Aspects of the Laplace transform isotherm migration method

Linda Radford

A thesis submitted in partial fulfilment of the requirements of the University of Hertfordshire for the degree of

Doctor of Philosophy

The programme of research was carried out in the
School of Physics, Astronomy & Mathematics
Faculty of Engineering and Information Sciences
University of Hertfordshire

May 2008

If I have seen further
it is by standing
on the shoulders of giants

Sir Isaac Newton (1642-1727)

Acknowledgements

It is my greatest sadness as I write this, that my father is not here to see the completion of my work, for he would have been so proud. I owe thanks to both him and my mother for the encouragement and support they gave me in my education, without which I am sure I would not have achieved so much.

I am grateful to my husband Russell for believing that I could gain a PhD and for never losing faith in me even when I had doubts about my ability. I am sure I was not an easy person to live with especially when I was nearing the end of the work, but he was always there for me.

I would also like to thank my friend and supervisor Professor Alan Davies for suggesting I attempt a research programme. This was something which I had never contemplated and I will be forever indebted to him for inviting me to apply for a research degree at the University of Hertfordshire.

I must also thank my other supervisor, Dr Steve Kane, for all his help and for offering a corner of his office where I could study without interruption or distraction. Having a quiet space meant I was able to have many hours of quality time without which it would have been difficult to complete my work.

I am deeply grateful to my friend Dr Diane Crann, who began her research degree at the same time and it was she who encouraged me to keep going after I had suspended my work for family reasons. Without her support I am sure I would not have resumed my research.

I would like to thank my examiners, Professor Bruce Christianson and Professor Choi-Hong Lai, for their interest in my work and for giving me the opportunity to talk about it at length. They put me at ease during my oral examination, which allowed me to express my ideas clearly.

Finally I must mention Dr Ian Nicolson, for it was he who suggested

I study mathematics with the Open University when my knowledge was found to be lacking during an interview for an astronomy course which I was hoping to join in 1978. I think neither he nor I would have imagined then that I would have come so far and been awarded a PhD for my work in mathematics.

Abstract

There are many different methods available for the solution of the heat equation and the choice of which to use is dependent upon the nature of the problem and the specific regions of the domain where the temperature is required. In the case of melting or freezing problems it is usual for the position of the boundary, at which change of physical state (phase change) occurs, to be of greater interest than the temperature at particular points. Again there are several solution methods enabling the tracking of the moving interface between the physical states of the material.

For this work we begin with the isotherm migration method, which first appeared in the 1970s but is less frequently cited now. We first solve problems in one dimension with no phase change using the isotherm migration method, which is in itself new work, since all references we have found allude to it as a tool for the solution of phase change problems. We test the method using a variety of examples to explore the difficulties and challenges it produces, and we find it to be robust and tolerant of errors.

We then combine it with the Laplace transform method, a well-established technique for solving ordinary and partial differential equations, in which the number of independent variables is reduced by one. The solution is then transformed back into the time domain using a suitable numerical process.

The Laplace transform isotherm migration method is a new process, not mentioned previously to our knowledge, and it produces results which are comparable with the isotherm migration method. The new process is applied to one-dimensional phase change problems,where we find that due to the mathematics at the phase change boundary, we are required to make a modification to the usual manner of operating the Laplace transform. This is novel as far as we are aware.

Our method is applied to a variety of problems and produces satisfactory

results. We then move on to a two-dimensional setting where we find the situation to be much more complex and challenging, as it requires interpolation and curve-fitting processes.

Finally we examine the possiblity of speeding up the calculation time using the Laplace transform isotherm migration method by setting problems in a parallel environment and using an MPI platform. This has not been previously attempted and we are able to show a measure of success in our objective.

Contents

List of Figures

List of Tables

Glossary

The terms included in this glossary refer to items defined in the Laplace transform isotherm migration method. For clarity we do not include symbols used in chapter 2, the review of other methods.

Bi	dimensionless Biot number, $h_c l_c / K$
c	specific heat
$f(t)$	function of time
$\bar{f}(\lambda)$	transformed function of time
$g(x, y)$	shape function of boundary condition
h	difference in temperatures of adjacent isotherms
h_c	heat transfer coefficient
K	thermal conductivity
K_{liq}	thermal conductivity in the liquid phase
K_{sol}	thermal conductivity in the solid phase
l	length
l_c	characteristic length, the volume of a body divided by its surface area
L	latent heat of fusion
$\mathcal{L}[f(t)]$	Laplace transform
M	number of Stehfest parameters
p	number of processors
s	dimensionless Stefan number, L/c
S_p	speed-up
t	time
$\tilde{t}$	dimensionless t co-ordinate
t_0	initial time
T	specific value of t at which a solution is required

u	temperature
$\tilde{u}$	dimensionless u co-ordinate
u_0	initial temperature
$\tilde{u}_0$	dimensionless u_0 co-ordinate
u_f	fusion temperature
$u_F(x, y, t) = 0$	contour on freezing front
u_i	isotherm of specific temperature
u_L	temperature of left boundary
u_{liq}	temperature in the liquid phase
$\tilde{u}_L$	dimensionless temperature of left boundary
u_R	temperature of right boundary
$\tilde{u}_R$	dimensionless temperature of right boundary
u_{sol}	temperature in the solid phase
U_i	approximation to u at grid point (x_i)
$U_{i,j}$	approximation to u at grid point (x_i, t_j)
V	constant temperature of liquid phase
u_v	steady temperature outside wall at x_L
u_w	temperature of wall at x_L
$\tilde{u}_w$	dimensionless temperature of wall at x_L
v	temperature function $f_1(\tilde{x}_1, \tilde{t})$
v_1	temperature of ice
v_1'	transformed temperature of ice
v_2	temperature of water
v_2'	transformed temperature of water
w	temperature function $f_2(\tilde{x}_2, \tilde{t})$
w_j	Stehfest weight
$\tilde{x}$	dimensionless x co-ordinate
x_0	position of isotherm with temperature u_0

$\bar{x}$	position under Laplace transform
$\tilde{x}_0$	dimensionless position of isotherm with temperature $\tilde{u}_0$
$\tilde{X}$	approximation to dimensionless position $\tilde{x}$
α	thermal diffusivity
β	constant
$\varepsilon(t)$	parameter in shape function of freezing front
θ	difference in temperature of wall $\tilde{u}_w^{(n)}$ and next isotherm $\tilde{u}_i$ in convection case
λ	Laplace transform parameter
λ_j	Stehfest Laplace transform parameter
ρ	density
ω	parameter used in interpolation

Chapter 1

Introduction

We begin our story with some background about what led us to choose the Laplace transform isotherm migration method as a topic for our research.

Work had been ongoing in the Department of Mathematics at the University of Hertfordshire around some of the various solution methods for the diffusion equation; examples of these are Davies (1993), Davies and Mushtaq (1996), Davies and Mushtaq (1998) and Honnor *et al.* (2003).

A paper written by Crank and Phahle (1973), in which the authors modelled a diffusion problem, that of a melting block of ice, and produced a solution using a technique known as the 'isotherm migration method', gave a different slant to the usual methods of solution to the heat equation, in that the movement of regions of the same temperature could be tracked rather than finding the temperature at particular points in the domain. The method appeared to be attractive, it seemed to be simple to set up the equations and produced results which compared well with the analytic solutions. A search of the literature showed that the method was the subject of some interest in the late 1970's. Crank and Gupta (1975) and Crank and Crowley (1978 and 1979) produced papers developing the method, but it became less frequently used although from time to time it still appears (Kutluay and Esen, 2004). The method was examined more

closely, to look at any advantages and disadvantages and to try to see why it was less used than other methods. Our intention was to develop the method further and produce a solution using the Laplace transform, which had also been a topic of research at the University of Hertfordshire, (Crann 1996, Honnor and Davies 2002 and Crann 2005), and which could then be extended from one-dimensional to two-dimensional problems. A finite difference method was chosen by Crank and Phahle (1973) in their solution to the melting ice block and at the outset we hoped to be able to provide an alternative in a boundary element method solution for the two-dimensional case. This was not possible, as we shall discuss later, since the boundary in the transformation of the problem is not always complete.

Another thread to our research would be to implement the Laplace transform isotherm migration method in a parallel environment, since the Department of Mathematics at the University of Hertfordshire/Hatfield Polytechnic has had an interest in parallel computation for more than twenty years. In the following we indicate some of the areas of interest and the parallel environments involved:

The early work was done on the ICL DAP, a distributed array machine with 4096 processors, an SIMD (Single Instruction Multiple Data) machine, (Dixon and Ducksbury 1985 and Davies 1989).

In the mid 1990's the Department had access to a small transputer network comprising four T800 transputers configured to run a version of FORTRAN. This is an MIMD (Multiple Instruction Multiple Data) environment running as SIMD. Davies and Mushtaq (1995 and 1996), Mushtaq and Davies (1996) and Davies *et al.* (1997) refer to work using the transputer.

The next progression was to use PVM and HPF software on a network of workstations including SUN workstation clusters and PC clusters. These are MIMD environments. Papers were written by Davies *et al.* (1997) and Mushtaq and Davies (1997a and 1997b) concerning work done on this

system.

In 1998 the Department was given an nCUBE machine with sixty-four processors arranged in a hypercube configuration. This provided an MIMD environment and was used by Davies and Mushtaq (1998), Crann *et al.* (1998) and Honnor and Davies (2002) in their work.

The most recent interest has been in the area of time domain-decomposition methods, which are MIMD in nature, for diffusion processes, in particular the use of the Laplace transform. Examples of work in this environment include Crann *et al.* (1998), Davies *et al.* (2004), Crann (2005) and Crann *et al.* (2007).

This thesis begins with a derivation of the heat equation and a review of some of the many methods for its solution. Simple heat conduction problems in one dimension are then solved with the isotherm migration method as in the model suggested by Crank and Phahle (1973) and we consider various scenarios in order to gain information regarding the robustness of the method and to see its limitations. This leads on to new work in which the Laplace transform and the isotherm migration method are combined and a comparison is made with the results obtained in the previous examples. The next step is to consider examples of heat problems with phase change, in particular melting and freezing, which are solved using the new method, before extending this to examples in two dimensions. Finally the work is carried out in a parallel environment with a view to increasing efficiency in calculation time.

Chapter 2

The construction of the heat equation and methods which have been used to solve it

2.1 The heat equation

We will be using the heat equation as the basis of our work and so we begin by giving a derivation in its most general form. The construction of the heat equation is to be found in many publications. Examples include Carslaw and Jaeger (1959), Spiegel (1959), Weinberger (1965), Kreider *et al.* (1966) and Crank (1979).

We consider a material, with density ρ, and specific heat c, which occupies a region of space, V, and which is bounded by a surface S. If the temperature of the material at any point in V is $u(\mathbf{r}, t)$, where $\mathbf{r}$ is the usual position vector, then the total heat energy contained in the solid is

$$\int_V \rho c u \, dV.$$

Heat may only enter or leave the region by flowing across the boundary S. We consider the heat flux vector, $\mathbf{q}$, which represents the rate of heat flow

at a point per unit area. If we have a small element of the surface area dS with outward unit normal vector $\hat{\mathbf{n}}$, then the rate of heat flow outward through this element of surface area is $\mathbf{q}.\hat{\mathbf{n}}\, dS$ and so over the whole surface the total rate of heat flow outwards is

$$\int_S \mathbf{q}.\hat{\mathbf{n}}\, dS$$

According to the law of conservation of energy, the rate at which the total heat energy within the region V changes must balance with the amount of heat crossing the boundary. Therefore

$$\frac{d}{dt}\int_V \rho cu\, dV + \int_S \mathbf{q}.\hat{\mathbf{n}} dS = 0$$

because, if the flux of heat out of the region is positive, then the total energy inside the region will decrease, so its derivative will be negative. We now apply the divergence theorem to the flux integral and take the time derivative inside the volume integral to obtain

$$\int_V \frac{\partial}{\partial t}(\rho cu)\, dV + \int_V \mathrm{div}\,\mathbf{q}\,\mathrm{dV} = 0$$

or

$$\int_V \left[\frac{\partial}{\partial t}(\rho cu) + \mathrm{div}\,\mathbf{q}\right] dV = 0$$

This expression has to be true for any volume V, which means that the integrand must be zero everywhere. Therefore we must have

$$\frac{\partial}{\partial t}(\rho cu) + \mathrm{div}\mathbf{q} = 0$$

From Fourier's law of heat flow we know that

$$\mathbf{q} = -K\mathbf{grad}u \tag{2.1}$$

where K is the thermal conductivity, which is not necessarily constant, and

$$\mathrm{div}\,(K\mathbf{grad}u) = \frac{\partial}{\partial t}(\rho cu)$$

This is the most general form of the heat equation.

In our work, we consider the case where K is constant and ρ and c are independent of time and we use the following form of the heat equation, which is the most convenient for our purposes:

$$\nabla^2 u = \frac{1}{\alpha}\frac{\partial u}{\partial t} \tag{2.2}$$

where $\alpha = \frac{K}{pc}$ is the thermal diffusivity.

The partial differential equation for diffusion problems is set up in a similar manner (Crank 1979) and its form is

$$\frac{\partial^2 C}{\partial x^2} = \frac{1}{D}\frac{\partial C}{\partial t}$$

C is the concentration of the diffusing substance and D is the diffusion coefficient and the law equivalent to Fourier's law in equation (2.1) is Fick's law

$$\mathbf{q} = -D\mathbf{grad}C$$

The heat equation is a partial differential equation and in the next section we look at the classification of partial differential equations, because the nature of the solution depends on the classification of the equation.

2.1.1 Classification of partial differential equations

In order to classify the equations, we will consider the linear partial differential operator $\mathcal{L}$ in two independent variables for simplicity, which can be extended to three or more variables (Weinberger 1965). We define the linear partial differential operator as

$$\mathcal{L}[u] \equiv A(x,y)\frac{\partial^2 u}{\partial y^2} + B(x,y)\frac{\partial^2 u}{\partial x \partial y} + C(x,y)\frac{\partial^2 u}{\partial x^2} + D(x,y)\frac{\partial u}{\partial y} + E(x,y)\frac{\partial u}{\partial x} + F(x,y)\,u$$

and if

$$\mathcal{L}[u] = G$$

where $G(x, y)$ is some function independant of u, this is called a *linear partial differential equation.* If

$$\mathcal{L}[u] = 0$$

then the equation is called a *homogeneous* equation. If any of the coefficients A, B, C, D, E and F are functions of u or its derivatives as well as x and y, then the equation becomes a *non-linear partial differential equation.*

If $B^2 - AC$ is negative, then the partial differential equation is said to be *elliptic*, an example of this being the Laplace equation

$$\nabla^2 u = 0$$

Elliptic equations are typically used for steady-state and potential problems and their solution depends only on values known at the boundary. If we consider a bounded region where C denotes the boundary, then the most commonly occurring boundary conditions are:
Dirichlet, where

$$u(\mathbf{r}) = u_1(s) \quad \text{on} \quad C_1$$

Neumann, where

$$\frac{\partial u}{\partial n} = q_2(s) \quad \text{on} \quad C_2$$

Robin, where

$$\frac{\partial u}{\partial n} + \sigma(s)u = h_3(s) \quad \text{on} \quad C_3$$

where σ is a function of s, which defines the position on $C = C_1 + C_2 + C_3$ and $\mathbf{r}$ is the usual position vector of a point on the boundary.

If $B^2 - AC$ is positive, then we have a *hyperbolic* equation, an example being the wave equation

$$\frac{\partial^2 u}{\partial t^2} = c\frac{\partial^2 u}{\partial x^2}$$

where c is the speed of propagation of the wave. Typically hyperbolic equations are concerned with the propogation of information through a system,

and in order to solve these equations, we need as well as boundary conditions, initial conditions of the form

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

and

$$\frac{\partial u}{\partial t}(x,0) = g(x)$$

When $B^2 - AC = 0$ we have a parabolic problem, an example of which is the heat equation whose form is shown in equation (2.2). Parabolic equations require boundary conditions and an initial condition of the form

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

The difference between parabolic and hyperbolic equations is that hyperbolic problems have a finite propagation speed, whereas in parabolic problems the effects of propagation are felt immediately throughout the domain.

2.1.2 Well-posed problems

In addition to the conditions described above, to be able to solve a partial differential equation, we should also have a well-posed problem, in the sense described by Hadamard (1923). We have a well-posed problem if

1. a solution to the problem exists.
2. the solution is unique.
3. the solution depends continuously on the problem data, that is, small changes in data yield small changes in the solution.

For an elliptic problem, we need to have the partial differential equation defined in the interior of some region, with the solution subject to a single boundary condition at each point of the boundary. The type of boundary

condition may vary from one point on the boundary to another, but only one condition may be specified at any point.

Hyperbolic equations require the same constraints on the boundary conditions as elliptic equations, but in addition they require two initial conditions, one to describe the initial state of the system and the other to describe the initial velocity.

Parabolic equations also require the boundary conditions described for the previous two cases, but they need only a single initial condition specifying the state of the system at $t = 0$.

2.2 Methods of solution for the heat equation

2.2.1 Analytic solutions

Assuming the thermal diffusivity α to be constant, we write the heat equation as

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2} \tag{2.3}$$

The first method we consider is the method of separation of variables, mentioned by Crank (1979) . In this case we look for solutions of the form

$$u = X(x)\,T(t) \tag{2.4}$$

where $X(x)$ and $T(t)$ are functions of x and t respectively. Substituting equation (2.4) into equation (2.3) we get

$$X \frac{dT}{dt} = \alpha T \frac{d^2 X}{dx^2}$$

and on separating the variables we have

$$\frac{1}{T}\frac{dT}{dt} = \frac{\alpha}{X}\frac{d^2 X}{dx^2}$$

so that the left hand side depends only on t and the right hand side depends only on x. Both sides therefore must be equal to the same constant and to

simplify the algebra we write this as $-\mu^2\alpha$. The solutions are then

$$T = T_0 e^{-\mu^2 \alpha t}$$

where T_0 is a constant and

$$X = A \sin \mu x + B \cos \mu x$$

with A and B constants. A solution of equation(2.3) is then

$$u = T_0 (A \sin \mu x + B \cos \mu x) \exp\left(-\mu^2 \alpha t\right) \tag{2.5}$$

and since the heat equation is a linear equation, the most general solution is obtained by summing solutions of equation (2.5) type to get

$$u = \sum_{m=1}^{\infty} T_{0,m} (A_m \sin \mu_m x + B_m \cos \mu_m x) \exp\left(-\mu_m^2 \alpha t\right)$$

The constants A_m,B_m,$T_{0,m}$ and μ_m are determined from the initial and boundary conditions and again, examples are to be found in Crank (1979). Another analytic method is the use of similarity solutions.

We define the dimensionless variable

$$\eta = \frac{x}{\sqrt{\alpha t}}$$

and then look for solutions of the form

$$u\,(x,t) = t^p g\,(\eta) \tag{2.6}$$

where the number p and the function $g(\eta)$ are to be found. Substituting equation (2.6) into equation (2.3) we find

$$t^{p-1}\left(pg - \frac{\eta}{2} g' - g''\right) = 0$$

which implies

$$g'' + \frac{\eta}{2} g' = pg \tag{2.7}$$

subject to appropriate boundary conditions using equation (2.6). This is difficult to solve in closed form for arbitrary values of p.

We can find the solution in different forms and we show the solution for two cases of p.

For $p = 0$:

Equation (2.7) may be solved to give

$$g'(\eta) = Ae^{\frac{-\eta^2}{4}}$$

where A is a constant. Integrating this gives

$$g(\eta) = A\int_{-\infty}^{\eta} e^{\frac{-\eta^2}{4}} d\eta'$$

which gives a full solution for $u(x,t)$

$$u(x,t) = A\int_{-\infty}^{\frac{x}{\sqrt{\alpha t}}} e^{\frac{-\eta'^2}{4}} d\eta' = 2a\sqrt{\pi}\text{erf}\left(\frac{x}{2\sqrt{\alpha t}}\right) \tag{2.8}$$

where the error function $\text{erf}(\xi)$ is defined as

$$\text{erf}(\xi) = \frac{1}{\sqrt{\pi}}\int_{-\infty}^{\xi} e^{-y^2} dy$$

For $p = -\frac{1}{2}$:

Another solution can be obtained by defining $G(\eta) = ge^{\frac{\eta^2}{4}}$ and then equation (2.7) can be written as

$$G'' - \frac{\eta}{2}G' = \left(p + \frac{1}{2}\right)G$$

This has the trivial solution

$$G(\eta) = b = \text{constant}$$

Therefore

$$g(\eta) = be^{\frac{-\eta^2}{4}}$$

which gives a full solution for $u(x,t)$ in the form

$$u(x,t) = bt^{-\frac{1}{2}}e^{-\frac{x^2}{4\alpha t}} \tag{2.9}$$

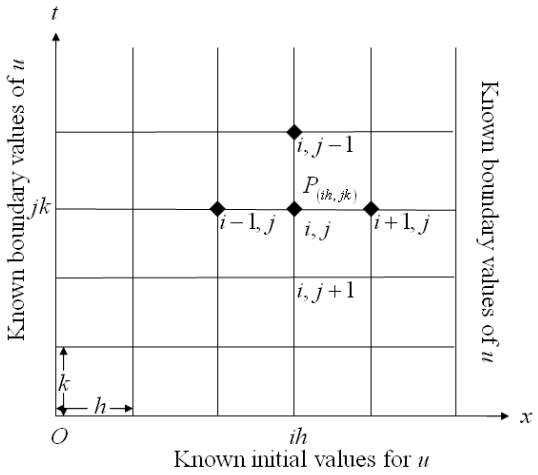

Figure 2.1: Grid for using the finite difference method

The constants A in equation (2.8) and b in equation (2.9) need to be found using the boundary and initial conditions.

Crank (1979) presents several mathematical solutions, mostly in the form of infinite series, but he notes that their use is restricted to simple geometries and constant thermal properties such as the thermal diffusivity. For most problems we need to turn to numerical methods to evaluate their solutions and we describe some well-known methods as follows.

2.2.2 The finite difference method

For a rectangular domain, probably the simplest method available is the finite difference method. This method is described fully by Smith (1978). Consider an example of heat conduction in which the heat equation is solved to find the temperature u at a distance x units of length from one end of a thermally insulated bar of length l after t seconds of heat conduction. In such problems the temperatures at the ends of the rod are often known for all time and these are the boundary conditions. Usually we also know the temperature distribution along the bar at some particular starting time, and

this is the initial condition. The solution to the problem gives the values of u for values of x between 0 and l and values of t from zero to infinity. Figure 2.1 shows the area of integration in the x, t-plane, that is the infinite area bounded by the x-axis and the parallel lines $x = 0$ and $x = l$. We see that the area is covered by a grid, and the size of a step, or mesh size, in the x direction is assigned the variable h, while the mesh size in the t direction is k. We use a truncated Taylor series to obtain an approximation to the temperature at the point P.

We write the approximated temperature as $U_{i,j}$, the subscripts denoting the position of the point on the grid. The subscript i tells us which row we are in and the subscript j, which column. So by knowing the mesh size we can pinpoint the position on the grid.

The temperature at a typical point such as $P_{i,j}$ is found by reference to its neighbouring points. Here we see a central difference approximation in space and a forward difference approximation in time. The temperature in this case is given by

$$U_{i,j+1} = U_{i,j} + r(U_{i-1,j} - 2U_{i,j} + U_{i+1,j})$$

where $r = \frac{k}{h^2}$ and $U_{i,j}$ is the approximation to the actual value $u_{i,j}$. The method has a limitation which is that the value of r is critical.

Smith (1978) describes an analysis of the stability of the method and finds that for values $r < \frac{1}{2}$ the finite difference solutions agree reasonably well with the analytical solutions, but for values $r > \frac{1}{2}$ the problem becomes unstable in the sense that errors increase without limit and this means that to use the method we are forced to take a large number of small time steps. The instability in the explicit method may be overcome by using the Crank-Nicolson implicit method (1947), which is unconditionally stable. The expression for this approximation is

$$-rU_{i-1,j+1} + (2 + 2r)\, U_{i.j+1} - rU_{i+1,j+1} = rU_{i-1,j} + (2 - 2r)\, U_{i,j} + rU_{i+1,j}$$

The left hand side contains three unknowns at time level $j+1$ and on the right hand side we have the three known values at time level j. This involves solving a set of simultaneous equations and obviously more work is involved at each time step, but the solution does remain stable for all values of r. We can thus proceed with larger and therefore fewer time steps, but we bear in mind that in developing the formula from the Taylor series, we have neglected higher order terms as being comparatively small in value which may influence the accuracy if the steps are too large.

