that the matrix A is still square). Figure 7.7(a) suggests that a higher value of M in (7.4) is needed for determining accurately the initial condition, whereas Figure 7.7(b) gives a very similar result to Figure 6.9(b) in Chapter 6, where the same amount of random noise has been applied.

Note that the variables used in this example are not optimised. Thus, another value for h,

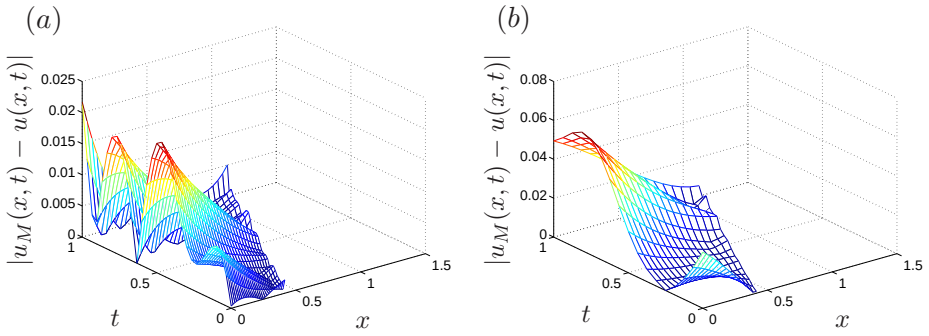

Figure 7.7: Plots of the the absolute error for all $(x,t) \in D$ for noise level $\delta = 5\%$ obtained with $h = 2.5$, $\lambda = 10^{-6}$ and (a) $M = 30$ and (b) $M = 16$, for Example 1.

changes in the number of points M placed in time and different choices for the regularization parameter λ may produce even better results.

7.3.2 Example 2

The following problem was investigated in [27], but with the direct Stefan problem being the focus of that paper. The free boundary in this example is given by the nonlinear function

$$s(t) = 2a\sqrt{t + t_0}, \quad t \in [0, T = 1], \tag{7.8}$$

where $t_0 = 0.162558$ and $a = 0.620063$. We shall place source points on the external boundaries $(-h, \tau)$, $\tau \in (-1, 1)$, $(s(\tau) + h, \tau)$, $\tau \in [0, 1)$ and $(s(0) + h, \tau)$, $\tau \in (-1, 0)$, for an illustration see Figure 7.8. We take the exact solution to the heat equation (6.1) given by

$$u(x,t) \;=\; 1 - \frac{\operatorname{erf}\Big(\frac{x}{2\sqrt{t+t_0}}\Big)}{\operatorname{erf}(a)}, \quad (x,t) \in [0, s(t)] \times [0, 1], \tag{7.9}$$

where

$$\operatorname{erf}(\xi) \;=\; \frac{2}{\sqrt{\pi}} \int_0^{\xi} e^{-\sigma^2}\, d\sigma \tag{7.10}$$

is the error function. From this we generate the following boundary conditions (6.3) and (6.4) on the free boundary:

$$u(s(t), t) \;=\; 0, \quad t \in (0, 1], \tag{7.11}$$

$$\frac{\partial u}{\partial x}(s(t), t) \;=\; -\frac{e^{-a^2}}{\sqrt{\pi(t + t_0)}\operatorname{erf}(a)}, \quad t \in (0, 1]. \tag{7.12}$$

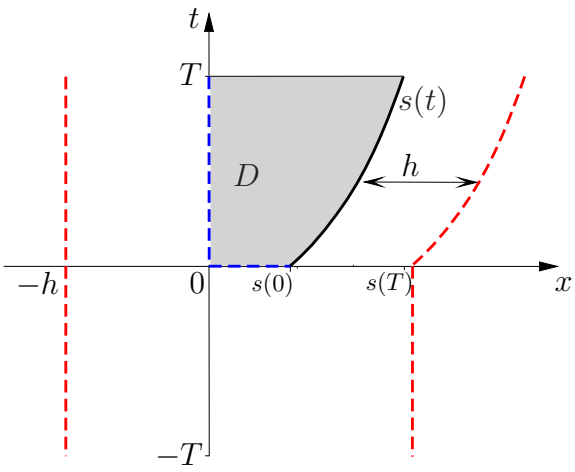

Figure 7.8: Particularisation of Figure 7.1 for $s(t)$ given by equation (7.8) in Example 2.

Noise is then added to the Neumann data (7.12) similar to (6.36)–(6.37).

We wish to determine the output data (7.1)–(7.3) given by

$$u(x,0) \;=\; u_0(x) \;=\; 1 - \frac{\operatorname{erf}\left(\frac{x}{2\sqrt{t_0}}\right)}{\operatorname{erf}(a)}, \quad x \in [0, s(0) = 2a\sqrt{t_0}\,], \tag{7.13}$$

$$\begin{cases} u(0,t) \;=\; f(t) \;=\; 1, & t \in [0,1], \\ \frac{\partial u}{\partial x}(0,t) \;=\; g(t) \;=\; -\frac{1}{\sqrt{\pi(t+t_0)}\,\operatorname{erf}(a)}, & t \in [0,1]. \end{cases} \tag{7.14}$$

Figure 7.9 contains plots of the L-curve for different noise levels; and we choose the regularization parameters $\lambda = 10^{-6}$ for $\delta = 1\%$, and $\lambda = 10^{-5}$ for $\delta = 3\%$ and 5%.

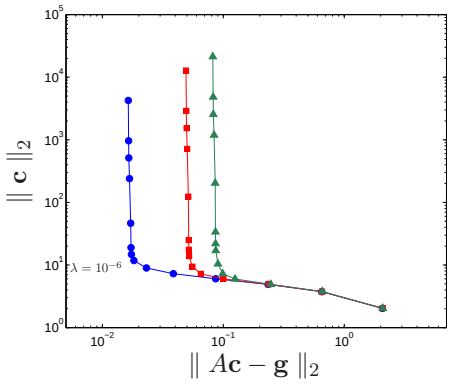

Figure 7.9: L-curve plots for $\delta = 1\%$ (●–●), $\delta = 3\%$ (■–■) and $\delta = 5\%$ (▲–▲) when $M = 30$, for Example 2.

In Figures 7.10(a), 7.10(b) and 7.10(c) the MFS approximations for the missing data $u(0,t)$,

$u_x(0,t)$, and $u(x,0)$, respectively, are plotted for noise levels $\delta \in \{1,3,5\}\%$. We obtain similar stable results as in the previous example. Also the initial condition, which is not supplied, is approximated with good accuracy.

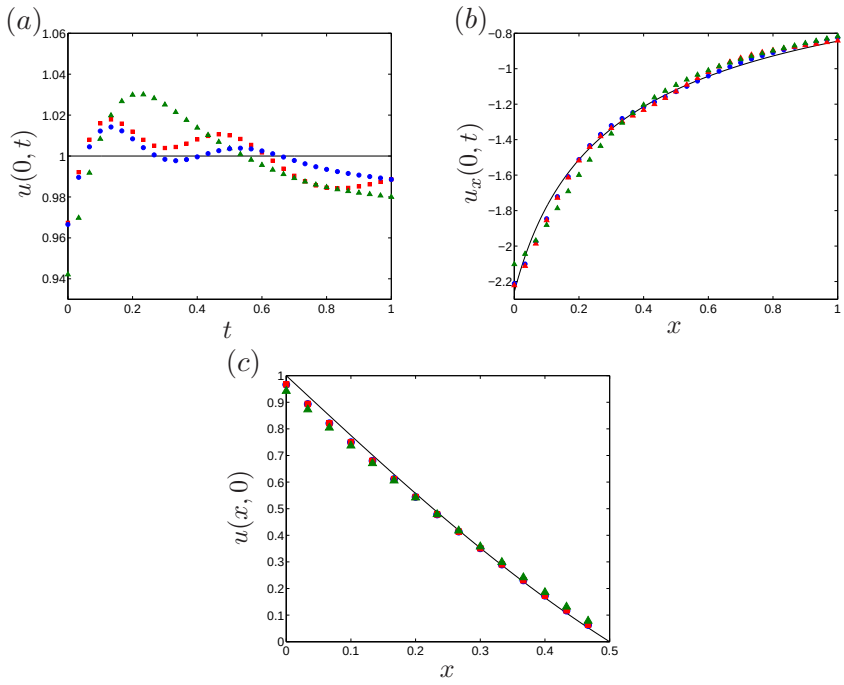

Figure 7.10: The exact solutions (——) and MFS approximations for: (a) $u(0,t)$, (b) $u_x(0,t)$, and (c) $u(x,0)$. All MFS approximations have been generated for noise levels $\delta = 1\%$ (●) with $\lambda = 10^{-6}$, $\delta = 3\%$ (■) with $\lambda = 10^{-5}$, and $\delta = 5\%$ (▲) with $\lambda = 10^{-5}$, and obtained with $h = 2$, and $M = 30$, for Example 2.

Figures 7.11(a) and 7.11(b) show the best and least accurate approximations after ten trials for the MFS approximation of $u(0,t)$ and $u(x,0)$, respectively, however, for one of the sets of random noise the maximum error is relatively large; this may be because the random data contains large values or a better choice of regularization parameter may be required with another plot of the L-curve for that data.

Finally, Figure 7.12(a) contains a plot of the exact solution, and we note that the solution has a relatively steep gradient which may affect the accuracy of the approximation at the initial and final time points. Figure 7.12(b) is a plot of the absolute error for the noise level $\delta = 5\%$, and the error is greatest at $t = 0$, where the solution has its largest gradient. The maximum error obtained, however, is again of the same order as the noise level. Example 2 displays similar features to those of Example 1 and it appears again that not supplying the initial condition has not impaired the results in any significant way.

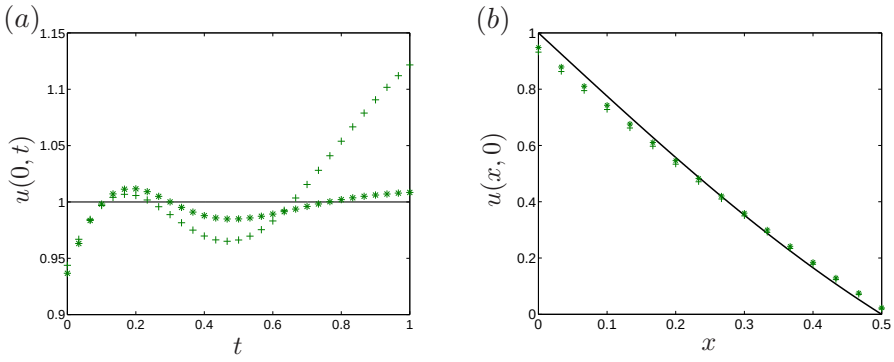

Figure 7.11: Plots of the exact solution (——) and the best (∗) and least (+) accurate MFS approximations from ten different sets of noisy data with noise level $\delta = 5\%$ for (a) $u(0,t)$ and (b) $u(x,0)$. Both plots obtained with $h = 2$, $M = 30$ and $\lambda = 10^{-4}$, for Example 2.

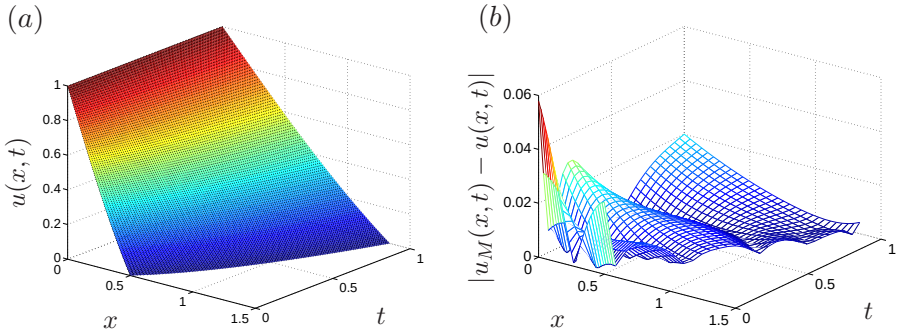

Figure 7.12: The exact solution $u(x,t)$ for all $(x,t) \in \overline{D}$ and (b) the absolute error for all $(x,t) \in D$ for noise level $\delta = 5\%$, obtained with $h = 2$, $\lambda = 10^{-5}$, $M = 30$, for Example 2.

7.3.3 Example 3

In this example the MFS is applied to a more difficult problem where we have a singularity in the solution at the initial point $(s(0), 0)$ of the closure of the moving boundary Γ_S. The problem we consider is a modification of Neumann's analytic solution; the unmodified Neumann's analytic solution was, for example, considered in [83]. The nonlinear function representing the free boundary is now given by

$$s(t) = 1 - 2b\sqrt{t}, \quad t \in [0, T = 1], \tag{7.15}$$

where $b = 1/4$. We place source points on the external boundaries $(-h, \tau)$, $\tau \in (-1, 1)$, $(s(\tau) + h, \tau)$, $\tau \in [0, 1)$ and $(s(0) + h, \tau)$, $\tau \in (-1, 0)$; for an illustration see Figure 7.13. We use the exact solution given by

$$u(x,t) \;=\; -1 + 2t + x^2 + \frac{\operatorname{erfc}\left(\frac{1-x}{2\sqrt{t}}\right)}{\operatorname{erfc}(b)}, \quad (x,t) \in [0, s(t)] \times [0, 1], \tag{7.16}$$

where $\operatorname{erfc}(\xi) = 1 - \operatorname{erf}(\xi)$ is the complementary error function. The solution u in (7.16) has a singularity located at $(s(0), 0) = (1, 0)$. From (7.16) we generate the following boundary conditions (6.3) and (6.4) on the free boundary:

$$u(s(t), t) = 2t + (1 - 2b\sqrt{t})^2, \quad t \in (0, 1], \tag{7.17}$$

$$\frac{\partial u}{\partial x}(s(t), t) = 2(1 - 2b\sqrt{t}) + \frac{e^{-b^2}}{\sqrt{\pi t}\operatorname{erfc}(b)}, \quad t \in (0, 1]. \tag{7.18}$$

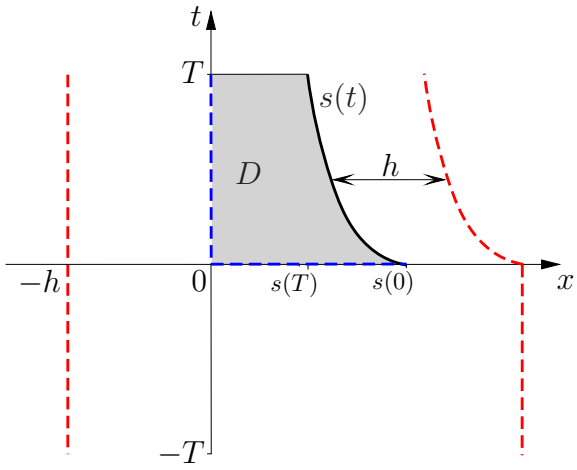

Figure 7.13: Particularisation of Figure 7.1 for $s(t)$ given by equation (7.15) in Example 3.

From (7.18) it can be seen that the heat flux on $x = s(t)$ blows up to infinity as t tends to zero. We wish to determine the output data (7.1)–(7.3) given by

$$u(x, 0) = u_0(x) = x^2 - 1, \quad x \in [0, s(0) = 1], \tag{7.19}$$

$$\begin{cases} u(0, t) = f(t) = -1 + 2t + \dfrac{\operatorname{erfc}\left(\frac{1}{2\sqrt{t}}\right)}{\operatorname{erfc}(b)}, & t \in [0, 1], \\ \dfrac{\partial u}{\partial x}(0, t) = g(t) = \dfrac{e^{-\frac{1}{4t}}}{\sqrt{\pi t}\operatorname{erfc}(b)}, & t \in [0, 1]. \end{cases} \tag{7.20}$$

Figures 7.14(a) and 7.14(b) contain plots of the MFS approximation of $u(0, t)$ and $u(x, 0)$, respectively, when noise has not been applied to the boundary data. In Figure 7.14(a) we can see that the error is largest for small t than elsewhere; this is due to the singularity at $(x, t) = (1, 0)$. Figure 7.14(b) confirms this, with results being unstable and oscillatory; different choices of λ, M and h did not improve the accuracy significantly.

To obtain better results we remove the singularity, as was suggested for different examples

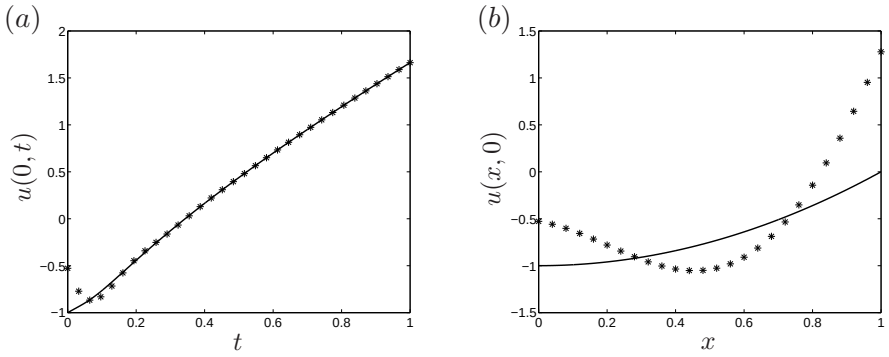

Figure 7.14: (a) The exact solution $u(0,t)$ (——) and the MFS approximation, and (b) the exact solution $u(x,0)$ (——) and the MFS approximation. Both plots obtained with $\lambda = 10^{-8}$, $h = 2$, and $M = 30$, for Example 3.

with singularities in [62, 79]. We define

$$u_{\text{sing}}(x,t) \;=\; \frac{\operatorname{erfc}\left(\frac{1-x}{2\sqrt{t}}\right)}{\operatorname{erfc}(b)}, \tag{7.21}$$

which satisfies the heat equation (6.1), and instead apply the MFS to the problem corresponding to the function $u^* = u - u_{\text{sing}}$. The function u^* is now regular and satisfies the heat equation (6.1), and the following modified Dirichlet and Neumann boundary conditions

$$u^*(s(t),t) \;=\; -1 + 2t + (1 - 2b\sqrt{t})^2, \qquad \frac{\partial u^*}{\partial x}(s(t),t) \;=\; 2(1 - 2b\sqrt{t}), \quad t \in (0,1]. \tag{7.22}$$

Figure 7.15 contains plots of the L-curve for different noise levels, and choose $\lambda = 10^{-6}$ for $\delta = 1\%$, $\lambda = 10^{-5}$ for $\delta = 3\%$, and $\lambda = 10^{-4}$ for $\delta = 5\%$.

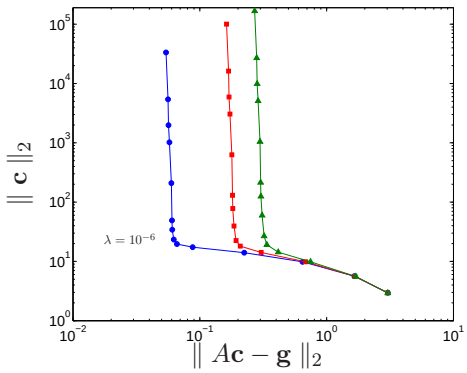

Figure 7.15: L-curve plots for $\delta = 1\%$ (—●—●—), $\delta = 3\%$ (—■—■—) and $\delta = 5\%$ (—▲—▲—) when $M = 30$, for Example 3.

In Figures 7.16(a), 7.16(b), and 7.16(c) the MFS approximations for the reconstructions of the data $u(0,t)$, $u(0,t)$, and $u(x,0)$, respectively, are plotted for noise levels $\delta \in \{1,3,5\}\%$, with the respective regularization parameters described above. By comparing Figures 7.14(a),(b) with Figures 7.16(a),(c), respectively, it can be seen that the removal of the singularity (7.21) has produced significantly more accurate results.