The advantage of using the finite difference method is that for rectangular domains it is easy to discretise using the grid construction, and the solution procedure is simple to operate. However the disadvantage is that it is not suitable for domains with a non-rectangular shape.

2.2.3 The finite element method

The finite element method has been used for some time by engineers interested in stress analysis problems and steady-state potential problems, but has also been used for transient heat problems. Full accounts of the method are to be found in Davies (1986) and Zienkiewicz and Taylor (2000). In this method, the domain is covered with a mesh, often triangular, in which the triangles may be of varying size thereby giving a better approximation to domains having an irregular shape, as shown in figure 2.2. We outline the theory, full details of which can be found in Davies (1986). Suppose that u satisfies Poisson's equation

$$\nabla^2 u = b\,(x, y) \quad \text{in} \quad D$$

subject to the Dirichlet condition

$$u = h\,(s) \quad \text{on} \quad C_1$$

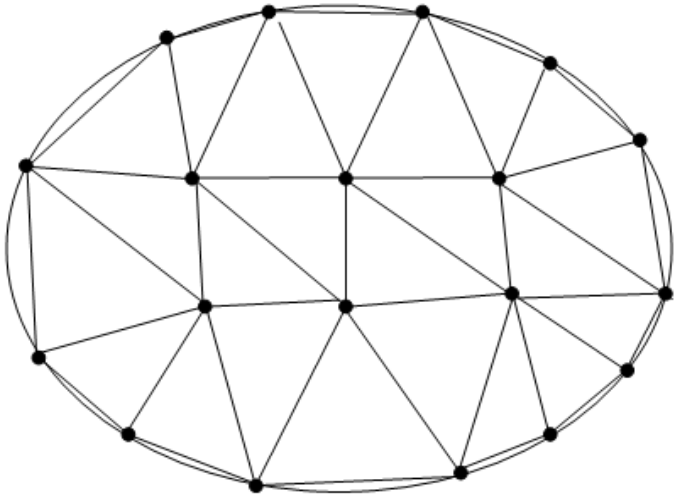

Figure 2.2: Typical grid for the finite element method

and the Robin condition

$$\frac{\partial u}{\partial n} + \sigma(s)\, u = g(s) \quad \text{on} \quad C_2$$

where D is a two-dimensional region bounded by a closed curve $C = C_1 + C_2$. A Neumann boundary condition is a special case with $\sigma \equiv 0$.

Suppose that D is subdivided into m elements, D^e, over each of which u is interpolated from its nodal values U_j, there being n nodes in all. A piecewise polynomial approximation to u of the form is obtained

$$u = \sum_{j=1}^{n} w_j(x, y)\, U_j$$

where $\{w_j(x, y) : j = 1, 2, \ldots, n\}$ is a set of linearly independent basis functions. A set of linear algebraic equations for the nodal values U_i $(i = 1, 2, \ldots, n)$ follows. The Dirichlet boundary condition is an essential condition which must be enforced, while the Robin condition is a natural condition and the values of U at the nodes which lie on C_2 will be found as part of the solution process.

The method leads to a system of algebraic equations having the form

$$\mathbf{KU} = \mathbf{F} \tag{2.10}$$

where $\mathbf{K}$ is a symmetric banded matrix, $\mathbf{U}$ is the vector containing the nodal values and $\mathbf{F}$ is a vector of known quantities obtained from non-homogeneous terms in the boundary-value problem. The global matrix $\mathbf{K}$ is formed using element matrices $\mathbf{k}$ by using the terms from the elements which contain both i and j and placing these into the i, j position in the global matrix. The matrix $\mathbf{F}$ is formed in a similar manner. Where a node, r, lies on the boundary and has an essential boundary condition, equation r is removed from the set of equations (2.10) and $U_r = h(s_r)$ is placed in the remaining equations. So if there are p boundary nodes with an essential condition, the global stiffness matrix will be of order $N \times N$ where $N = n - p$. The terms in the element matrices are of the following forms:

$$k_{i,j} = \iint\limits_{D^e} \left(\frac{\partial w_i}{\partial x}\frac{\partial w_j}{\partial x} + \frac{\partial w_i}{\partial y}\frac{\partial w_j}{\partial y} \right) \mathrm{d}x\mathrm{d}y + \int\limits_{C_2^e} \sigma w_i w_j \mathrm{d}s$$

$$f_{i,j} = \iint\limits_{D^e} f w_i \mathrm{d}x\mathrm{d}y + \int\limits_{C_2^e} g w_i \mathrm{d}s$$

where C_2^e is that part of the boundary of element e which lies on C_2.

Other meshes are possible, for example those using isoparametric elements, so that a curved boundary can be more accurately represented because a polynomial is used to approximate it.

2.2.4 The boundary integral equation method

By way of example we consider a region D on which Laplace's equation

$$\nabla^2 u = 0$$

holds, this region being bounded by a closed curve. The boundary conditions may be Dirichlet, Neumann or Robin (mixed condition), and we choose a

fundamental solution which satisifies the Laplace equation on the region, usually

$$u^*(r) = -\frac{1}{2\pi} \ln r$$

where r is the distance between one point and another. Application of Green's second theorem leads to an expression in which the potential at any point may be expressed as an integral equation.

We have already stated in subsection 2.1.2 that for a well-posed problem, only one condition may be specified at each point of the boundary. When the boundary integral expression has been formulated, we are in a position to find both the potential and flux at all points on the boundary and from there we can find the potential at any internal points required.

2.2.5 The boundary element method

Jaswon and Symm (1977) discuss the numerical solution of boundary integral equations in which they approximate the boundary of the region by a polygon and they choose the solution to the problem and its normal derivative on the boundary to be constants on each polygon side. Fairweather *et al.* (1979) consider a method in which the approximations to the solution and the flux on the boundary of the region are generated from piecewise quadratic polynomial functions. However, the familiar term 'boundary element method' was first used by Brebbia and Dominguez (1977) and a full description of the method is to be found in Brebbia and Dominguez (1989).

The method involves collocation between a base node and a target element as shown in figure 2.3 and once the system of equations has been solved so that the values on the boundary are known, the user may then find the solutions at particular internal points. The boundary element method has emerged as a powerful alternative to the finite difference and finite element methods. The method has the advantage that the user can avoid a grid method where solutions have to be found at each mesh point irrespective

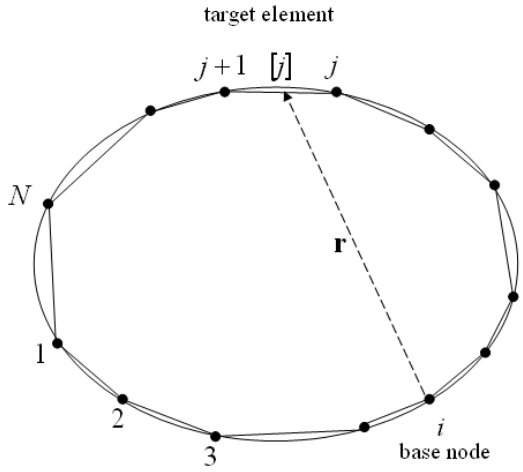

Figure 2.3: Diagram showing the discretisation of the boundary into elements and the collocation of a typical base node with a target element for the boundary element method

of whether they are required. It may be used on domains of most shapes, provided the elements used are sufficiently small to accurately represent the boundary.

However, difficulties arise when the boundary contains corners and points with discontinuous boundary conditions. The problem with boundaries containing corners is the ambiguity of the direction of the normal derivative at the corner. An approach to overcoming this is discussed by Toutip (2001) in which he compares the multiple node method described by Mitra and Ingber (1993) and the gradient approach of Alarcón *et al.* (1979) and Paris and Cañas (1997). He concludes that both methods produce equally acceptable results, but the multiple node method is simpler from a programming point of view.

Although the boundary element method is attractive in having smaller amounts of data which need to be processed, there are difficulties in ex-

tending the technique to non-homogeneous, non-linear and time-dependent problems, because the domain in these problems needs to be discretised into a series of internal cells to deal with terms which are not taken to the boundary by applying the fundamental solution. A typical example is the Poisson equation

$$\nabla^2 u = b(x, y)$$

where we now need to carry out a domain integral on the term b. According to Partridge *et al.* (1992), the simplest way of computing the domain term is to use a cell integration approach by subdividing the region into a series of internal cells and carrying out a numerical integration such as Gauss quadrature on each. Another method they note is the Monte Carlo method (Gipson 1985) which uses random integration points within the domain rather than the regular grid of the cell integration method; they report this method as being expensive in computer time, as a large number of points is needed to compute the domain term. We describe the most commonly used 'domain term' method in the next section.

2.2.6 The dual reciprocity method.

We describe how the dual reciprocity method (DRM) works for a general Poisson equation, a full account being available in Partridge *et al.* (1992). We begin with the usual form of the Poisson equation

$$\nabla^2 u = b$$

which may be considered as the sum of the solution to the Laplace equation and a particular solution $\hat{u}$ so that

$$\nabla^2 \hat{u} = b$$

We approximate the term b by

$$b \approx \sum_{j=1}^{N+L} \beta_j f_j \tag{2.11}$$

where β_j are a set of initially unknown constants, f_j are approximating functions, usually radial basis functions, N is the number of boundary nodes and L is the number of internal nodes. The particular solutions $\hat{u}_j$ and the f_j are related by

$$\nabla^2 \hat{u}_j = f_j$$

After some algebraic manipulation we arrive at the expression

$$\nabla^2 u = \sum_{j=1}^{N+L} \beta_j \left(\nabla^2 \hat{u}_j \right)$$

which is then multiplied by the fundamental solution and integrated over the domain. We apply Green's second theorem, as before, but this time it must be applied to both sides of the equation, hence the name 'dual' reciprocity. Thus, as in the boundary element method, we are able to find the potential and flux at all points on the boundary and then to find the potential at points of interest.

The dual reciprocity method allows the solution of a variety of problems where b may be a constant or a function of any of x, y, u and t. Naturally the method becomes increasingly complex to use when b is a function of more variables. The heat equation may be solved using the dual reciprocity method and this is described in Partridge *et al.* (1992).

Tanaka *et al.* (2003) solved transient heat conduction problems in three dimensions using a method similar to that described above, and they used a finite difference scheme to approximate the time derivative. Each time step related back to the previous result as a kind of new initial condition. They noted that the time-step width was an important factor for accuracy and stability and suggested that this was considered when setting up the problem. They concluded that very accurate results can be obtained if appropriate computational conditions are selected.

2.2.7 The method of separation of variables with the finite difference method

The method of separation of variables has already been discussed in subsection (2.2.1). Here we describe a less frequent manner of using this which was developed following a method described by Brunton and Pullan (1996) in which they used the method of separation of variables and a modal-decomposition solution based on an eigenvalue problem developed using the dual reciprocity method in the space variables. Davies and Radford (2001) followed the same approach but used a finite difference process in the space variables.

We take the usual heat equation in two dimensions with constant thermal diffusivity α

$$\nabla^2 u = \frac{1}{\alpha}\frac{\partial u}{\partial t} \quad \text{in the region} \quad D \tag{2.12}$$

subject to the usual boundary conditions

$$\left.\begin{array}{lll} u = 0 & \text{on} & C_1 \\ q \equiv \frac{\partial u}{\partial n} = 0 & \text{on} & C_2 \end{array}\right\}$$

and the initial condition

$$u\left(x, y, 0\right) = u_0\left(x, y\right)$$

We use a standard separation of variables approach to get a solution to the heat equation of the form

$$u\left(x, y, t\right) = P\left(x, y\right) T\left(t\right) \tag{2.13}$$

where $P\left(x, y\right)$ is a function of position only and $T\left(t\right)$ is a function of time only. Substituting into equation (2.12)

$$T\left(t\right) \nabla^2 P\left(x, y\right) = \frac{1}{\alpha} P\left(x, y\right) \frac{dT\left(t\right)}{dt}$$

gives

$$\frac{1}{P\left(x, y\right)} \nabla^2 P\left(x, y\right) = \frac{1}{\alpha T\left(t\right)} \frac{dT\left(t\right)}{dt} \tag{2.14}$$

The left-hand side of equation (2.14) is a function of space only, while the right-hand side is a function of time only. This means that both sides are equal to some constant, say μ, thus producing two equations which independently describe the effects of varying time and space:

$$\frac{dT(t)}{dt} = -\mu\alpha T(t) \tag{2.15}$$

and

$$\nabla^2 P(x,y) = -\mu P(x,y) \tag{2.16}$$

The analytic solution to equation (2.15) is

$$T(t) = A\exp(-\mu\alpha t)$$

where A is a constant. Equation (2.16) is the usual Helmholz equation, and together with the boundary conditions, gives an eigenvalue problem.

The eigenvalues are the values of μ for which equation (2.16) has a non-trivial solution for P and the eigenfunctions are the corresponding values $P_i(x,y)$. A Helmholtz equation in a finite domain has an infinite number of non-negative eigenvalues, $\mu_i, i = 1, 2, ...$, which are real, discrete and non-degenerate, (Courant and Hilbert 1953). The corresponding eigenfunctions form a complete orthogonal set and it follows that a general solution to the heat equation with homogeneous boundary conditions may be written in the form

$$u(x,y,t) = \sum_{i=1}^{\infty} a_i P_i(x,y)\exp(-\mu_i\alpha t) \tag{2.17}$$

where the constants a_i are determined by the initial conditions and are given by

$$a_i = \frac{\int\limits_D u_0 P_i dA}{\int\limits_D P_i^2 dA} \tag{2.18}$$

and the integration is carried out over the bounded region D. Davies and Radford (2001) solved equation (2.16) by using a finite difference approximation, restricting the problem to two-dimensional rectangular regions leading

to a set of linear equations of the form

$$\mathbf{AP} = -\mu h^2 \mathbf{P}$$

$\mathbf{A}$ is an $N \times N$ square matrix whose elements depend on the form of the finite difference approximation used, $\mathbf{P}$ is a vector of nodal values of $P(x, y)$ and h is the finite difference mesh-size parameter. If μ_i and $P_i, i = 1, 2, ..., N$ are the eigenvalues and eigenvectors respectively of the matrix $\mathbf{A}$ then the approximate separated solution, equivalent to equation (2.17)

$$\mathbf{U} = \sum_{i=1}^{N} a_i \mathbf{P}_i \exp\left(-\mu_i k t\right)$$

where $\mathbf{U}$ is the vector of approximate values of u at the nodes.

The initial condition $u\left(x, y, 0\right) = u_0\left(x, y\right)$ leads to

$$\mathbf{U}_0 = \sum_{i=1}^{N} a_i \mathbf{P}_i$$

so that, since the $\mathbf{P}_i$ are orthogonal,

$$a_i = \frac{\mathbf{P}_i^T \mathbf{U}_0}{\mathbf{P}_i^T \mathbf{P}_i}$$

which is the discrete analogue of equation (2.18).

In this case the Laplacian operator is replaced by the 'five-point' formula

$$\frac{u_S + u_W + u_E + u_N - 4u_i}{h^2}$$

and the eigenvalues and eigenvectors are found using the Jacobi method. Although this method is interesting and produces results of good accuracy it is rarely used. The fact that it requires a rectangular mesh to cover the domain means that its use is limited.

2.2.8 The method of fundamental solutions

This method was introduced by Kupradze (1964) and is discussed in detail by Golberg (1995). It is of interest because unlike the finite element method

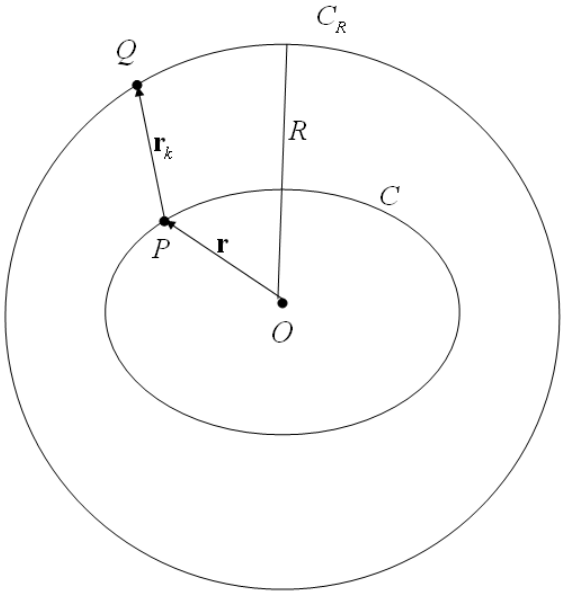

Figure 2.4: Region of geometry for the method of fundamental solutions

and the boundary element method, it requires neither domain nor boundary discretisation. We consider the method for solving a Poisson equation.

In figure 2.4 we show how to set up the geometry of the method. We have the domain D enclosed by the boundary C in which the Poisson equation is satisfied. First we choose a fundamental solution to the equation, and as before, the usual choice for the Laplace operator in two dimensions being

$$u^*(r) = \frac{1}{2\pi} \ln(r)$$

where r is the distance between two points . We then enclose the domain D entirely within a circle of radius R, and it is suggested by Golberg that the radius of this circle should be at least four times the maximum distance of the boundary C from the origin.

Having decided on the number of boundary points, n, on C in which we are interested, we place $n-1$ points on the curve C_R. These points are

labelled Q_k, where $k = 1, 2, ..., n-1$ and we use these for collocation.

Let P be any point in D then u may be approximated by the function

$$U_n(\mathbf{r}) = \sum_{k=1}^{n} c_k u^*(r_k) + c \tag{2.19}$$

where r_k represents the distance between P and Q_k and c_k and c are constants, for a Dirichlet boundary condition.

A Neumann boundary condition is approximated by

$$Q_n(\mathbf{r}) = \sum_{k=1}^{n} c_k \frac{\partial u^*(r_k)}{\partial n}$$

and a Robin condition is approximated by

$$G_n(\mathbf{r}) = \sum_{k=1}^{n} c_k \left[\frac{\partial u^*(r_k)}{\partial n} + \sigma(\mathbf{r}) u^*(r_k) \right] + \sigma(\mathbf{r}) c$$

The c_k are calculated by solving a system of equations and we can then find all the unknown values on the boundary and calculate the internal solutions required. Golberg (1995) compared the results using this method with the dual reciprocity method and concluded that the numerical results obtained were superior.

The method was one of five methods compared by Davies *et al.* (1997), which provided a solution to the heat equation, using a Laplace transform method, which will be discussed later, in a parallel environment. It performed as well as any other in terms of speed-up in a parallel environment. Fairweather and Karageorghis (1998) described the development of the method of fundamental solutions over the previous three decades and discussed several applications. They concluded that the method is easy to implement and requires relatively few boundary points to produce accurate results. They also found that corners in a region which may cause problems in the boundary element method, are not a specific source of inaccuracy in this method. It neither needs discretisation of the boundary, nor does it involve integrals on the boundary. Furthermore, to find a solution for a point

in the domain we only need to evaluate the approximate solution, whereas the boundary element method requires us to use numerical quadrature. The method of fundamental solutions is one of a family of methods known as mesh-free methods (Liu 2003). These techniques are not yet widely used but there is an increasing interest in them since they offer advantages from the point of view of problem set-up.

2.2.9 The isotherm migration method

Generally this method is particularly useful for solving problems involving phase change, that is a change in the material from one state to another, for example, a change from ice to water. However, there might be situations when rather than finding the temperature at certain points in the domain, we would like to know the movement of lines passing through points having the same temperature. These are known as isotherms. The isotherm migration method enables us to do this, but first we have to re-formulate the heat equation. The heat equation (2.2) defines the temperature u as a function of space (x, y) and time t. For this method we need to re-write the heat equation so that position is now a function of temperature and time. This leads to a non-linear partial differential equation. The method was proposed by Chernous'ko (1970) where he described the method for one-dimensional problems and the ideas were further developed by Crank and Phahle (1973). Since we shall be referring to the isotherm migration method throughout our work, we shall not discuss it here, but a full explanation of its operation will be provided further on.

2.3 Moving boundary problems

Moving boundaries occur frequently in diffusion problems. Such problems may involve a change of state which occurs on the interface, for example,

in the case of melting ice (Crank and Phahle 1973). Diffusion in a medium where the concentration of substance is higher in one region than another may also be modelled in this way. Crank and Gupta (1972a) described a moving boundary problem arising from the diffusion of oxygen in absorbing tissue, and Voller *et al.* (2006) produced a model to track the movement of the shoreline of a sedimentary ocean basin, one feature of which was sediment transportation via diffusion. The applications of moving boundary problems are therefore varied. Their common feature is that they are known as Stefan problems, since they were first referred to by Stefan (1891) in his study of the thickness of the polar ice cap, and they involve situations in which there is a phase change, which occurs when a material exists in two states on each side of a boundary. A new condition arises on the moving boundary as a result of the phase change, the so-called Stefan condition. We shall return to this topic in greater depth in chapter 5. In the following sections we consider methods available to solve the Stefan moving boundary problem.

2.3.1 Similarity solutions

There are very few analytical solutions and they are mainly for the one-dimensional cases of an infinite region with simple initial and boundary conditions and constant thermal properties. These exact solutions take the form of functions of $\frac{x}{\sqrt{t}}$. There are many examples of these to be found in Carslaw and Jaeger (1959).

One important solution is that due to Neumann, which solves the problem for a substance in a region $x > 0$ initially liquid at a constant temperature u_c with the surface $x = 0$ maintained at zero for $t > 0$. The solutions u_1 and u_2 for the temperatures in the solid and liquid phases respectively are given by

$$u_1 = \frac{u_f}{\operatorname{erf}(\mu)} \operatorname{erf}\left(\frac{x}{2\left(\alpha_1 t\right)^{\frac{1}{2}}}\right)$$

and

$$u_2 = u_c - \frac{(u_c - u_f)}{\operatorname{erfc}\left(\mu\left(\alpha_1/\alpha_2\right)^{\frac{1}{2}}\right)} \operatorname{erfc}\left(\frac{x}{2\left(\alpha_2 t\right)^{\frac{1}{2}}}\right)$$

where μ is a constant to be determined, u_f is the solidifying point of the material and α_1 and α_2 are the thermal constants associated with the solid and liquid phases respectively.

Lightfoot (1929) used an integral method in which he assumed that the thermal properties of the solid and liquid were the same. He considered the surface of solidification which was moving and liberating heat and this led to an integral equation for the temperature.

2.3.2 The heat-balance integral method

Goodman (1958) integrated the one-dimensional heat flow equation with respect to x and inserted boundary conditions to produce an integral equation which expressed the overall heat balance of the system. Goodman says that although the solution was approximate it provided good accuracy and the problem was reduced from that of solving a partial differential equation to one of solving an ordinary differential equation. Poots (1962) extended the heat-balance integral method to study the movement of a two-dimensional solidification front in a liquid contained in a uniform prism.

2.3.3 Front tracking methods

These are methods which compute the position of the moving boundary at each step in time. If we use a fixed grid in space-time, then in general, the position of the moving boundary will fall between two grid points. To resolve this, we either have to use special formulae which allow for unequal space intervals or we have to deform the grid in some way so that the moving boundary is always on a gridline. Several numerical solutions based on the finite difference method have been proposed. Their approach to the grid

and the moving boundary differs. In general, the moving boundary will not coincide with a gridline if we take δt to be constant.

Douglas and Gallie (1955) chose each δt iteratively so that the boundary always moved from one gridline to the next in an interval δt.

Murray and Landis (1959) kept the number of space intervals between the fixed and moving boundary constant and equal to some parameter, r. So for equal space intervals,

$$\delta x = \frac{x_0}{r}$$

where x_0 is the position of the moving boundary. The moving boundary is always on grid line r. They differentiated partially with respect to time t, following a given grid line instead of at constant x. They compared this method with a fixed grid approach and concluded that the variable space grid is preferable if we want to continuously track the fusion front travel, but the fixed space network is more convenient if we wish to know temperatures within the domain.

Crank and Gupta (1972a) described a moving boundary problem arising from the diffusion of oxygen in absorbing tissue by using Lagrangian-type formulae and a Taylor series near the boundary. They subsequently (1972b) developed a method making use of a grid system which moved with the boundary. This had the effect of transferring the unequal space interval from the neighbourhood of the moving boundary to the fixed surface boundary and resulted in an improved smoothness in the calculated motion of the boundary when compared with the results using the Lagrange interpolation. Other methods involving grids are discussed in Crank (1984).

Furzeland (1980) describes another method, the method of lines, which he attributes to Meyer (1970). In this method, by discretising the time variable the Stefan problem is reduced to a sequence of free boundary value problems for ordinary differential equations which are solved by conversion to initial value problems.