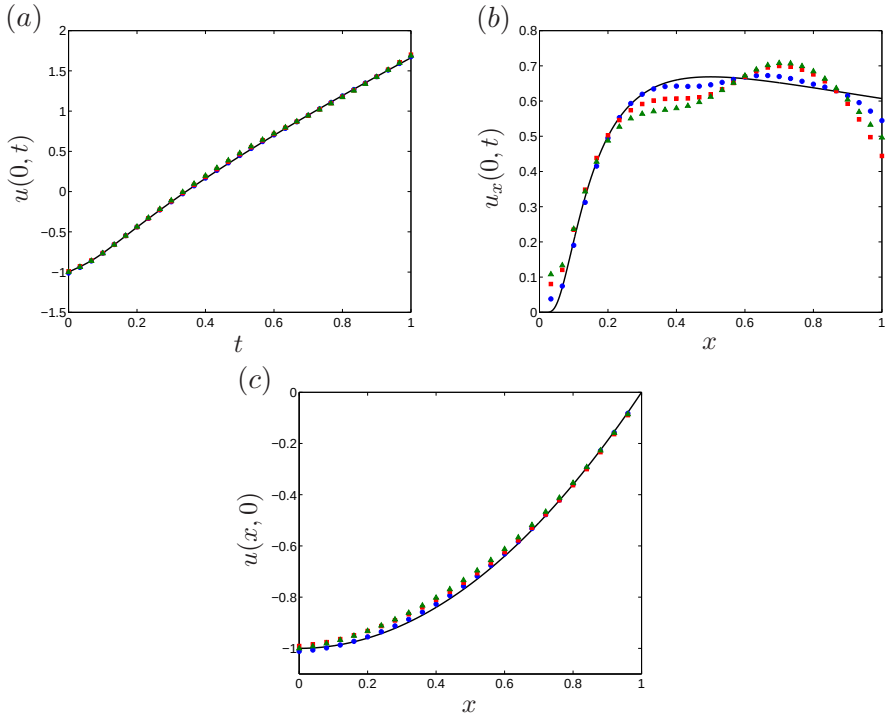

Figure 7.16: The exact solutions (——) and MFS approximations with singularity removed for: (a) $u(0,t)$, (b) $u_x(0,t)$, and (c) $u(x,0)$. All MFS approximations have been generated for noise levels $\delta = 1\%$ (●) with $\lambda = 10^{-6}$, $\delta = 3\%$ (■) with $\lambda = 10^{-5}$, and $\delta = 5\%$ (▲) with $\lambda = 10^{-5}$, and obtained with $h = 2$, and $M = 30$, for Example 3.

Figures 7.17(a) and 7.17(b) show the best and least accurate approximations after ten trials for the MFS approximation of $u(0,t)$ and $u(x,0)$, respectively, and results are very accurate for all the sets of random data applied to the Neumann boundary data (7.18).

Lastly, Figure 7.18(a) contains a plot of the exact solution, and Figure 7.18(b) is a plot of the absolute error for the noise level $\delta = 5\%$. Again, as in the previous examples, we obtain accurate results at the output comparable with the level of noise in the input data.

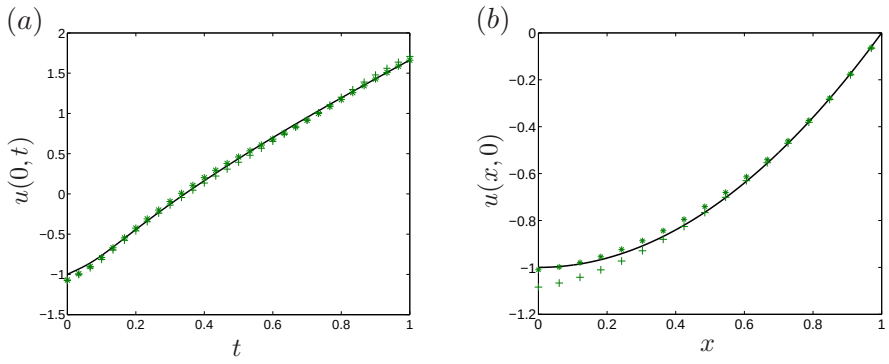

Figure 7.17: Plots of the exact solution (——) and the best (∗) and least (+) accurate MFS approximations from ten different sets of noisy data with noise level $\delta = 5\%$ for (a) $u(0,t)$ and (b) $u(x,0)$. Both plots obtained with singularity removed, $h = 2$, $M = 30$ and $\lambda = 10^{-4}$, for Example 3.

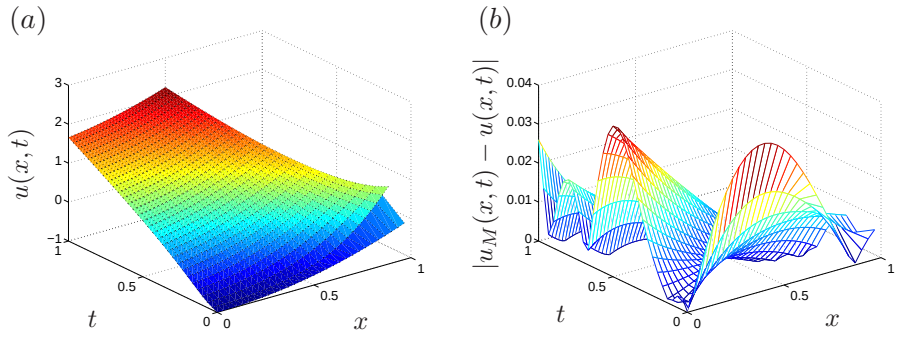

Figure 7.18: (a) The exact solution $u(x,t)$ for all $(x,t) \in \overline{D}$ and (b) the absolute error for all $(x,t) \in D$ for noise level $\delta = 5\%$, obtained with the singularity removed, $h = 2$, $\lambda = 10^{-5}$, $M = 30$, for Example 3.

7.4 Summary of Chapter 7

In this chapter an MFS was proposed and investigated for the one-dimensional inverse Cauchy-Stefan problem, where both initial and boundary data need to be reconstructed. Formal solutions to the non-characteristic Cauchy problem and the inverse Stefan problem were given, however, as was pointed out in Section 0.2.7 of the Introduction, the solutions are not viable numerically since both require infinite differentiation of often non-smooth functions.

The MFS was then applied to three numerical examples, showing that the MFS given in Chapter 6 to find boundary data can be adjusted to the inverse Cauchy-Stefan problem with some changes to the implementation. The first example considered is also studied in Chapter 6, and the results obtained here demonstrate that knowledge of the initial data is not crucial to get an accurate reconstruction. Comparing the results obtained in this chapter and Chapter

6 we find that the collocation points placed on the free boundary have a greater effect on the accuracy than collocation points placed on the initial base. The third and final example was a more challenging problem since the solution has a singularity located on the boundary of the solution domain. To obtain an accurate MFS approximation it was then necessary to first remove this singularity, as was suggested in [62, 79].

CHAPTER 8

AN INVERSE TWO-PHASE ONE-DIMENSIONAL NONLINEAR STEFAN PROBLEM

8.1 Introduction

This inverse two-phase one-dimensional nonlinear Stefan problem (0.11), (0.12), (0.2), (0.13)–(0.17) was formulated in Section 0.2.8 of the Introduction, and requires finding the triplet solution (u_1, u_2, s) (u_1 and u_2 the temperatures in the domains D_T^1 and D_T^2, respectively, and s the free boundary), and the Dirichlet (0.18a), Neumann (0.18b) and mass (0.18c) data at a fixed boundary $x = 0$. We note that the problem is nonlinear due to the free surface being considered unknown. We remind the reader that we wish to find an approximation which satisfies the heat equations

$$\frac{\partial u_1}{\partial t} = \alpha_1 \frac{\partial^2 u_1}{\partial x^2} \quad \text{in } 0 < x < s(t),\ 0 < t \leq T, \tag{0.12a}$$

$$\frac{\partial u_2}{\partial t} = \alpha_2 \frac{\partial^2 u_2}{\partial x^2} \quad \text{in } s(t) < x < l,\ 0 < t \leq T, \tag{0.12b}$$

and the initial and boundary data presented in Figure 8.1. A discussion on uniqueness can be found in the introduction.

8.2 The MFS for an inverse two-phase one-dimensional nonlinear Stefan problem

The fundamental solution of the heat equations (0.12a) and (0.12b) is given by

$$F_n(x,t;y,\tau) = \frac{H(t-\tau)}{\sqrt{4\alpha_n \pi (t-\tau)}} \exp\left(-\frac{(x-y)^2}{4\alpha_n (t-\tau)}\right), \quad n = 1, 2, \tag{8.1}$$

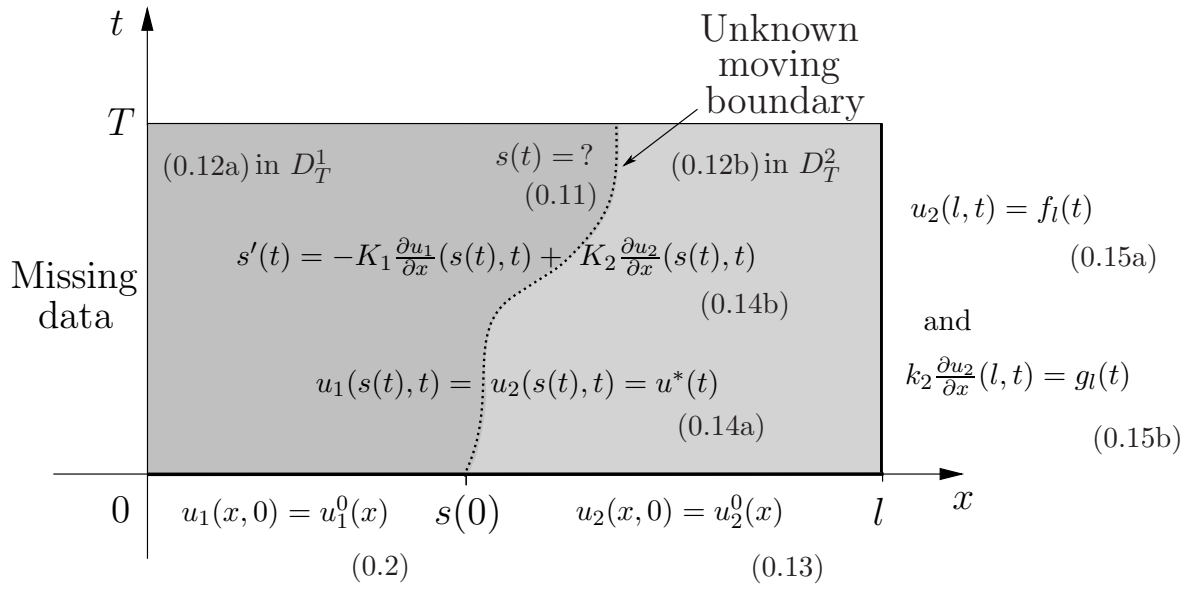

Figure 8.1: Representation of the initial and boundary conditions for the inverse two-phase one-dimensional nonlinear Stefan problem.

with spatial derivative

$$\frac{\partial F_n}{\partial x}(x,t;y,\tau) = -\frac{(x-y)H(t-\tau)}{2\sqrt{4\pi\alpha_n^3(t-\tau)^3}} \exp\left(-\frac{(x-y)^2}{4\alpha_n(t-\tau)}\right), \quad n = 1,2,$$

and the MFS approximation, in the domains D_T^1 and D_T^2, given by

$$u_n(x,t) = \sum_{j=1}^{M} c_j^{(n)} F_n(x,t;y_j^{(n)},\tau_j), \quad (x,t) \in \overline{D_T^n},\ n = 1,2. \tag{8.2}$$

In expression (8.2), the source points $(y_j^{(n)},\tau_j)$ for $j = 1,\ldots,M$ and $n = 1,2$, are located outside the solution domains $\overline{D_T^n}$ in the following way:

$$y_j^{(1)} = \begin{cases} -h, & j = 1,\ldots,2M_1 \\ h + s_{3M_1-j+1}, & j = 2M_1+1,\ldots,3M_1 \\ h + s_{4M_1-j+1}, & j = 3M_1+1,\ldots,4M_1 \end{cases} \tag{8.3a}$$

$$y_j^{(2)} = \begin{cases} -h + s_{M_1-j+1}, & j = 1,\ldots,M_1 \\ -h + s_{2M_1-j+1}, & j = M_1+1,\ldots,2M_1 \\ l + h, & j = 2M_1+1,\ldots,4M_1 \end{cases} \tag{8.3b}$$

where $h > 0$, as in previous chapters, is a preassigned parameter giving the distance between

the source points and the boundary, $M = 4M_1$,

$$\tau_j = \frac{(2j - 1 - 2M_1)T}{2M_1} \quad \text{for } j = 1, \ldots, 2M_1,$$

$\tau_j = \tau_{j-2M_1}$ for $j = 2M_1 + 1, \ldots, 4M_1$, and $s_j = s(\tau_{j+M_1})$ for $j = 1, \ldots, M_1$, see Figure 8.2 for a representation of the domains, boundaries and placement of source points.

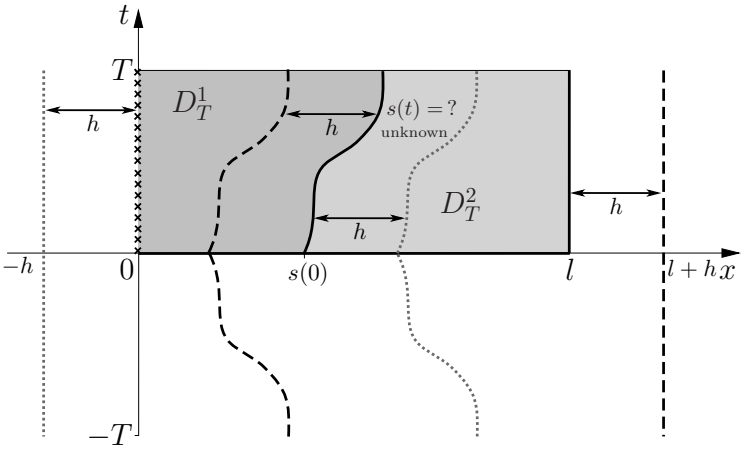

Figure 8.2: Representation of the two-phase problem with domains D_T^1 and D_T^2, unknown boundary data (×), and source points (·····) and (**---**) placed externally to the domains D_T^1 and D_T^2, respectively.

The $2M$ unknown coefficients $\mathbf{c} = (c_j^{(n)})_{j=1,\ldots,M}^{n=1,2}$ and the $M_1 = M/4$ discretization points $\mathbf{s} = (s_j)_{j=1,\ldots,M_1}$ of the unknown free boundary are determined by collocating conditions (0.2), (0.13)–(0.17), as described next. Let us select a uniform distribution of collocation points given by

$$t_0 = 0, \quad t_i = \frac{(2i-1)T}{2M_1} = |\tau_{2M_1-i+1}|, \quad i = 1, \ldots, M_1,$$

$$\tilde{t}_i = \frac{iT}{M_2} \quad \text{for } i = 0, \ldots, M_2,$$

$$x_1^{(k)} = \frac{ks(0)}{K+1}, \quad x_2^{(k)} = s(0) + \frac{k(l - s(0))}{K+1}, \quad k = 1, \ldots, K, \tag{8.4}$$

and collocate equations (0.2), (0.13)–(0.17) as follows:

$$u_n(x_n^{(k)}, 0) = u_n^0(x_n^{(k)}), \quad k = 1, \ldots, K, \ n = 1, 2, \tag{8.5}$$

$$u_1(s_i, t_i) = u_2(s_i, t_i) = u^*(t_i), \quad i = 0, \ldots, M_1, \tag{8.6}$$

$$-K_1 \frac{\partial u_1}{\partial x}(s_i, t_i) + K_2 \frac{\partial u_2}{\partial x}(s_i, t_i) = s'(t_i) = \begin{cases} \frac{s(t_1) - s(0)}{t_1}, & i = 1 \\ \frac{s(t_i) - s(t_{i-1})}{t_i - t_{i-1}}, & i = 2, \ldots, M_1 \end{cases}, \quad i = 1, \ldots, M_1, \tag{8.7}$$

$$u_2(l,\tilde{t}_i) = f_l(\tilde{t}_i), \quad k_2\frac{\partial u_2}{\partial x}(l,\tilde{t}_i) = g_l(\tilde{t}_i), \quad i = 0,\ldots,M_2. \tag{8.8}$$

We note that in (8.4) the time points t_i (time points for the free boundary) have been placed at the same times as the source points so that the position of the pseudoboundary can change with $s(t)$. In total, via (8.2), equations (8.5)–(8.8) form a system of $(2K+3M_1+2M_2+4)$ equations with $9M_1$ unknowns $(\mathbf{c},\mathbf{s})$. Therefore, we require $2M_2+2K+4 \geq 6M_1$, or $M_2+K+2 \geq 3M_1$. Note that this system is linear in $\mathbf{c}$, but it is nonlinear in $\mathbf{s}$. We apply the nonlinear Tikhonov regularization method based on minimizing the non-linear regularized least-squares functional

$$\begin{aligned}
\mathcal{F}(\mathbf{c},\mathbf{s}) = &\sum_{n=1}^{2}\sum_{k=1}^{K}\left(\sum_{j=1}^{M} c_j^{(n)} F_n(x_n^{(k)},0;y_j^{(n)},\tau_j) - u_n^0(x_n^{(k)})\right)^2 \\
&+ \sum_{i=0}^{M_1}\left(\sum_{j=1}^{M} c_j^{(1)} F_1(s_i,t_i;y_j^{(1)},\tau_j) - u^*(t_i)\right)^2 \\
&+ \sum_{i=0}^{M_1}\left(\sum_{j=1}^{M} c_j^{(2)} F_2(s_i,t_i;y_j^{(2)},\tau_j) - u^*(t_i)\right)^2 \\
&+ \sum_{i=1}^{M_1}\left(\sum_{j=1}^{M}\left[K_2 c_j^{(2)}\frac{\partial F_2}{\partial x}(s_i,t_i;y_j^{(2)},\tau_j) - K_1 c_j^{(1)}\frac{\partial F_1}{\partial x}(s_i,t_i;y_j^{(1)},\tau_j)\right] - s'(t_i)\right)^2 \\
&+ \sum_{i=0}^{M_2}\left(\sum_{j=1}^{M} c_j^{(2)} F_2(l,\tilde{t}_i;y_j^{(2)},\tau_j) - f_l(\tilde{t}_i)\right)^2 \\
&+ \sum_{i=0}^{M_2}\left(\sum_{j=1}^{M} c_j^{(2)} k_2\frac{\partial G_2}{\partial x}(l,\tilde{t}_i;y_j^{(2)},\tau_j) - g_l(\tilde{t}_i)\right)^2 \\
&+ \lambda_1|\mathbf{c}|^2 + \lambda_2|\mathbf{s}|^2,
\end{aligned} \tag{8.9}$$

where $\lambda_1,\lambda_2 \geq 0$ are regularization parameters to be prescribed according to some criterion, e.g. the L-surface. In the functional (8.9) the squared terms on the first six lines represent goodness-to-fit terms between the approximation and the boundary data. In particular, the first line represents the discrepancy between the approximations and the initial data (8.5), the second and third lines are for the error between the approximations and the Dirichlet (Stefan) condition (8.6) on the free boundary, and the fourth line is for the Neumann (Stefan) condition (8.7). The fifth and sixth lines correspond to the discrepancy between the approximation and Dirichlet and Neumann boundary data (8.8). The final two terms in the functional are the regularization terms for the unknown constants $\mathbf{c}$ and the free boundary $\mathbf{s}$.

The minimization of the functional (8.9) is performed using the MATLAB toolbox routine *lsqnonlin*, which was also used in Section 6.5 (where details for the toolbox and some of the variables can be found). In MATLAB we use the commands

```
Options=
   optimset('Display','iter','MaxFunEvals',MFE,'MaxIter',MI,'TolFun',TF,'TolX',TX)
[cs,resnorm,residual,exitflag,output]=
   lsqnonlin(@(cs) functionalcs(cs,parameters),cs0,LBcs,UBcs,Options)
```

where `cs` is the solution vector consisting of the vectors $\mathbf{c}$ and $\mathbf{s}$. The initial guess, given by `cs0` above, to start the iterative process is arbitrary and in this study we take $\mathbf{c}^0 = \mathbf{0}$ and $\mathbf{s}^0 = s(\mathbf{0})$. We note that the gradient does not need to be supplied by the user and the simple bounds on the variables

$$0 < s_i < l \quad \text{for } i = 1, \ldots, M_1 \tag{8.10}$$

are also allowed, therefore, we set the entries corresponding to s_i in `LBcs` (vector representing the lower bound of `cs`) and `UBcs` (vector representing the upper bound of `cs`) to be 0 and l, respectively. No bound is enforced for the solution vector $\mathbf{c}$. Unless otherwise stated we set `MFE=10000`, `MI=1000`, and `TF` and `TX` to be their default values (`1.0e-06` $= 10^{-6}$).

8.3 Numerical results

In the first two examples we have analytical solutions available and we also investigate for one of them the case when the initial conditions (0.2) and (0.13) are not given. In the third example, an analytical solution is not available.