2.3.4 Front-fixing methods

We have already discussed the isotherm migration method in a previous section, and this is one example of a front-fixing method.
The simplest case of front-fixing suitable for the one-dimensional case was proposed by Landau (1950). He suggested making the transformation

$$\xi = \frac{x}{x_0\left(t\right)}$$

which fixes the melting boundary at $\xi = 1$ for all t. The heat equation and the equation for the moving boundary are also transformed, before being solved using some method such as the finite difference method, as described by Crank (1957).

Another approach is to use so-called body-fitted curvilinear co-ordinates. In this method a curve-shaped region is transformed into a fixed rectangular domain by transforming to a new co-ordinate system. This is useful, because, when working on Stefan problems in two dimensions, the shape of the region is continuously changing as the phase-change boundary moves. However the transformed partial differential equation does increase in complexity because a change of variable has to be used,

$$x = x\left(\xi, \eta\right)$$

and

$$y = y\left(\xi, \eta\right)$$

This leads to the Laplace equation being transformed to an expression with five partial derivatives in both ξ and η which have constants which need to be solved using a system of simultaneous equations. The curvilinear mesh also has to be generated at each time step. This method was used by Furzeland (1977).

2.3.5 Fixed-domain methods

In certain cases it might be difficult to track the moving boundary and one way to overcome this is to reformulate the problem so that the Stefan condition is implicitly bound up in a new form of the equations, which applies over the whole of a fixed domain. The position of the moving boundary then appears as a component of the solution after the problem has been solved. To do this, a total heat function or so-called enthalpy function is introduced.

The use of enthalpy was proposed by Eyres *et al.* (1946). Later, Price and Slack (1954) looked at the solution of the heat equation where the latent heat of freezing was a factor and in which they considered the total heat content of the system. The enthalpy function describes the total heat content of the system, which is the sum of the specific heat and the latent heat needed for a phase change. Therefore when shown graphically, this is a step function, the step occurring at the boundary where we have a phase change. In the case where we have a mushy region, a region where material exists in both solid and liquid forms, the step will not be so steep. We shall not consider problems of this type.

Crowley (1979) solved the Stefan problem using the enthalpy method together with a weak solution method. A weak solution is a general solution to a partial differential equation, for which the derivatives in the equation may not all exist, but which is still deemed to satisfy the partial differential equation in some way. To find the weak solution, the differential equation is first rewritten in such a way so that no partial derivatives show up. This is usually achieved by multiplying it by a suitable test function, writing the integral form and changing the order of integration. The solutions to this new form of the equation are the weak solutions, because although they satisfy the equation in the second form, they may not satisfy the original equation. A differential equation may have solutions which are not differentiable and the weak formulation allows one to find such solutions. Crowley compared

her results with other numerical methods and concluded that results using this method are in good agreement.

Furzeland (1980) produced a comparative study of numerical methods for moving boundary problems. He considered the method of lines in time, the co-ordinate transformation method, a method combining both of these and the enthalpy method. He used four different examples and concluded that the enthalpy method was very attractive because it was easy to program and there was no extra computation involved in tracking the moving boundary and it could be used for mushy phase-change problems and for complicated shaped domains. However for high accuracy it needs many space points and it is not suitable for all problems. The 'front-tracking' methods produce very accurate solutions both for the moving boundary and other temperatures, but cannot be used for mushy problems.

Chun and Park (2000) developed a modification to the enthalpy method, which avoided oscillations in temperature and phase front which can be observed in certain cases. They introduced a fictitious temperature on the phase-change front based on values obtained at the previous time step, and then used finite difference equations to solve across the interface. Their results compared favourably with two other methods.

The enthalpy method has been used together with the finite element method (Elliott 1981), the finite difference method (Voller 1985) and more recently with the boundary element dual reciprocity method (Honnor *et al.* 2003 and Kane *et al.* 2004), which indicates that it is still a favourable method.

Another fixed-domain method is the method of variational inequalities (Elliott 1980). The variational expressions refer to a fixed domain and explicit use of the Stefan condition is avoided.

2.4 Summary of Chapter 2

In this chapter we began by giving a derivation of the heat equation and we discussed the class of partial differential equations to which it belongs.

We then described the many methods of solving the heat equation, starting with analytic solutions for very simple cases and then giving brief descriptions of the numerical methods available. These included the finite difference, finite element, boundary element and dual reciprocity methods. Less common methods were also mentioned, in particular the method of separation of variables in combination with the finite difference method and the method of fundamental solutions.

We will elaborate on the isotherm migration method and the Laplace transform methods later, as these form the basis of our work.

We moved on to discuss situations in which we have a phase change and gave brief details of different methods which may be employed to solve such problems.

2.4.1 Contribution

We have compared many methods available for solving the heat equation and commented on their suitabilty for use in various scenarios.

Chapter 3

The isotherm migration method for one-dimensional problems with no phase change

3.1 Background to the isotherm migration method

Our work is based on the solution of the heat equation using the isotherm migration method. We described in the previous chapter how this method is usually used to solve problems involving a phase change. However, we first note another method for dealing with moving boundary problems, the level set method. The level set method of Osher and Sethian (1988) tracks the motion of an interface by embedding the interface as the zero level set of the distance function. The motion of the interface is matched with the zero level set of the level set function, and the resulting initial value partial differential equation for the evolution of the level set function resembles a Hamilton-Jacobi equation. In this setting, curvatures and normals may

be easily evaluated, topological changes occur in a natural manner, the technique extends to three dimensions.

We also note that there are different approaches to solving the heat equation. The Eulerian method considers changes as they occur at a fixed point in the domain while the Lagrangian method considers changes which occur as a particle is followed along a trajectory. The Eulerian derivative is the rate of change at a fixed position and by definition this is the usual partial derivative $\frac{\partial}{\partial t}$. The Lagrangian derivative is normally written as $\frac{D}{Dt}$.

The relationship between the Eulerian and Lagrangian deriviatives is such that

$$\frac{Du}{Dt} = \frac{\partial u}{\partial t} + v\frac{\partial u}{\partial x}$$

where v is the velocity of the medium, for example fluid flow.

We return to the isotherm migration method which was proposed independently by Chernous'ko (1970) and Dix and Cizek (1970) and is an effective solution technique for solving moving boundary problems. Several authors have produced efficient numerical solution processes based on the isotherm migration approach including Crank and Phahle (1973), Crank and Gupta (1975), Crank and Crowley (1978 and 1979), Wood (1991a, 1991b and 1991c) and Kutluay and Esen (2004).

In the first instance, we use the method to solve one-dimensional problems with no phase change. This is because we wish to look at several aspects of the method to see how robust it is and to understand any difficulties which have to be overcome.

Before we use the isotherm migration method, we need to rewrite the heat equation so that rather than giving the temperature as a function of position and time, the equation gives us position as a function of temperature and time. To get the heat equation in the correct form we perform a mapping process.

3.2 The mapping of the heat equation

In heat transfer problems it is usual to express the temperature as a function of space and time i.e. $u = u(x,t)$ where u is the temperature, x is the space variable, and t is the time. In the isotherm migration method we map the heat equation so that x is the dependent variable and u and t are the independent variables, so that $x = x(u,t)$ and in this way we are able to find the positions of the isotherms, which move across the domain with time. This mapping was discussed by Dix and Cizek (1970). Rose (1967) derived a mapped equation but did not develop a numerical method. Crank and Phale (1973) discuss a mapping and then go on to solve a problem in melting ice using a finite difference method.

A convenient way of describing the mapping is to consider a rod, of length a, initially at temperature u_0, which is held at a constant temperature at each end. The temperature, u, of the rod satisifes the usual heat equation given in equation (2.2) which we will write as

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2} \tag{3.1}$$

together with the boundary conditions

$$u(0,t) = u_L, \quad u(a,t) = u_R$$

where u_L and u_R are the temperatures of the left and right hand ends of the rod respectively and the initial condition

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

We use the change of variables

$$\tilde{u} = u, \quad \tilde{x} = \frac{x}{a}, \quad \tilde{t} = \frac{\alpha t}{a^2}, \quad \tilde{u}_0 = u_0, \quad \tilde{u}_L = u_L, \quad \tilde{u}_R = u_R \tag{3.2}$$

which leads to the following dimensionless equation:

$$\frac{\partial \tilde{u}}{\partial \tilde{t}} = \frac{\partial^2 \tilde{u}}{\partial \tilde{x}^2}, \quad 0 < \tilde{x} < 1, \quad \tilde{t} \geqslant 0 \tag{3.3}$$

We wish to write the heat flow equation (5.21) so that $\tilde{x}$ is expressed as a function of $\tilde{u}$ and $\tilde{t}$.

Since

$$\delta\tilde{x} = \frac{\partial\tilde{x}}{\partial\tilde{u}}\delta\tilde{u} + \frac{\partial\tilde{x}}{\partial\tilde{t}}\delta\tilde{t}$$

to first order, if $\tilde{t}$ is constant then

$$\delta\tilde{x} = \frac{\partial\tilde{x}}{\partial\tilde{u}}\delta\tilde{u}$$

Hence

$$1 = \frac{\partial\tilde{x}}{\partial\tilde{u}}\frac{\partial\tilde{u}}{\partial\tilde{x}}$$

which implies

$$\frac{\partial\tilde{u}}{\partial\tilde{x}} = \frac{1}{\frac{\partial\tilde{x}}{\partial\tilde{u}}} = \left(\frac{\partial\tilde{x}}{\partial\tilde{u}}\right)^{-1} \tag{3.4}$$

Similarly,

$$\delta\tilde{u} = \frac{\partial\tilde{u}}{\partial\tilde{x}}\delta\tilde{x} + \frac{\partial\tilde{u}}{\partial\tilde{t}}\delta\tilde{t}$$

where, on an isotherm, $\tilde{u}$ is constant so that $\delta\tilde{u}$ is 0. Therefore, on an isotherm

$$0 = \frac{\partial\tilde{u}}{\partial\tilde{x}}\delta\tilde{x} + \frac{\partial\tilde{u}}{\partial\tilde{t}}\delta\tilde{t}$$

$$-\frac{\partial\tilde{u}}{\partial\tilde{x}}\delta\tilde{x} = \frac{\partial\tilde{u}}{\partial\tilde{t}}\delta\tilde{t}$$

$$-\left(\frac{\partial\tilde{x}}{\partial\tilde{u}}\right)^{-1}\delta\tilde{x} = \frac{\partial\tilde{u}}{\partial\tilde{t}}\delta\tilde{t}$$

Hence

$$\frac{\partial\tilde{x}}{\partial\tilde{t}} = -\frac{\partial\tilde{u}}{\partial\tilde{t}}\frac{\partial\tilde{x}}{\partial\tilde{u}} \tag{3.5}$$

$$= -\frac{\partial^2\tilde{u}}{\partial\tilde{x}^2}\frac{\partial\tilde{x}}{\partial\tilde{u}}$$

$$= -\frac{\partial}{\partial\tilde{x}}\left(\frac{\partial\tilde{u}}{\partial\tilde{x}}\right)\frac{\partial\tilde{x}}{\partial\tilde{u}}$$

Using equation (3.4) this becomes

$$\frac{\partial \tilde{x}}{\partial \tilde{t}} = -\frac{\partial}{\partial \tilde{x}} \left(\frac{\partial \tilde{x}}{\partial \tilde{u}} \right)^{-1} \frac{\partial \tilde{x}}{\partial \tilde{u}} \tag{3.6}$$

We now consider

$$\begin{aligned} \frac{\partial}{\partial \tilde{x}} \left(\frac{\partial \tilde{x}}{\partial \tilde{u}} \right)^{-1} &= \frac{\partial}{\partial \tilde{u}} \left(\frac{\partial \tilde{x}}{\partial \tilde{u}} \right)^{-1} \frac{\partial \tilde{u}}{\partial \tilde{x}} \\ &= -\left(\frac{\partial \tilde{x}}{\partial \tilde{u}} \right)^{-2} \frac{\partial^2 \tilde{x}}{\partial \tilde{u}^2} \frac{\partial \tilde{u}}{\partial \tilde{x}} \\ &= -\left(\frac{\partial \tilde{x}}{\partial \tilde{u}} \right)^{-3} \frac{\partial^2 \tilde{x}}{\partial \tilde{u}^2} \end{aligned} \tag{3.7}$$

Substituting this into equation (3.6) gives

$$\frac{\partial \tilde{x}}{\partial \tilde{t}} = \left(\frac{\partial \tilde{x}}{\partial \tilde{u}} \right)^{-3} \frac{\partial^2 \tilde{x}}{\partial \tilde{u}^2} \left(\frac{\partial \tilde{x}}{\partial \tilde{u}} \right)$$

so that

$$\frac{\partial \tilde{x}}{\partial \tilde{t}} = \left(\frac{\partial \tilde{x}}{\partial \tilde{u}} \right)^{-2} \frac{\partial^2 \tilde{x}}{\partial \tilde{u}^2} \tag{3.8}$$

When we consider the heat equation with u as a function of x and t the quantities $\frac{\partial u}{\partial x}$ and $\frac{\partial^2 u}{\partial x^2}$ represent the rate of change of temperature with respect to distance and diffusion respectively. However under the isotherm migration mapping there is no equivalent meaning for the terms $\frac{\partial \tilde{x}}{\partial \tilde{u}}$ and $\frac{\partial^2 \tilde{x}}{\partial \tilde{u}^2}$ and this is one example of the difficulty that when using this method it is difficult to visualise the problem in a physical sense.

The boundary conditions are

$$\tilde{x}\left(\tilde{u}_L, \tilde{t}\right) = 0, \quad \tilde{t} \geqslant 0 \tag{3.9}$$

$$\tilde{x}\left(\tilde{u}_R, \tilde{t}\right) = 1, \quad \tilde{t} \geqslant 0 \tag{3.10}$$

and the initial condition is

$$\tilde{x}\left(\tilde{u}, 0\right) = \tilde{x}_0, \quad \tilde{u}_L < \tilde{u} < \tilde{u}_R \tag{3.11}$$

Equations (3.8), (3.9), (3.10) and (3.11) form the system describing the isotherm migration method and will be used for our work in the one-dimensional case.

3.3 A method to solve the transformed equation

We can see that equation (3.8) is in a form which makes it suitable to solve using an explicit finite difference method. This method was discussed by Crank and Phahle (1973). We use a forward difference in $\tilde{t}$ and a central difference in $\tilde{u}$. This gives an explicit finite difference approximation $\tilde{X}_i$ for the position, of the isotherms:

$$\tilde{X}_i^{(n+1)} = \tilde{X}_i^n + 4\delta\tilde{t}\left\{\frac{\tilde{X}_{i-1}^{(n)} - 2\tilde{X}_i^{(n)} + \tilde{X}_{i+1}^{(n)}}{\left(\tilde{X}_{i-1}^{(n)} - \tilde{X}_{i+1}^{(n)}\right)^2}\right\} \tag{3.12}$$

where $\delta\tilde{t}$ is a suitable time-step size.

We have not attempted an analysis of the stability of the finite difference equation.

If $\tilde{u}_L$ is greater than $\tilde{u}_R$, then the isotherms move along the positive x-axis. For the isotherms to move forward in the correct way as time increases, $\tilde{X}_i^{(n+1)}$ must be greater than $\tilde{X}_i^{(n)}$. The expression

$$4\delta\tilde{t}\left\{\frac{\tilde{X}_{i+1}^{(n)} - 2\tilde{X}_i^{(n)} + \tilde{X}_{i-1}^{(n)}}{\left(\tilde{X}_{i-1}^{(n)} - \tilde{X}_{i+1}^{(n)}\right)^2}\right\}$$

must therefore be positive.

$$4\delta\tilde{t}\left\{\frac{\tilde{X}_{i+1}^{(n)} + \tilde{X}_{i-1}^{(n)}}{\left(\tilde{X}_{i-1}^{(n)} - \tilde{X}_{i+1}^{(n)}\right)^2}\right\}$$

is always positive and so if we ensure that

$$\tilde{X}_i^{(n)} + 4\delta\tilde{t}\left\{\frac{-2\tilde{X}_i^{(n)}}{\left(\tilde{X}_{i-1}^{(n)} - \tilde{X}_{i+1}^{(n)}\right)^2}\right\}$$

is also positive, then $\tilde{X}_i^{n+1}$ will be greater than $\tilde{X}_i^{(n)}$.

We therefore have a condition on $\delta\tilde{t}$ such that

$$1 - \frac{8\delta\tilde{t}}{\left(\tilde{X}_{i-1}^{(n)} - \tilde{X}_{i+1}^{(n)}\right)^2} > 0$$

which leads to

$$\delta\tilde{t} < \frac{1}{8}\left(\tilde{X}_{i-1}^{(n)} - \tilde{X}_{i+1}^{n}\right)^2$$

and this allows $\delta\tilde{t}$ to be determined as the solution progresses rather than being fixed for all time. According to Dix and Cizek (1970) the truncation error is proportional to $\delta\tilde{t}$ and $(\delta\tilde{u})^2$.

Example 3.1

We consider a rod of unit length, initially at uniform temperature $\tilde{u}_0$. A constant heat source is applied so that the left-hand end of the rod is maintained at a temperature $\tilde{u}_L$, while the right-hand end is maintained at $\tilde{u}_R$. The boundary conditions for the problem are

$$\tilde{u}(0,\tilde{t}) = \tilde{u}_L, \quad \tilde{u}(1,\tilde{t}) = \tilde{u}_R \tag{3.13}$$

and the initial condition is

$$\tilde{u}(\tilde{x},0) = \tilde{u}_0 \tag{3.14}$$

The analytic solution to this problem is

$$\tilde{u}(\tilde{x},\tilde{t}) = \tilde{u}_L + (\tilde{u}_R - \tilde{u}_L)\,\tilde{x} + \sum_{n=1}^{\infty} b_n \sin(n\pi\tilde{x}) \exp(-n^2\pi^2\tilde{t}) \tag{3.15}$$

where

$$b_n = \frac{2}{n\pi}\left\{(\tilde{u}_0 - \tilde{u}_L)\,(1-(-1)^n) + (\tilde{u}_R - \tilde{u}_L)\,(-1)^n\right\}$$

This analytic solution is derived using the separation of variables and expressing the solution as a series, which we mentioned in Chapter 2.

In this example we consider the case where $\tilde{u}_L = 10$, $\tilde{u}_R = 0$ and $\tilde{u}_0 = 0$. We notice that no isotherms other than $u = 0$ exist at $\tilde{t} = 0$, and so we must have some means of generating a set of initial positions for the isotherms at some small time.

In all our work, if an analytic solution is available we shall use it to provide starting values and as a benchmark with which to compare our solutions. When we do not have such a solution, we use a numerical method to provide starting values and we compare our results with those produced by another method, or by looking at expected trends.

The analytic solution provides a method for starting this problem as well as a method for checking the accuracy of the solution. As $\tilde{u}(\tilde{x}, 0) = 0$ throughout the region when $\tilde{t} = 0$, we need to use the solution to equation (3.15) at a small time, say $\tilde{t} = 0.1$, to find a starting position for the isotherms. Under the mapping equations described in equations (3.9) and (3.10), equations (3.13) become

$$\tilde{x}(10, \tilde{t}) = 0, \quad \tilde{x}(0, \tilde{t}) = 1$$

and from equation (3.11), the initial condition, equation (3.14) is

$$\tilde{x}(\tilde{u}, 0) = 0$$

and we consider isotherms of temperature 2 units apart. When we refer to 'isotherm N', we mean the contour having a temperature value of N units. The positions of isotherms 2 and 8, for times up to $\tilde{t} = 1.5$ are shown in tables 3.1 and 3.2 respectively.

We note that the results are in excellent agreement with the analytic solutions. We also show the movement of isotherms 2, 4, 6 and 8 with time in figure 3.1, where IMM refers to the results obtained with the isotherm migration method, and we note that the steady state is reached at about $\tilde{t} = 0.6$ and that our calculated solutions closely follow the analytic solutions.

Table 3.1: The position of isotherm 2 with increasing time for example 3.1

time	IMM	analytic	% error
0.1	0.5713	0.5713	0.00
0.2	0.7320	0.7346	0.35
0.3	0.7787	0.7789	0.02
0.4	0.7927	0.7925	0.02
0.5	0.7974	0.7973	0.01
0.6	0.7990	0.7990	0.01
0.7	0.7997	0.7996	0.00
0.8	0.7999	0.7999	0.00
0.9	0.8000	0.7999	0.00
1.0	0.8000	0.7999	0.00
1.1	0.8000	0.8000	0.00
1.2	0.8000	0.8000	0.00
1.3	0.8000	0.8000	0.00
1.4	0.8000	0.8000	0.00
1.5	0.8000	0.8000	0.00

Table 3.2: The position of isotherm 8 with increasing time for example 3.1

time	IMM	analytic	% error
0.1	0.1133	0.1133	0.00
0.2	0.1582	0.1578	0.24
0.3	0.1823	0.1822	0.09
0.4	0.1933	0.1930	0.17
0.5	0.1974	0.1973	0.05
0.6	0.1991	0.1990	0.03
0.7	0.1997	0.1996	0.01
0.8	0.1999	0.1999	0.00
0.9	0.2000	0.1999	0.00
1.0	0.2000	0.2000	0.00
1.1	0.2000	0.2000	0.00
1.2	0.2000	0.2000	0.00
1.3	0.2000	0.2000	0.00
1.4	0.2000	0.2000	0.00
1.5	0.2000	0.2000	0.00

Example 3.2

In this example we consider the case when $\tilde{u}_L = 10, \tilde{u}_R = 5$ and $\tilde{u}_0 = 0$

As shown in section 3.2 equations (3.13) and (3.14) become

$$\tilde{u}(0,\tilde{t}) = 10, \tilde{u}(1,\tilde{t}) = 5 \tag{3.16}$$

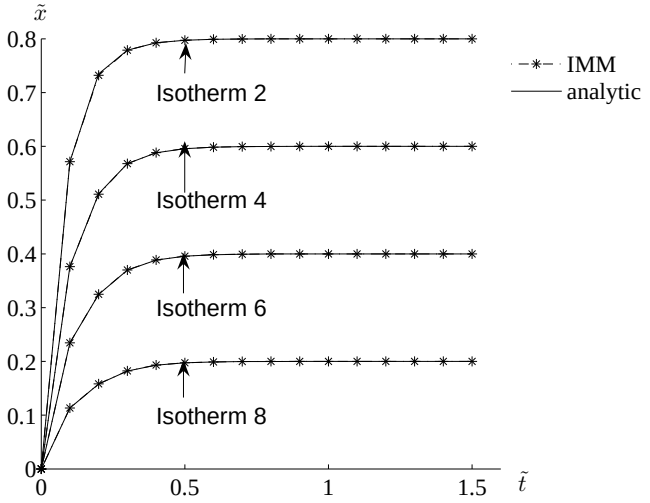

Figure 3.1: The movement of the isotherms with time in example 3.1

and

$$\tilde{u}(\tilde{x}, 0) = 0 \tag{3.17}$$

When $\tilde{t} > 0$, there will be temperatures $5 \leqslant \tilde{u} \leqslant 10$ along the rod. We notice that for some values of $\tilde{u}$ there will not be a unique position, $\tilde{x}$. We show this in figure 3.2 where we can see that, for example, isotherm 4 is at $\tilde{x} = 0.25$ and $\tilde{x} = 0.9$ approximately. This shows that $\tilde{x} = \tilde{x}(\tilde{u}, \tilde{t})$ may not have a unique solution for all $\tilde{u}$.

A further difficulty arises because some of the isotherms exist for only very short times. The analytic solution in equation (3.15) with $\tilde{u}_L = 10$ and $\tilde{u}_R = 5$ shows that the mid-point of the rod, $\tilde{x} = 0.5$, reaches a temperature, $\tilde{u} = 7.5$ at the steady state. We consider the region $0 \leqslant \tilde{x} \leqslant 0.5$. Here isotherms for temperatures $\tilde{u}$ in the interval $[0, 7.5)$ will exist only for a finite time and we show this in figure 3.3 where we plot the positions of the isotherms as a function of time. Some of the isotherms exist only for very brief times. The points shown in figure 3.3 are the positions of the isotherms from $\tilde{t} = 0.005$ to $\tilde{t} = 0.15$. For example, isotherm 2 and isotherm 4 clearly do not exist for time $\tilde{t} = 0.15$.