8.3.1 Example 1

Let us take $T = 1$, $l = 2$, $\alpha_1 = 2$, $k_1 = 1$, $\alpha_2 = 1$, $k_2 = 2$, $K_1 = 1$, $K_2 = 2$. We also choose in equations (0.2) and (0.13) the initial conditions

$$u_1(x, 0) = u_1^0(x) = 2\left[\exp\left(\frac{0.5 - x}{2}\right) - 1\right], \quad x \in [0, s(0) = 0.5] \tag{8.11a}$$

$$u_2(x, 0) = u_2^0(x) = \exp(0.5 - x) - 1, \quad x \in [0.5 = s(0), l = 2]. \tag{8.11b}$$

We also take $u^*(t) = 0$ such that (0.14a) becomes

$$u_1(s(t), t) = u_2(s(t), t) = 0, \quad t \in (0, T = 1], \tag{8.12}$$

and take the Cauchy boundary conditions (0.15a) and (0.15b) specified by

$$u_2(2, t) = f_l(t) = \exp(t - 1.5) - 1, \quad t \in [0, T = 1], \tag{8.13a}$$

$$k_2 \frac{\partial u_2}{\partial x}(2, t) = g_l(t) = -2 \exp(t - 1.5), \quad t \in [0, T = 1]. \tag{8.13b}$$

Then the analytical solution of the inverse Stefan problem (0.11), (0.12), (0.14), (8.11)–(8.13) is given by

$$u_1(x, t) = 2 \left(\exp\left(\frac{t + 0.5 - x}{2} \right) - 1 \right), \quad (x, t) \in \overline{D_T^1}, \tag{8.14a}$$

$$u_2(x, t) = \exp(t + 0.5 - x) - 1, \quad (x, t) \in \overline{D_T^2}, \tag{8.14b}$$

$$s(t) = t + 0.5, \quad t \in [0, T = 1], \tag{8.14c}$$

which can be verified by direct substitution.

Of particular interest is to recover the missing data at the inaccessible hostile boundary $x = 0$, given by the boundary temperature

$$u_1(0, t) = f_0(t) = 2 \left(\exp\left(\frac{t + 0.5}{2} \right) - 1 \right), \quad t \in [0, T = 1] \tag{8.15a}$$

the heat flux

$$-k_1 \frac{\partial u_1}{\partial x}(0, t) = g_0(t) = \exp\left(\frac{t + 0.5}{2} \right), \quad t \in [0, T = 1] \tag{8.15b}$$

and the mass

$$\int_0^{s(t)} u_1(x, t)\, dx = E(t) = 2 \left(2 \exp\left(\frac{t + 0.5}{2} \right) - t - 2.5 \right), \quad t \in [0, T = 1]. \tag{8.15c}$$

If we consider the boundary temperature (8.13a) at $x = l = 2$ as the quantity which is physically measured in practice, whilst the heat flux (8.13b) is prescribed, in order to take into account the uncertainties in the measurement we add to it random additive noise generated as

$$u_2^\delta(1, t) = f_l^\delta(t) = f_l(t) + N(0, \sigma^2), \tag{8.16}$$

where $N(0,\sigma^2)$ represents the normal distribution with mean zero and standard deviation σ which is taken to be

$$\sigma = \delta \times \max_{t\in[0,1]} |f_l(t)|, \tag{8.17}$$

where δ is the percentage of noise.

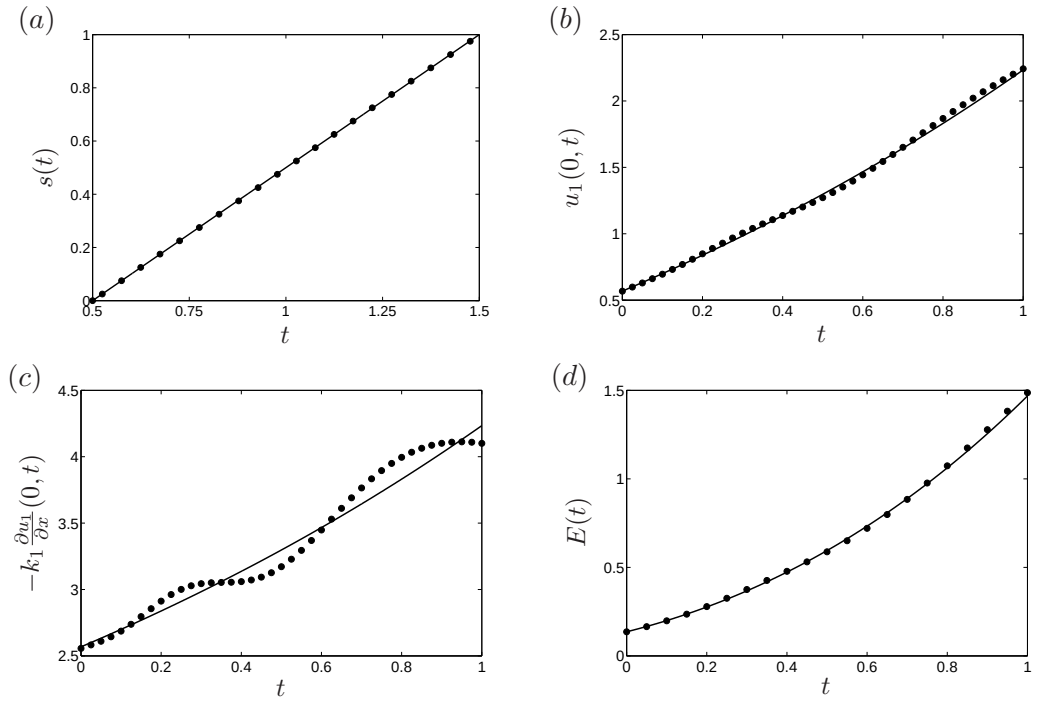

Figure 8.3: The numerical (•••) MFS solutions for: (a) $s(t)$, (b) $u_1(0,t)$, (c) $-k_1\frac{\partial u_1}{\partial x}(0,t)$, and (d) $E(t)$, as a function of $t \in [0,1]$, obtained with $M_1 = 20$, $M_2 = 40$, $K = 40$, $h = 3$ for no noise $\delta = 0$ and $\lambda_1 = \lambda_2 = 0$ in comparison with the exact (——) solutions (8.14c), 8.15(a)–(c), respectively, for Example 1.

Figures 8.3(a)–(d) show the numerical MFS solutions for $s(t)$, $u_1(0,t)$, $-k_1\frac{\partial u_1}{\partial x}(0,t)$ and $E(t)$, as a function of $t \in [0,1]$, obtained with $M_1 = 20$, $M_2 = 40$, $K = 40$, i.e. $2K+3M_1+2M_2+4 = 224$ equations with $9M_1 = 180$ unknowns, for no noise, i.e. $\delta = 0$, and $\lambda_1 = \lambda_2 = 0$, in comparison with the exact solutions (8.14c), (8.15a)–(8.15c), respectively. In calculating the mass (0.18c) using the MFS the following integral is needed:

$$\int_0^{s(t)} F_1(x,t;y,\tau)\,dx = \frac{H(t-\tau)}{2}\left[\operatorname{erf}\left(\frac{s(t)-y}{\sqrt{4\alpha_1(t-\tau)}}\right) - \operatorname{erf}\left(\frac{-y}{\sqrt{4\alpha_1(t-\tau)}}\right)\right], \tag{8.18}$$

where erf is the error function.

Figures 8.4(a)–(d) show the same as Figures 8.3(a)–(d), but for $\delta = 1\%$ and $\delta = 5\%$ noise and $\lambda_1 = \lambda_2 = 10^{-6}$. In Figures 8.3 and 8.4 it can be seen that stable and accurate numerical

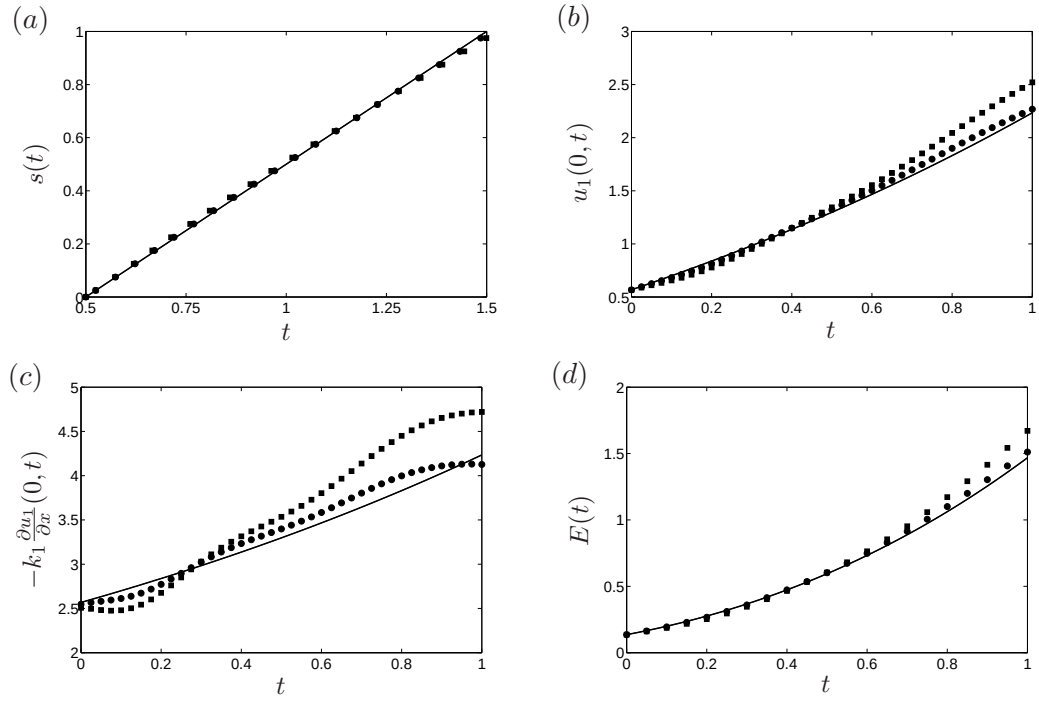

Figure 8.4: The numerical MFS solutions for: (a) $s(t)$, (b) $u_1(0,t)$, (c) $-k_1\frac{\partial u_1}{\partial x}(0,t)$, and (d) $E(t)$, as a function of $t \in [0,1]$, obtained with $M_1 = 10$, $M_2 = 20$, $K = 20$, $h = 3$ for $\delta = 1\%$ (•) and $\delta = 5\%$ (■) noise and $\lambda_1 = \lambda_2 = 10^{-6}$ in comparison with the exact (——) solutions (8.14c), 8.15(a)–(c), respectively, for Example 1.

MFS solutions are obtained.

Next, we look at the case when the initial conditions (8.11a) and (8.11b) are not imposed. In this case the first term in the right-hand side of (8.9) drops out, i.e. equation (8.5) is not imposed, such that, via (8.2), equations (8.6)–(8.8) form now a smaller system of $(3M_1+2M_2+4)$ equations with $9M_1$ unknowns, and we require $M_2 + 2 \geq 3M_1$.

Figures 8.5(a)–(d) and 8.6(a)–(d) represent the same quantities as Figures 8.3(a)–(d) and 8.4(a)–(d), respectively, but obtained with $M_1 = 10$ and $M_2 = 40$, i.e. $3M_1 + 2M_2 + 4 = 114$ equations with $9M_1 = 90$ unknowns. In addition, Figures 8.5(e) and 8.6(e) present the numerical results for the initial temperature

$$u(x,0) = \begin{cases} u_1(x,0), & x \in [0, s(0) = 0.5] \\ u_2(x,0), & x \in [0.5 = s(0), l = 2] \end{cases} \tag{8.19}$$

in comparison with the exact solution given by equations (8.11a) and (8.11b). Again, even when we are missing the initial data (0.2) and (0.13), we obtain accurate and stable results in all

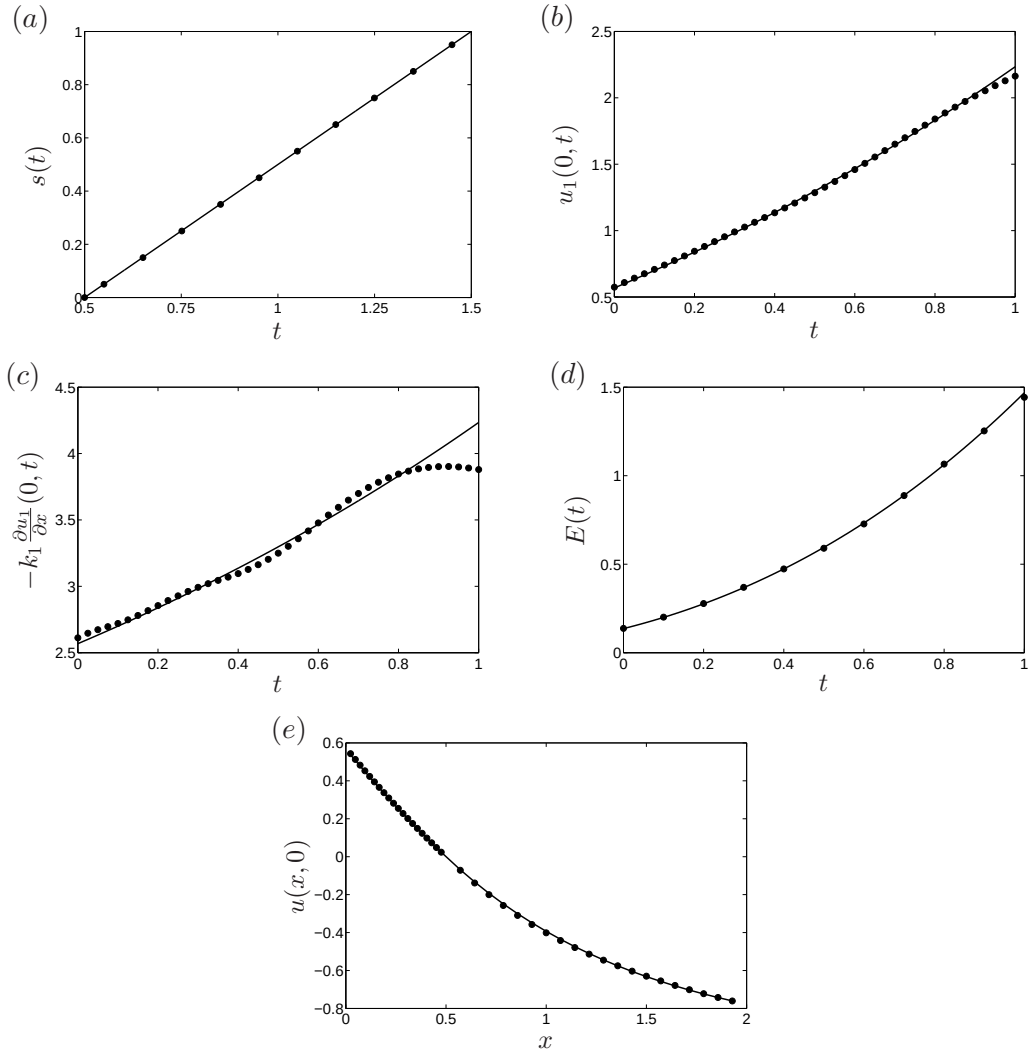

Figure 8.5: The numerical (●●●) MFS solutions for: (a) $s(t)$, (b) $u_1(0,t)$, (c) $-k_1\frac{\partial u_1}{\partial x}(0,t)$, and (d) $E(t)$, as a function of $t \in [0,1]$, and (e) $u(x,0)$ as a function of $x \in [0,2]$, obtained with $M_1 = 10$, $M_2 = 40$, $h = 3$ for no noise $\delta = 0$ and $\lambda_1 = \lambda_2 = 0$ in comparison with the exact (——) solutions (8.14c), 8.15(a)–(c), respectively, for Example 1.

figures, although some instabilities start to manifest in the retrieved heat flux in Figure 8.6(c).

So far, Example 1 has investigated retrieving a linear function of t for the free surface (8.14c). In the next example we consider a nonlinear variation.

8.3.2 Example 2

Let us take $T = 1$, $l = 1$, $\alpha_1 = 0.5$, $\alpha_2 = 1$, $k_1 = 0.5$, $k_2 = 1$, $L = 1$, $\rho_2 = 1$, $K_1 = 0.5$, $K_2 = 1$, and $u^* = 0$. We also choose in equations (0.2) and (0.13) the initial conditions

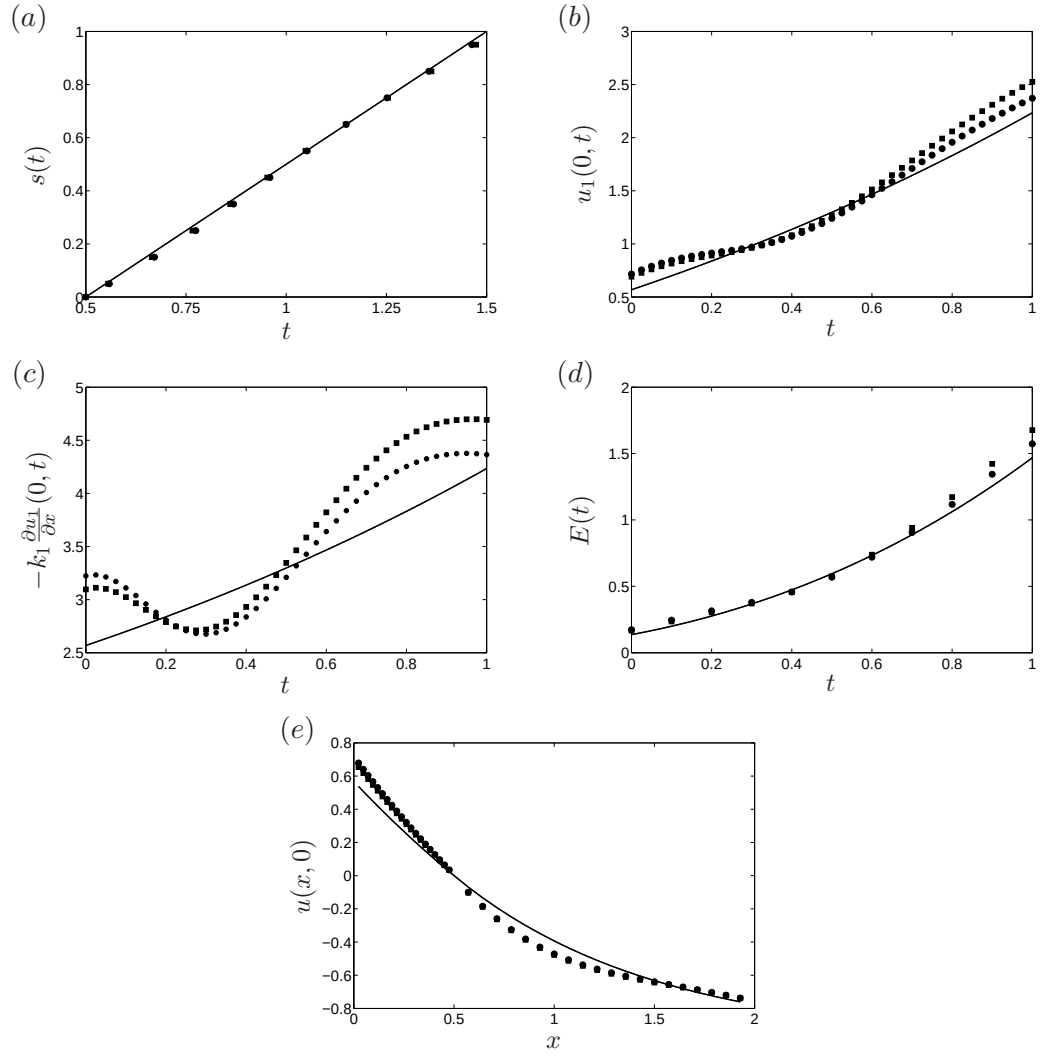

Figure 8.6: The numerical MFS solutions for: (a) $s(t)$, (b) $u_1(0,t)$, (c) $-k_1\frac{\partial u_1}{\partial x}(0,t)$, and (d) $E(t)$, as a function of $t \in [0,1]$, and (e) $u(x,0)$ as a function of $x \in [0,2]$, obtained with $M_1 = 10$, $M_2 = 40$, $h = 3$ for $\delta = 1\%$ (•) and $\delta = 5\%$ (■) noise with $\lambda_1 = \lambda_2 = 10^{-6}$ in comparison with the exact (——) solutions (8.14c), 8.15(a)–(c), respectively, for Example 1.

$$u_1(x,0) = u_1^0(x) = 1 - \frac{\operatorname{erf}\left(\frac{x}{2\sqrt{\alpha_1 t^0}}\right)}{\operatorname{erf}\left(\frac{\gamma}{2\sqrt{\alpha_1}}\right)}, \quad x \in [0, s(0) = 0.4] \tag{8.20a}$$

$$u_2(x,0) = u_2^0(x) = 1 + \frac{\operatorname{erfc}\left(\frac{x}{2\sqrt{\alpha_2 t^0}}\right)}{\operatorname{erfc}\left(\frac{\gamma}{2\sqrt{\alpha_2}}\right)}, \quad x \in [0.4 = s(0), l = 1] \tag{8.20b}$$

where $\gamma = 0.479611$, $t^0 = 0.695571$ and erfc is the complementary error function. We also take the interface condition (8.12) and the Cauchy boundary conditions (0.15a) and (0.15b) specified by

$$u_2(1,t) = f_l(t) = -1 + \frac{\operatorname{erfc}\left(\frac{1}{2\sqrt{\alpha_2(t+t^0)}}\right)}{\operatorname{erfc}\left(\frac{\gamma}{2\sqrt{\alpha_2}}\right)}, \quad t \in (0, T=1], \tag{8.21a}$$

$$k_2 \frac{\partial u_2}{\partial x}(1,t) = g_l(t) = -\frac{k_2 \exp\left(-\frac{1}{4\alpha_2(t+t^0)}\right)}{\sqrt{\pi\alpha_2(t+t^0)}\,\operatorname{erfc}\left(\frac{\gamma}{2\sqrt{\alpha_2}}\right)}, \quad t \in (0, T=1], \tag{8.21b}$$

Then the analytic solution of the inverse Stefan problem (0.11), (0.12), (0.14), (8.12), (8.20) and (8.21) is given by, see [27, 32, 64],

$$u_1(x,t) = 1 - \frac{\operatorname{erf}\left(\frac{x}{2\sqrt{\alpha_1(t+t^0)}}\right)}{\operatorname{erf}\left(\frac{\gamma}{2\sqrt{\alpha_1}}\right)}, \quad (x,t) \in \overline{D_T^1}, \tag{8.22a}$$

$$u_2(x,t) = -1 + \frac{\operatorname{erfc}\left(\frac{x}{2\sqrt{\alpha_2(t+t^0)}}\right)}{\operatorname{erfc}\left(\frac{\gamma}{2\sqrt{\alpha_2}}\right)}, \quad (x,t) \in \overline{D_T^2}, \tag{8.22b}$$

$$s(t) = \gamma\sqrt{t+t^0}, \quad t \in [0, T=1], \tag{8.22c}$$

which can be verified by direct substitution. Of particular interest is to recover the missing data at the inaccessible hostile boundary $x = 0$, given by the boundary temperature

$$u_1(0,t) = f_0(t) = 1, \quad t \in [0, T=1] \tag{8.23a}$$

the heat flux

$$-k_1 \frac{\partial u_1}{\partial x}(0,t) = g_0(t) = \frac{k_1}{\sqrt{\pi\alpha_1(t+t^0)}\,\operatorname{erf}\frac{\gamma}{2\sqrt{\alpha_1}}}, \quad t \in [0, T=1] \tag{8.23b}$$

and the mass

$$\int_0^{s(t)} u_1(x,t)\,dx = E(t) = \frac{2}{\operatorname{erf}\left(\frac{\gamma}{2\sqrt{\alpha_1}}\right)} \sqrt{\frac{\alpha_1(t+t^0)}{\pi}} \left(1 - \exp\left(-\frac{\gamma^2}{4\alpha_1}\right)\right), \quad t \in [0, T=1]. \tag{8.23c}$$

Noise is added in the boundary temperature (8.21a), as in (8.16).

Figures 8.7 and 8.8 for Example 2 are the analogous of Figures 8.3 and 8.4 for Example 1 and the same conclusions about the good performance of the method are obtained. To further improve the results obtained in Figure 8.7, in Figure 8.9 we increase the number of collocation and source points.

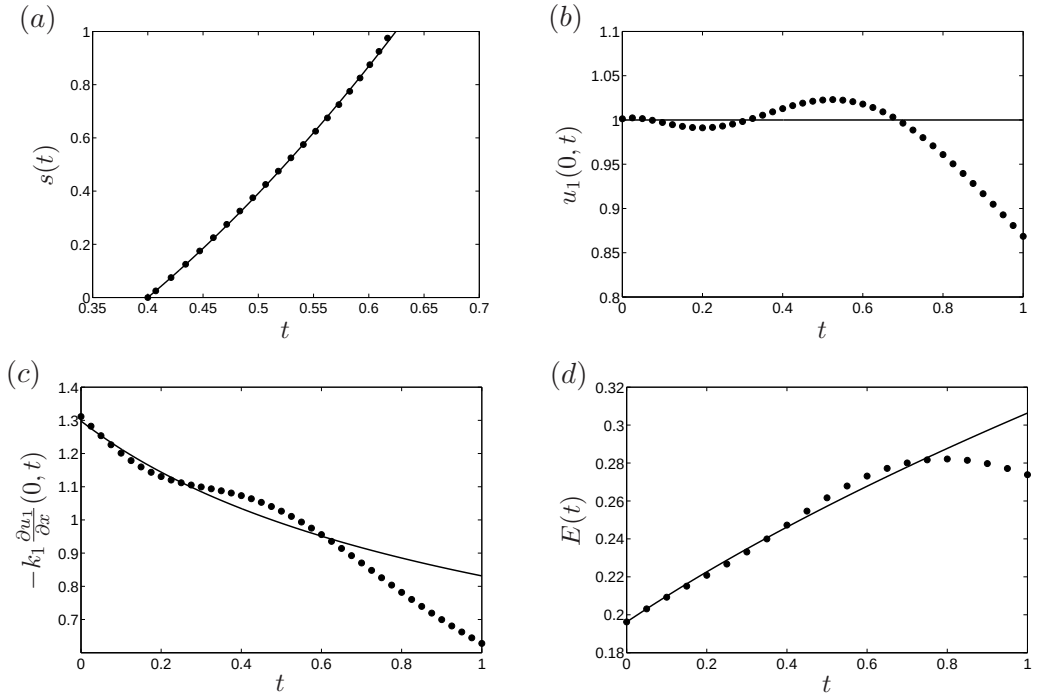

Figure 8.7: The numerical (•••) MFS solutions for: (a) $s(t)$, (b) $u_1(0,t)$, (c) $-k_1\frac{\partial u_1}{\partial x}(0,t)$, and (d) $E(t)$, as a function of $t \in [0,1]$, obtained with $M_1 = 20$, $M_2 = 40$, $K = 40$, $h = 2$ for no noise $\delta = 0$ and $\lambda_1 = \lambda_2 = 0$ in comparison with the exact (——) solutions (8.22c), 8.23(a)–(c), respectively, for Example 2.

In both Examples 1 and 2 analytic solutions were available and this enabled assessing the accuracy of the numerically obtained results. The next example considers the case when an analytic solution is not available.

8.3.3 Example 3

In this example, partially taken from [2] (partially since we do not impose the initial data in the domain D_T^1 due to problems with the existence of a solution, see [64]), where a linear inverse Stefan problem was addressed, an analytic solution is not available. We take $T = 1$, $l = \pi/2$, $\alpha_1 = 2$, $\alpha_2 = 1$, $s(0) = \pi/4$, $K_1 = 1$, $K_2 = 2$, $k_1 = 1$, $k_2 = 2$, $u^* = 0$, $g_l = 0$, and $u_2^0(x) = \cos(2x)$ for $x \in [\frac{\pi}{4}, \frac{\pi}{2}]$. Since an analytic solution is not available the data (0.15a) is numerically simulated by solving separately using the MFS the direct problem in the domain D_T^2 given by equations (0.12b), namely

$$\frac{\partial u_2}{\partial t} = 2\frac{\partial^2 u_2}{\partial x^2}, \quad (x,t) \in D_T^2 = (s(t), l) \times (0, T], \tag{8.24}$$

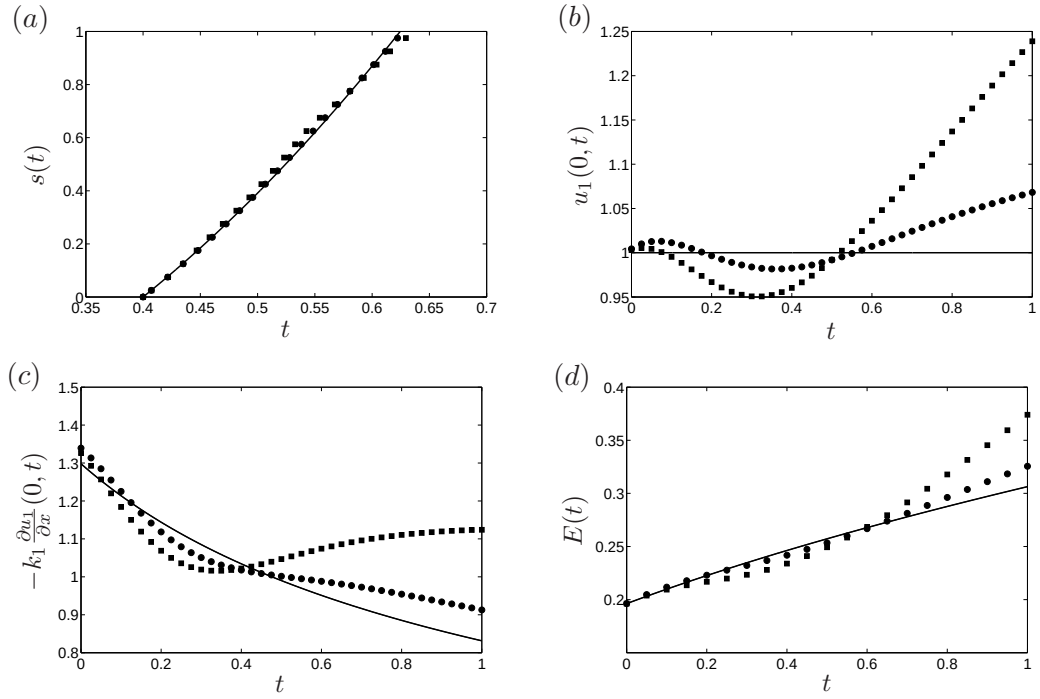

Figure 8.8: The numerical MFS solutions for: (a) $s(t)$, (b) $u_1(0,t)$, (c) $-k_1\frac{\partial u_1}{\partial x}(0,t)$, and (d) $E(t)$, as a function of $t \in [0,1]$, obtained with $M_1 = 20$, $M_2 = 40$, $K = 40$, $h = 2$ for $\delta = 1\%$ (•) and $\delta = 5\%$ (■) noise and $\lambda_1 = \lambda_2 = 10^{-6}$ in comparison with the exact (——) solutions (8.22c), 8.23(a)–(c), respectively, for Example 2.

equation (0.13) given by

$$u_2(x,0) = u_2^0(x) = \cos(2x), \quad x \in \left[\frac{\pi}{4}, \frac{\pi}{2}\right], \tag{8.25}$$

condition (0.14a) given by

$$u_2(s(t),t) = u^*(t) = 0, \quad t \in (0,1], \tag{8.26}$$

the boundary condition (0.15b) given by

$$k_2\frac{\partial u_2}{\partial x}\left(\frac{\pi}{2}, t\right) = g_l(t) = 0, \quad t \in (0,1], \tag{8.27}$$

when the free surface is known and is given by

$$s(t) = \arctan(1+t), \quad t \in (0,1]. \tag{8.28}$$

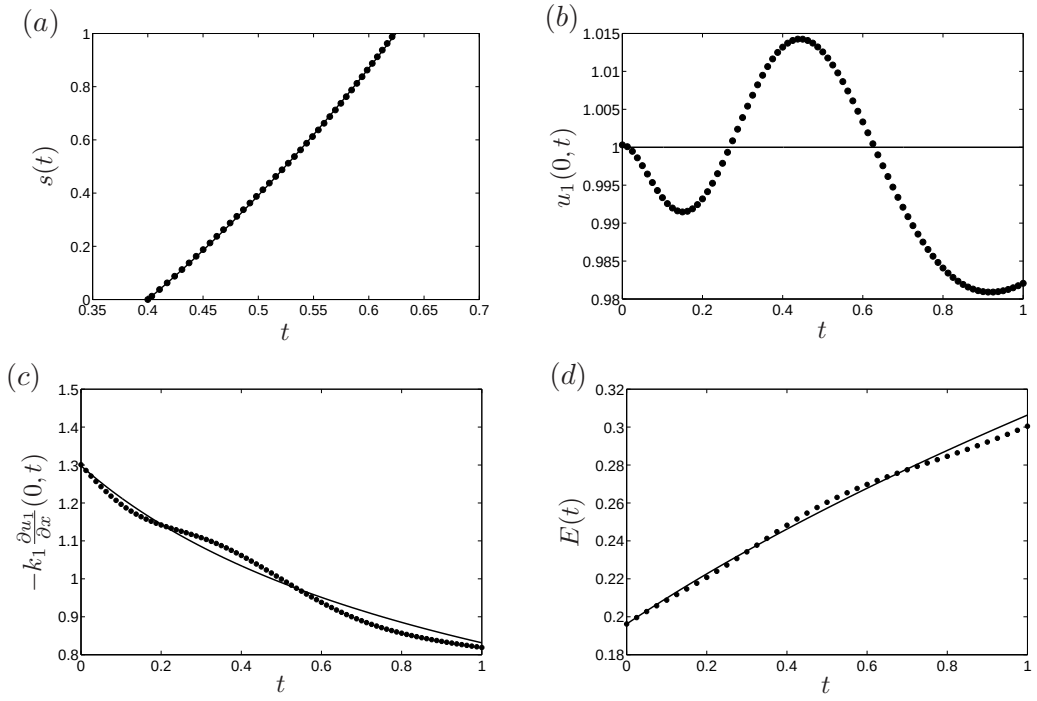

Figure 8.9: The numerical (•••) MFS solutions for: (a) $s(t)$, (b) $u_1(0,t)$, (c) $-k_1\frac{\partial u_1}{\partial x}(0,t)$, and (d) $E(t)$, as a function of $t \in [0,1]$, obtained with $M_1 = 40$, $M_2 = 80$, $K = 80$, $h = 2$ for no noise $\delta = 0$ and $\lambda_1 = \lambda_2 = 10^{-12}$ in comparison with the exact (——) solutions (8.22c), 8.23(a)–(c), respectively, for Example 2.

Observe that the compatibility conditions in (0.16) and (0.17) given by

$$u_2^0(s(0)) = u^*(0), \quad k_2\frac{du_2^0}{dx} = g_l(0) \tag{8.29}$$

are automatically satisfied by the data (8.25)–(8.28). In the direct problem we collocate the equations (8.25)–(8.27) and (8.29) as

$$u_2(x_2^{(k)}, 0) = u_2^0(x_2^{(k)}), \quad k = 1, \ldots, K \tag{8.30}$$

$$u_2(s_i, t_i) = u^*(t_i), \quad i = 0, \ldots, M_1 \tag{8.31}$$

$$k_2\frac{\partial u_2}{\partial x}(l, \tilde{t}_i) = g_l(\tilde{t}_i), \quad i = 0, \ldots, M_2 \tag{8.32}$$

resulting in, via (8.2), a system of $(K + M_1 + M_2 + 2)$ linear equations with $4M_1$ unknowns $\mathbf{c}^{(2)} = (c_j^{(2)})_{j=1,\ldots,M}$. A necessary condition for a unique solution is $K + M_2 + 2 \geq 3M_1$. This system of equations is ill-conditioned and therefore we employ a linear Tikhonov regularization with regularization parameter λ. The numerical results obtained for the boundary temperature $u_2(\pi/2, t)$, as a function of $t \in [0,1]$, are shown in Figure 8.10. This data, to which noise is

added as in (8.16), is then used as the input (0.15a) in the inverse problem.

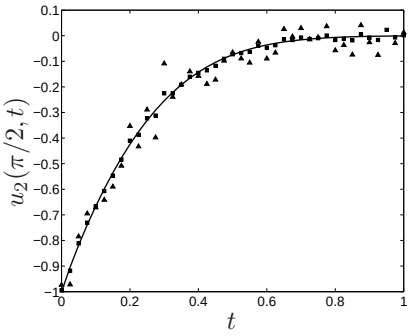

Figure 8.10: The direct problem MFS for $u_2(\pi/2,t)$ (——), as a function of $t \in [0,1]$, obtained with $h=2$, $M_1=40$, $M_2=60$, $K=58$ (i.e. $K+M_1+M_2+2=160$ equations with $4M_1=160$ unknowns) and $\lambda=10^{-14}$, and the perturbed data for $\delta=1\%$ (■) and $\delta=5\%$ (▲), for Example 3.

The imposition of the initial condition (0.2) cannot be arbitrary and in order to ensure that the solution exists we simply do not impose it. In this case the first term in the right-hand side of (8.9) for $n=1$ drops out, i.e. equation (8.5) for $n=1$ is not imposed, such that, via (8.2), equations (8.5) for $n=2$, (8.6)–(8.8) form a system of $(3M_1+2M_2+K+4)$ equations with $9M_1$ unknowns, and we require $2M_2+K+4 \geq 6M_1$.

Figures 8.11(a)–(e) and 8.12(a)–(e) show the MFS solutions for $s(t), u_1(0,t), -k_1\frac{\partial u_1}{\partial x}(0,t)$, $E(t)$ and $u_1(x,0)$ obtained for $M_1=20, M_2=40, K=40$ (i.e. $3M_1+2M_2+K+4=184$ equations with $9M_1=180$ unknowns) for $\delta=0$ and $\lambda_1=\lambda_2=0$, $\delta=1\%$ and $\lambda_1=\lambda_2=10^{-6}$. We note that to generate the accurate results in Figure 8.11 the variable and function tolerances used in the MATLAB toolbox *lsqnonlin* were reduced from 10^{-6} to 10^{-9}. Reducing the variable tolerance reduces the lower bound on the step size and lowering the function tolerance reduces the lower bound for the calculation of the minimum of the functional (8.9), which should generate a closer approximation to $\mathbf{c}$. For the free boundary, comparison between the MFS solution and the exact solution (8.28) made in Figure 8.11(a) shows good agreement. We point out that the reconstructions become rather inaccurate when increasing the noise level, and for $\delta=5\%$ it is not possible to obtain a reasonable approximation. Moreover, by inspecting Figures 8.12(a)–(e) (see also Figures 8.6(c) and 8.8(c)) we note that the numerical solutions seem to deviate from the 'exact' MFS approximations that were generated in [64], as the time t approaches the final time T. This is to be somewhat expected since if we want to recover the heat flux at $x=0$ over the whole time interval $[0,T]$ then we need to use the Cauchy data (0.15a) and (0.15b) over an

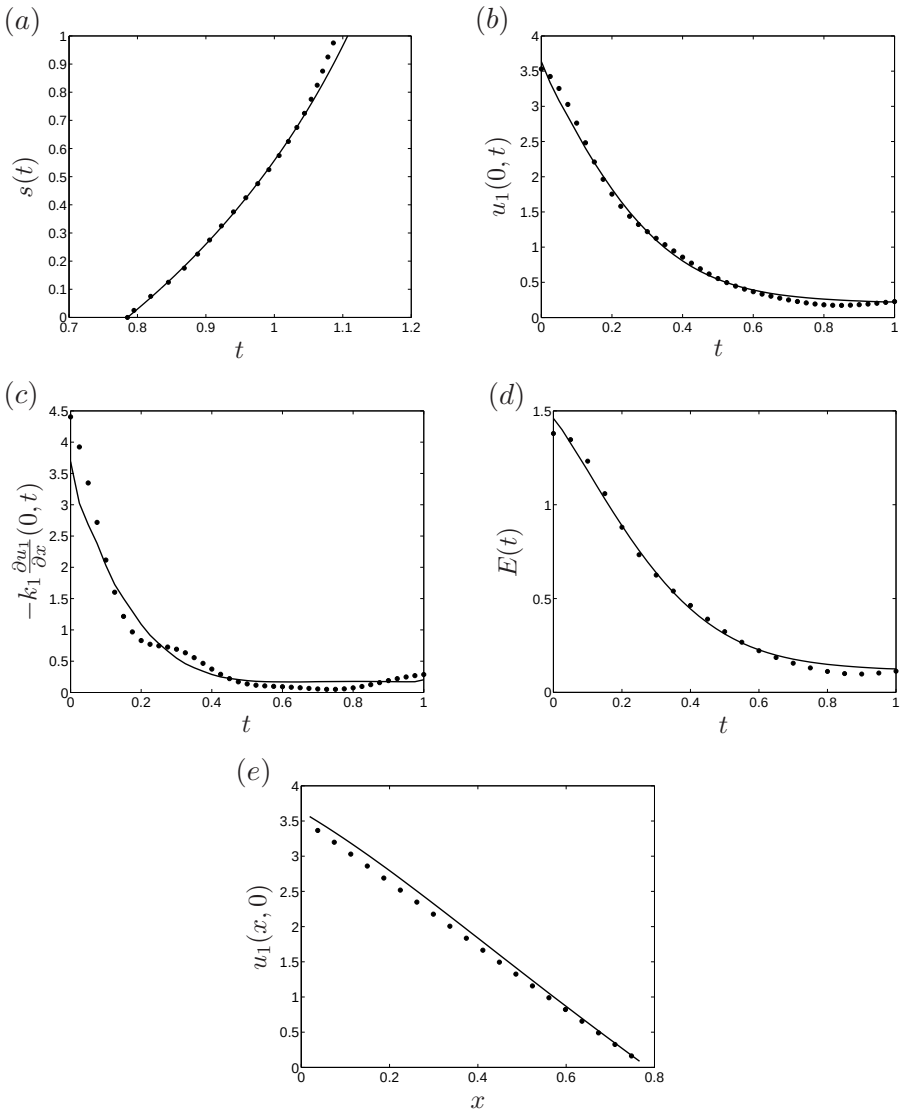

Figure 8.11: The numerical MFS solutions (●●●) for: (a) $s(t)$, (b) $u_1(0,t)$, (c) $-k_1\frac{\partial u_1}{\partial x}(0,t)$, and (d) $E(t)$, as a function of $t \in [0,1]$, and (e) $u_1(x,0)$, as a function of $x \in [0,\pi/4]$ obtained with $M_1 = 20$, $M_2 = 40$, $K = 40$, for no noise $\delta = 0$ and $\lambda_1 = \lambda_2 = 0$, for Example 3. In case (a), the comparison with the exact solution (8.28) for $s(t)$ is also shown, and in cases (b)–(e) comparisons are made with the MFS approximation (——) that was generated in [64].

extended interval $[0, T+r]$, where $r > 0$ is related to the concept of 'future times' in the inverse heat conduction literature, see [6, 52].