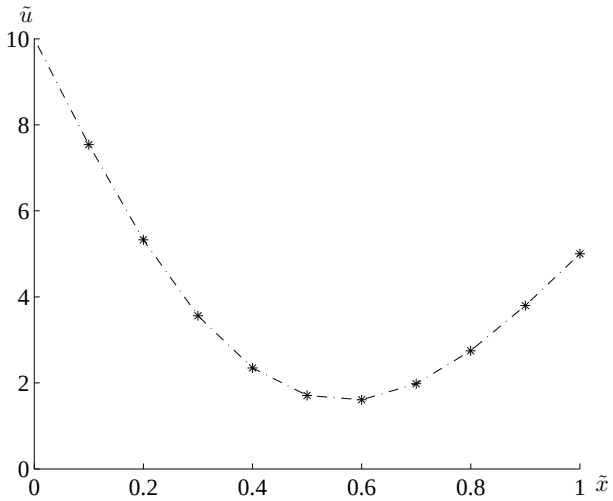

Figure 3.2: Temperature plotted as a function of position in example 3.2 for $\tilde{t} = 0.05$

Considering the region $0 \leqslant \tilde{x} \leqslant 1$, if we use the mapping in the way described in example 3.1, only isotherms representing regions of temperature between $\tilde{u} = 5$ and $\tilde{u} = 10$ lie within the bounded region in $\tilde{u}, \tilde{t}$-space. We cannot therefore use the isotherm migration method to solve the problem directly.

To overcome this, we consider the case as two problems. Because the heat equation is linear, we may write

$$\frac{\partial v}{\partial \tilde{t}} = \frac{\partial^2 v}{\partial \tilde{x}^2} \tag{3.18}$$

with boundary and initial conditions

$$v_L = 10, v_R = 0 \text{ and } v_0 = 0$$

and

$$\frac{\partial w}{\partial \tilde{t}} = \frac{\partial^2 w}{\partial \tilde{x}^2} \tag{3.19}$$

with boundary and initial conditions

$$w_L = 0, w_R = 5 \text{ and } w_0 = 0$$

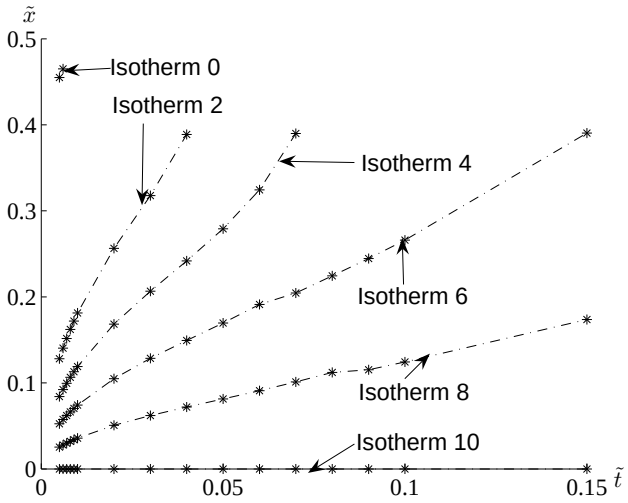

Figure 3.3: The positions of the isotherms for $0 \leqslant \tilde{x} \leqslant 5$ plotted as a function of time for example 3.2

and then the required solution is the sum of the solutions to these equations, $\tilde{u} = v + w$. Now, although the heat equation in $\tilde{x}, \tilde{t}$-space is linear, the mapped equation in $\tilde{u}, \tilde{t}$-space is not linear and therefore we cannot simply add the values of $\tilde{x}(v, \tilde{t})$ and $\tilde{x}(w, \tilde{t})$. In order to solve the problem, we plot the functions $v = f_1(\tilde{x}_1, \tilde{t})$ and $w = f_2(\tilde{x}_2, \tilde{t})$ for each time to which we seek a solution. We find the trendlines for the graphs obtained and then find the value of $\tilde{u}$, the required temperature, by interpolation.

Figures 3.4 and 3.5, show the graphs of $v = f_1(\tilde{x}_1, \tilde{t})$ and $w = f_2(\tilde{x}_2, \tilde{t})$ at time $\tilde{t} = 0.1$, and in figure 3.6 we show the graph obtained by summing these to get $\tilde{u}(\tilde{x}, \tilde{t}) = f_1(\tilde{x}_1, \tilde{t}) + f_2(\tilde{x}_2, \tilde{t})$ which also shows the analytic solution for $\tilde{u}$ for comparison. The legend IMM refers to the isotherm migration method. Similarly, figures 3.7 and 3.8 show the graphs of $v = f_1(\tilde{x}_1, \tilde{t})$ and $w = f_2(\tilde{x}_2, \tilde{t})$ at time $\tilde{t} = 0.5$, and in figure 3.9 we show the graph obtained by summing these to get $\tilde{u}(\tilde{x}, \tilde{t}) = f_1(\tilde{x}_1, \tilde{t}) + f_2(\tilde{x}_2, \tilde{t})$ which again shows the analytic solution for $\tilde{u}$ for comparison.

The results agree favourably with the analytic solutions at both times.

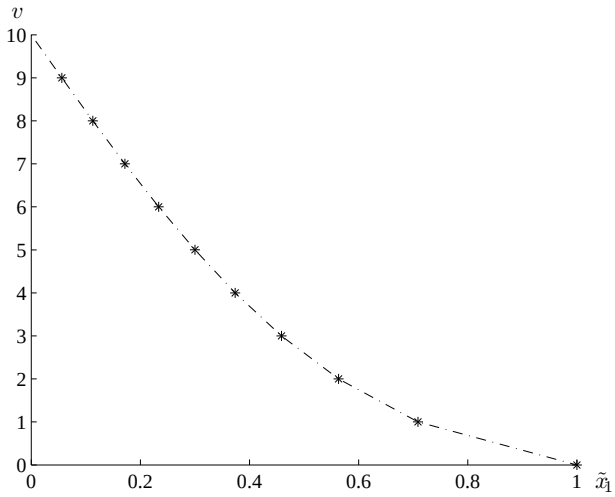

Figure 3.4: The graph of $f_1(\tilde{x}_1, \tilde{t})$ for $\tilde{t} = 0.1$ in example 3.2

We include tables 3.3 and 3.4 respectively showing the percentage error in using the isotherm migration method at times $\tilde{t} = 0.1$ and $\tilde{t} = 0.5$ respectively. We see that the isotherm migration method gives accurate results, particularly for the larger value of $\tilde{t}$.

Table 3.3: Error in value of $\tilde{u}$ at $\tilde{t} = 0.1$ for example 3.2

$\tilde{x}$	IMM	analytic	% error
0.1	8.2422	8.3818	1.67
0.2	6.7205	6.8783	2.29
0.3	5.4902	5.5913	1.81
0.4	4.5513	4.5973	1.00
0.5	3.9039	3.9413	0.95
0.6	3.5479	3.6334	2.35
0.7	3.4834	3.6497	4.56
0.8	3.7102	3.9368	5.76
0.9	4.2285	4.4179	4.29

3.4 The effect of errors in the initial data

Generally we will need to use a numerical method to find some starting values at a small initial time. As the results of the numerical method will

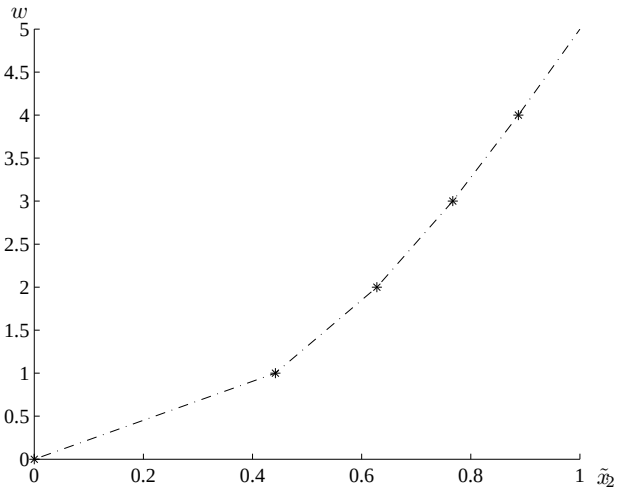

Figure 3.5: The graph of $f_2(\tilde{x}_2, \tilde{t})$ for $\tilde{t} = 0.1$ in example 3.2

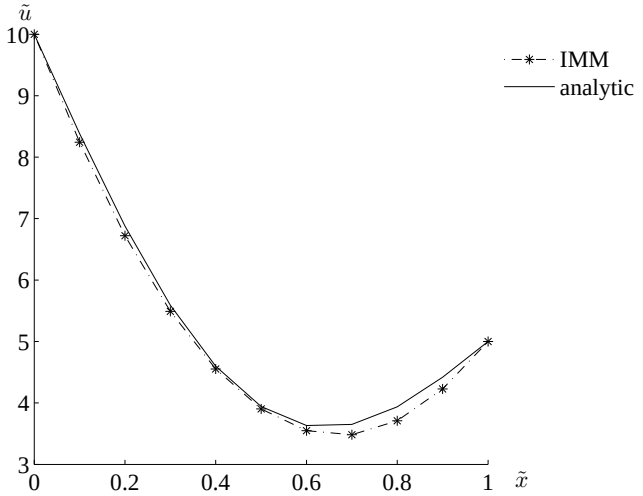

Figure 3.6: The graphs of $\tilde{u}(\tilde{x}, \tilde{t}) = f_1(\tilde{x}_1, \tilde{t}) + f_2(\tilde{x}_2, \tilde{t})$ and the analytic solution at $\tilde{t} = 0.1$ in example 3.2

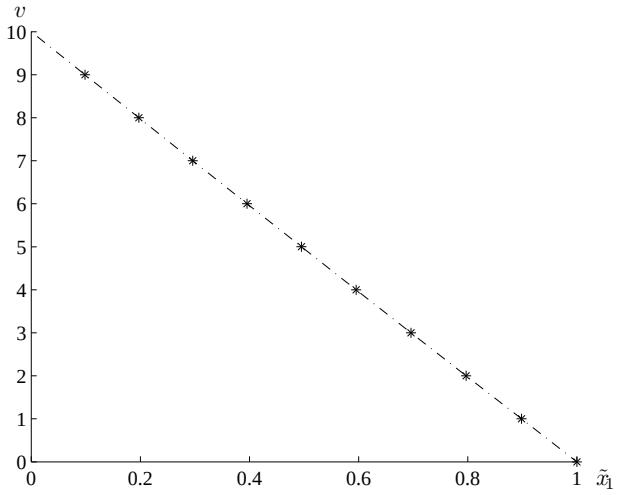

Figure 3.7: The graph of $f_1(\tilde{x}_1, \tilde{t})$ for $\tilde{t} = 0.5$ in example 3.2

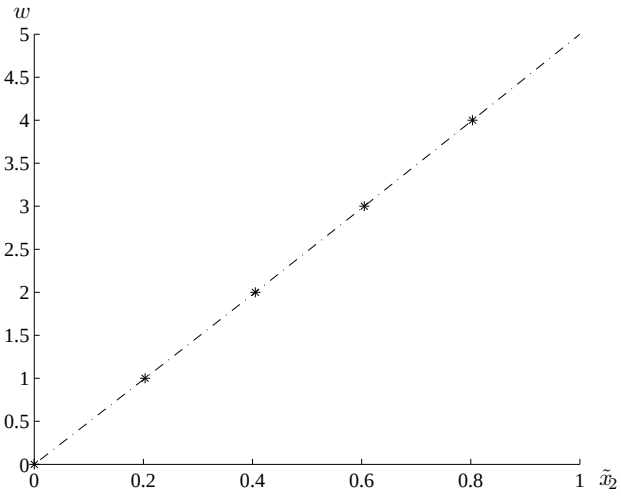

Figure 3.8: The graph of $f_2(\tilde{x}_2, \tilde{t})$ for $\tilde{t} = 0.5$ in example 3.2

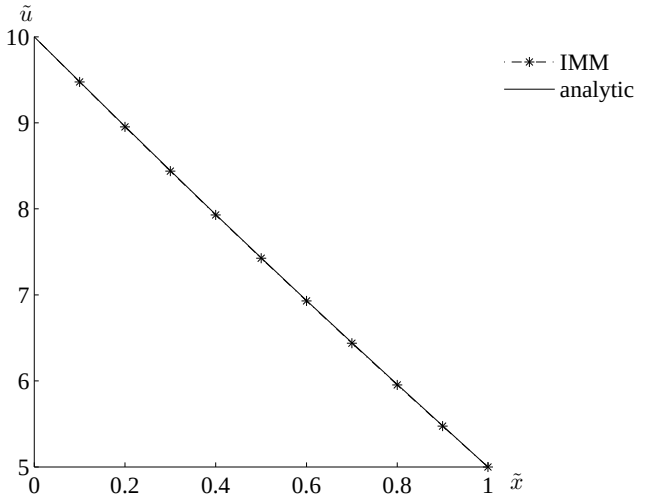

Figure 3.9: The graphs of $\tilde{u}(\tilde{x},\tilde{t}) = f_1(\tilde{x}_1,\tilde{t}) + f_2(\tilde{x}_2,\tilde{t})$ and the analytic solution at $\tilde{t} = 0.5$ in example 3.2

themselves contain errors, we wish to know how the isotherm migration method will perform under these conditions.

Example 3.3

We revisit the problem in Example 3.1 and consider the effect of introducing errors of $\pm 1\%$, $\pm 5\%$ and $\pm 10\%$ in the initial data. We show the results for the positions of isotherms 2 and 8 in tables 3.5 and 3.7 respectively. We also show the percentage errors in the calculated positions for isotherms 2 and 8 respectively in tables 3.6 and 3.8. In all cases, as time progresses, the calculated solution rapidly approaches the analytic solution, suggesting that the isotherm migration method is tolerant of errors in the initial data.

3.5 The case when α is not constant

Since α depends on the physical properties of specific heat, density and thermal diffusivity, it is possible that α might not be constant, for example, if the material is non-homogeneous. In this section we consider the case when α depends on position, that is it varies with x and in particular we

Table 3.4: Error in value of $\tilde{u}$ at $\tilde{t} = 0.5$ for example 3.2

$\tilde{x}$	IMM	analytic	% error
0.1	9.4744	9.4788	0.05
0.2	8.9534	8.9596	0.07
0.3	8.4385	8.4444	0.07
0.4	7.9295	7.9347	0.07
0.5	7.4265	7.4313	0.06
0.6	6.9294	6.9347	0.08
0.7	6.4383	6.4444	0.09
0.8	5.9530	5.9596	0.11
0.9	5.4741	5.4788	0.09

look at the case when $\alpha = 1 + \beta x$ where β is another constant.

We need to re-write equation (3.3) as

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left((1+\beta x)\frac{\partial u}{\partial x}\right) \tag{3.20}$$

$$= (1+\beta x)\frac{\partial^2 u}{\partial x^2} + \beta\frac{\partial u}{\partial x} \tag{3.21}$$

We then apply the change of variables $\tilde{x} = x/a, \tilde{t} = t/a^2, \tilde{x}_0 = x_0/a$, $\tilde{u} = u$ to equation (3.21), so that

$$\frac{\partial u}{\partial t} = \frac{1}{a^2}\frac{\partial \tilde{u}}{\partial \tilde{t}} \tag{3.22}$$

$$\frac{\partial u}{\partial x} = \frac{1}{a}\frac{\partial \tilde{u}}{\partial \tilde{x}} \tag{3.23}$$

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{a^2}\frac{\partial^2 \tilde{u}}{\partial \tilde{x}^2} \tag{3.24}$$

$$1 + \beta x = 1 + \beta a\tilde{x} \tag{3.25}$$

Substituting equations (3.22), (3.23), (3.24) and (3.25) into equation (3.21) we have

$$\frac{1}{a^2}\frac{\partial \tilde{u}}{\partial \tilde{t}} = (1+\beta a\tilde{x})\frac{1}{a^2}\frac{\partial^2 \tilde{u}}{\partial \tilde{x}^2} + \frac{\beta}{a}\frac{\partial \tilde{u}}{\partial \tilde{x}}$$

$$\frac{\partial \tilde{u}}{\partial \tilde{t}} = (1+\beta a\tilde{x})\frac{\partial^2 \tilde{u}}{\partial \tilde{x}^2} + \beta a\frac{\partial \tilde{u}}{\partial \tilde{x}}, \qquad 0 < \tilde{x} < 1,\ \tilde{t} \geqslant 0 \tag{3.26}$$

Table 3.5: Effect of errors in starting values for isotherm 2 in example 3.3

Time	Analytic	1%	5%	10%	-1%	-5%	-10%
0.1	0.5713	0.5771	0.5999	0.6285	0.5656	0.5428	0.5142
0.2	0.7346	0.7338	0.7407	0.7487	0.7302	0.7228	0.7129
0.3	0.7789	0.7793	0.7812	0.7835	0.7783	0.7762	0.7735
0.4	0.7925	0.7928	0.7935	0.7942	0.7925	0.7918	0.7910
0.5	0.7973	0.7974	0.7976	0.7979	0.7973	0.7971	0.7968
0.6	0.7990	0.7991	0.7991	0.7992	0.7990	0.7989	0.7988
0.7	0.7996	0.7997	0.7997	0.7997	0.7996	0.7996	0.7996
0.8	0.7999	0.7999	0.7999	0.7999	0.7999	0.7999	0.7998
0.9	0.7999	0.8000	0.8000	0.8000	0.8000	0.8000	0.7999
1.0	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000
1.1	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000
1.2	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000
1.3	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000
1.4	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000
1.5	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000

At first it appears that equation (3.26) is no less complicated than equation (3.21). However, we now working with non-dimensionalised or scaled variables. Now we have shown in section (3.2) that

$$\frac{\partial \tilde{u}}{\partial \tilde{t}} = -\frac{\frac{\partial \tilde{x}}{\partial \tilde{t}}}{\frac{\partial \tilde{x}}{\partial \tilde{u}}}$$

$$\frac{\partial^2 \tilde{u}}{\partial \tilde{x}^2} = -\frac{\frac{\partial^2 \tilde{x}}{\partial \tilde{u}^2}}{\left(\frac{\partial \tilde{x}}{\partial \tilde{u}}\right)^3}$$

and

$$\frac{\partial \tilde{u}}{\partial \tilde{x}} = \frac{1}{\frac{\partial \tilde{x}}{\partial \tilde{u}}}$$

Therefore equation (3.26) can be written

$$-\frac{\frac{\partial \tilde{x}}{\partial \tilde{t}}}{\frac{\partial \tilde{x}}{\partial \tilde{u}}} = -\left(1+\beta a \tilde{x}\right)\frac{\frac{\partial^2 \tilde{x}}{\partial \tilde{u}^2}}{\left(\frac{\partial \tilde{x}}{\partial \tilde{u}}\right)^3} + \beta a \frac{1}{\frac{\partial \tilde{x}}{\partial \tilde{u}}}$$

which on simplifying becomes

$$\frac{\partial \tilde{x}}{\partial \tilde{t}} = \left(1+\beta a \tilde{x}\right)\frac{\partial^2 \tilde{x}}{\partial \tilde{u}^2}\left(\frac{\partial \tilde{x}}{\partial \tilde{u}}\right)^{-2} - \beta a \tag{3.27}$$

Table 3.6: Percentage errors in solution for isotherm 2 in example 3.3

Time	1%	5%	10%	-1%	-5%	-10%
0.1	1.00	5.00	10.00	1.00	5.00	10.00
0.2	0.11	0.82	1.91	0.60	1.61	2.96
0.3	0.04	0.29	0.58	0.08	0.35	0.69
0.4	0.04	0.11	0.21	0.01	0.09	0.20
0.5	0.02	0.05	0.08	0.00	0.02	0.06
0.6	0.01	0.02	0.03	0.00	0.01	0.02
0.7	0.00	0.01	0.01	0.00	0.00	0.01
0.8	0.00	0.00	0.00	0.00	0.00	0.00
0.9	0.00	0.00	0.00	0.00	0.01	0.00
1.0	0.00	0.00	0.00	0.00	0.00	0.00
1.1	0.00	0.00	0.00	0.00	0.00	0.00
1.2	0.00	0.00	0.00	0.00	0.00	0.00
1.3	0.00	0.00	0.00	0.00	0.00	0.00
1.4	0.00	0.00	0.00	0.00	0.00	0.00
1.5	0.00	0.00	0.00	0.00	0.00	0.00

We then express equation (3.27) in the finite difference form

$$X_i^{(n+1)} = X_i^{(n)} + \left((1+\beta a X)\, X_i^{(n)}\right) 4\delta\tilde{t} \left\{ \frac{X_{i-1}^{(n)} - 2X_i^n + X_{i+1}^{(n)}}{\left(X_{i-1}^{(n)} - X_{i+1}^{(n)}\right)^2} \right\} - \beta a \delta\tilde{t} \tag{3.28}$$

Example 3.4

In this example we consider the case when $\tilde{u}_L = 10, \tilde{u}_R = 0$, $\tilde{u}_0 = 0$ and $a = 1$ as in example 3.1, but where the heat conductivity varies as in equation (3.20), with $\alpha = 1 + \beta\tilde{x}$.

We consider the cases when $\beta = 0.1, 0.5, 1, 2$ and 5. In this case, we do not know the analytic solution and we need some initial values at a small time to start the isotherm migration method. To do this, we use equation (3.20), which is the usual form of the heat equation, and we use a finite difference method with $\delta\tilde{t} = 0.0001$ and $\delta\tilde{x} = 0.1$ to generate a set of temperatures at intervals of $\tilde{x} = 0.1$ in the domain. From this the values u at positions $\tilde{x}$ may found for our chosen starting time. As β increases, the rate of heating also increases and we find that the starting time has to be

Table 3.7: Effect of errors in starting values for isotherm 8 in example 3.3

Time	Analytic	1%	5%	10%	-1%	-5%	-10%
0.1	0.1133	0.1144	0.1190	0.1246	0.1122	0.1076	0.1020
0.2	0.1578	0.1589	0.1617	0.1652	0.1575	0.1547	0.1513
0.3	0.1821	0.1827	0.1840	0.1857	0.1820	0.1806	0.1788
0.4	0.1930	0.1933	0.1938	0.1945	0.1930	0.1924	0.1917
0.5	0.1973	0.1975	0.1977	0.1980	0.1974	0.1972	0.1969
0.6	0.1990	0.1991	0.1992	0.1992	0.1990	0.1990	0.1988
0.7	0.1996	0.1997	0.1997	0.1997	0.1996	0.1996	0.1996
0.8	0.1999	0.1999	0.1999	0.1999	0.1999	0.1999	0.1998
0.9	0.1999	0.2000	0.2000	0.2000	0.2000	0.1999	0.1999
1.0	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000
1.1	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000
1.2	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000
1.3	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000
1.4	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000
1.5	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000

tailored to the particular value of β. We find for $\beta = 5$ we need a much smaller starting time of $\tilde{t} = 0.01$ to capture the events before the steady-state is reached. We find the starting positions as follows. The variable $\tilde{x}$ is plotted as a function of $\tilde{u}$. The trendline together with its equation is added, and the positions of the isotherms required may be found from the equation, for a small time. It is possible that there will be some error in finding the starting values in this way, but example 3.3 showed that this error should decrease with increasing time. The steady-state analytic value for the positions of each isotherm is found as follows:

The steady state is reached when

$$\frac{\partial}{\partial \tilde{x}} \left((1 + \beta \tilde{x}) \frac{\partial \tilde{u}}{\partial \tilde{x}} \right) = 0$$

and integrating twice leads to

$$\tilde{x} = \frac{1}{\beta} \left(\exp \left(\frac{(10 - \tilde{u}) \ln (1 + \beta)}{10} \right) - 1 \right)$$

We show the results for isotherms 2 and 8 respectively in tables 3.9 and 3.10. We show the analytic steady-state position at the bottom of each table. We

Table 3.8: Percentage errors in solution for isotherm 8 in example 3.3

Time	1%	5%	10%	-1%	-5%	-10%
0.1	1.00	5.00	10.00	1.00	5.00	10.00
0.2	0.68	2.46	4.70	0.21	1.97	4.16
0.3	0.28	1.03	1.95	0.10	0.86	1.82
0.4	0.15	0.43	0.78	0.00	0.30	0.68
0.5	0.07	0.18	0.31	0.02	0.09	0.24
0.6	0.04	0.08	0.12	0.02	0.03	0.08
0.7	0.02	0.03	0.05	0.01	0.01	0.03
0.8	0.01	0.01	0.02	0.00	0.00	0.01
0.9	0.00	0.00	0.01	0.00	0.00	0.00
1.0	0.00	0.00	0.00	0.00	0.00	0.00
1.1	0.00	0.00	0.00	0.00	0.00	0.00
1.2	0.00	0.00	0.00	0.00	0.00	0.00
1.3	0.00	0.00	0.00	0.00	0.00	0.00
1.4	0.00	0.00	0.00	0.00	0.00	0.00
1.5	0.00	0.00	0.00	0.00	0.00	0.00

note that the larger the value of β the more quickly the solution settles, which we would expect because increasing β and hence α, will increase the thermal diffusivity so that we would expect heat to flow more rapidly. We show the results for values of $\beta = 0.1, \beta = 1$ and $\beta = 5$ in figures 3.10 and 3.11 for isotherms 2 and 8 respectively.