8.4 Summary of Chapter 8

In this final chapter we have investigated an inverse two-phase one-dimensional nonlinear Stefan problem, which involved finding the temperature in both domains, and the free boundary s. The

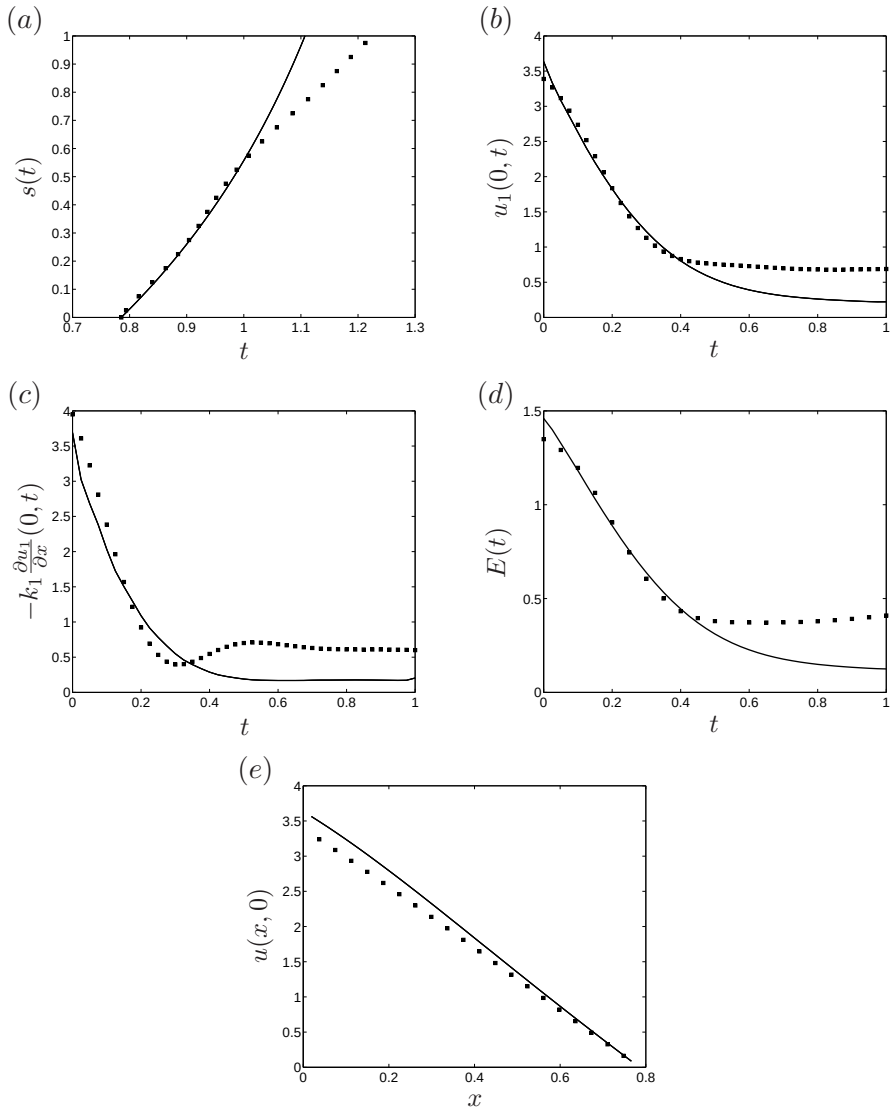

Figure 8.12: The numerical MFS solutions for: (a) $s(t)$, (b) $u_1(0,t)$, (c) $-k_1\frac{\partial u_1}{\partial x}(0,t)$, and (d) $E(t)$, as a function of $t \in [0,1]$, and (e) $u_1(x,0)$, as a function of $x \in [0,\pi/4]$ obtained with $M_1 = 20$, $M_2 = 40$, $K = 40$, for $\delta = 1\%$ (■) noise and $\lambda_1 = \lambda_2 = 10^{-6}$, for Example 3. In case (a), the comparison with the exact solution (8.28) for $s(t)$ is also shown, and in cases (b)–(e) comparisons are made with the MFS approximation (——) that was generated in [64].

approximation was sought by applying the nonlinear Tikhonov regularization method based on minimizing the nonlinear regularized least-squares functional (8.9), which was accomplished using the MATLAB toolbox *lsqnonlin*.

We numerically tested three examples, the first two had analytic solutions available and results were reasonable for most figures, although there were instabilities in the recovered Neumann data and close to the final time point, which was seen previously in the MFS approximations for other Stefan type problems in Chapters 6 and 7. In the third example an analytic solution

was not available, and results were compared with those generated in [64], where the MFS was used to solve the inverse two-phase one-dimensional linear Stefan problem, and it was observed that when noise was added to the boundary the approximation deteriorated for $t > 0.5$.

CHAPTER 9

CONCLUSION

The aim of this thesis was to investigate the application of the method of fundamental solutions to different heat conduction problems. In previous work (see reviews [36, 40]) the method was capable of generating accurate solutions to linear PDEs as well as being computationally inexpensive. However, most of the literature on the MFS had predominantly focused on stationary heat flow problems governed by elliptic PDEs, and formulations (different placement of source points, transforming the heat equation into an elliptic equation using the Laplace transform, etc.) of the MFS for the parabolic heat equation (given in [26, 41, 103, 110, 71]) were performed without theoretical justification (linear independence and denseness results).

This thesis involved extending the MFS proposed in [62], where one-dimensional heat conduction was investigated, with the sources placed outside the space domain of interest, but in time, i.e. placed in the interval $(-T, T)$ (other studies have placed source points in the domain but at an earlier or later time, see Figure 6 in Section 0.4). It is well established that for direct heat conduction problems popular methods, including the FEM, FDM and BEM, readily produce accurate results, even over complicated domains. However, for inverse heat conduction problems (IHCPs) the application of these methods is studied considerably less (and the process of modifying these methods is often rather complicated), see [46] for a review. For examples, a deforming FEM analysis of inverse Stefan problems was studied in [112], and the BEM has been successfully applied to a range of inverse heat conduction problems, see [54, 78, 80], however, the MFS does not require boundary element integrations (although once the integrals have been calculated for the BEM we solve a linear system like the MFS).

Two of the advantages with the MFS is that it can be applied to many IHCPs and is relatively simple to understand and program into a computer, which is useful for engineers.

Inverse problems are mostly ill-posed, with problems which involve no continuous dependence of the solution on the input data requiring regularization techniques to obtain stable solutions, in this work we have used Tikhonov regularization, in conjunction with the L-curve criterion to obtain a suitable choice for the regularization parameter.

In Chapter 1, theoretical results (including linear independence and denseness) were proved for linear combinations of fundamental solutions, and five numerical examples were investigated, for the direct two-dimensional heat conduction problem. In Example 2 we tested the accuracy of the MFS when the source points located below the initial time $t = 0$ are placed uniformly over intervals of different lengths below the initial time $t = 0$. The reason for testing this was to see if a smaller interval resulted in the condition number of the system decreasing, and hence an increase in accuracy, however, for this example, this was not the case. For the errors at $t = 0.2$ the accuracy for larger intervals ($\tilde{T} < 50$) stayed comparable with those when $\tilde{T} = 1$. In Example 3, using the exact solution from Example 2, we briefly investigated the absolute error when source points are placed in different shapes around a domain, and for this domain we found source points placed on a circle were preferable, however, in general, placing source points on a pseudo-boundary via dilatation by expanding the boundary (which is an understandable choice) results in reasonable approximations. Examples 4 and 5 considered square domains, and in Example 4 we had a problem with a very large thermal diffusivity constant, and found that approximations were only possible over a very small time interval (which makes sense). In Example 5 we considered our first inverse problem, with data missing from one of the boundaries, and we note that modifying the code for this scenario was straightforward.

With the success of applying the MFS to the inverse problem in Example 5 of Chapter 1 we moved on to proving (where necessary) linear independence and denseness results, and numerically solving IHCPs, in the remaining chapters. Often we have produced chapters in this thesis to display the applicability of the MFS or, after reviewing papers, (which use either complicated methods, or different (unjustified) formulations of the MFS), and implement our MFS and see if our results are comparable or better (see, for example Chapters 2 and 5). In Chapter 2 we produced a comparison study for the backward heat conduction problem (for problems in one and two-dimensions), which involved finding the temperature at the initial time point $t = 0$. As with the compared papers, there was a value of T for which reconstructions were not possible, however, in the numerics we found our errors comparable to those in papers

[86, 82, 59]. We also performed an analysis of what impact the parameters h and T have on the approximation.

In Chapters 4 and 5 we moved onto radial and axisymmetric problems, which are usually used to model heat flow in pipes, and the symmetry in such models reduces the dimension by one. In Chapter 4 we examined the radially symmetric backward heat conduction problem (extending the work performed in Chapter 2, part of the work used in this chapter formed a proceedings of the 8th UK Conference on Boundary Integral Methods, University of Leeds, UK, 4-5th July 2011. To prove linear independence in Theorems 4.3.3 and 4.3.5 we needed additional results, which were proved and given in Lemmas 4.3.1 and 4.3.2. The point of the lemmas is that we can find a curve from (r, t) to (ρ, τ) (a source point) such that a certain ratio becomes unbounded, see the proofs of Theorems 4.3.3 and 4.3.5. With Chapter 4 as justification for using the MFS for inverse radial problems we studied the radially symmetric IHCP in Chapter 5, which involved finding Dirichlet and Neumann data at an inner boundary r_0, when Dirichlet and Neumann data is given on an outer boundary R. As mentioned previously, we justify this work based on the fact that this problem was considered in the recent paper [109], which applied a quasi-reversibility regularization method, and after numerically implementing the same examples we found that RMSE and RRMSE values were better when using our MFS.

The rest of the thesis involved studying inverse Stefan problems, where a free boundary is present, which is caused by a freezing or melting process. Chapters 6 and 7 featured design problems, where the free boundary is known, and data is recovered on the free boundary $x = 0$. These problems, unlike direct Stefan problems, ask: what temperature and flux need to be applied to the fixed boundary to obtain a certain melting front? In the numerics at the end of Chapter 6 we applied the dynamic MFS in an attempt to find an optimal value for h (the distance the source points are from the boundary). Other nonlinear solvers are available, and might produce better results, however, for simplicity and ease of use the MATLAB toolbox *lsqnonlin*, which requires only simple bounds on the data, performs very well. We continued the work in Chapter 6 to the inverse Cauchy-Stefan problem in Chapter 7, where the initial data is missing, and showed that results were still accurate and stable when this information is not given.

To finish, in Chapter 8 we examined the inverse two-phase one-dimensional nonlinear Stefan problem, which required the determination of the position of the free boundary, along with the

unknown boundary data at $x = 0$, where we applied a nonlinear Tikhonov regularization method based on minimizing the nonlinear regularized least squares functional which is accomplished by again using the MATLAB toolbox *lsqnonlin*. This is a particularly difficult problem to solve, and we were impressed with the results that were obtained. However, we had to be careful with the choice of certain parameters, in particular, h and λ. Also when noise was applied to the boundary data in Example 3 stable results were only possible up to the time point $t = 0.5$.

I have also contributed to four other papers (which are listed in the section "Outcome of the Scientific Work" at the beginning of this thesis). Three of the papers were in collaboration with my supervisor Dr B. T. Johansson and Professor D. Lesnic and involved solving Stefan problems of different types, in particular, the inverse two-phase one-dimensional linear Stefan problem, the two-dimensional inverse Stefan problem and the two-dimensional two-phase linear inverse Stefan problem. The other paper was in collaboration with Dr B. T. Johansson and Professor D. J. Needham called "The development of a wax layer on the interior wall of a circular pipe transporting heated oil." The paper contained modelling, theory, asymptotics, as well as numerics which were carried out using the MFS.

This is an initial study and I think it shows promising results, but to be a valuable method the following will need to be done:

- Extend to three-dimensional problems.
- Multiphase inverse Stefan problems.
- Theoretical investigations for optimal placement of source points, which would extend the work on elliptic problems produced in [1].
- Error estimates between the MFS approximation and the exact solution showing the dependence of the accuracy on the number of source and collocation points.
- Applying it to real world problems, such as the transporting of heated oil.

APPENDIX

Notation and definitions

In this section we give definitions of some spaces and sets that are used throughout the thesis.

If we denote D a bounded (or unbounded) domain in $\mathbb{R}^n$, an n-dimensional Euclidean space, the closure of D will be denoted by $\overline{D}$. The 2-norm of $\mathbf{x} \in D \subset \mathbb{R}^n$ will be given by

$$\|\mathbf{x}\| = \left(\sum_{i=1}^{n} x_i^2 \right)^{\frac{1}{2}}.$$

The space of continuous functions will be denoted by $C^0(D)$, and the space of functions with all derivatives continuous up to order j will be denoted by $C^j(D)$. The space of functions with all derivatives continuous up to order j of the variable $\mathbf{x}$ (for example), and order k for another variable $\mathbf{t}$ will be denoted $C^{j,k}(D)$. The space of square-integrable functions is denoted by $L^2(D)$, i.e. the real or complex space $L^2(D)$ is a set of finite measurable functions on a measurable space such that

$$\langle f, f \rangle = \left(\int |f|^2 \, d\mu \right)^{\frac{1}{2}} < \infty,$$

for a measure μ.

Lastly, a subset X of D is dense if the closure of X contains D.

Distributions and the Dirac delta function

In this section we define a certain class of linear functionals, called distributions.

Definition A.1. *A test function is a function $\phi(\mathbf{x})$, with $\mathbf{x} \in \mathbb{R}^n$, where the following hold:*
(i) *$\phi(\mathbf{x})$ is continuously differentiable to all orders,*
(ii) *$\phi(\mathbf{x})$ has compact support, i.e. there exists a constant $c \in \mathbb{R}$ such that $\phi(\mathbf{x}) = 0$ for every $|\mathbf{x}| > c$. The space of test functions is denoted by $\mathcal{D}$.*

To derive the fundamental solution of the heat equation we need the Dirac delta function, $\delta(\mathbf{x})$, which represents a unit source at $\mathbf{x} = \mathbf{0}$. The delta function is not actually a function, but instead is known as a distribution.

Definition A.2. *The element f is a functional on $\mathcal{D}$ if there is a rule that maps each $\phi \in \mathcal{D}$ to a real number $\langle f, \phi \rangle$, i.e. $f : \mathcal{D} \to \mathbb{R}$. Furthermore, such a functional is denoted a distribution if additionally it is linear and continuous. We have linearity if for every $c_1, c_2 \in \mathbb{R}$ and for every $\phi_1, \phi_2 \in \mathcal{D}$*

$$\langle f,\, c_1\phi_1 + c_2\phi_2 \rangle \;\; = \;\; c_1\langle f, \phi_1 \rangle + \;\; c_2\langle f, \phi_2 \rangle$$

and continuity, for $\phi_k \to \phi$ as $k \to \infty$, if

$$\langle f, \phi_k \rangle \;\to\; \langle f, \phi \rangle \;\; \textit{as } k \to \infty.$$

The set of all distributions on $\mathcal{D}$ is denoted by $\mathcal{D}'$.

Distributions can be generated by locally integrable functions $f(\mathbf{x})$, which are readily shown to be linear and continuous functionals, given by the formula

$$\begin{aligned} \langle f, \phi \rangle = & \int_{\mathbb{R}^n} f(\mathbf{x})\phi(\mathbf{x})\, d\mathbf{x} = \\ & \int_{\mathbb{R}} \dots \int_{\mathbb{R}} f(x_1, x_2, \dots, x_n)\phi(x_1, x_2, \dots, x_n)\; dx_1 dx_2 \dots dx_n. \end{aligned} \tag{A1}$$

Equation (A1), where f is locally integrable, defines a regular distribution, with other distributions called singular.

The Dirac delta function, $\delta(\mathbf{x})$, is singular and the notation used, for $\phi \in \mathcal{D}$, is given by

$$\int_{\mathbb{R}^n} \delta(\mathbf{x} - \mathbf{y})\phi(\mathbf{x}) d\mathbf{x} \;\; = \;\; \phi(\mathbf{y}). \tag{A2}$$

It is understood that equation (A2) is only a formal notation since δ is not a function and $\delta(\mathbf{x} - \mathbf{y})$ represents a unit source at $\mathbf{x} = \mathbf{y}$. Additional details on distributions can be found in [98, 100].

We define the Heaviside function, which is the integral of the Dirac delta function, to be

$$H(x) = \begin{cases} 1 & \text{if } x > 0 \\ \\ 0 & \text{if } x \leq 0. \end{cases} \tag{A3}$$

The fundamental solution of the heat equation

The fundamental solution of the heat equation can be defined as follows:

Definition A.3. *Let* $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ *and* $t, \tau \in (0, T)$, *where* τ *is considered an initial time point and* $T > 0$ *a fixed final time point* $(0 \leq \tau < T)$. *A function* $F(\mathbf{x}, t; \mathbf{y}, \tau)$ *is a fundamental solution to the heat equation,*

$$\frac{\partial u}{\partial t} = \Delta u, \tag{A4}$$

where the thermal diffusivity has been set equal to 1 *(via a variable substitution), if the following hold:*

(i) F *is defined and continuous for every* $(\mathbf{x}, t), (\mathbf{y}, \tau) \in \mathbb{R}^n \times (0, T)$ *apart from when* $(\mathbf{x}, t) = (\mathbf{y}, \tau)$.

(ii) $F(\mathbf{x}, t; \mathbf{y}, \tau) = 0$ *for every* $t \leq \tau$.

(iii) $\int_{\mathbb{R}^n} |F(\mathbf{x}, t; \mathbf{y}, \tau)| \, d\mathbf{y} \leq c$, *for some constant* $c \in \mathbb{R}$.