Table 3.9: The positions of isotherm 2 for linear variation in α in example 3.4

time	$\beta = 0.1$	$\beta = 0.5$	$\beta = 1$	$\beta = 2$	$\beta = 5$
0.10	0.6201	0.7175	0.6436	0.6328	0.6168
0.20	0.7448	0.7562	0.7196	0.6913	0.6297
0.30	0.7776	0.7635	0.7357	0.7007	0.6307
0.40	0.7874	0.7655	0.7395	0.7023	0.6308
0.50	0.7906	0.7660	0.7405	0.7026	0.6308
0.60	0.7917	0.7662	0.7408	0.7026	0.6308
0.70	0.7921	0.7662	0.7408	0.7027	0.6308
0.80	0.7922	0.7663	0.7408	0.7027	0.6308
0.90	0.7923	0.7663	0.7408	0.7027	0.6308
1.00	0.7923	0.7663	0.7408	0.7027	0.6308
s-state	0.7923	0.7663	0.7411	0.7041	0.6386

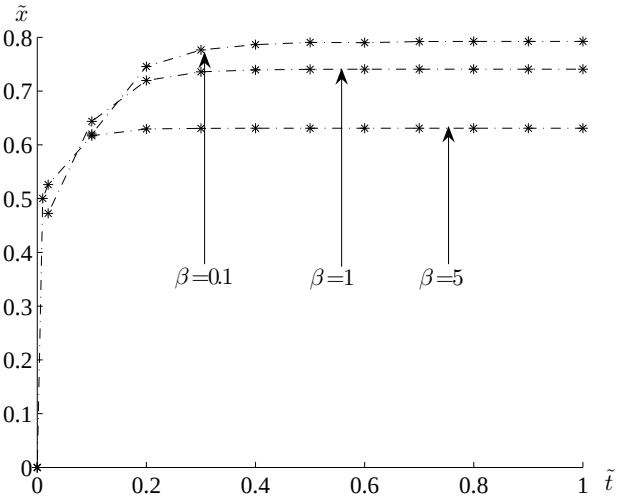

Figure 3.10: The positions of isotherm 2 for linear variation in α in example 3.4

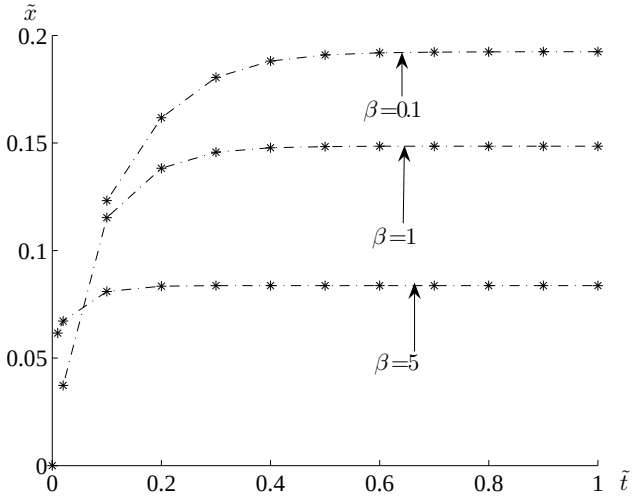

Figure 3.11: The positions of isotherm 8 for linear variation in α in example 3.4

Table 3.10: The positions of isotherm 8 for linear variation in α in example 3.4

time	$\beta = 0.1$	$\beta = 0.5$	$\beta = 1$	$\beta = 2$	$\beta = 5$
0.10	0.1232	0.1542	0.1153	0.1029	0.0810
0.20	0.1617	0.1623	0.1382	0.1180	0.0835
0.30	0.1805	0.1669	0.1457	0.1213	0.0837
0.40	0.1881	0.1683	0.1478	0.1220	0.0837
0.50	0.1909	0.1687	0.1483	0.1221	0.0837
0.60	0.1919	0.1689	0.1485	0.1221	0.0837
0.70	0.1923	0.1689	0.1485	0.1221	0.0837
0.80	0.1924	0.1689	0.1485	0.1221	0.0837
0.90	0.1924	0.1689	0.1485	0.1221	0.0837
1.00	0.1924	0.1689	0.1485	0.1221	0.0837
s-state	0.1924	0.1689	0.1487	0.1229	0.0862

3.6 The effect of approximating the initial values on the accuracy of the solution

Example 3.5

We use the same problem as that in example 3.4, but rather than calculating the initial positions using $\alpha = 1 + \beta\tilde{x}$, we use an average value for α across the region, so that we take $\alpha = 1 + \beta/2$. This means that we can treat α as a constant and use the analytic solution to equation (3.20) which is

$$u(x,t) = u_L + (u_R - u_L)\frac{x}{a} + \sum_{n=1}^{\infty} b_n \sin\left(\frac{n\pi x}{a}\right) \exp\left(\frac{-\alpha n^2 \pi^2 t}{a^2}\right)$$

to find the starting values at $\tilde{t} = 0.1$

We show the results for isotherms 2 and 8 with the results when we take the average value of $\alpha = 1 + \beta\tilde{x}$ to start the problem, in table 3.11 for $\beta = 0.1$ and table 3.12 for $\beta = 5$. Again the starting times need to be tailored to suit the particular values of β. We see that after a short time, the solutions using the smaller value of β appear to have converged to the steady-state value by $\tilde{t} = 1.0$ while the larger value of β causes the solution to settle more quickly, but it is not so close to the steady-state value.

This indicates that we can use this average method with confidence, and

it is much more straightforward to find the starting values in this way, as the analytic solution is used directly and we can therefore avoid using the method described in example 3.4, in which we have to find temperatures as a function of position, then use these to plot a graph of position as a function of temperature from which we attempt to find suitable positions of the isotherms.

Table 3.11: The positions of isotherms 2 and 8 when an average value of β is taken to find starting values for $\beta = 0.1$ in example 3.5

Time	Isotherm 2	Isotherm 8
0.1	0.6201	0.1240
0.2	0.7415	0.1605
0.3	0.7760	0.1794
0.4	0.7867	0.1874
0.5	0.7903	0.1906
0.6	0.7916	0.1918
0.7	0.7920	0.1922
0.8	0.7922	0.1924
0.9	0.7923	0.1924
1.0	0.7923	0.1924
s-state	0.7923	0.1925

Table 3.12: The positions of isotherms 2 and 8 when an average value of β is taken to find starting values for $\beta = 5$ in example 3.5

time	isotherm2	isotherm 8
0.1	0.5761	0.0749
0.2	0.6266	0.0828
0.3	0.6305	0.0836
0.4	0.6308	0.0837
0.5	0.6308	0.0837
0.6	0.6308	0.0837
0.7	0.6308	0.0837
0.8	0.6308	0.0837
0.9	0.6308	0.0837
1.0	0.6308	0.0837
s-state	0.6386	0.0862

3.7 Summary of Chapter 3

In this chapter we have described the background to the isotherm migration method and we have shown how to carry out the mapping of the heat equation so that position is expressed as a function of temperature and time. We have discussed the use of the finite difference method to solve the mapped equation and we have tested the method using simple heating examples. We noted that difficulties may arise where we have isotherms appearing or disappearing during heating, but we illustrated a modification with which this may be overcome. We believe the method to be simple to operate, robust and tolerant of errors in the initial data, which we have demonstrated using suitable examples. Using the case where the heat conductivity is not constant we have described two methods for finding initial data when an analytic solution in not available and we conclude that simplifying the procedure for finding the initial data as far as possible still enables accurate solutions to be found.

3.7.1 Contribution

In general the isotherm migration method is a solution method for problems involving phase change. We have used it in a different setting to solve simple heat-conduction problems, identifying and attempting to overcome any difficulties which arise. We have tested the method and found it to be robust, tolerant of errors in initial data and we have examined cases with non-constant heat conductivity and shown the method performs well.

Chapter 4

The Laplace transform isotherm migration method for one-dimensional problems with no phase change

The Laplace transform is a well established technique for the solution of ordinary and partial differential equations for initial value problems. In the context of the heat equation, using the Laplace transform removes the time variable, leaving an ordinary differential equation in Laplace space which we solve using the initial and boundary conditions. This solution then requires an inverse transform to recover the solution in the usual time and space variables.

We showed in chapter 3 that when using the isotherm migration method, we have to consider the condition placed on the size of the time step to avoid instability, and that this time step is necessarily very small and therefore a large number of calculations are required for solution of the problem. We are therefore interested in whether using the Laplace transform will enable

us to reduce the number of calculations by allowing us to remove the time stepping aspect. Full accounts of the Laplace transform and its applications are to be found in Carslaw and Jaeger (1959) and Spiegel (1965). We shall give a brief outline of the process here.

4.1 The Laplace transform definition

Given f, a function of time with value $f(t)$ at time t, the Laplace transform is defined as

$$\mathcal{L}\left[f(t)\right] \equiv \bar{f}(\lambda) = \int_0^\infty f(t)e^{-\lambda t}dt \tag{4.1}$$

where $f(t)$ is defined for $t \geqslant 0$, as stated by Abramovitz and Stegun (1972). The Laplace transforms of the derivatives $f'(t), f''(t), ..., f^{(n)}(t), ...$ are given by the following formula (Spiegel 1965)

$$\mathcal{L}\left[f^{(n)}(t)\right] = \lambda^n \bar{f}(s) - \lambda^{n-1}f(0) - \lambda^{n-2}f'(0) - ... - \lambda f^{(n-2)}(0) - f^{(n-1)}(0) \tag{4.2}$$

Equation (4.1) is a Fredholm equation of the first kind. These equations contain an integral which is the product of a kernel, which is a known function, and an unknown function which is the function we seek. Fredholm equations are inherently ill-posed and the solution is extremely sensitive to arbitrarily small perturbations of the system (Hansen 1992).

4.2 The inverse transform

We now need a method to perform an inverse transformation to find the required solution. Simple cases may be solved algebraically, and inverted by using tables, but in many situations it is impossible to arrange the transform so that the terms in the expression correspond to the standard transforms

available in tables. We do have an inversion formula (Spiegel 1965),

$$f(t) = \frac{1}{2\pi i} \int_{\gamma - i\infty}^{\gamma + i\infty} \bar{f}(\lambda) e^{\lambda t} d\lambda$$

an integral in the complex plane, which we would need to evaluate using a suitable contour, the Bromwich contour. This would provide an exact solution, but for any other than very simple cases, it is very difficult to use. Therefore we need to consider the use of a numerical inversion method.

Rizzo and Shippy (1970) described a method for solving heat conduction boundary value problems using the Laplace transform and a method of inversion due to Schapery (1962). They chose an arbitrary Laplace transform parameter, which is required for a curve fitting process to determine a sequence used in Schapery's inversion method. A poor choice of parameter resulted in unstable solutions or a badly defined curve and so their results lacked accuracy. Piessens (1972) described a method which approximates the transform by truncated polynomials related to the Laguerre polynomials and obtained results which compared favourably with other methods involving interpolating functions. Zakian and Littlewood (1973) described a numerical inversion using a weighted least-squares approximation with Legendre polynomials. The inversion formula is expressed as the sum of a series of terms which are a combination of a polynomial in space and an exponential in time. The total error in inversion is stated to be the sum of roundoff and truncation errors and they found that there was an optimum value for the number of terms taken in the series, above which the total error begins to rise. Lachat and Combescure (1977) used a method of inversion involving a series of Legendre polynomials which resulted in a triangular matrix which they found was very ill-conditioned. The method was suitable for examples involving thermal shocks at initial time, a thermal shock being a situation where a region initially at some temperature is subjected to an instantaneous change of temperature on its boundary. They found that when a second thermal shock was applied at a later time, the results

using their inversion method were not as accurate as the results found by the boundary integral method. They therefore suggested that the method is limited to those problems involving shocks at initial times.

Davies and Martin (1979) tested a variety of numerical methods for inverting the Laplace transform. They used several criteria, including the applicability of the method to the problem, accuracy, computation time and ease of programming and implementation. The methods they tested were

1. Methods which compute a sample and involve choosing an integral

$$I_n(t) = \int_0^\infty \delta_n(t,u) f(u)\, du$$

such that I_n tends to $f(t)$, the inversion integral, with increasing n. This includes the extrapolation formula derived by Stehfest (1970), which will be described in more detail later.

2. Expanding $f(t)$ in exponential functions, usually by introducing $\exp(-t)$ as a new independent variable. These methods include the use of the Legendre polynomials, trigonometric functions, and Schapery's (1962) expansion

$$f(t) = A + bt + \sum_{k=1}^{N} a_k e^{-b_k t}$$

where the exponent weights b_k are chosen to suit the expected form of the function $f(t)$. Laplace transformation yields

$$\lambda F(\lambda) = A + b\lambda^{-1} + \sum_{k=1}^{N} a_k \left(1 + b_k \lambda^{-1}\right)^{-1}$$

and these equations are solved for a_k by substituting

$$\lambda = b_k, \quad k = 1, ...N$$

and also using the identities

$$B = \lim_{\lambda \to 0} \lambda^2 F(\lambda)$$

$$A = -\sum_{k=1}^{N} a_k$$

3. Gauss quadrature, which may be used only for Laplace transforms of a particular form.

4. Bilinear transformations, expanding the function $f(t)$ in a series of Laguerre functions or Jacobi polynomials.

5. Representation by Fourier series.

Their results showed that there is no simple conclusion to be drawn as to which method is better than another, and that each method has advantages and disadvantages, which they outline in an attempt to help make a suitable choice for a particular problem. They say that all methods have something to offer but because it is simple to use and in most cases provides accurate results, they recommend the Stehfest method. Moridis and Reddell (1991a, 1991b and 1991c) used the Laplace Transform with the finite difference method, the finite element method and the boundary element method respectively to solve diffusion-type problems and the method of inversion they used is due to Stehfest (1970). This method seems to be the method of choice for diffusion-type problems as the procedure is stable and errors are considerably smaller than those in other approaches. It has been used by Zhu *et al.* (1994), Davies *et al.* (1997), Crann *et al.* (1998), Honnor and Davies (2002) and Crann *et al.* (2005), and continues to be routinely used. It is this method which we use in our work and so we give a description of it in the next section.

4.3 The Stehfest numerical inversion method

The Stehfest (1970) formula is a weighted sum of transform values at a discrete set of transform parameters.

If $\bar{f}(\lambda)$ is the Laplace transform of $f(t)$ then we seek a value $f(T)$ for the specific time value $t = T$. To use the inversion algorithm, we choose a discrete set of transform parameters

$$\lambda_j = j\frac{\ln 2}{T}, \qquad j = 1, 2, ..., M$$

where M is even. The approximate numerical inversion is given by

$$f(T) \approx \frac{\ln 2}{T} \sum_{j=1}^{M} w_j \overline{f}(\lambda_j)$$

The weights are given by

$$w_j = (-1)^{\left(\frac{M}{2}\right)+j} \sum_{k=\left[\frac{1}{2}(1+j)\right]}^{\min\left(j,\frac{M}{2}\right)} \frac{(2k)!k^{\frac{M}{2}}}{\left(\frac{M}{2}-k\right)!k!\,(k-1)!\,(j-k)!\,(2k-j)!} \tag{4.3}$$

In table 4.1 we show the numerical values for the Stehfest weights for $M = 4, 6, 8, 10, 12, 14$ and 16.

Table 4.1: Stehfest weights for $M = 4, 6, 8, 10, 12, 14$ and 16

M=4	M=6	M=8	M=10	M=12	M=14	M=16
-2	1	0.3333333	0.083333	-0.01667	0.0027778	0.00039683
26	-49	48.333333	-32.0833	16.01667	-6.4027778	2.13372957
-48	366	-906	1279	-1247	924.05	-551.016667
24	-858	5464.6667	-15623.7	27554.33	-34597.929	33500.1667
	810	-14376.67	84244.17	263280.8	540321.11	-812665.111
	-270	18730	-23957.5	1324139	-4398346.4	10076183.8
		-11946.67	375912	-3891706	21087592	-73241383
		2986.6667	-340072	7053286	-63944913	339059632
			164062.5	-8005337	127597580	-1052539536
			-32812.5	5552831	-170137188	2259013329
				-2155507	150327467	-3399701984
				359251.2	-84592162	3582450462
					27478885	-2591494081
					-3925555	1227049829
						-342734555
						42841819.4

The value of M is chosen by the user. We see that the numerical values of the weights become very large as M increases and this means that, unless high precision arithmetic is used, there will be round-off errors in the inversion process.

Moridis and Reddell (1991a,1991b and 1991c) tested the inversion method in three diffusion type problems using values $6 \leqslant M \leqslant 20$, and they concluded that although the accuracy of the solution increases as M increases, the improvement for $M > 6$ is marginal and insufficient to justify the additional execution time. Crann (1996) suggests that accuracy decreases for $M \geqslant 10$ and Zhu *et al.* (1994) suggest that a value of $M = 6$ gives the best accuracy. From this it seems clear that the choice of M depends on the problem being solved and that it is not possible to state categorically an ideal value before starting to solve the problem.

We follow the suggestion of Davies and Crann (1999) and choose a value of $M = 8$ for our work, since we require a value for M which will give us accuracy, while at the same time, not require more calculations than are necessary.

It is worth mentioning that the Stehfest inversion method does not work well with problems where the solution is oscillatory or if it behaves like $\exp(\kappa t)$ where $\kappa > 0$. This was confirmed by Crann (1996) who showed that for an example involving the wave equation, the solution only compared well for the first quarter period of the vibration, after which time it became progressively worse and ultimately bore no resemblance to the analytic solution.

4.4 The Laplace transform method of solution for linear diffusion problems

The Laplace transform is a linear operator and therefore it is applied to linear differential equations.

The diffusion equation, equation (4.4), is not periodic and is linear, and therefore is a suitable choice to illustrate the Laplace transform method of solution using Stehfest inversion.

$$\frac{\partial}{\partial x}\left(\alpha\left(x\right)\frac{\partial u}{\partial x}\right)=\frac{\partial u}{\partial t}\quad 0<x<l,\quad t>0 \tag{4.4}$$

Consider equation (4.4) subject to Dirichlet boundary conditions

$$u\left(0,t\right)=u_{L}\left(t\right),\quad u\left(l,t\right)=u_{R}\left(t\right),\quad t<0 \tag{4.5}$$

and the initial condition

$$u\left(x,0\right)=u_{0}\left(x\right)\quad 0<x<l \tag{4.6}$$

Taking the Laplace transform we obtain from equations (4.4), (4.5) and (4.6)

$$\frac{d}{dx}\left(\alpha\left(x\right)\frac{d\bar{u}}{dx}\right)=\lambda\bar{u}-u_{0}\left(x\right) \tag{4.7}$$

together with

$$\bar{u}_{0}\left(0\right)=\bar{u}_{L}\quad \bar{u}\left(l\right)=\bar{u}_{R}$$

This example is solved by Davies and Crann (1999), both for α constant and α as a linear function of x. They use the finite difference method to solve equation (4.7), and at the transform parameter they substitute $\lambda=\lambda_{j}$. This leads to a tri-diagonal system to be solved to find the approximations $\bar{U}_{i,j}$ to the solutions u at the required grid points. A Gauss-Seidel approach is employed to solve the tri-diagonal system and solutions

$$U_{i}\approx\frac{\ln 2}{T}\sum_{j=1}^{M}w_{j}\bar{U}_{i,j} \tag{4.8}$$

are found where w_j are the Stehfest weights. Davies and Crann (1999) compare their results with the analytic solution for α constant and are able to deduce expected behaviour for the case when α is not constant, and they conclude that the method produces expected results and is very easy to apply. Full details are to be found in Davies and Crann (1999) and we will not consider the solution of linear problems further here.

4.5 The Laplace transform method of solution for non-linear problems

We have stated that the Laplace transform method can only be applied to linear problems. However we would like to use the method in combination with the isotherm migration method. The transformed heat equation under the isotherm migration mapping is non-linear but we do have a linearisation process to overcome this which will be described in the next example.

Example 4.1

We consider the problem described in example 3.1, and for clarity we repeat the description of the problem here. In addition, since we now have to write the transformed variable with a bar above the symbol, we will drop the use of the tilde to represent the dimensionless variable, and when we write u, x, t, or any of these with a subscript, we mean the dimensionless form. We consider a rod of unit length, initially at uniform temperature u_0. A constant heat source is applied so that the left-hand end of the rod is maintained at a temperature u_L, while the right-hand end is maintained at u_R. The boundary conditions for the problem are

$$u(0,t) = u_L, \quad u(1,t) = u_R \tag{4.9}$$

and the initial condition is

$$u(x,0) = u_0 \tag{4.10}$$

The analytic solution to this problem is given by equation (4.11)

$$u(x,t) = u_L + (u_R - u_L)\, x + \sum_{n=1}^{\infty} b_n \sin(n\pi x) \exp(-n^2\pi^2 t) \tag{4.11}$$

where

$$b_n = \frac{2}{n\pi}\left\{(u_0 - u_L)\left(1 - (-1)^n\right) + (u_R - u_L)\left(-1\right)^n\right\}$$

In this example we consider the case where $u_L = 10$, $u_R = 0$ and $u_0 = 0$.

We begin with the heat equation which has undergone the isotherm mapping process

$$\frac{\partial x}{\partial t} = \left(\frac{\partial x}{\partial u}\right)^{-2} \frac{\partial^2 x}{\partial u^2} \tag{4.12}$$

This is a non-linear equation and so we need a method to linearise it before we can use the Laplace transform method to solve it. We use the direct iteration method suggested by Zhu (1999) and Crann (2005). We put the previous numerical result for

$$\left(\frac{\partial x}{\partial u}\right)^{-2}$$

into the next iteration so that

$$\frac{\partial x^{(n)}}{\partial t} = \left(\left(\frac{\partial x}{\partial u}\right)^{-2}\right)^{(n-1)} \frac{\partial^2 x^{(n)}}{\partial u^2}$$

and then take the Laplace transform of this linear equation to get

$$\lambda \bar{x}^{(n)} - x\left(t_0\right) = \left(\left(\frac{\partial x}{\partial u}\right)^{-2}\right)^{(n-1)} \frac{\partial^2 \bar{x}^{(n)}}{\partial u^2} \tag{4.13}$$

We represent the position of the isotherm with temperature u_i by x_i, so for any particular isotherm, equation (4.13) becomes

$$\lambda \bar{x}_i^{(n)} - x_i\left(t_0\right) = \left(\left(\frac{\partial x}{\partial u}\right)^{-2}\right)_i^{(n-1)} \frac{\partial^2 \bar{x}_i^{(n)}}{\partial u^2}$$

We use a central difference approximation with $\bar{X}_i$ being the approximation to $\bar{x}_i$,

$$\lambda \bar{X}_i^{(n)} - X_i\left(t_0\right) = \left(\left(\frac{\partial X}{\partial u}\right)^{-2}\right)_i^{(n-1)} \frac{\bar{X}_{i+1}^{(n)} - 2\bar{X}_i^{(n)} + \bar{X}_{i-1}^{(n)}}{h^2}$$

where h is the difference in the value of succesive temperatures in whose isotherms we are interested.

The algorithm to find the next position of the isotherm with temperature u_i is

$$\bar{X}_i^{(n+1)} = \frac{1}{\left(\lambda + \frac{2}{h^2}\left(\left(\frac{\partial X}{\partial u}\right)^{-2}\right)_i^{(n)}\right)}\left[X_i^{(n)}(t_0) + \left(\left(\frac{\partial X}{\partial u}\right)^{-2}\right)_i^{(n)}\left(\frac{\bar{X}_{i+1}^{(n)} - \bar{X}_{i-1}^{(n)}}{h^2}\right)\right] \tag{4.14}$$

As in example 3.1, we use the analytic solutions at $t = 0.1$ to obtain the starting positions for the isotherms at the required temperatures. Before performing the Laplace transform, we obtain a numerical value for the term $\left(\left(\frac{\partial X}{\partial u}\right)^{-2}\right)_i^{(n)}$ from the starting values. We now enter a loop to perform the Laplace transform, using the Stehfest method. At this point we have to choose a suitable value for the parameter T which is needed to calculate the Stehfest values λ_j. Clearly this must be chosen carefully, and it cannot be too small, otherwise the value of λ_j could become extremely large and lead to inaccuracies, particularly as there is a wide variation in the range of Stehfest weights. Crann (2005) suggests that values of T less than 0.1 should not be used, and so in this example we use $T = 0.1$. The next stage is to transform the boundary conditions into Laplace space. We now use a Gauss-Seidel solver to perform the iteration. We use equation (4.14) for each λ_j, $j = 1, 2, ..., M$ and we obtain $\bar{X}_i(\lambda_j)$, the value of $\bar{X}_i$ for each transform parameter λ_j. We continue the iteration until agreement to an acceptable tolerance has been reached. On achieving convergence, we perform the inverse transform using the Stehfest method. This inversion process takes the form

$$X_i \approx \frac{\ln 2}{T}\sum_{j=1}^{M} w_j \bar{X}_i(\lambda_j) \tag{4.15}$$

where the w_j are the Stehfest weights.

Because we are using direct iteration to linearise the isotherm migration equation, we now test for convergance of $X_i^{(n+1)}$ and $X_i^{(n)}$. If the solutions

do not agree to an acceptable tolerance, we recalculate $\left(\left(\frac{\partial X}{\partial u}\right)^{-2}\right)^{(n)}$ and continue the calculation, by reapplying the Laplace transform.

Table 4.2: Comparison of the results using the isotherm migration method alone and the Laplace transform with the isotherm migration method for isotherm 2 in example 4.1

Time	IMM	% Error	LTIMM	% Error	Analytic
0.1	0.5713	0.00	0.5713	0.00	0.5713
0.2	0.7320	0.35	0.7333	0.18	0.7346
0.3	0.7788	0.02	0.7784	0.07	0.7789
0.4	0.7927	0.02	0.7919	0.08	0.7925
0.5	0.7974	0.01	0.7968	0.06	0.7973
0.6	0.7990	0.01	0.7987	0.03	0.7990
0.7	0.7997	0.00	0.7995	0.01	0.7996
0.8	0.7999	0.00	0.7999	0.00	0.7999
0.9	0.8000	0.00	0.8000	0.00	0.7999
1.0	0.8000	0.00	0.8000	0.00	0.8000
1.1	0.8000	0.00	0.8000	0.00	0.8000
1.2	0.8000	0.00	0.8000	0.00	0.8000
1.3	0.8000	0.00	0.8000	0.00	0.8000
1.4	0.8000	0.00	0.8000	0.00	0.8000
1.5	0.8000	0.00	0.8000	0.00	0.8000

In tables 4.2 and 4.3 we show the results for the isotherms with temperatures 2 and 8 respectively, which we obtain using the Laplace transform isotherm migration method and the analytic solutions, and we include the results which we obtain with the isotherm migration method in example 3.1 for comparison. We see that the Laplace transform method compares well with the analytic solutions and is just as accurate as the isotherm migration method alone. Figure 4.1 shows a plot of the solutions obtained using the Laplace transform isotherm migration method (LTIMM) for isotherms with temperatures 2 and 8, together with a plot of the analytic values for these isotherms. We see that the Laplace transform method gives very accurate results.

Table 4.3: Comparison of the results using the isotherm migration method alone and the Laplace transform with the isotherm migration method for isotherm 8 in example 4.1

Time	IMM	% Error	LTIMM	% Error	Analytic
0.1	0.1133	0.00	0.1133	0.00	0.1133
0.2	0.1582	0.24	0.1552	1.64	0.1578
0.3	0.1823	0.09	0.1810	0.65	0.1822
0.4	0.1933	0.17	0.1923	0.37	0.1930
0.5	0.1974	0.05	0.1969	0.21	0.1973
0.6	0.1991	0.03	0.1988	0.08	0.1990
0.7	0.1997	0.01	0.1996	0.00	0.1996
0.8	0.1999	0.01	0.2000	0.05	0.1999
0.9	0.2000	0.00	0.2001	0.06	0.1999
1.0	0.2000	0.00	0.2001	0.06	0.2000
1.1	0.2000	0.00	0.2001	0.05	0.2000
1.2	0.2000	0.00	0.2001	0.04	0.2000
1.3	0.2000	0.00	0.2001	0.03	0.2000
1.4	0.2000	0.00	0.2000	0.00	0.2000
1.5	0.2000	0.00	0.2000	0.00	0.2000

4.6 The solution of non-linear boundary value problems using the Laplace transform isotherm migration method

We consider the problem

$$\frac{\partial u}{\partial t} = \frac{\partial}{dx}\left(\alpha(u)\frac{\partial u}{\partial x}\right) \tag{4.16}$$

which we expand as

$$\frac{\partial u}{\partial t} = \frac{d\alpha}{du}\frac{\partial u}{\partial x}\frac{\partial u}{\partial x} + \alpha\left(u\right)\frac{\partial^2 u}{\partial x^2}$$

$$= \frac{d\alpha}{du}\left(\frac{\partial u}{\partial x}\right)^2 + \alpha(u)\frac{\partial^2 u}{\partial x^2} \tag{4.17}$$

The isotherm migration mapping for this case is carried out as follows. We write equation (4.17) as

$$-\left(\frac{\partial x}{\partial t}\right)\left(\frac{\partial x}{\partial u}\right)^{-1} = \left(\frac{d\alpha}{du}\right)\left(\frac{\partial x}{\partial u}\right)^{-2} - \alpha(u)\left(\frac{\partial x}{\partial u}\right)^{-3}\frac{\partial^2 x}{\partial u^2}$$

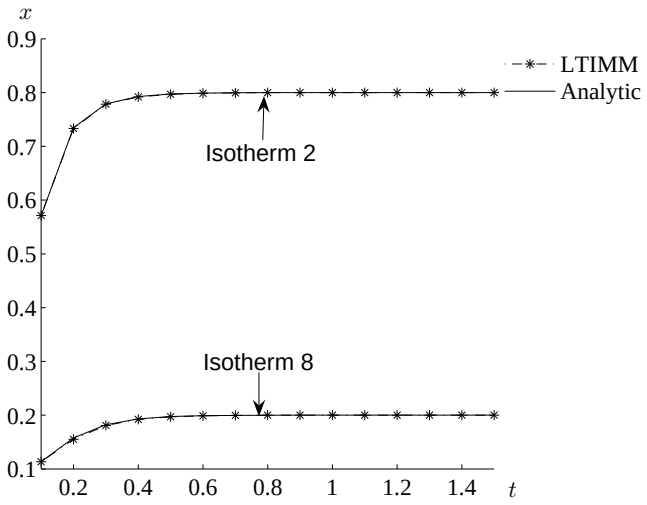

Figure 4.1: Comparison of the Laplace transform isotherm migration method with the analytic solution for isotherms 2 and 8 in example 4.1

Multiplying through by

$$\left(\frac{\partial x}{\partial u}\right)$$

leads to

$$-\frac{\partial x}{\partial t} = \left(\frac{d\alpha}{du}\right)\left(\frac{\partial x}{\partial u}\right)^{-1} - \alpha(u)\left(\frac{\partial x}{\partial u}\right)^{-2}\frac{\partial^2 x}{\partial u^2}$$

and hence

$$\frac{\partial x}{\partial t} = \alpha(u)\left(\frac{\partial x}{\partial u}\right)^{-2}\frac{\partial^2 x}{\partial u^2} - \left(\frac{d\alpha}{du}\right)\left(\frac{\partial x}{\partial u}\right)^{-1} \tag{4.18}$$

After linearising the equation by using direct iteration, we are able to take the Laplace transform of equation (4.18):

$$\lambda\bar{x}^{(n)} - x\left(t_0\right) = \alpha\left(u\right)\left(\left(\frac{\partial x}{\partial u}\right)^{-2}\right)^{(n-1)}\left(\frac{\partial^2\bar{x}}{\partial u^2}\right)^{(n-1)} - \frac{1}{\lambda}\left(\left(\frac{d\alpha}{du}\right)\left(\frac{\partial x}{\partial u}\right)^{-1}\right)^{(n-1)} \tag{4.19}$$

Example 4.2

We take equation (4.16) and consider a rod of unit length, initially at uniform temperature $u_0 = u(x, t_0)$ with boundary conditions $u_L = u(0, t)$ and $u_R = u(1, t)$. In this problem $u_L = 0$ and $u_R = 1$. We solve the problem for

$\alpha = 1 + u$. For the steady-state solution

$$\frac{\partial u}{\partial t} = 0$$

and therefore

$$x = \frac{2u + u^2}{3}$$

which gives the positions of the isotherms in the steady-state.
Since we do not know an analytic solution to this problem, this will give us some indication of the accuracy of our method.

The Laplace transform of the linearised problem is

$$\lambda \bar{x}^{(n)} - \tilde{x}\left(t_0\right) = (1+u)\left(\left(\frac{\partial x}{\partial u}\right)^{-2}\right)^{(n-1)}\left(\frac{\partial^2 \bar{x}}{\partial u^2}\right)^{(n-1)} - \frac{1}{\lambda}\left(\left(\frac{\partial x}{\partial u}\right)^{-1}\right)^{(n-1)}$$

Using a finite difference method as described before we have

$$\bar{X}_i^{(n+1)} = \frac{1}{\lambda + \frac{2(1+u_i)}{h^2}\left(\left(\frac{\partial X}{\partial u}\right)^{-2}\right)^{(n)}}$$

$$\times \left\{ X\left(t_0\right) + (1+u_i)\left(\left(\frac{\partial X}{\partial u}\right)^{-2}\right)^{(n)}\left(\frac{\left(\bar{X}_{i+1}^{(n)} - \bar{X}_{i-1}^{(n)}\right)}{h^2} - \frac{1}{\lambda}\left(\left(\frac{\partial X}{\partial u}\right)^{-1}\right)^{(n)}\right)\right\}$$

which we iterate to convergence and we use Stehfest inversion as given by equation (4.15).

In order to start the problem, we use an accurate finite difference method with $\delta x = 0.01$ and $\delta t = 0.00002$ and we find the temperature at various positions. We choose a starting time of $t = 0.1$ and we plot the position as a function of temperature and we use the equation of the best fit trend line to calculate the starting values.

In table 4.4 we show the positions of the isotherms at $t = 0.1, 0.2, 0.3$ and 1.5 and in table 4.5 we show that the steady-state positions are achieved when $t = 1.5$. These results are shown graphically in figure 4.2. Table 4.5 shows that the method gives very accurate results.

We also wish to see how the solution is affected by introducing some errors into the starting values, and so we use an average value of α to

Table 4.4: The positions of the isotherms at $t = 0.1, 0.2, 0.3$ and 1.5 when $\alpha = 1 + u$ in example 4.2

u	$t = 0.1$	$t = 0.2$	$t = 0.3$	$t = 1.5$
0.0	0.0000	0.0000	0.0000	0.0000
0.2	0.2749	0.1700	0.1519	0.1467
0.4	0.4927	0.3600	0.3293	0.3200
0.6	0.6821	0.5616	0.5302	0.5199
0.8	0.8432	0.7734	0.7535	0.7466
1.0	1.0000	1.0000	1.0000	1.0000

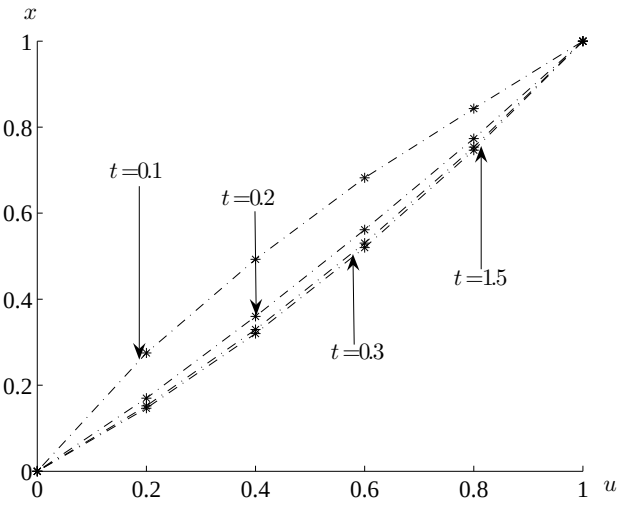

Figure 4.2: Position as a function of temperature, u, for times $t = 0.1, 0.2, 0.3$ and 1.5 when $\alpha = 1 + u$ in example 4.2

calculate the starting values. We have $0 \leqslant u \leqslant 1$ so that the average value of α is 1.5. We obtain the values for u when $t = 0.1$ as before, using the finite difference method, but with $\alpha = 1.5$, that is, α is constant. We see that the isotherms in figure 4.3 show a similar trend to those in figure 4.2 and the steady-state values are reached when $t = 1.5$.

We show in tables 4.7 and 4.8 that altering the start values by $\pm 1\%$ and $\pm 5\%$ respectively has little effect on the steady state solution, thus indicating that this is a robust method which is apparently not overly sensitive to changes in the initial data. This is a useful property as we may not know

Table 4.5: The positions of the isotherms when the steady-state is reached for $\alpha = 1 + u$ in example 4.2

u	Analytic	LTIMM	% error
0.0	0.0000	0.0000	0.00
0.2	0.1467	0.1467	0.01
0.4	0.3200	0.3200	0.01
0.6	0.5200	0.5199	0.01
0.8	0.7467	0.7466	0.01
1.0	1.0000	1.0000	0.00

Table 4.6: The positions of the isotherms at $t = 0.1, 0.2, 0.3$ and 1.5 when the average value of α is used to start the problem in example 4.2

u	$t = 0.1$	$t = 0.2$	$t = 0.3$	$t = 1.5$
0.00	0.0000	0.0000	0.0000	0.0000
0.20	0.3022	0.1773	0.1534	0.1467
0.40	0.5375	0.3717	0.3321	0.3200
0.60	0.7290	0.5729	0.5331	0.5199
0.80	0.8768	0.7802	0.7554	0.7466
1.00	1.0000	1.0000	1.0000	1.0000

an exact solution to start these problems, and we may be fairly confident that a small error in the initial data will not produce an unacceptable error in the solution.

Table 4.7: The effect of changing the starting values, by $\pm 1\%$ on the steady state solution for $\alpha = 1 + u$ in example 4.2

Analytic	SV-1%	SV+1%
0.0000	0.0000	0.0000
0.1467	0.1467	0.1467
0.3200	0.3200	0.3200
0.5200	0.5200	0.5200
0.7467	0.7466	0.7466
1.0000	1.0000	1.0000

Example 4.3

The boundary and initial conditions for this problem are the same as those in example 4.2 that is, the uniform temperature is $u_0 = u(x, t_0)$ initially and the boundary conditions are $u_L = u(0, t)$ and $u_R = u(1, t)$. We consider the

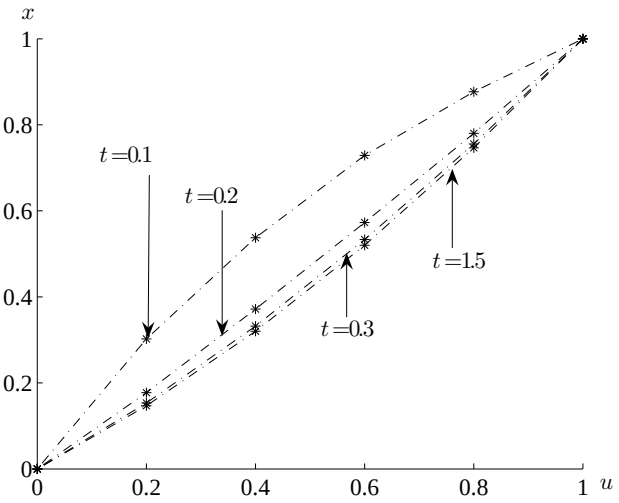

Figure 4.3: Position as a function of temperature, u, for times $t = 0.1, 0.2, 0.3$ and 1.5 when the average value of α is used to start the problem in example 4.2

Table 4.8: The effect of changing the starting values, by $\pm 5\%$ on the steady state solution for $\alpha = 1 + u$ in example 4.2

Analytic	SV-5%	SV+5%
0.0000	0.0000	0.0000
0.1467	0.1467	0.1467
0.3200	0.3200	0.3200
0.5200	0.5199	0.5199
0.7467	0.7467	0.7466
1.0000	0.9999	0.9999

case when $\alpha = \exp(-u)$.

In this example the Laplace transformed equation (4.19) is

$$\lambda \bar{x}^{(n)} - x\left(t_0\right) = e^{-u}\left(\left(\frac{\partial x}{\partial u}\right)^2\right)^{(n-1)} \frac{\partial^2 \bar{x}^{(n)}}{\partial u^2} + e^{-u}\left(\left(\frac{\partial x}{\partial u}\right)^{-1}\right)^{(n-1)}$$

Using a finite difference method, the algorithm is

$$\bar{X}_i^{(n+1)} = \frac{1}{\lambda + \frac{2e^{-u_i}}{h^2}\left(\left(\frac{\partial X}{\partial u}\right)^{-2}\right)^{(n)}}$$

$$\times \left\{ X\left(t_0\right) + e^{-u_i}\left(\left(\frac{\partial X}{\partial u}\right)^{-2}\right)^{(n)}\left(\frac{\bar{X}_{i+1}^{(n)} + \bar{X}_{i-1}^{(n)}}{h^2}\right) + e^{-u_i}\left(\left(\frac{\partial X}{\partial u}\right)^{-1}\right)^{(n)} \right\}$$

To start this problem, we use an accurate finite difference method, with $\delta x = 0.01$ and $\delta t = 0.000025$ and we choose a lower starting time of $t = 0.05$ because the isotherms move rapidly. To find the starting positions of the isotherms we plot x as a function of u at $t = 0.05$ but we cannot obtain a good polynomial trendline to fit the resulting curve. We therefore use a cubic spline interpolating function to estimate the positions of the required isotherms. This is a piecewise continuous curve passing through each of the temperature values found from the finite difference method, and knowing the positions for the temperatures either side of the required isotherm, we are able to deduce the position of the isotherm by calculating the coefficients of the cubic polynomial.

Again, we do not know the analytic solution to this problem, but we can find the positions of the isotherms accurately for the steady-state situation. We have

$$\frac{d}{dx}\left(e^{-u}\frac{du}{dx}\right) = 0$$

and we can deduce that

$$x = \frac{1 - e^{-u}}{1 - e^{-1}}$$

when the steady-state is reached.

Table 4.9: The positions of the isotherms at $t = 0.05, 0.15, 0.25$ and 1.45 when $\alpha = \exp(-u)$ in example 4.3

u	$t = 0.05$	$t = 0.15$	$t = 0.25$	$t = 1.45$
0.0	0.0000	0.0000	0.0000	0.0000
0.2	0.6908	0.4879	0.3792	0.2868
0.4	0.8186	0.7073	0.6274	0.5215
0.6	0.8987	0.8389	0.7920	0.7137
0.8	0.9549	0.9303	0.9096	0.8710
1.0	1.0000	1.0000	1 .0000	1.0000

Table 4.9 shows the positions of the isotherms at $t = 0.05, 0.15, 0.25$ and 1.45 and the movement of the isotherms with time is shown in figure 4.4. We also show in table 4.10 that the calculated positions for the steady state

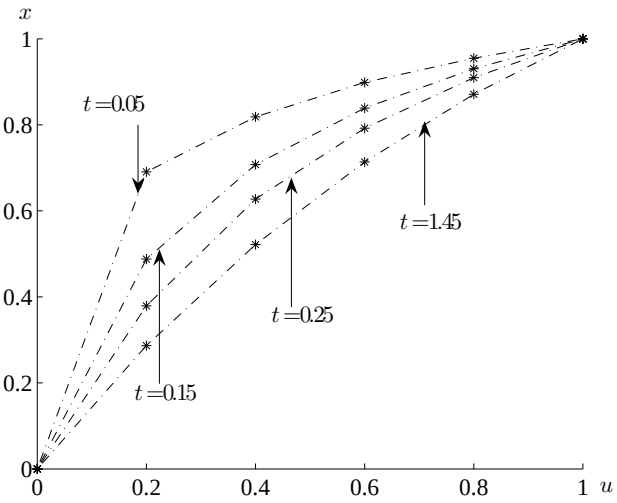

Figure 4.4: Position as a function of temperature, u, for times $t = 0.05, 0.15, 0.25$ and 1.45 when $\alpha = \exp(-u)$ in example 4.3

Table 4.10: The positions of the isotherms when the steady state is reached for $\alpha = \exp(-u)$ in example 4.3

u	Analytic	LTIMM	% error
0.0	0.0000	0.0000	0.00
0.2	0.2868	0.2868	0.01
0.4	0.5215	0.5215	0.00
0.6	0.7138	0.7137	0.01
0.8	0.8711	0.8711	0.01
1.0	1.0000	1.0000	0.00

isotherms are very close to the expected values, which confirms that this method is likely to be useful for solving non-linear problems.

We have noted that we need some numerical values to start the calculations in the Laplace transform isotherm migration method, and we have either used the analytic solution if one exists, or a numerical method which might involve finding a trendline or a cubic spline approximation. This means that it would be difficult to automate the method as the operator needs to use one of these methods to start a problem. Automation could be a future trend of research.

4.7 Summary of Chapter 4

In this chapter we have shown how to use the Laplace transform together with the isotherm migration method.

In example 4.1 we revisited the problem described in example 3.1 and compared the results obtained in example 3.1 with those obtained using the Laplace transform isotherm migration method.

Example 4.2 showed the use of the Laplace transform isotherm migration method to solve a problem which was non-linear, that is where $\alpha = 1 + u$, and we looked at the effect of using an average value for α to find the starting values and of introducing an error into the starting values. Another non-linear problem was used to illustrate the method in example 4.3, this time with a cubic spline interpolation to provide suitable starting values.

We concluded that the method appears to be tolerant of errors and gives results which compare favourably with those expected.

We have shown that we have a robust numerical method and when dealing with complex problems we can consider simpler problems to set up the required starting values.

4.7.1 Contribution

The combination of the Laplace transform and the isotherm migration method is a new idea. This new solution method was tested against the isotherm migration method model described in chapter 3 and was found to be of similar accuracy and tolerance to error. Further examples showed that its use was not restricted to linear problems.

Chapter 5

The Laplace transform isotherm migration method for one-dimensional problems with phase change

We have already discussed briefly, in chapter 2, the idea of problems involving a moving boundary on which there is a phase change, Stefan problems. We now return to such problems and describe how they may solved using the Laplace transform isotherm migration method.

We consider the melting of a block of ice by raising its surface to a temperature above 0. We have two phases, water and ice, which are separated by a boundary, on which melting occurs. The boundary moves across the domain as time progresses. Solving this system mathematically requires that the motion of the boundary is detemined and the usual heat flow equations are solved in the water and ice. The solutions and the boundary movement are dependent upon each other. Crank and Phahle (1973) describe a method to solve such a problem using the isotherm migration method using finite

differences, and we shall show how this problem can be solved using the Laplace transform together with a Gauss Seidel iterative method.

Following the problem described by Crank and Phahle (1973) we have a plane sheet of ice initially occupying the region $x_L \leqslant x \leqslant x_R$, where $x_L = 0$ and $x_R = a$, and being melted by the application of a constant temperature, u_0, on the surface $x = x_L$. At any time, t, the moving boundary separating the water from the ice is at $x = x_0(t)$. The region $x_L \leqslant x \leqslant x_0(t)$ consists of water with specific heat, density and thermal conductivity denoted by c, ρ and K respectively. The temperature of the water satisfies the heat flow equation

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2} \tag{5.1}$$

We take the ice to be initially at temperature 0 thoughout. At the melting boundary, $x_0(t)$, the heat flowing per unit area from the water into the ice in a short time, δt, is, using Fourier's law, $-(K\partial u/\partial x)\,\delta t$. If the boundary moves a distance δx_0 in time δt, the heat required to melt the mass $\rho \delta x_0$ of ice per unit area is $L\rho\delta x_0$ where L is the latent heat of fusion for ice. Equating these and taking the limit as $\delta t \to 0$, we see that the first condition to be satisfied on the moving boundary is the so-called Stefan condition

$$L\rho \frac{dx_0}{dt} = -K \frac{\partial u}{\partial x} \tag{5.2}$$

A second condition since ice melts at 0 is

$$u = 0, \qquad x = x_0, \qquad t \geqslant 0$$

This is shown diagrammatically in figure 5.1.

We make the problem dimensionless by using the variables

$$\tilde{x} = x/a,\ \tilde{t} = \alpha t/a^2,\ \tilde{x}_0 = x_0/a,\ \tilde{u} = u \text{ and } s = L/c$$

where s is the Stefan number which is a dimensionless constant. Details of how to calculate the Stefan number are to be found in Carslaw and Jaeger (1959).