(iv) *For every continuous and bounded function* $\phi(\mathbf{y}) \in \mathbb{R}$ *the function*

$$u(\mathbf{x}, t) = \int_{\mathbb{R}^n} F(\mathbf{x}, t; \mathbf{y}, \tau) \phi(\mathbf{y}) \, d\mathbf{y}$$

is a solution to (A4) *for* $(\mathbf{x}, t) \in \mathbb{R}^n \times (\tau, T)$. *We finally require that*

$$\lim_{(\mathbf{x}, t) \to (\mathbf{x_0}, \tau^+)} u(\mathbf{x}, t) = \phi(\mathbf{x_0}).$$

The fundamental solution can be defined in a more abstract setting equivalently, as seen in [90], by

Definition A.4. *A distribution* $F \in \mathcal{D}'(\mathbb{R}^n)$ *is called a fundamental solution of* $\mathcal{L}$, *a linear differential operator, if and only if the equation* $\mathcal{L}(F) = \delta$ *holds in* $\mathcal{D}'(\mathbb{R}^n)$, *where* δ *is the Dirac delta function.*

In our case for the heat equation we have the solution space as a product $\mathbb{R}^n \times \mathbb{R}$ and therefore we use $\delta(\mathbf{x} - \mathbf{y})\delta(t - \tau)$, which, physically represents a unit source of heat at position $\mathbf{x} = \mathbf{y}$ and time $t = \tau$. Hence, for the heat conduction equation, given by (A4), to construct the

fundamental solution, $F(\mathbf{x}, t; \mathbf{y}, \tau)$, we consider

$$\frac{\partial F}{\partial t} - \Delta F = \delta(\mathbf{x} - \mathbf{y})\delta(t - \tau). \tag{A5}$$

Therefore, we now state the definition, given in ([98], p.59), of the causal fundamental solution (causal being defined as a function which vanishes for $t < 0$ or $t > 0$):

Definition A.5. *The causal fundamental solution $F_0(\mathbf{x}, t)$ satisfies*

$$\frac{\partial F_0}{\partial t} - \Delta F_0 = \delta(\mathbf{x})\delta(t), \qquad F_0 \equiv 0 \quad \textit{when} \quad t < 0. \tag{A6}$$

A derivation of the fundamental solution of the n-dimensional heat equation (A4) can be found in [98], and is given by

$$F(\mathbf{x}, t; \mathbf{y}, \tau) = F_0(\mathbf{x} - \mathbf{y}, t - \tau) = \frac{H(t - \tau)}{(4\pi(t - \tau))^{\frac{n}{2}}} e^{-\frac{|\mathbf{x}-\mathbf{y}|^2}{4(t-\tau)}}. \tag{A7}$$

LIST OF REFERENCES

[1] C. J. S. Alves. On the choice of source points in the method of fundamental solutions. *Eng. Anal. Boundary Elements*, 33:1348–1361, 2009.

[2] D. D. Ang, A. Pham Ngoc Dinh, and D. N. Tranh. Regularization of an inverse two-phase Stefan problem. *Nonlinear Analysis*, 34:719–731, 1998.

[3] D. D. Ang, A. Pham Ngoc Dinh, and D.N. Tranh. An inverse Stefan problem: identification of boundary value. *J. Comput. Appl. Math.*, 66:75–84, 1996.

[4] W.-T. Ang and E.-H. Ooi. A dual-reciprocity boundary element approach for solving axisymmetric heat equation subject to specification of energy. *Eng. Anal. Boundary Elements*, 32:210–215, 2008.

[5] K. Balakrishnan and P. A. Ramachandran. The method of fundamental solutions for linear diffusion-reaction equations. *Math. Comput. Model.*, 31:221–237, 2000.

[6] J. V. Beck, B. Blackwell, and C. R. St-Clair Jr. *Inverse Heat Conduction*. J. Wiley-Intersc. Publ., New York, 1985.

[7] J. V. Beck, K. D. Cole, A. Haji-Sheikh, and B. Litkouhi. *Heat Conduction Using Green's Functions*. Hemisphere Publishing Co., Washington D.C., 1992.

[8] M. Belge, M. E. Kilmer, and E. L. Miller. Efficient determination of multiple regularization parameters in a generalized L-curve framework. *Inverse Problems*, 18:1161–1183, 2002.

[9] A. Ben-Israel and T.N.E. Greville. *Generalized Inverses: Theory and Applications*, volume 15. Springer Verlag, New York, 2003.

[10] J.R. Berger and A. Karageorghis. The method of fundamental solutions for layered elastic materials. *Eng. Anal. Boundary Elements*, 25(10):877–886, 2001.

[11] J.R. Berger, A. Karageorghis, and P.A. Martin. Stress intensity factor computation using the method of fundamental solutions: mixed-mode problems. *Int. J. Numer. Meth. Eng.*, 69(3):469–483, 2007.

[12] B. Bialecki, G. Fairweather, and A. Karageorghis. Matrix decomposition algorithms for elliptic boundary value problem: a survey. *Numer. Algorithms*, 56:253–295, 2011.

[13] A. Bogomolny. Fundamental solutions method for elliptic boundary value problems. *SIAM J. Numer. Anal.*, 22:644–669, 1985.

[14] J. R. Cannon. A Cauchy problem for the heat equation. *Ann. Mat. Pura Appl.*, 66:155–166, 1964.

[15] J. R. Cannon. *The One-dimensional Heat Equation*. Addison-Wesley, Menlo Park, California, 1984.

[16] J. R. Cannon, D. B. Henry, and D. B. Kotlow. Classical solutions of the one-dimensional, two-phase Stefan problem. *Ann. Mat. Pura Appl.*, 107:311–341, 1975.

[17] J. R. Cannon and C. D. Hill. Existence, uniqueness, stability, and monotone dependence in a Stefan problem for the heat equation. *J. Math. Mech.*, 17:1–19, 1967.

[18] J. R. Cannon and J. Douglas Jr. The Cauchy problem for the heat equation. *SIAM J. Numer. Anal.*, 4(3):317–336, 1967.

[19] J. R. Cannon and M. Primicerio. A two phase Stefan problem: regularity of the free boundary. *Ann. Mat. Pura Appl.*, 88:217–228, 1971.

[20] J. R. Cannon and M. Primicerio. A two phase Stefan problem with flux boundary conditions. *Ann. Mat. Pura Appl.*, 88:193–205, 1971.

[21] J. R. Cannon and M. Primicerio. A two phase Stefan problem with temperature boundary conditions. *Ann. Mat. Pura Appl.*, 88:177–191, 1971.

[22] J. R. Cannon and M. Primicerio. Remarks on the one-phase Stefan problem for the heat equation with the flux prescribed on the fixed boundary. *J. Math. Anal. Appl.*, 35:361–373, 1971.

[23] J. R. Cannon and J. van der Hoek. The classical solution of the one-dimensional two-phase Stefan problem with energy specification. *Ann. Mat. Pura Appl.*, 130:385–398, 1982.

[24] J. R. Cannon and J. van der Hoek. The one phase Stefan problem subject to the specification of energy. *J. Math. Anal. Appl.*, 86:281–291, 1982.

[25] H. S. Carslaw and J. C. Jaeger. *Conduction of Heat in Solids*. Oxford University Press, London, 1959.

[26] S. Chantasiriwan. Methods of fundamental solutions for time-dependent heat conduction problems. *Int. J. Numer. Meth. Eng.*, 66:147–165, 2006.

[27] S. Chantasiriwan, B. T. Johansson, and D. Lesnic. The method of fundamental solutions for free surface Stefan problems. *Eng. Anal. Boundary Elements*, 33:529–538, 2009.

[28] A. H.-D. Cheng and D. T. Cheng. Heritage and early history of the boundary element method. *Eng. Anal. Boundary Elements*, 29:268–302, 2005.

[29] W. Cheng, C.-L. Fu, and Z. Qian. Two regularization methods for a spherically symmetric inverse heat conduction problem. *Appl. Math. Modelling*, 32:432–442, 2008.

[30] T.F. Coleman and Y. Li. On the convergence of reflective Newton methods for large-scale nonlinear minimization subject to bounds. *Math. Program.*, 67:189–224, 1994.

[31] T.F. Coleman and Y. Li. An interior, trust region approach for nonlinear minimization subject to bounds. *SIAM J. Optim.*, 6:418–445, 1996.

[32] J. Crank. *Free and Moving Boundary Problems*. Clarendon Press, Oxford, 1984.

[33] A. El Badia and F. Moutazaim. A one-phase inverse Stefan problem. *Inverse Problems*, 15:1507–1522, 1999.

[34] L. Eldén. The numerical solution of a non-characteristic Cauchy problem for a parabolic equation. In: *Numerical Treatment of Inverse Problems in Differential Equations and Integral Equations*, (eds. P. Deuflhard and E. Hairer), pp. 246–268, Birkhäuser, Basel, 1983.

[35] L. C. Evans. *Partial Differential Equations*. American Mathematical Society, Providence, Rhode Island, 1998.

[36] G. Fairweather and A. Karageorghis. The method of fundamental solutions for elliptic boundary value problems. *Adv. Comput. Math.*, 9:69–95, 1998.

[37] J.B.J. Fourier. *The Analytical Theory of Heat (1878)*. Kessinger Publishing, Montana, 2008. English translation: translated by A. Freeman.

[38] A. Friedman. *Partial Differential Equations of Parabolic Type*. Prentice-Hall Inc., Englewood Cliffs, NJ, 1964.

[39] B.D. Geelen. Accurate solution for the modified Bessel function of the first kind. *Adv. Eng. Software*, 23(2):105–109, 1995.

[40] M. A. Golberg and C. S. Chen. The method of fundamental solutions for potential, Helmholtz and diffusion problems. In: *Boundary Integral Methods: Numerical and Mathematical Aspects*, (ed. M.A. Golberg), pp. 103-176, Computational Mechanics Publications, Southampton, 1998.

[41] M. A. Golberg, C. S. Chen, and A. S. Muleshkov. The method of fundamental solutions for time-dependent problems. In: *Proc. 13th Int. Conf. Boundary Element Technology*, (eds. C. S. Chen, C. A. Brebbia and D. W. Pepper), pp. 377–386, Southampton, WIT Press, 1999.

[42] N. L. Gol'dman. *Inverse Stefan Problems*. Kluwer Academic Publ., Dordrecht, 1997.

[43] N. L. Gol'dman. Inverse problems with phase transitions. In: *Proceedings of 3rd International Conference on Dynamic Systems Identification and Inverse Problems*, (eds. O. M. Alifanov, N. V. Kerov, V. V. Michailov and A. N. Nenarokomov), pp. 128–139, Moscow-St. Petersburg, Russia WIT Press, 1998.

[44] N. L. Gol'dman. Properties of solutions of the inverse Stefan problem. *Differ. Equ.*, 39:66–72, 2003.

[45] P. Gorzelanczyk and J. A. Kolodziej. Some remarks concerning the shape of the source contour with application of the method of fundamental solutions to elastic torsion of prismatic rods. *Eng. Anal. Boundary Elements*, 32:64–75, 2008.

[46] K. Grysa. Inverse Heat Conduction Problems. Heat Conduction - Basic Research, Vyacheslav S. Vikhrenko (Ed.). Available from:
`http://www.intechopen.com/books/heat-conduction-basic-research/inverse-heat-conduction-problems`
ISBN: 978-953-307-404-7, InTech, DOI: 10.5772/26575, 2011.

[47] R. Grzymkowski and D. Slota. One-phase inverse Stefan problem solved by Adomain decomposition method. *Comput. Math. Appl.*, 51:33–40, 2006.

[48] J. Hadamard. *Lectures on Cauchy's problem in linear partial differential equations*. Yale University Press, New Haven, 2003.

[49] H. Han, D.B. Ingham, and Y. Yuan. The boundary element method for the solution of the backward heat conduction equation. *J. Comput. Phys.*, 116:292–299, 1995.

[50] P. C. Hansen. Analysis of discrete ill-posed problems by means of the L-curve. *SIAM Rev.*, 34:561–580, 1992.

[51] P.C. Hansen. *Discrete Inverse Problems: Insight and Algorithms*, volume 7. Society for Industrial and Applied Mathematics, Philadelphia, 2010.

[52] Dinh Nho Hào. *Methods for Inverse Heat Conduction Problems*. Peter Lang, Frankfurt am Main, 1998.

[53] Dinh Nho Hào, Nguyen Van Duc, and D. Lesnic. Regularization of parabolic equations backward in time by a non-local boundary value problem method. *IMA J. Appl. Math.*, 75:291–315, 2010.

[54] Dinh Nho Hào, Phan Xuan Thanh, D. Lesnic, and B.T. Johansson. A boundary element method for a multi-dimensional inverse heat conduction problem. *Int. J. Comput. Math.*, 89(11):1540–1554, 2012.

[55] A. Hasanov and J. L. Mueller. A numerical method for backward parabolic problems with non-self adjoint elliptic operators. *Appl. Numer. Math.*, 37:55–78, 2001.

[56] C. D. Hill. Parabolic equations in one space variable and the non-characteristic Cauchy problem. *Comm. Pure Appl. Math.*, 20:619–633, 1967.

[57] E. Holmgren. Sur l'equation de la propagation de la chaleur. *Arkiv. Mat. Fys.*, 14:1–11, 1908.

[58] Y. C. Hon and M. Li. A computational method for inverse free boundary determination problem. *Int. J. Numer. Meth. Eng.*, 73:1291–1309, 2008.

[59] Y. C. Hon and M. Li. A discrepancy principle for the source points location in using the MFS for solving the BHCP. *Int. J. Comput. Meth.*, 6:181–197, 2009.

[60] L. Hui and L. Jijun. Solution of backward heat problem by Morozov discrepancy principle and conditional stability. *Numer. Math. J. Chin. Univ.* (*Engl. Ser.*), 14:180–192, 2005.

[61] P. Jochum. The numerical solution of the inverse Stefan problem. *Numer. Math.*, 34:411–429, 1980.

[62] B. T. Johansson and D. Lesnic. A method of fundamental solutions for transient heat conduction. *Eng. Anal. Boundary Elements*, 32:697–703, 2008.

[63] B. T. Johansson and D. Lesnic. A method of fundamental solutions for transient heat conduction in layered materials. *Eng. Anal. Boundary Elements*, 33:1362–1367, 2009.

[64] B. T. Johansson, D. Lesnic, and T. Reeve. A meshless method for an inverse two-phase one-dimensional linear Stefan problem. *Inverse Problems Sci. Eng.*, 21(1):17–33, 2013.

[65] F. John. Numerical solution of the equation of heat conduction for preceding times. *Ann. Mat. Pura Appl. ser. IV*, 40:129–142, 1955.

[66] R. Johnson. A priori estimates and unique continuation theorems for second order parabolic equations. *Trans. Amer. Math. Soc.*, 158:167–177, 1971.

[67] M. Jourhmane and N. S. Mera. An iterative algorithm for the backward heat conduction problem based on variable relaxation factors. *Inverse Problems Eng.*, 10:293–308, 2002.

[68] S.I. Kabanikhin. Definitions and examples of inverse and ill-posed problems. *J. Inv. Ill-Posed Problems*, 16:317–357, 2008.

[69] A. Karageorghis. A practical algorithm for determining the optimal pseudo-boundary in the method of fundamental solutions. *Adv. Appl. Math. Mech.*, 1:510–528, 2009.

[70] A. Karageorghis, D. Lesnic, and L. Marin. The method of fundamental solutions for the detection of rigid inclusions and cavities in plane linear elastic bodies. *Computers & Structures*, http://dx.doi.org/10.1016/j.compstruc.2012.05.001, 2012.

[71] J. A. Kolodziej, J. Stefaniak, and M. Kleiber. Transient heat conduction by boundary collocation methods and FEM – A comparison study. In: *Notes on Numerical Fluid Mechanics*, pp. 104–115, Braunschweig, Vieweg Verlag, 1995.

[72] V. D. Kupradze. A method for the approximate solution of limiting problems in mathematical physics. *USSR Comput. Maths. Math. Phys.*, 4:199–205, 1964.

[73] V. D. Kupradze and M. A. Aleksidze. The method of functional equations for the approximate solution of certain boundary value problems. *USSR Comput. Math. Math. Phys.*, 4:82–126, 1964.

[74] P. K. Kythe. *Fundamental solutions for differential operators and applications*. Birkhäuser, Boston, 1996.

[75] G. Lamé and B. P. Clapeyron. Mémoire sur la solidification par refroidissement d'un globe solide. *Ann. Chem. Phys.*, 47:250–256, 1831.

[76] E. M. Landis. *Second-Order Equations of Elliptic and Parabolic Type* [*in Russian*]. Nauka, Moscow, 1971. Translated by American Mathematical Soc.

[77] R. Lattes and J.-L. Lions. *The Method of Quasi-Reversibility.* Amer. Elsevier Publ., New York, 1969.

[78] D. Lesnic, L. Elliott, and D. B. Ingham. Boundary element method for solving an inverse problem in radial heat conduction. *Proc. ASME/JSME Thermal Eng. Joint Conf.*, Hawaii, USA, 19–24 March 1995, pp. 71–78.

[79] D. Lesnic, L. Elliott, and D. B. Ingham. Treatment of singularities in time-dependent problems using the boundary element method. *Eng. Anal. Boundary Elements*, 16:65–70, 1995.

[80] D. Lesnic, L. Elliott, and D. B. Ingham. Application of the boundary element method to inverse heat conduction problems. *Int. J. Heat Mass Transfer*, 39:1503–1517, 1996.

[81] D. Lesnic, L. Elliott, and D. B. Ingham. An iterative boundary element method for solving the backward heat conduction problem using an elliptic approximation. *Inverse Problems Eng.*, 6:255–279, 1998.

[82] M. Li, T. Jiang, and Y. C. Hon. A meshless method based on RBFs method for nonhomogeneous backward heat conduction problem. *Eng. Anal. Boundary Elements*, 34:785–792, 2010.

[83] J. Liu and B. Guerrier. A comparative study of domain embedding methods for regularized solutions of inverse Stefan problems. *Int J. Numer. Meth. Eng.*, 40:3579–3600, 1997.

[84] L. Marin. Relaxation procedures for an iterative MFS algorithm for two-dimensional steady-state isotropic heat conduction Cauchy problems. *Eng. Anal. Boundary Elements*, 35:415–429, 2011.

[85] R. Mathon and R. L. Johnston. The approximate solution of elliptic boundary-value problems by fundamental solutions. *SIAM J. Numer. Anal.*, 14:638–650, 1977.

[86] N. S. Mera. The method of fundamental solutions for the backward heat conduction problem. *Inverse Problems Sci. Eng.*, 13:65–78, 2005.

[87] N. S. Mera, L. Elliott, and D. B. Ingham. An iterative boundary element method for solving the one-dimensional backward heat conduction problem. *Int. J. Heat Mass Transfer*, 44:1937–1946, 2001.

[88] W. L. Miranker. A well posed problem for the backward heat equation. *Proc. Amer. Math. Soc.*, 12:243–254, 1961.

[89] D. A. Murio. Solution of inverse heat conduction problems with phase changes by the mollification method. *Comput. Math. Appl.*, 24:45–57, 1992.

[90] N. Ortner and P. Wagner. A survey on explicit representation formulae for fundamental solutions of linear partial differential operators. *Acta Appl. Math.*, 47:101–124, 1997.

[91] R. Penrose. A generalized inverse for matrices. In *Proc. Cambridge Philos. Soc*, volume 51, pages 406–413. Cambridge Univ Press, 1955.

[92] D. Redekop and R.S.W. Cheung. Fundamental solutions for the collocation method in three-dimensional elastostatics. *Computers & structures*, 26(4):703–707, 1987.

[93] R. Reemtsen and A. Kirsch. A method for the numerical solution of the one-dimensional inverse Stefan problem. *Numer. Math.*, 45:253–273, 1984.

[94] L. I. Rubinsteĭn. *The Stefan Problem*. American Mathematical Society, Providence, 1971.

[95] J.-C. Saut and B. Scheurer. Unique continuation for some evolution equations. *J. Differential Equations*, 66:118–139, 1987.

[96] D. Slota. Solving the inverse Stefan design problem using genetic algorithms. *Inverse Problems Sci. Eng.*, 16:829–846, 2008.

[97] D. Slota. Using genetic algorithms for the determination of an heat transfer coefficient in three-phase inverse Stefan problem. *Int. Commun. Heat Mass Transfer*, 35:149–156, 2008.

[98] I. Stakgold. *Boundary Value Problems of Mathematical Physics*, volume 2. Macmillan; Collier-Macmillan, New York, 1968.

[99] J. Stefan. Über einige Probleme der Theorie der Wärmeleitung. *S.-B. Wien. Akad. Mat. Natur.*, 98:473–484, 1889.