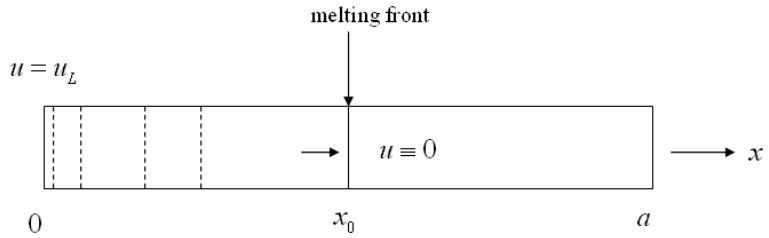

Figure 5.1: Diagram showing the problem described by Crank and Phale (1973)

We then have following the system of equations

$$\frac{\partial \tilde{u}}{\partial \tilde{t}} = \frac{\partial^2 \tilde{u}}{\partial \tilde{x}^2}, \quad 0 < \tilde{x} < \tilde{x}_0, \quad \tilde{t} \geqslant 0, \tag{5.3}$$

$$s\frac{d\tilde{x}_0}{d\tilde{t}} = -\frac{\partial \tilde{u}}{\partial \tilde{x}}, \quad \tilde{x} = \tilde{x}_0, \quad \tilde{t} \geqslant 0 \tag{5.4}$$

$$\tilde{u} = 0, \quad \tilde{x} = \tilde{x}_0, \quad \tilde{t} \geqslant 0$$

$$\tilde{u} = \tilde{u}_L, \quad \tilde{x} = 0, \quad \tilde{t} \geqslant 0$$

$$\tilde{u} = 0, \quad 0 < \tilde{x} < 1, \quad \tilde{t} = 0$$

In chapter 3 we give details of how to perform the isotherm migration mapping of equation (5.3) so that we have

$$\frac{\partial \tilde{x}}{\partial \tilde{t}} = \left(\frac{\partial \tilde{x}}{\partial \tilde{u}}\right)^{-2} \frac{\partial^2 \tilde{x}}{\partial \tilde{u}^2} \tag{5.5}$$

We also need to apply the mapping to equation (5.4) which becomes

$$s\frac{d\tilde{x}_0}{d\tilde{t}} = -\left(\frac{\partial \tilde{x}}{\partial \tilde{u}}\right)^{-1} \quad \tilde{u} = 0, \quad \tilde{t} \geqslant 0 \tag{5.6}$$

and the other boundary conditions which become

$$\tilde{x} = \tilde{x}_0, \quad \tilde{u} = 0, \quad \tilde{t} \geqslant 0$$

$$\tilde{x} = 0, \quad \tilde{u} = \tilde{u}_L, \quad \tilde{t} \geqslant 0$$

As in chapter 4 for clarity, we shall drop the use of the tilde for the non-dimensional variables and assume a dimensionless equation We now transform our system of equations into Laplace space. We describe how to do this for the mapped heat equation in chapter 4, but in addition, we now have to consider the melting boundary, which is a boundary condition involving derivatives, and which is non-linear. We use the method of direct iteration described in chapter 4 to make equation (5.6) linear. So we have

$$s\frac{dx_0{}^{(n)}}{dt} = -\left(\left(\frac{\partial x}{\partial u}\right)^{-1}\right)^{(n-1)} \qquad u = 0, \quad t \geqslant 0 \tag{5.7}$$

and we take the Laplace transform to obtain

$$s\left(\lambda\bar{x}_0^{(n)} - x_0^{(n)}(t_0)\right) = \frac{1}{\lambda}\left(\left(\frac{\partial x}{\partial u}\right)^{-1}\right)^{(n-1)}$$

Hence the algorithm to calculate the position of the melting front in Laplace space is

$$X_0{}^{(n)} = \frac{X_0{}^{(n)}(t_0)}{\lambda} + \frac{1}{s\lambda^2}\left(\left(\frac{\partial X}{\partial u}\right)^{-1}\right)^{(n-1)} \tag{5.8}$$

Equation (5.5) is rewritten under the Laplace transform as described in chapter 4 and together with equation (5.8) replacing the right hand boundary condition, is solved using a Gauss Seidel solver, as before. We now apply the method to the problem discussed by Crank and Phahle (1973), which they solved using an explicit finite difference method with the isotherm migration method.

Example 5.1

We consider a block of ice, of unit thickness and initially at zero temperature throughout. One face, x_L, is maintained at a temperature of 10. The analytic solution of this problem is given by Carslaw and Jaeger (1959). It is

$$\left.\begin{array}{l} u = u_0 - \frac{u_0}{\mathrm{erf}(\phi)}\mathrm{erf}\left(\frac{x}{2t^{\frac{1}{2}}}\right), 0 < x < x_0, t \geqslant 0 \\ \\ u = 0, \quad x_0 < x < 1, \quad t > 0 \end{array}\right\} \tag{5.9}$$

$$x_0 = 2\phi t^{\frac{1}{2}}, \tag{5.10}$$

where ϕ is given by

$$\pi^{\frac{1}{2}}\phi\mathrm{erf}(\phi)\exp\left(\phi^2\right) = u_0/s \tag{5.11}$$

and u_0 is the temperature at $x = 0$, that is at x_L. The analytic solution provides a means of checking the accuracy of our method as well as providing the numerical values at some small time, to start the solution to the problem.

To re-iterate the method described in chapter 4, we use a central difference approximation with $\bar{X}_i$ being the approximation to $\bar{x}_i$,

$$\lambda\bar{X}_i^{(n)} - X_i\left(t_0\right) = \left(\left(\frac{\partial X}{\partial u}\right)^{-2}\right)_i^{(n-1)} \frac{\bar{X}_{i+1}^{(n)} - 2\bar{X}_i^{(n)} + \bar{X}_{i-1}^{(n)}}{h^2}$$

where h is the difference in the value of successive temperatures in whose isotherms we are interested.

The algorithm to find the next position of the isotherm with temperature u_i is

$$\bar{X}_i^{(n+1)} = \frac{1}{\left(\lambda + \frac{2}{h^2}\left(\left(\frac{\partial X}{\partial u}\right)^{-2}\right)_i^{(n)}\right)}\left[X_i^{(n)}\left(t_0\right) + \left(\left(\frac{\partial X}{\partial u}\right)^{-2}\right)_i^{n}\left(\frac{\bar{X}_{i+1}^{(n)} - \bar{X}_{i-1}^{(n)}}{h^2}\right)\right] \tag{5.12}$$

We follow Crank and Phale (1973) and find the positions of isotherms at intervals of 2; however we take time steps of 0.1 because of the limitation in choice of step size required by the Stehfest inversion method described in chapter 4.

We use the Laplace transform in two diferent ways:
In the first method (LT1), we keep the same initial value of $x_0\left(t_0\right)$ throughout the calculation, which is the usual way of using the Laplace transform.

However, we find that doing this causes the position of the melting front to fall behind its true value and consequently the other isotherms are affected, and the error becomes larger as time progresses.

In the second method (LT2), we update the initial condition of the melting boundary at each step before finding the positions of the other isotherms. That is, we consider the previous position of the melting boundary to be the initial value at the next step. This greatly improves the accuracy of the solutions, which can been seen from tables (5.1) and (5.2) which show the results for the positions of the melting front and isotherm with temperature 8 respectively.

Table 5.1: The position of the melting front calculated using the Laplace transform without updating (LT1) and with updating (LT2)in example 5.1

Time	Analytic	LT1	% Error	LT2	% Error
0.5	0.361	0.324	10.22	0.356	1.39
1.0	0.510	0.434	14.81	0.505	0.98
1.5	0.625	0.518	17.15	0.620	0.80
2.0	0.722	0.588	18.61	0.717	0.69
2.5	0.807	0.649	19.59	0.803	0.50
3.0	0.884	0.704	20.34	0.879	0.57
3.5	0.955	0.755	20.95	0.950	0.52
3.8	0.995	0.784	21.24	0.990	0.50

Table 5.2: The position of the isotherm with temperature $u = 8$ calculated using the Laplace transform without updating (LT1) and with updating (LT2) in example 5.1

Time	Analytic	LT1	% Error	LT2	% Error
0.5	0.071	0.063	10.284	0.070	1.567
1.0	0.100	0.085	14.955	0.099	1.058
1.5	0.123	0.101	17.243	0.122	0.813
2.0	0.141	0.115	18.669	0.140	0.690
2.5	0.158	0.127	19.668	0.157	0.628
3.0	0.173	0.139	19.822	0.172	0.599
3.5	0.187	0.148	21.008	0.186	0.595
3.8	0.195	0.153	21.308	0.194	0.601

In figures 5.2 and 5.3 we show the graphs of the positions of the melting

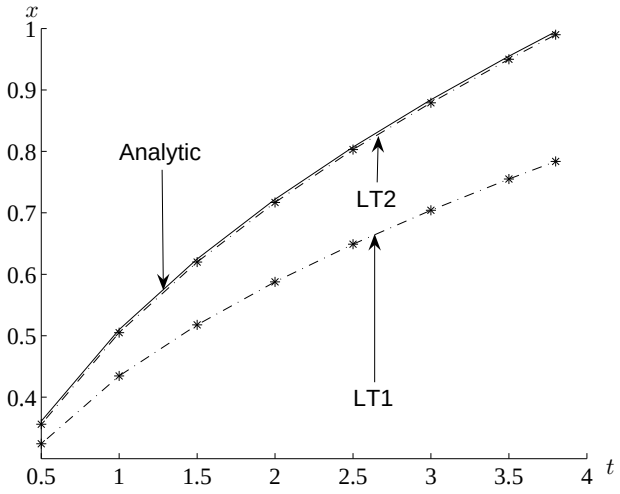

Figure 5.2: The position of the melting front calculated using the Laplace transform without updating (LT1) and with updating (LT2) in example 5.1

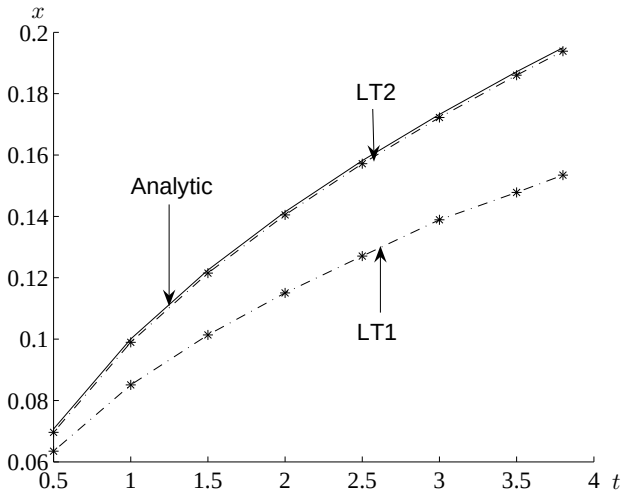

Figure 5.3: The position of isotherm with temperature $u = 8$ calculated using the Laplace transform without updating (LT1) and with updating (LT2)in example 5.1

front and isotherm 8 respectively using methods LT1 and LT2 and we also show the plot of the analytic solution for comparison. These figures confirm that to achieve meaningful results it is necessary to use method LT2.

We saw in chapter 4, that we could use the Laplace transform with the isotherm migration method in the usual way when the moving boundary was not one on which a phase change occurred.

The difference in this problem, is that we have a phase change, and moreover the boundary condition has a non-linear derivative term, and therefore we have to apply direct iteration to it before we can use the Laplace transform. The term

$$\left(\left(\frac{\partial x}{\partial u}\right)^{-1}\right)^{(n-1)}$$

in equation (5.24) is calculated from

$$\frac{h}{x_0 - x_1}$$

where h is the difference in temperatures of successive isotherms, and x_0 and x_1 are the positions of the freezing front and the isotherm next to it. This numerical quantity has to be calculated at the start of each iteration loop. We can see in equation (5.8) that in this case the initial condition now has an iterative subscript and so it is unsurprising that it will need updating.

On first sight we might then question whether there is any benefit in using the isotherm migration method with the Laplace transform, as in a sense, each time step is a new problem with a new initial condition and we need to perform several calculations when using the Stehfest inversion method. However we need to balance this, by remembering that when using the finite difference method, we have to apply the stability condition to the time step, and this means that in the early stages of the calculation, the time step is extremely small, again resulting in many calculations.

We therefore should perform an analysis on the efficiency of each method.

Considering the method of Crank and Phale (1973), the finite difference method solution of the isotherm migration method, we count the number of arithmetic operations at each time step, remembering that the time step is itself calculated, and we find the number of operations needed to proceed from a time of 0.1 to a time of 0.2. The time step requires three operations and the calculation of the new position of the freezing front requires five operations. The remaining positions of the remaining four isotherms each require nine operations, making a total of seventeen operations at each time step. To find the positions of the isotherms at a time of 0.2,which is the time step needed for the Laplace transform method, takes a total of 2329 arithmetic operations. The structure of the Laplace transform isotherm migration method can be sub-divided as an outer loop of direct iteration (which linearises the problem), a middle loop of Stehfest conversion to and inversion from Laplace space, and an inner loop which is the Gauss-Seidel solver. The Gauss-Seidel loop has eleven arithmetic operations to be processed on 4 inner isotherms, and takes forty-five passes to achieve convergence using double precision arithmetic. This loop then, accounts for 1980 operations. The Stehfest loop outside this, has 248 operations in setting up the eight parameters and inverting back and the Gauss-Seidel loop is within this, which means that at this stage we have now 16,088 operations.

To find the result at a time of 0.2, seventeen iterations of the direct iteration loop are needed for convergence, so the 16,088 operations from the previous step are carried out seventeen times, making 273,496 together with 612 operations to update the non-linear quantities. This means a final total of 274,108 arithmetic operations. The Laplace transform isotherm migration method therefore increases the number of calculations by a factor of approximately seventeen in this example.

However, this is a relatively simple example, which can be solved using the finite difference method. More complex examples may not be so

straightforward and might require other methods of solution. We are also looking towards speeding up the calculation using a parallel computing environment and we will be discussing how the Laplace transform method is ideally suited to this.

At this stage, we feel that although the Laplace transform isotherm migration method appears to be expensive in terms of numbers of calculations, it is nevertheless a useful method for solving Stefan problems and later in our work we shall show that the method is amenable to load sharing among multiple processors, and this will in itself represent a time saving in calculation.

5.1 A freezing problem solved using the Laplace transform isotherm migration method

Example 5.2

We consider a problem similar to that in Example 5.1. Whereas earlier we obtained the solutions when a plane sheet of ice was melted by applying a constant temperature at $x = 0$, we now find solutions when we have a region of water, initially at the melting temperature and by applying a constant temperature of u_L at $x = 0$, we track the freezing front and the movement of the isotherms in the solid phase.

The heat equation for this system is the same as equation (5.1) but equation (5.2) becomes

$$L\rho\frac{dx_0}{dt} = K\frac{\partial u}{\partial x} \tag{5.13}$$

that is, the sign is reversed, because heat is now flowing in the opposite direction. Therefore after applying the isotherm migration mapping we have

$$\frac{\partial x}{\partial t} = \left(\frac{\partial x}{\partial u}\right)^{-2}\frac{\partial^2 x}{\partial u^2} \tag{5.14}$$

$$s\frac{d\tilde{x}_0}{dt} = \left(\frac{\partial x}{\partial u}\right)^{-1} \quad u = 0, \quad t \geq 0 \tag{5.15}$$

$$x = x_0, \quad u = 0, \quad t \geqslant 0$$

$$x = 0, \quad u = u_L, \quad t \geqslant 0$$

We consider the case where we have a bounded region containng water at 0. The face at $x = 0$ is maintained at a temperature of -10, that is $u_L = -10$ so that the freezing front progresses forward from $x = 0$ and when it reaches $x = 1$ the region is completely frozen. We need to establish the correct value for s, the Stefan number for this problem, and this is calculated from the thermal properties of water and ice provided in Carslaw and Jaeger (1959). As in previous examples, we need a method to find the positions of some isotherms at a small time, $t = 1.0$.

An analytic solution for this case does exist and we follow the method in Carslaw and Jaeger (1959) to derive this. As well as providing the starting values for the problem, it also enables us to compare our results for accuracy. The analytic solution is based upon Neumann's solution for a region $x > 0$ initially liquid at constant temperature V with the surface at $x = 0$ maintained at zero for $t > 0$. The boundary conditions are $v_2 \to V$, as $x \to \infty$ and $v_1 = 0$ when $x = 0$, where v_2 and v_1 are the temperatures of the water and ice respectively.

To use this method for our case, we need to carry out a transformation so that we have

$$v_2' = v_2 + u_L$$

and

$$v_1' = v_1 + u_L$$

where u_L is the temperature at $x = 0$. We also remember that we have non-dimensionalised our problem by writing

$$\tilde{t} = \frac{\alpha t}{a^2}$$

but we will now drop the tilde as before and we find that the position of the

freezing front is given by

$$x_0 = 2\phi t^{\frac{1}{2}}$$

where ϕ is a numerical constant given by

$$\pi^{\frac{1}{2}}\phi \text{erf}(\phi) \exp\left(\phi^2\right) = -u_L/s$$

Having found ϕ we may then find the solutions

$$v_1' = u_L - \frac{u_L}{\text{erf}(\phi)} \text{erf}\left(\frac{x}{2t^{\frac{1}{2}}}\right), \quad 0 < x < x_0, \quad t \geqslant 0$$

and

$$v_2' = 0, \quad x_0 < x < 1, \quad t > 0$$

We now solve the problem with the Laplace transform isotherm migration method as described in example 5.1, in which we update the initial conditions after each time step. The algorithms are similar to those in the melting case, but care must be taken with signs as the direction of heat flow is opposite to that in the melting problem.

We compare the results obtained for the freezing front and isotherm with temperature -8 with those obtained from the analytic solution. Table 5.3 shows the percentange error in using the Laplace transform. We see that the results are acceptable, generally showing less than two percent error. In figure 5.4 we show the analytic solution plotted together with the solution from the Laplace transform isotherm migration method. This illustrates that the results using our method compare well with the analytic solution and provides further assurance that we may proceed with the Laplace transform isotherm migration method with some confidence.

Table 5.3: The positions of the freezing front and the isotherm with temperature $u = -8$ for example 5.2

	Freezing front			Isotherm with temperature -8		
Time	LTIMM	analytic	% error	LTIMM	analytic	% error
0.1	0.111	0.111	0.00	0.022	0.022	0.00
0.5	0.239	0.248	0.00	0.047	0.049	4.08
1.0	0.342	0.350	2.29	0.068	0.069	1.45
2.0	0.487	0.495	1.62	0.097	0.098	1.02
3.0	0.599	0.607	1.32	0.119	0.120	0.83
4.0	0.693	0.700	1.00	0.137	0.139	1.44
5.0	0.776	0.783	0.89	0.154	0.155	0.65
6.0	0.850	0.858	0.93	0.168	0.170	1.18
7.0	0.918	0.927	0.97	0.182	0.183	0.55
7.6	0.957	0.966	0.93	0.189	0.191	1.05
8.1	0.988	1.000	1.20	0.196	0.198	1.01

5.2 A Stefan problem with convective boundary conditions

We have seen in example 3.2 that situations may occur, where the number of isotherms cannot be decided beforehand and do not remain fixed until the end of computations. We use a method described by Gupta and Kumar (1988), which we adapt to include the Laplace transform and we show that this can be used to solve some problems where the number of isotherms varies during computation.

Gupta and Kumar consider an infinite cylinder in which a coolant is flowing with uniform temperature throughout. Outside the cylinder there is a liquid at its fusion temperature which freezes owing to the coolant. They describe this process in terms of radial co-ordinates which they map to a one-dimensional form. Because the condition on the boundary wall of the cylinder is not of a Dirichlet type, the temperature there will be a function of time, and so the number of isotherms will not remain the same as that chosen in the beginning. In the case they consider, as the liquid outside the cylinder freezes, new negative valued isotherms will be generated at the wall and will

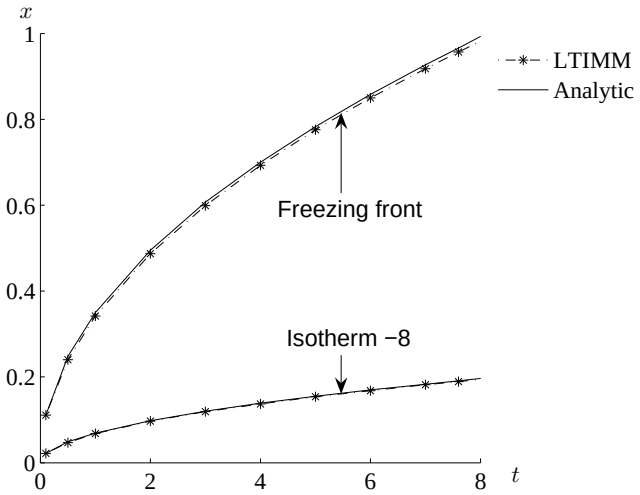

Figure 5.4: The position of the melting front and isotherm with temperature $u = -8$ with increasing time in example 5.2

move outwards with increasing time. Shih and Tsay (1971) solved the same problem using an analytic iteration method and Gupta and Kumar conclude that their method provides results which are not significantly different.

We describe the problem in two-dimensions in a general case. We consider a plane sheet of solid material at its fusion temperature, u_f, initially occupying the region $x_L \leqslant x \leqslant x_R$, where $x_L = 0$ and $x_R = a$, and with a wall in contact with the face x_L, outside which there is a steady temperature of u_v, which is greater than the fusion temperature. This means that heat will pass through the wall and melt the material, such that the melting front will progress from x_L to x_R.

The temperature at the wall at x_L at any time is given by u_w. The situation may be modelled by the following equations:

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}, \quad x_L < x < x_0 \tag{5.16}$$

$$K \frac{\partial u}{\partial x} = h_c \left(u_w - u_v \right), \quad x = x_L, \quad t > 0 \tag{5.17}$$

$$u = u_f, \quad x = x_0, \quad t \geqslant 0 \tag{5.18}$$

$$\rho L \frac{\partial x}{\partial t} = K \frac{\partial u}{\partial x}, \quad x = x_0, \quad t > 0 \tag{5.19}$$

$$u = u_f, \quad x_L < x < x_R, \quad t = 0 \tag{5.20}$$

where $u = u(x,t)$ denotes the temperature at any point x at any time t, h_c is the convective heat transfer coefficient, and x_0 is the position of the melting front. Equations (5.16) to (5.20)are converted to dimensionless form using the following changes of variable:

$$\tilde{u} = \frac{u - u_f}{u_f - u_v}$$

$$\tilde{t} = \frac{\alpha t}{a^2}$$

$$\tilde{u}_w = \frac{u_w - u_f}{u_f - u_v}$$

$$\tilde{x} = \frac{x}{a}$$

$$\tilde{x}_0 = \frac{x_0}{a}$$

$$s = \frac{L}{c(u_f - u_v)}$$

$$Bi = \frac{h_c l_c}{K}$$

The quantity Bi is known as the Biot number, depends on the thermal resistance of the material at the wall, where l_c is the characteristic length, and this depends on the physical characteristics of the material outside the wall. The system of equations is now

$$\frac{\partial \tilde{u}}{\partial \tilde{t}} = \frac{\partial^2 \tilde{u}}{\partial \tilde{x}^2}, \quad 0 < \tilde{x} < \tilde{x}_0, \quad \tilde{t} \geqslant 0 \tag{5.21}$$

$$\frac{\partial \tilde{u}}{\partial \tilde{x}} = Bi(\tilde{u}_w + 1), \quad \tilde{x} = 0, \quad \tilde{t} > 0 \tag{5.22}$$

$$\tilde{u}(\tilde{x},\tilde{t}) = 0, \quad \tilde{x} = \tilde{x}_0, \quad \tilde{t} \geqslant 0 \tag{5.23}$$

$$s\frac{d\tilde{x}_0}{d\tilde{t}} = -\frac{\partial \tilde{u}}{\partial \tilde{x}}, \quad \tilde{x} = \tilde{x}_0, \quad \tilde{t} \geqslant 0 \tag{5.24}$$

$$\tilde{u} = 0, \quad 0 < \tilde{x} < 1, \quad \tilde{t} = 0 \tag{5.25}$$

We now write equations (5.21), (5.23), (5.24) and(5.25) so that $\tilde{x}$ is expressed as a function of $\tilde{u}$ and $\tilde{t}$, so that we may use the isotherm migration method to solve the problem after performing the mapping process.