[100] W.A. Strauss. *Partial Differential Equations - An Introduction*. John Wiley & Sons, New York, 1992.

[101] D. Tataru. Unique continuation for PDE's. *IMA Vol. Math. Appl.*, 137:239–255, 2004.

[102] A. N. Tikhonov and V. Y. Arsenin. *Solutions of Ill-Posed Problems*. John Wiley & Sons, New York, Toronto, London, 1977.

[103] S. S. Valtchev and N. C. Roberty. A time-marching MFS scheme for heat conduction problems. *Eng. Anal. Boundary Elements*, 32:480–493, 2008.

[104] S. Vessella. Quantitative estimates of unique continuation for parabolic equations, determination of unknown time-varying boundaries and optimal stability estimates. *Inverse Problems*, 24:023001, 2008.

[105] P. Wagner. On the explicit calculation of fundamental solutions. *J. Math. Anal. Appl.*, 297:404–418, 2004.

[106] Y. B. Wang, J. Cheng, J. Nakagawa, and M. Yamamoto. A numerical method for solving the inverse heat conduction problem without the initial value. *Inverse Problems Sci. Eng.*, 18:655–671, 2010.

[107] T. Wei and M. Yamamoto. Reconstruction of a moving boundary from Cauchy data in one-dimensional heat equation. *Inverse Problems Sci. Eng.*, 17:551–567, 2009.

[108] L. C. Wrobel and C. A. Brebbia. A formulation of the boundary element method for axisymmetric transient heat conduction. *Int. J. Heat Mass Transfer*, 24(5):843–850, 1981.

[109] X.-T. Xiong. On a radially symmetric inverse heat conduction problem. *Appl. Math. Modelling*, 34:520–529, 2010.

[110] D. L. Young, C. C. Tsai, K. Murugesan, C. M. Fan, and C. W Chen. Time-dependent fundamental solutions for homogeneous diffusion problems. *Eng. Anal. Boundary Elements*, 28:1463–1473, 2004.

[111] E.C. Young. *Partial Differential Equations: An Introduction*. Allyn and Bacon, Boston, 1972.

[112] N. Zabaras and Y. Ruan. A deforming finite element method analysis of inverse Stefan problems. *Internat. J. Numer. Methods Engrg.*, 28(2):295–313, 1989.

MATLAB CODE

The final parts of this thesis include the MATLAB code for a function called 'MFS1DStefan', and the random matrices (both called 'randMatrix') which contain noisy data, and will be saved in .mat files called 'randvars1' and 'randvars2'. The code was used for the numerical results in Chapter 6, where the one-dimensional inverse Stefan problem was investigated. The code is fully commented and almost self contained, the only function needed is *lsqnonlin* in the optimization toolbox, and is only needed if you solve the nonlinear least squares problem, which involves finding an 'optimal' value of `h` (if you do not have access to this then in the code leave `Solver='GE'` (set as default) and this will solve the problem using Gaussian elimination instead, and `h` will remain fixed).

Copy all the text starting on the next page to the end of this section (before Section 'MATLAB .mat file: randvars1' starts) into a blank m-file and save the file as MFS1DStefan.m and put this into an empty folder. Unfortunately if you are copying the text from a PDF then carriage returns and indents will be lost, however, in the MATLAB editor if you highlight all the code (ctrl+A), right click on the mouse and select smart indent (or press crtl+I) then most indents will be restored, but you will still need to include your own carriage returns if you so desire.

After this is done move on to the next two sections called 'MATLAB .mat file: randvars1' and 'MATLAB .mat file: randvars2'. Note that if the matrices found in the next two sections, which contain the random data which was used in Chapter 6, are not included, then the code will still run, but with unassigned random noise applied to the Neumann data as in (6.36) (i.e. you will not be using the noise that was used in Chapter 6 and you will not have control over the noise for testing purposes). As a test (after the m-file is created) enter the following into the command window:

```
MFS1DStefan(3,1,10^(-14),30,16,0.0,1)
```

After the .mat files (randvars1 and randvars2) have been saved, then you have all of the code. In the comments you will find descriptions of the input variables and what happens at each step of the code, and examples of what you could enter for the input variables for the function 'MFS1DStefan' are described in the comments near the top of the script.

```
function [] = MFS1DStefan(h,T,lambda,K,M,delta,randCol)

% MFS1dStefan is a function which approximates the solution of the classical
% one-dimensional inverse Stefan problem (specifically for the recovery of
% the data on the fixed boundary x=0) using the direct MFS with initial
% condition given by u0 (at time=0) and boundary conditions given by h1 and
% h2 on the free boundary s(t) t in [0,T].  There is also an option to
% solve a nonlinear least squares problem which involves trying to find the
% optimal h (the distance the source points are from the boundary).
%
% Description of parameters:
% h:       Is the distance the source points are from the fixed boundary x=0
%          and the free boundary s(t), see figure 99 when code has run.
%          (Usually chosen in the interval 1 to 3)
% T:       Is the final time point.
%          (Set to be 1 in the Examples)
% lambda:  Is the Tikhonov regularization parameter.
%          (10^(-14) and 10^(-6) without and with noise, respectively, are
%          usually good choices, use L-curve plot to confirm - set
%          plotLP = 'P'; below)
% K:       Is the number of points placed on the initial base.
%          (Set to be 30 and 60 in the Examples)
% M:       (M+1) is the number of points placed in time on the free boundary
%          s(t), and there 4*M source points altogether.
%          (Set to be 16 and 31 in the Examples)
% delta:   is the percentage of noise added to the boundary.  Type 0.05 for
%          5% noise, for example.
% randCol: Is the set of random noise you want to use where the entries are
%          contained in a matrix with (M+1)x10 entries called randMatrix
%          (which at the time of writing was contained in either the MATLAB
%          file randvars1 or randvars2).  Therefore this picks a column from
%          this noise matrix.
%          (Choose a value between 1 and 10)
%
% We will obtain a linear system Ac=g, however, the matrix A will be
% ill-conditioned and we apply Tikhonov regularization.
%
% A typical example that could be entered into the command window would be:
%
% (With no noise)
% MFS1DStefan(3,1,10^(-14),30,16,0.0,1)
%
% (With 5% noise (for example))
% MFS1DStefan(3,1,10^(-6),30,16,0.05,1)

%Below are additional options for the program:

%Set if you want the run of the program timed (yes 'Y' or no 'N'):
TimeIT = 'Y';
if(TimeIT=='Y')
    tic
end
```

```
%Choose if you want to solve the problem using Gaussian Elimination
%('GE') or Least Squares ('LS'):
Solver = 'GE';


%Choose if you want the condition number of the matrix A printed to the
%screen yes ('Y') OR  no ('N') when using Gaussian elimination.
CondA = 'Y';


%Select (by setting Example appropriately) which example you want to use
%from the thesis:
%'Ex1'  :  s(t)=sqrt(2)-1+t/sqrt(2), u(x,t)=-1+exp(1-1/sqrt(2)+t/2-x/sqrt(2))
%'Ex2'  :  s(t)=2-sqrt(3-2t),        u(x,t)=-x^2/2+2x-1/2-t
Example = 'Ex1';


%Choose if you want to plot the L-curve ('L') or plots of the results
%('P'):
plotLP = 'P';


%Allowing an initial choice of h to be entered for h in the least squares
%solver

if strcmp(Solver,'LS')==1
    h = input('Least squares method chosen. Please enter an initial value for h = ');
    if isempty(h)
        h=10;
        fprintf('Default value of h=%g chosen.\n',h)
    end
end


%Note that the above can be set to be entered in the command window.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  END OF INPUT  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Checks to make sure values above are entered correctly

if CondA~='Y' && CondA~='N'
    CondA = 'N';
end

if strcmp(Solver,'GE')==0 && strcmp(Solver,'LS')==0
    fprintf('  The solver variable must be set to solve using Gaussian elimination (GE)')
    fprintf('  or the least squares method (LS), please check mfile\n')
    return
end

if strcmp(Example,'Ex1')==0 && strcmp(Example,'Ex2')==0
```

```
    fprintf('  Please choose Example 1 (Ex1) or Example 2 (Ex2), please check mfile.\n')
    return
end

if strcmp(plotLP,'P')==0 && strcmp(plotLP,'L')==0
    fprintf('  You must either choose to plot data (P) or produce an L-curve plot (L),')
    fprintf('  please check mfile.\n')
    return
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%



disp('-------------------------------------------------------------------------------------')

fprintf('Paramters: h=%g, T=%g, lambda=%g, K=%d, M=%d, delta=%g.\n',h,T,lambda,K,M,delta)
if strcmp(Solver,'GE')==1
    fprintf('Solving using Gaussian elimination.\n')
elseif strcmp(Solver,'LS')==1
    fprintf('Solving using least squares.\n')
end

if strcmp(Example,'Ex1')==1
    fprintf('Using data from Example 1.\n')
elseif strcmp(Example,'Ex2')==1
    fprintf('Using data from Example 2.\n')
end

if strcmp(plotLP,'P')==1
    fprintf('Producing plots...\n')
elseif strcmp(plotLP,'L')==1
    fprintf('Producing the L-curve plot...\n')
end

disp('-------------------------------------------------------------------------------------')

if strcmp(plotLP,'P')==1
    fprintf('Figure 99: A plot of the boundary points and source points.\n')
    fprintf('           Collocation points = blue *, Source points = red o, and\n')
    fprintf('           Unknown data = - - - \n')
    fprintf('Figure 1:  The exact (-) and the MFS approximation (*) of the Dirichlet data at x=0.\n')
    fprintf('Figure 2:  The exact (-) and the MFS approximation (*) of the Neumann data at x=0.\n')
    fprintf('Figure 3:  The exact data in the entire domain [0,s(t)]x[0,%g] t in [0,%g].\n',T,T)
    fprintf('Figure 4:  The MFS approximation in the entire domain [0,s(t)]x[0,%g] t in [0,%g].\n',T,T)
    fprintf('Figure 5:  The absolute error in the entire domain [0,s(t)]x[0,%g] t in [0,%g].\n',T,T)
end

disp('-------------------------------------------------------------------------------------')

%Below we define the time coordinates of the source points (split into
%taum1 (for time < 0) and taum2 (for time > 0), and these are then combined
```

```
%into the vector taum with 2*M time points.
m1 = (-(M-1):0);
m2 = (1:M);
taum1 = ((2*m1-1)/(2*M))*T;
taum2 = ((2*m2-1)/(2*M))*T;
taum = [taum1,taum2];


%M+1 time coordinates for the collocation points on s(t).
i = (0:M);
ti = (i/M)*T;


%Points along the 'base' for imposing the initial condition
l = (1:K);
xl = l.*s(0,Example)./(K+1);

%Defining all collocation points and source points.
%Source points have been placed at a distance h along from the boundary and
%the placement below t=0 is symmetric wrt the x axis.

PI = [xl',zeros(K,1)];
PB = [s(ti',Example),ti'];

PS1 = [repmat(-h,length(taum),1),taum'];
PS2 = [s(-taum1',Example)+h,taum1';s(taum2',Example)+h,taum2'];
PS = [PS1;PS2];

%Next we construct the matrix A (see thesis for details).

A = zeros(length(PI(:,1))+2*length(PB(:,1)),length(PS(:,1)));

for j = 1:length(PS(:,1))
    for i = 1:length(PI(:,1))
        A(i,j) = F([PI(i,:),PS(j,:)]);
    end
    for i = 1:length(PB(:,1))
        A(i+length(PI(:,1)),j) = F([PB(i,:),PS(j,:)]);
        A(i+length(PI(:,1))+length(PB(:,1)),j) = dF([PB(i,:),PS(j,:)]);
    end
end

%Printing out the number of equations and unknowns.
fprintf('There are %d equations and %d unknowns.\n',length(A(:,1)),length(A(1,:)))
disp('------------------------------------------------------------------------------------')

%If required, the condition number of A is printed out
if(CondA=='Y' && strcmp(Solver,'GE')==1)
    fprintf('For h=%g the condition number of A = %g.\n',h,cond(A))
end

%Transforming the matrix A into a sparse matrix
```

```
As = sparse(A);

%Impose the initial (g1) and boundary conditions (g2 for Dirichlet and g3
%for Neumann) below.

g1 = u0(PI,Example);

g2 = h1(PB,Example);

%Below we impose noise to the boundary data h2 (if delta~=0).  We have
%two options for testing purposes there is preassigned noise when K==30 and
%M==16, or K==60 and M=31, saved in randvars1 and randvars2, respectively.
%If randvars1 or randvars2 do not exist in the directory then
%unassigned random variables are used instead.
%Define other variables for other preassigned noise.  Other choices for M
%and K not preassigned will use non preassigned random noise (see else
%statement below).
if K==30 && M==16
    if length(dir('randvars1.mat'))==1
        %randvars1 file exists
        load randvars1
        g3 = h2(PB,Example) + delta.*(max(abs(h2(PB,Example)))).*randMatrix(:,randCol);
    else
        % File does not exist.
        fprintf('Warning: randvars1 file does not exist, using unassigned random variables\n');
        g3 = h2(PB,Example) + delta.*(max(abs(h2(PB,Example)))).*randn(length(PB),1);
    end
elseif K==60 && M==31
    if length(dir('randvars2.mat'))==1
        %randvars1 file exists
        load randvars2
        g3 = h2(PB,Example) + delta.*(max(abs(h2(PB,Example)))).*randMatrix(:,randCol);
    else
        % File does not exist.
        fprintf('Warning: randvars1 file does not exist, using unassigned random variables\n');
        g3 = h2(PB,Example) + delta.*(max(abs(h2(PB,Example)))).*randn(length(PB),1);
    end
else
    g3 = h2(PB,Example) + delta.*(max(abs(h2(PB,Example)))).*randn(length(PB),1);
end
%#ok<*NODEF>

%Putting all the boundary data together...

g = [g1;g2;g3];


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%The following is a plot of the L-curve for varying lambda, if necessary to
%calculate.  (Choose if you want to do this using options above.)
```

```
if plotLP == 'L'

    lambda1 = 10.^(-16:1:0);
    cnorm = zeros(size(lambda1));
    Residual = zeros(size(lambda1));

    for i=1:length(lambda1);
        c1= (As'*As + lambda1(1,i).*eye(size(A'*A)))\(As'*g);
        cnorm(1,i) = norm(c1,2);
        Residual(1,i) = norm(As*c1-g,2);
    end

    loglog(Residual,cnorm,'ko');

end

%If the L-curve is not being plotted then choosing 'P' for plotLP will
%calculate c and produce the plots below.

if plotLP == 'P'
    switch Solver
        case 'GE'
            %Solving using Guassian elimination...
            %Tikhonov Regularisation using the backslash command.
            c = (As'*As + lambda.*eye(size(A'*A)))\(As'*g);

        case 'LS'
            %Solving a nonlinear least squares problem using lsqnonlin to
            %determine an "optimal" h...

            %Below we define cs0 the initial guess to start the iterative
            %process of finding the minimization using lsqnonlin.
            ch0 = [h,0*ones(1,length(PS))];

            %Below are the bounds for the parameter h and the constants c,
            %(LBcs represents the lower bound and UBcs the upper bound).
            LBch = [0,-10^(308)*ones(1,length(PS))];
            UBch = [10^(308),10^(308)*ones(1,length(PS))];

            %See thesis and documentation for details on lsqnonlin.
            Options=optimset('Display','iter','MaxFunEvals',50000,'MaxIter',500,'TolFun',1e-8,'TolX',1e-8);

            %Introduce an array of arrays to store the parameters that are
            %used in the function functionalch
            parameters{1} = taum; parameters{2} = taum1; parameters{3} = taum2;
            parameters{4} = Example; parameters{5} = PI; parameters{6} = PB;
            parameters{7} = g1; parameters{8} = g2; parameters{9} = g3;
            parameters{10} = lambda;

            [ch,resn,res,exitflag,output]=lsqnonlin(@(ch) funch(ch,parameters),ch0,LBch,UBch,Options);

            fprintf('The value of the squared 2-norm of the residual is %g.\n',resn)
```

```
            fprintf('The value of the exitflag is %g.\n',exitflag)
            disp('Additional information:')
            disp(output)

            h = ch(1);
            c = ch(2:end)';

            fprintf('h=%g is the "optimal" value obtained after running lsqnonlin.\n',h)

            PS1 = [repmat(-h,length(taum),1),taum'];
            PS2 = [s(-taum1',Example)+h,taum1';s(taum2',Example)+h,taum2'];
            PS = [PS1;PS2];

    end
end


%Below we plot the position of the boundaries and the position of the
%source points, in Figure 99.

figure(99)
plot(PI(:,1),PI(:,2),'b*')
hold on
plot(PB(:,1),PB(:,2),'b*')
plot(PS(:,1),PS(:,2),'ro')
plot([0,0],[0,1],'k--')
hold off
xlabel('x','FontSize',12,'FontAngle','italic')
ylabel('t','FontSize',12,'FontAngle','italic')


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Next we produce plots for the Dirichlet and Neumann data at x=0, as well
%as plots in the whole domain.

if plotLP == 'P'

    %Setting variables for use with the plots:

    x = 0.0;
    t = 0:0.0625:1;
    t2 = t;

    xt1 = [repmat(x,length(PS),length(t)),repmat(t,length(PS),1)];
    PE = reshape(xt1,length(t)*length(PS),2);

    %Plot of the exact and MFS approximation of the Dirichlet data along
    %the fixed boundary x=0 for varying t.

    figure(01)
```

```
    f1FPE = (F([PE,repmat(PS,length(t),1)]));
    f1FPE1 = reshape(f1FPE,length(PS),length(t));
    f1FPE2 = f1FPE1';

    f1uapprox = (f1FPE2*c)';

    switch Example
        case 'Ex1'
            f1uexact = -1+exp(1-(2^(-1/2))+(t2/2)-(x/(2^(1/2))));
        case 'Ex2'
            f1uexact = -(x^2)/2 + 2*x - t2 -1/2;
    end

    plot(t,f1uapprox,'k*',t2,f1uexact,'b-')
    xlabel('t','FontSize',12,'FontAngle','italic')
    ylabel('u(0,t)','FontSize',12,'FontAngle','italic')


%Plot of the exact and MFS approximation of the Neumann data along
%the fixed boundary x=0 for varying t.

figure(02)

    f2FPE = (dF([PE,repmat(PS,length(t),1)]));
    f2FPE1 = reshape(f2FPE,length(PS),length(t));
    f2FPE2 = f2FPE1';

    f2uapprox = (f2FPE2*c)';

    switch Example
        case 'Ex1'
            f2uexact = -(1/(2^(1/2)))*exp(1-(2^(-1/2))+(t2/2));
        case 'Ex2'
            f2uexact = 2.*ones(1,length(t2));
    end

    plot(t,f2uapprox,'k*',t2,f2uexact,'b-')
    xlabel('t','FontSize',12,'FontAngle','italic')
    ylabel('u_x(0,t)','FontSize',12,'FontAngle','italic')


clear t x


%Three-dimensional plot of the exact, MFS approximation and absolute
%error, over the entire domain.

tc = (0:0.0625:1)';
tcmatrix = repmat(tc,1,length(tc));

x=zeros(length(tc),length(tc));
```

```
for i=1:length(tc)
    x(i,:) = 0:s(tc(i),Example)/(length(tc)-1):s(tc(i),Example);
end

uapprox = zeros(length(tc),length(tc));

for i=1:length(tc)
    for j=1:length(tc)
        PE=repmat([x(i,j),tcmatrix(i,j)],length(PS),1);
        uapprox(i,j) = (F([PE,PS]))'*c;
    end
end

switch Example
    case 'Ex1'
            uexact = -1 + exp(1-(2^(-1/2))+(tcmatrix/2)-(x./(2^(1/2))));
    case 'Ex2'
        uexact = -((x.^2)./2) + 2.*x - tcmatrix - 1/2;
end

uerror = abs(uexact-uapprox);

%Plots of the exact solution (figure(04)), approximate solution
%(figure(05)) and absolute error (figure(06)) for the entire domain.
figure(03)
    surf(x,tc,uexact)
    box off
    xlabel('x','FontSize',12,'FontAngle','italic')
    ylabel('t','FontSize',12,'FontAngle','italic')
    zlabel('u(x,t)','FontSize',12,'FontAngle','italic')

figure(04)
    mesh(x,tc,uapprox)
    box off
    xlabel('x','FontSize',12,'FontAngle','italic')
    ylabel('t','FontSize',12,'FontAngle','italic')
    zlabel('u(x,t)','FontSize',12,'FontAngle','italic')

figure(05)
    mesh(x,tc,uerror)
    box off
    xlabel('x','FontSize',12,'FontAngle','italic')
    ylabel('t','FontSize',12,'FontAngle','italic')
    zlabel('u(x,t)','FontSize',12,'FontAngle','italic')




disp('-------------------------------------------------------------------------------------------')

%The mean error, max error, RMSE and RRMSE printed out for the Dirichlet
```

```
    %data at x=0;

    f1MeanError = mean(mean(abs(f1uexact-f1uapprox)));
    fprintf('The mean absolute error for the Dirichlet data is = %g.\n',f1MeanError)

    f1MaxError = max(max(abs(f1uexact-f1uapprox)));
    fprintf('The max absolute error for the Dirichlet data is = %g.\n',f1MaxError)

    f1RMSE = sqrt((1./numel(f1uexact)).*sum(sum((f1uexact-f1uapprox).^2)));
    fprintf('The RMSE of the approximated unknown Dirichlet data = %g.\n',f1RMSE)

    f1RRMSE = sqrt(sum(sum((f1uexact-f1uapprox).^2))./sum(sum((f1uexact).^2)));
    fprintf('The RRMSE of the approximated unknown Dirichlet data = %g.\n',f1RRMSE)

    disp('-------------------------------------------------------------------------------------')

    %The mean error, max error, RMSE and RRMSE printed out for the Neumann
    %data at x=0;

    f2MeanError = mean(mean(abs(f2uexact-f2uapprox)));
    fprintf('The mean absolute error for the Neumann data is = %g.\n',f2MeanError)

    f2MaxError = max(max(abs(f2uexact-f2uapprox)));
    fprintf('The max absolute error for the Neumann data is = %g.\n',f2MaxError)

    f2RMSE = sqrt((1./numel(f2uexact)).*sum(sum((f2uexact-f2uapprox).^2)));
    fprintf('The RMSE of the approximated Neumann data = %g.\n',f2RMSE)

    f2RRMSE = sqrt(sum(sum((f2uexact-f2uapprox).^2))./sum(sum((f2uexact).^2)));
    fprintf('The RRMSE of the approximated Neumann data = %g.\n',f2RRMSE)

    disp('-------------------------------------------------------------------------------------')

    %The mean error, max error, RMSE and RRMSE printed out for the entire domain;

    TDMeanError = mean(mean(abs(uexact-uapprox)));
    fprintf('The mean absolute error over the entrie domain is = %g.\n',TDMeanError)

    TDMaxError = max(max(abs(uexact-uapprox)));
    fprintf('The max absolute error over the entrie domain is = %g.\n',TDMaxError)

    TDRMSE = sqrt((1./numel(uexact)).*sum(sum((uexact-uapprox).^2)));
    fprintf('The RMSE over the entrie domain = %g.\n',TDRMSE)

    TDRRMSE = sqrt(sum(sum((uexact-uapprox).^2))./sum(sum((uexact).^2)));
    fprintf('The RRMSE over the entrie domain = %g.\n',TDRRMSE)

    disp('-------------------------------------------------------------------------------------')

end

if TimeIT=='Y'
```

```
    toc
end

disp('--------------------------------------------------------------------------------')
disp(' ')
disp(' ')
disp('--------------------------------------------------------------------------------')

%In the two functions below we define the fundamental solution and the
%spatial derivative of the fundamental solution of the heat equation in
%one-dimension (thermal diffusivity has been set equal to 1).  This
%function has one input, an array which has 4 columns, columns 1 and 2
%correspond to the space and time coordinates, respectively, in the domain
%or on the boundary of interest, and columns 3 and 4 correspond to the
%space and time coordinates of the source points.

function [f] = F(xt)
    %fundamental solution split up for clarity.
    f1 = heaviside(xt(:,2)-xt(:,4));
    f2 = (4.*pi.*(xt(:,2)-xt(:,4))).^(1/2);
    f3 = exp(-((xt(:,1)-xt(:,3)).^2)./(4.*(xt(:,2)-xt(:,4))));

    f = (f1./f2).*f3;

    f(isnan(f)) = 0 ;


function [df] = dF(xt)
    %fundamental solution split up for clarity
    f1 = heaviside(xt(:,2)-xt(:,4));
    f2 = ((4.*pi.*(xt(:,2)-xt(:,4))).^(1/2));
    f3 = (-xt(:,1)+xt(:,3))./(2.*(xt(:,2)-xt(:,4)));
    f4 = exp(-((xt(:,1)-xt(:,3)).^2)./(4.*(xt(:,2)-xt(:,4))));

    df = (f1./f2).*f3.*f4;

    df(isnan(df)) = 0 ;



%s(t) (the free boundary) is defined below.  The first input parameter is
%an array of time coordinates and the second input parameter is a string
%representing which example we are considering (see options near top of
%mfile).

function [s1] = s(t,Example)
    switch Example
        case 'Ex1'
            s1 = (2^(1/2))-1+(t./(2^(1/2)));
        case 'Ex2'
            s1 = 2 - ((3-2.*t).^(1/2));
    end
```

```
%The initial condition u0 is defined below.  The first input parameter is
%an array of size n x 2 with column 1 and column 2 the space and time
%coordinates, respectively, and the second input parameter is a string
%representing which example we are considering (see options near top of
%mfile).

function [u1] = u0(xt,Example)
    switch Example
        case 'Ex1'
            u1 = -1+exp(1-(2^(-1/2))-(xt(:,1)./(2^(1/2))));
        case 'Ex2'
            u1 = (-(xt(:,1).^2)./2) + (2.*xt(:,1)) - (1/2);
    end

%The Dirichlet condition h1 along the boundary s(t).  The first input
%parameter is an array of size n x 2 with column 1 and column 2 the space
%and time coordinates, respectively, and the second input parameter is
%string representing which example we are considering (see options near top
%of mfile).

function [h] = h1(xt,Example)
    switch Example
        case 'Ex1'
            h = zeros(size(xt(:,1)));
        case 'Ex2'
            h = zeros(size(xt(:,1)));
    end


%The Neumann boundary condition h2 along the boundary s(t).  The first
%input parameter is an array of size n x 2 with column 1 and column 2 the
%space and time coordinates, respectively, and the second input parameter
%is string representing which example we are considering (see options near
%top of mfile).

function [h] = h2(xt,Example)
    switch Example
        case 'Ex1'
            h = -((2)^(-1/2)).*ones(size(xt(:,1)));
        case 'Ex2'
            h = ((3-2.*xt(:,2)).^(1/2));
    end


%Below we define the type of Heaviside function we use.

function [h] = heaviside(x)
    h = (x > 0);
```

```
%The function below is used when we are solving the nonlinear least squares
%problem associated with finding an optimal value for h.
% First input parameter (ch):  The vector containing the scalar h (located
%   in the first entry ch(1)) and the constant coefficients (the remaining
%   entries (located in ch(2:end)).
% See code above for details of the remaining input variables stored in
% parameters.

function [fun] = funch(ch,parameters)

    taum = parameters{1}; taum1 = parameters{2}; taum2 = parameters{3};
    Example = parameters{4}; PI = parameters{5}; PB = parameters{6};
    g1 = parameters{7}; g2 = parameters{8}; g3 = parameters{9};
    lambda = parameters{10};

    %Below we split up the vector into the scalar h (representing the distance
    %the source points are from the boundary) and c (the constant coefficients).
    h = ch(1);
    c = ch(2:end);

    %Below we define the positions of the source points.
    PS1 = [repmat(-h,length(taum),1),taum'];
    PS2 = [s(-taum1',Example)+h,taum1';s(taum2',Example)+h,taum2'];
    PS = [PS1;PS2];

    %The vector fun represents the vector-valued function, which lsqnonlin will
    %solve as a optimization problem (preassign to save memory).
    fun = zeros(length(PI)+2*length(PB));

    %Below we define the vector-valued functions corresponding to the initial
    %condition.
    for k = 1:length(PI)
        PEfun1 = [repmat(PI(k,:),length(PS),1),PS];
        fun(k) = c*F(PEfun1)-g1(k);
    end

    %Below we define the vector-valued functions corresponding to the Dirichlet
    %boundary condition on s(t).
    for k = 1:length(PB)
        PEfun2 = [repmat(PB(k,:),length(PS),1),PS];
        fun(k+length(PI)) = c*F(PEfun2)-g2(k);
    end

    %Below we define the vector-valued functions corresponding to the Neumann
    %condition on the boundary s(t).
    for k = 1:length(PB)
        PEfun3 = [repmat(PB(k,:),length(PS),1),PS];
        fun(k+length(PI)+length(PB)) = c*dF(PEfun3)-g3(k);
    end


    %lambda is a regularization parameter.
```

```
if lambda~=0
    fun(length(PI)+2*length(PB)+2) = sqrt(lambda)*norm(c,2);
end
```

MATLAB .mat file: randvars1

Before starting, clear the variables in the MATLAB workspace by entering `clear all` in the command window, then type `clc` to clear the screen. In the directory (folder) where you saved the file 'MFS1DStefan' (given in the previous section) copy and paste the following two commands into the command window, which will generate a 17×10 matrix called `randMatrix`, then type `save randvars1` into the command window. Then enter `clear all` into the command window again. Move onto the next section called 'MATLAB .mat file: randvars2'.

```
randMatrix(:,1:5) = [-2.35952351095849,-1.28838600294029,-1.38526268319524,-0.411250925532220,-0.845944212336118;
                     -0.509972045799939,-0.371221247497632,0.310508318600499,-0.368010733040955,-0.172913844259168;
                     -1.32162559081010,-0.757791916498819,-0.249489064473939,-1.36096312939369,-1.20865205474020;
                     -0.636128249660041,-0.563968917027427,0.503744055737328,0.779567428734972,-0.297126799995404;
                     0.317851419059697,0.555138560001846,-0.892661403477038,0.439411111553712,-3.23203779594001;
                     0.138047974449526,-0.556778063950366,1.90851233035822,-0.0896224837412097,-1.08695922946133;
                     -0.710735074811226,-0.895113135482436,0.122230700504887,1.02118011801418,-1.42643615947972;
                     0.777003526719788,-0.409327721634070,1.04703326689793,-0.873979465028903,-1.01445076770532;
                     0.622393924172013,-0.160886766738443,-0.226920199323937,0.414700293047627,-0.213267188307358;
                     0.647380884516047,0.409334430455455,-0.162501941548495,0.348441199952281,-0.325347780360491;
                     -0.425631681660351,-0.952635997119978,0.690051897857806,0.349254416663714,1.94439779196628;
                     1.04858076053644,0.317317472324612,0.555756771412105,-0.729247267629503,-0.571773218349456;
                     0.660707086367175,0.0780200809780775,-1.12025500148841,0.326840248762991,-0.250032278035044;
                     2.50877247318511,1.32438544916022,-1.53269301392951,-0.514881632926476,-1.56931547034070;
                     1.06345963904102,-0.213170486421586,-1.09786776954067,-0.896446150502486,-0.477382663694067;
                     1.15692165332739,-0.134478644555426,-1.41577332156794,-1.20326818641502,-1.33797665356226;
                     0.0529788273256128,-1.17135581792030,0.0595705885660458,1.03781563948524,0.0302990237881831];

randMatrix(:,6:10) = [0.853086767518578,0.496684391636530,1.05330457914955,-1.60580220462075,-0.652771012058928;
                      0.404253465109360,1.08224059616070,-0.748876752543159,0.661536244123409,0.477227285007673;
                      -0.700620213418206,0.970447788354572,-0.936326497116021,2.13850225382851,-0.0713196500274540;
                      -1.63054289274725,-0.568569570389605,-1.26908677286849,0.541139413417132,-0.938301288268817;
                      1.46001318514265,0.809972073318538,0.497980623046732,-1.54087718185788,0.161363534650636;
                      2.05004273293787,0.173247371832088,2.78908112764392,-0.203142794821834,-0.268182881913388;
                      0.120500599699389,-0.505542568515379,0.727572036915723,-0.499965221779343,-0.409872647756742;
                      -0.989901604221187,-1.19330579089020,-0.773064102152237,0.383023912780829,-0.711322710526139;
                      1.19777147431488,0.646970938881150,0.836633754476153,0.412035378111750,0.0614454835840975;
                      -0.592656218169254,-0.353622599299341,-1.12833031008360,0.405492550516440,-1.84612923609859;
                      -0.469809363224455,0.0464345266878230,-1.42447009141638,-0.363780750881015,-0.398333119478132;
                      0.886377377072694,-0.792947502201808,0.717442315507358,-0.599272040041946,-0.543548121612387;
                      -1.38521980111084,-1.55051447477956,-0.777905520185398,-0.589588991418034,-0.911898504288660;
                      -1.95675395423192,0.171586355268628,0.315985880708604,0.853540834064481,0.652698590995241;
                      0.420683700201250,-0.0621391248043618,1.40653513807679,-1.85300807141960,-0.734271258590912;
                      0.400737998146962,1.19902787729197,0.401124635731864,-0.207303159647625,0.540633089814446;
                      0.0951421578641687,0.801704070327277,0.929660284419656,0.270378202689916,0.975840884581828];
```

MATLAB .mat file: randvars2

Again, clear the variables in the MATLAB workspace by entering `clear all` in the command window, then type `clc` to clear the screen. In the directory (folder) where you saved the file 'MFS1DStefan' copy and paste the following two commands into the command window, which will generate a 32×10 matrix called `randMatrix`, then type `save randvars2` into the command window.

```
randMatrix(:,1:5) = [-0.433592022305684,1.37029854009523,-0.615601881466894,0.100092833139322,0.281984063670556;
                     0.342624466538650,-1.71151641885370,0.748076783703985,-0.544528929990548,0.0334798822444514;
                     3.57839693972576,-0.102242446085491,-0.192418510588264,0.303520794649354,-1.33367794342811;
                     2.76943702988488,-0.241447041607358,0.888610425420721,-0.600326562133734,1.12749227834159;
                     -1.34988694015652,0.319206739165502,-0.764849236567874,0.489965321173948,0.350179410603312;
                     3.03492346633185,0.312858596637428,-1.40226896933876,0.739363123604474,-0.299066030332982;
                     0.725404224946106,-0.864879917324457,-1.42237592509150,1.71188778298155,0.0228897927516298;
                     -0.0630548731896562,-0.0300512961962686,0.488193909859941,-0.194123535758265,-0.261995434966092;
                     0.714742903826096,-0.164879019209038,-0.177375156618825,-2.13835526943994,-1.75021236844679;
                     -0.204966058299775,0.627707287528727,-0.196053487807333,-0.839588747336614,-0.285650971595330;
                     -0.124144348216312,1.09326566903948,1.41931015064255,1.35459432800464,-0.831366511567624;
                     1.48969760778547,1.10927329761440,0.291584373984183,-1.07215528838425,-0.979206305167302;
                     1.40903448980048,-0.863652821988714,0.197811053464361,0.960953869740567,-1.15640165566400;
                     1.41719241342961,0.0773590911304249,1.58769908997406,0.124049800003193,-0.533557109315987;
                     0.671497133608081,-1.21411704361541,-0.804465956349547,1.43669662271894,-2.00263573588306;
                     -1.20748692268504,-1.11350074148676,0.696624415849607,-1.96089999936503,0.964229422631628;
                     0.717238651328839,-0.00684932810334806,0.835088165072682,-0.197698225974150,0.520060101455458;
                     1.63023528916473,1.53263030828475,-0.243715140377952,-1.20784548525980,-0.0200278516425381;
                     0.488893770311789,-0.769665913753682,0.215670086403744,2.90800803072936,-0.0347710860284830;
                     1.03469300991786,0.371378812760058,-1.16584393148205,0.825218894228491,-0.798163584564142;
                     0.726885133383238,-0.225584402271252,-1.14795277889859,1.37897197791661,1.01868528212858;
                     -0.303440924786016,1.11735613881447,0.104874716016494,-1.05818025798736,-0.133217479507735;
                     0.293871467096658,-1.08906429505224,0.722254032225002,-0.468615581100624,-0.714530163787158;
                     -0.787282803758638,0.0325574641649735,2.58549125261624,-0.272469409250188,1.35138576842666;
                     0.888395631757642,0.552527021112224,-0.666890670701386,1.09842461788862,-0.224771056052584;
                     -1.14707010696915,1.10061021788087,0.187331024578940,-0.277871932787639,-0.589029030720801;
                     -1.06887045816803,1.54421189550395,-0.0824944253709554,0.701541458163284,-0.293753597735416;
                     -0.809498694424876,0.0859311331754255,-1.93302291785099,-2.05181629991115,-0.847926243637934;
                     -2.94428416199490,-1.49159031063761,-0.438966153934773,-0.353849997774433,-1.12012830124373;
                     1.43838029281510,-0.742301837259857,-1.79467884145512,-0.823586525156853,2.52599969211831;
                     0.325190539456198,-1.06158173331999,0.840375529753905,-1.57705702279920,1.65549759288735;
                     -0.754928319169703,2.35045722400204,-0.888032082329010,0.507974650905946,0.307535159238252];

randMatrix(:,6:10) = [-1.25711835935205,-0.209713338388737,0.391894209432449,-0.590034564205222,0.00116208348351385;
                      -0.865468030554804,0.625190357087626,-1.25067890682641,-0.278064163765309,-0.0708372131604802;
                      -0.176534114231451,0.183227263001437,-0.947960922331432,0.422715691220478,-2.48628392070328;
                      0.791416061628634,-1.02976754356662,-0.741106093940412,-1.67020069785047,0.581172322675923;
                      -1.33200442131525,0.949221831131023,-0.507817550278174,0.471634326416303,-2.19243491996591;
                      -2.32986715580508,0.307061919146703,-0.320575506600239,-1.21284719967446,-2.31928030664330;
                      -1.44909729283874,0.135174942099456,0.0124690413616180,0.0661900484246114,0.0799337102984397;
                      0.333510833065806,0.515246335524849,-3.02917734140415,0.652355888661374,-0.948480983570505;
                      0.391353604432901,0.261406324055383,-0.457014640871583,0.327059967177088,0.411490621423374;
                      0.451679418928238,-0.941485770955434,1.24244840639074,1.08263350423676,0.676977805684030;
                      -0.130284653145721,-0.162337672803828,-1.06670139898475,1.00607711081905,0.857732545205355;
                      0.183689095861942,-0.146054634331526,0.933728162671239,-0.650907736597753,-0.691159125382991;
                      -0.476153016619074,-0.532011376808821,0.350321001356112,0.257056157433969,0.449377623166851;
```

0.862021611556922,1.68210359466318,-0.0290057637087263,-0.944377806404219,0.100633350315076;
-1.36169447087075,-0.875729346160017,0.182452167505983,-1.32178852139256,0.826069998469923;
0.455029556444334,-0.483815050110121,-1.56505601415073,0.924825933493706,0.536157079925919;
-0.848709379933659,-0.712004549027423,-0.0845394798177242,4.98490752508133e-05,0.897888425985076;
-0.334886938964048,-1.17421233145682,1.60394635060288,-0.0549189146094067,-0.131937867924581;
0.552783345944550,-0.192239517539275,0.0983477746401080,0.911127265653860,-0.147201456151267;
1.03909065350496,-0.274070229932602,0.0413736134896147,0.594583697409052,1.00777340530544;
-1.11763868326521,1.53007251442410,-0.734169112696739,0.350201173874535,-2.12365546241575;
1.26065870912090,-0.249024742513714,-0.0308137300123200,1.25025122830500,-0.504586405514010;
0.660143141046978,-1.06421341288933,0.232347012624477,0.929789458557716,-1.27059444980866;
-0.0678655535426873,1.60345729812004,0.426387557408945,0.239763257058580,-0.382584802707648;
-0.195221197898754,1.23467914689078,-0.372808741723504,-0.690361103111226,0.648679262048621;
-0.217606350143192,-0.229626450963181,-0.236454583757186,-0.651553641750281,0.825727149241758;
-0.303107621351741,-1.50615970397972,2.02369088660305,1.19210187053127,-1.01494364268014;
0.0230456244251053,-0.444627816446985,-2.25835397049619,-1.61183038867781,-0.471069912683167;
0.0512903558487747,-0.155941035724769,2.22944568045690,-0.0244619366359185,0.137024874130050;
0.826062790211596,0.276068253931536,0.337563700613106,-1.94884717689890,-0.291863375753573;
1.52697668673337,-0.261163645776479,1.00006081958912,1.02049801445265,0.301818555261006;
0.466914435684700,0.443421912904091,-1.66416447498706,0.861716302393419,0.399930942955802];