Equation (5.22) is not used directly in the isotherm migration method and so does not need to be changed. Following the usual process for the isotherm migration mapping we see that

$$\frac{\partial \tilde{x}}{\partial \tilde{t}} = \left(\frac{\partial \tilde{x}}{\partial \tilde{u}}\right)^{-2} \frac{\partial^2 \tilde{x}}{\partial \tilde{u}^2}, \quad \tilde{u}_w(\tilde{t}) < \tilde{u} < 0, \quad \tilde{t} > 0 \tag{5.26}$$

where $\tilde{u}_w(\tilde{t})$ is the temperature at $\tilde{x} = 0$ at any time $\tilde{t}$, and

$$s\frac{d\tilde{x}_0}{d\tilde{t}} = -\left(\frac{\partial \tilde{x}}{\partial \tilde{u}}\right)^{-1} \quad \tilde{u} = 0, \quad \tilde{t} \geqslant 0 \tag{5.27}$$

in the present case.

We now consider how to manage the additional isotherms which will be generated due to convection across the wall at $\tilde{x} = 0$. Suppose that at any time $\tilde{t} = \tilde{t}_n = n\delta\tilde{t}$, where $\delta\tilde{t}$ is the time step, there is a total of $(i+1)$ isotherms in the region $\left[\tilde{u}_w(\tilde{t}), 0\right]$. This implies that $\tilde{u}_w^{(n)} > \tilde{u}_i$, where $\tilde{u}_w^{(n)} = \tilde{u}_w(\tilde{t}_n)$. The temperature of an isotherm is determined as $\tilde{u}_i = \tilde{u}_0 - i\delta\tilde{u}, i = 0, 1, 2...(\tilde{u}_0 = 0)$, where $\delta\tilde{u}$ is the temperature step. If we denote the position of an isotherm $\tilde{u}_i$ at a time $\tilde{t}_n$ by $X_i^{(n)}$ then we may compute the approximate positions, the X values at time level $(n+1)$ for the isotherms $\tilde{u}_i, i = 0, 1, ...(i-1)$ using a finite difference method with the Laplace transform. The last isotherm, $\tilde{u}_i$, is treated differently. We use a

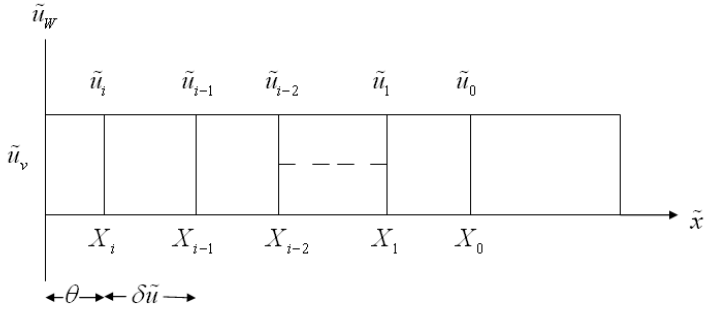

Figure 5.5: Diagram showing the approximate positions of the isotherms in the case when there is convection across the wall at $\tilde{x} = 0$

finite difference approximation with unequal intervals, as shown in figure 5.5 and we have, writing $\theta = \tilde{u}_i - \tilde{u}_w^{(n)}$,

$$\frac{\partial X}{\partial \tilde{u}} = \frac{\theta^2 X_{i-1} - (\theta^2 - \delta\tilde{u}^2)X_i}{\delta\tilde{u}\theta(\theta + \delta\tilde{u})} \tag{5.28}$$

and

$$\frac{\partial^2 X}{\partial \tilde{u}^2} = \frac{2\left[\theta X_{i-1} - (\theta + \delta\tilde{u})\, X_i\right]}{\delta\tilde{u}\theta(\theta + \delta\tilde{u})} \tag{5.29}$$

and these can be used in equation (5.26) directly. We now have the position of the isotherm with temperature $\tilde{u}_i$, and we use this to see whether any new isotherms have appeared. Following the suggestion of Gupta and Kumar (1988), we assume that the behaviour of the temperature is smooth in the region near the fixed boundary, and we fit a quadratic curve, which passes through the X values at the last two isotherms at temperatures $\tilde{u}_i$ and $\tilde{u}_{i-1}$, which can be written as

$$\tilde{u} = aX^2 + bX + c \tag{5.30}$$

where the coefficients a, b and c are to be determined. When $X = 0$ then $c = \tilde{u}_w$ and from equation (5.22)

$$Bi(\tilde{u}_w + 1) = 2aX + b$$

so that when $X = 0$

$$b = Bi(\tilde{u}_w + 1)$$

Eliminating $\tilde{u}_w$ gives

$$c = \frac{b}{Bi} - 1$$

which we substitute into equation (5.30) and we then have just a and b unknown. We find that

$$a = \frac{(BiX_i + 1)(1 + \tilde{u}_{i-1}) - (1 + \tilde{u}_i)(BiX_{i-1} + 1)}{(BiX_i + 1)X_{i-1}^2 - X_i^2(BiX_{i-1} + 1)}$$

$$b = \frac{BiX_i^2(1 + \tilde{u}_{i-1}) - (1 + \tilde{u}_i)BiX_{i-1}^2}{(BiX_{i-1} + 1)X_i^2 - X_{i-1}^2(BiX_i + 1)}$$

c can then be found using b and hence the temperature at the wall, $\tilde{u}_w$ is easily determined.

Before proceeding to the next time level we examine whether the new temperature at the fixed boundary exceeds the temperature of the nearest isotherm by an amount $\delta\tilde{u}$ or not. In the case $\tilde{u}_w^{(n+1)} > \tilde{u}_i + \delta\tilde{u}$, we look for the largest integer N_I which satisfies

$$\tilde{u}_w^{(n+1)} > \tilde{u}_i + N_I\delta\tilde{u}$$

We introduce N_I additional isotherms so that the number of isotherms at $\tilde{t} = \tilde{t}_{n+1}$ becomes $(i + N_I + 1)$ and at time level $n + 1$, we compute the movement of $(j + N_I + 1)$ isotherms. The X co-ordinate of these additional isotherms can be fixed from equation (5.30) by substituting the values for $\tilde{u}$, a, b and c and finding the roots of the resulting quadratic equation in X.

Example 5.3

We solve the problem described in example 4.1, but with a convective boundary condition at $\tilde{x} = 0$ and with $u_v = 10$. We do not have an analytic solution to this problem and so we look at expected trends to evaluate the method. We also need some values to start the problem and we begin by assuming that the wall temperature, $\tilde{u}_w$, has a value of 5 initially and the positions

of the isotherms are calculated as if the wall temperature is fixed. We have seen in previous work that the isotherm migration method is fairly tolerant of errors in the initial conditions, so that starting the calculation in this way should be acceptable.

We need a value for Bi, the Biot number, the ratio of the heat transfer coefficient, h_c, to the thermal conductivity, K. This depends on the physical properties of the materials used, the thickness of the wall, and the material outside the wall. A large Biot number implies heat transfer is faster on the surface than inside the material. Using typical values for water from Carslaw and Jaeger (1959) we use a value of 5.0 for the Biot number in this example.

Table 5.4: The evolution of the isotherms in Example 5.3

	isotherm								
time	0	1	2	3	4	5	6	7	8
0.1	0.161	0.128	0.096	0.063	0.036	0			
0.3	0.225	0.180	0.134	0.089	0.045	0.003			
0.6	0.310	0.258	0.205	0.154	0.102	0.051	0.001		
1.4	0.490	0.419	0.348	0.277	0.208	0.138	0.070	0.001	
3.5	0.822	0.717	0.612	0.508	0.406	0.304	0.202	0.101	0.000
3.8	0.863	0.763	0.665	0.568	0.472	0.377	0.282	0.188	0.094
4.9	1.000	0.885	0.772	0.660	0.548	0.438	0.328	0.218	0.109

In table 5.4, we show how the isotherms are generated with increasing time. In this example they appear slowly, and we do not find a new one with every time step. The ice bar takes a time of 4.9 to be completely melted and at this point the wall has reached a temperature of 8.7.

Figure 5.6 shows a comparison of how the melting front progresses in the case when the wall is held at a steady temperature and when there is convection of heat across the wall. We see that when the temperature is steady, the ice takes a time of 3.8 to melt, whereas where there is convection a time of 4.9 is needed. This is what we would expect from the physical

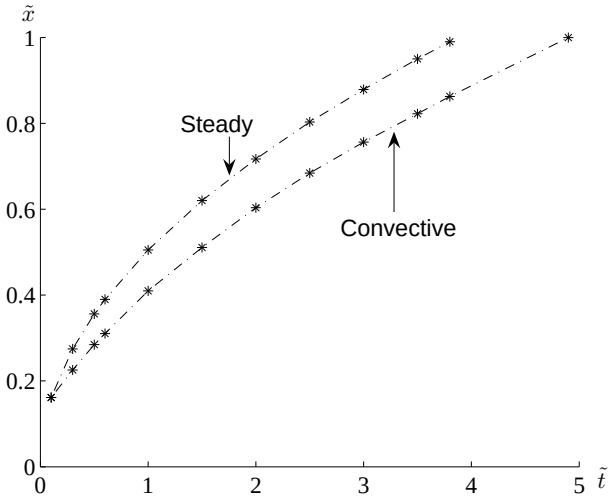

Figure 5.6: Comparison of the position of the melting front for the cases when the wall temperature is held steady and when there is convection across the wall, $Bi = 5$, in example 5.3

situation.

Example 5.4

We consider the same problem as in example 5.3, but in this case $Bi = 50$. This means that the heat transfer is increased and so we expect heating to be more rapid and that the new isotherms will appear more quickly and the melting front will propagate more quickly.

We see that when the conductivity is increased, the isotherms are generated much more quickly, with three appearing in the first time step. This is shown in table 5.5. When the time reaches 0.4, we already have all the isotherms and when the ice bar is completely melted, the temperature at the wall is 9.8, so that the case quickly becomes similar to the case when the wall has a steady temperature. We would therefore expect the movement of the melting front to be similar to that in the steady temperature case and we see this in figure 5.7, where the two paths almost coincide, and the bar is completely melted in a time of 3.9, compared with 3.8 in the steady

Table 5.5: The evolution of the isotherms in Example 5.4

	isotherm									
time	0	1	2	3	4	5	6	7	8	9
0.1	0.161	0.128	0.096	0.063	0.036					
0.2	0.196	0.156	0.117	0.078	0.039	0.025	0.014	0.004		
0.3	0.240	0.209	0.178	0.148	0.118	0.089	0.059	0.029	0.010	
0.4	0.282	0.249	0.217	0.186	0.154	0.123	0.092	0.061	0.031	0.004
1.0	0.479	0.430	0.380	0.332	0.284	0.236	0.188	0.141	0.094	0.047
1.5	0.598	0.536	0.475	0.414	0.354	0.294	0.235	0.176	0.117	0.059
2.0	0.697	0.625	0.553	0.483	0.412	0.343	0.274	0.205	0.137	0.068
2.5	0.784	0.703	0.622	0.543	0.464	0.386	0.308	0.231	0.154	0.077
3.0	0.862	0.773	0.684	0.597	0.510	0.424	0.339	0.254	0.169	0.084
3.5	0.934	0.837	0.741	0.647	0.553	0.459	0.367	0.275	0.183	0.091
3.9	0.988	0.885	0.784	0.684	0.584	0.486	0.388	0.291	0.193	0.097

situation.

We conclude that this method for dealing with isotherms which are generated or disappear during a melting or freezing problem appears to give satisfactory results for cases where we have a convective condition on the boundary. It is simple to operate and works well with the Laplace transform isotherm method.

5.3 Summary of Chapter 5.

We have introduced the idea of a moving boundary problem where there is a phase change involved across the moving boundary.

We have shown how to perform the isotherm mapping on the set of equations which describe this system and how the Laplace transform may be applied to the mapped system.

In example 5.1 we described a melting problem and solved it using the Laplace transform in the usual way (LT1) and we showed that this gave unsatisfactory numerical results and we then described a different way of using the Laplace transform (LT2), in which the initial values are updated

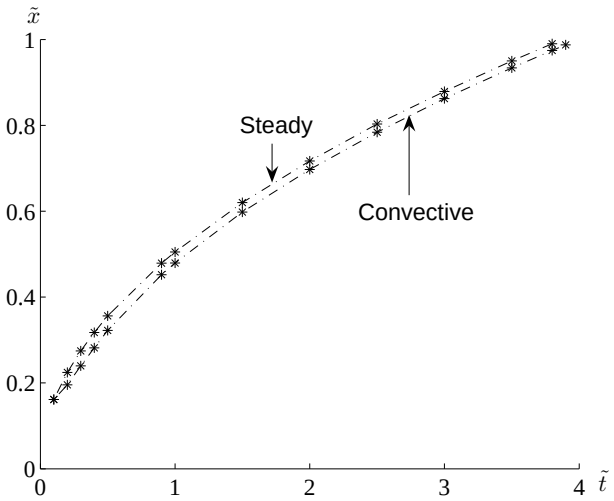

Figure 5.7: Comparison of the position of the melting front for the cases when the wall temperature is held steady and when there is convection across the wall, $Bi = 50$, in example 5.4

at each time step and we showed that this gave acceptable results.We gave some explanation as to why (LT2) produces better results.

We described the reverse problem, a freezing problem and have used the Laplace transform isotherm migration method to solve it in example 5.2.

We conclude that the Laplace transform migration method is a useful method for solving Stefan problems in one dimension.

Examples 5.3 and 5.4 described a modification to the method which allows us to deal with problems having a convective boundary condition where isotherms may either be generated or disappear. We concluded that this method was suitable for dealing with these cases, was simple to operate and could be easily used with the Laplace transform isotherm migration method.

5.3.1 Contribution

The Laplace transform isotherm migration method was tested to assess its usefulness in solving problems involving phase change, including examples

with a convective boundary condition. Results were compared with those obtained from the isotherm migration method and we concluded that the method provides an acceptable alternative for solving phase change problems in one dimension.

The use of LT2, in which we update the intial values at each time step, is a novel modification to the usual Laplace transform method.

Chapter 6

The Laplace transform isotherm migration method for two-dimensional problems with phase change

In this work, we extend the one-dimensional freezing problem discussed in example 5.2 in chapter 5, to a two-dimensional problem, using a method described by Crank and Gupta (1975). For clarity, in our work in two dimensions, we shall take the variables x, y, t to be the dimensionless form of the variables, rather than using a tilde superscript. In the first instance, we keep the problem deliberately simple, because we want to examine the limitations of the method and see whether it might be suitable to use with the Laplace transform, to solve more complex problems.

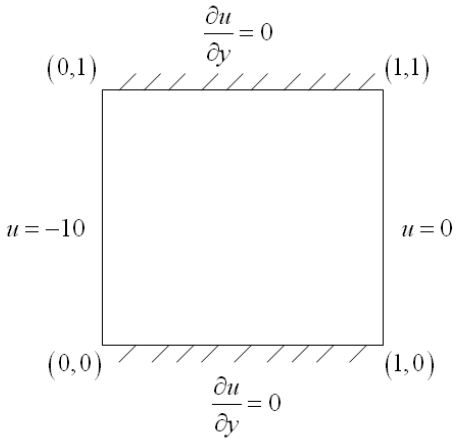

Figure 6.1: Diagram to show a square region of water, insulated on two parallel sides, in example 6.1

6.1 The freezing problem in two-dimensions.

Example 6.1

We previously considered a problem in which we had a region of water at temperature 0 extending from $x = 0$ to $x = 1$ and we applied a constant temperature to the boundary at $x = 0$ and evaluated the time for the freezing front to cross the region and reach the boundary $x = 1$. We also noted the movement of the isotherms, and were able to assess the accuracy of the methods we used, the finite difference method and the Laplace transform with a Gauss-Seidel solver, a standard method for solving a tri-diagonal system of equations.

We now consider the case in which we have a square region of water, which is insulated on two of its parallel boundaries. This is shown in figure 6.1. Because of symmetry, lines perpendicular to the x-axis will have the same temperature along their length, and so these are the isotherms. The freezing front, which is perpendicular to the x-axis, will propagate from $x = 0$ to $x = 1$. The boundary and initial conditions are

$$\begin{array}{llll} \frac{\partial u}{\partial y} = 0 & y = 0 & 0 \leqslant x \leqslant 1 & t \geqslant 0 \\ \frac{\partial u}{\partial y} = 0 & y = 1 & 0 \leqslant x \leqslant 1 & t \geqslant 0 \\ u = -10 & x = 0 & 0 \leqslant y \leqslant 1 & t \geqslant 0 \\ u = 0 & 0 < x < 1 & 0 < y < 1 & t = 0 \end{array}$$

Because the position of an isotherm varies with respect to the x co-ordinate, and the temperature with respect to the y co-ordinate is constant, this two-dimensional problem is essentially a one-dimensional problem.

6.2 The mapping of the equations in two dimensions

Although previously we have used $(\tilde{x}, \tilde{y})$ to represent the non-dimensional form of the cartesian co-ordinates, for clarity, we shall drop the tilde and when using (x, y), we mean the non-dimensional form of the variables. The usual heat conduction equation in two dimensions using non-dimensional space and time co-ordinates is

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \tag{6.1}$$

As the temperature is constant along an isotherm we have

$$du = \left(\frac{\partial u}{\partial x}\right)_{y,t} dx + \left(\frac{\partial u}{\partial t}\right)_{x,y} dt = 0$$

so that

$$\left(\frac{\partial x}{\partial t}\right)_{u,y} = -\left(\frac{\partial u}{\partial t}\right)_{x,y} \Bigg/ \left(\frac{\partial u}{\partial x}\right)_{y,t} = -\left(\frac{\partial u}{\partial t}\right)_{x,y} \left(\frac{\partial x}{\partial u}\right)_{y,t} \tag{6.2}$$

We substitute equation (6.2) into equation(6.1) and drop the suffices to get

$$\frac{\partial x}{\partial t} = -\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)\left(\frac{\partial x}{\partial u}\right) \tag{6.3}$$

We know that

$$\frac{\partial^2 u}{\partial x^2} = \frac{\partial}{\partial x}\left(\frac{\partial x}{\partial u}\right)^{-1} = -\frac{\partial^2 x}{\partial u^2}\left(\frac{\partial x}{\partial u}\right)^{-3}$$

which when substituted into equation (6.3) gives us

$$\frac{\partial x}{\partial t} = -\left\{\frac{\partial^2 u}{\partial y^2} - \frac{\partial^2 x}{\partial u^2}\left(\frac{\partial x}{\partial u}\right)^{-3}\right\}\left(\frac{\partial x}{\partial u}\right) \tag{6.4}$$

In this way we have expressed x as a function of u, y and t.

We now choose a u-y grid such that $u_i = u_0 + i\delta u, \quad i = 1, 2, ..., N$ and $y_j = y_0 + j\delta x, \quad j = 1, 2, ..., M$. The net rate at which heat becomes available at the interface is given by

$$K_{sol}\frac{\partial u_{sol}}{\partial n} - K_{liq}\frac{\partial u_{liq}}{\partial n}$$

where u_{liq} and u_{sol} are the temperatures in the liquid and solid phases respectively and K_{liq} and K_{sol} are the corresponding thermal conductivities. When the interface moves a distance dx, a quantity of heat $L\rho dx$ per unit area is liberated and must be removed by conduction. For heat balance this requires

$$K_{sol}\frac{\partial u_{sol}}{\partial n} - K_{liq}\frac{\partial u_{liq}}{\partial n} = L\rho\frac{dx}{dt} \tag{6.5}$$

where L is the latent heat of fusion and ρ is the density.

Crank and Gupta (1975) say that Patel (1968) showed that equation (6.5) can be written in a more convenient form, the revised equation being

$$\frac{\partial x}{\partial t} = \frac{1}{s}\left\{1 + \left(\frac{\partial x}{\partial y}\right)^2\right\}\left\{K_{sol}\left(\frac{\partial x}{\partial u_{sol}}\right)^{-1} - K_{liq}\left(\frac{\partial x}{\partial u_{liq}}\right)^{-1}\right\} \tag{6.6}$$

where s is the Stefan number. He did this by using a function, $f(x, y, t) = 0$, to describe the solid/liquid interface and evaluating $\frac{\partial x}{\partial t}$ for points on this interface. On the interface, equation (6.6) replaces equation (6.4) which holds at all other points.

We are able to use the shorter form of equation (6.6)

$$\frac{\partial x}{\partial t} = \frac{1}{s}\left\{1 + \left(\frac{\partial x}{\partial y}\right)^2\right\}\left(\frac{\partial x}{\partial u_{sol}}\right)^{-1} K_{sol} \tag{6.7}$$

because the liquid phase is always at the uniform temperature $u = 0$ and consequently there is no temperature gradient in the liquid phase. For convenience, we now drop the subscript and write u rather than u_{sol}.

6.3 The finite difference form

We evaluate the numerical solution on a u, y-grid, choosing δu and δy such that

$$u_i = u_0 + i\delta u, \quad i = 1, 2, ..., N \quad (u_0 = -10, \quad u_N = 0)$$

where $\delta u = 2$ and

$$y_j = y_0 + j\delta y, \quad j = 1, 2, ..., M \quad (y_0 = 0, \quad y_M = 1)$$

where $\delta y = 2$. We need to calculate the approximate values $X_{i,j}$, of the positions of the isotherms $x_{i,j}$, on this grid for successive values of δt.

We calculate the new position of the freezing front first, this being the isotherm with temperature u_N. We represent equation (6.7) by the finite difference form

$$\frac{X_{N,j}^{(n+1)} - X_{N,j}^{(n)}}{\delta t} = \frac{1}{s}\left\{1 + \left(\frac{X_{N,j}^{(n)} - X_{N,j-1}^{(n)}}{\delta y}\right)^2\right\}\frac{\delta u}{X_{N,j}^{(n)} - X_{N-1,j}^{(n)}} \tag{6.8}$$

which gives us $x_{N,j}^{(n+1)}$ for $j = 1, 2, ..., M$. We use backward difference for $\left(\frac{\partial x}{\partial u}\right)^{-1}$ because at the interface we need to refer back to the previous isotherm as there are no forward isotherms in the liquid phase so that a central difference is not appropriate here.

The positions of the remaining isotherms are found from equation (6.4) which may be simplified to

$$\frac{\partial x}{\partial t} = -\left(\frac{\partial x}{\partial u}\right)\frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 x}{\partial u^2}\left(\frac{\partial x}{\partial u}\right)^{-2} \tag{6.9}$$

and in finite difference form is represented by

$$\frac{X_{i,j}^{(n+1)} - X_{i,j}^{(n)}}{\delta t} = -\left(\frac{X_{i,j}^{(n)} - X_{i-1,j}^{(n)}}{\delta u}\right)\frac{\partial^2 U}{\partial y^2} + \frac{X_{i-1,j}^{(n)} - 2X_{i,j}^{(n)} + X_{i+1,j}^{(n)}}{\left(X_{i,j}^{(n)} - X_{i-1,j}^{(n)}\right)^2}$$

To remain consistent we use a backward difference for the term $\frac{\partial x}{\partial u}$.

In addition, we have a term $\frac{\partial^2 u}{\partial x^2}$ and we deal with this by interpolating or extrapolating linearly the values of u corresponding to $x_{i,j}$ at $y_{j-1}$and y_{j+1}. The formulae we use are for u at y_{j-1}

$$U = \frac{U_{i+1}\left(X_{i,j-1} - X_{i,j}\right) - U_i\left(X_{i+1,j-1} - X_{i,j}\right)}{X_{i,j-1} - X_{i+1,j-1}}$$

and for u at y_{j+1}

$$U = \frac{U_{i-1}\left(X_{i,j+1} - X_{i,j}\right) - U_i\left(X_{i-1,j+1} - X_{i,j}\right)}{X_{i,j+1} - X_{i-1,j+1}}$$

and we can then apply a central difference formula to find a value for $\frac{\partial^2 U}{\partial y^2}$ at the grid point we are interested in. This applies to the general case, but in this instance $\frac{\partial u}{\partial y} = 0$ and therefore $\frac{\partial^2 U}{\partial y^2} = 0$ and so we do not have to include this step.

In the general case, $\frac{\partial u}{\partial x}$ is undefined on $y = 0$ and $y = 1$, and we need another method to find the values on these boundaries. We overcome this difficulty by considering the problem in the x, y-plane. We use the fact that the flux is zero on these boundaries and use the two neighbouring points on the isotherm to fit a quadratic equation which cuts the boundaries. In practice, for the case we are considering, the fitted curve is linear, since the isotherms are perpendicular to the x-axis.

As before, we need some values to start the problem, and we use the analytic solutions found at $t = 0.1$ for the one-dimensional problem, remembering that the isotherms are parallel to the y-axis, so for a particular isotherm, the x co-ordinate is the same for all values of y. Furthermore, we need to decide on a time-step and we recall that in the one-dimensional case we used the condition

$$\delta t < \frac{\left(X_{i-1} - X_{i+1}\right)^2}{8}$$

The results of a variety of tests suggest that for this problem, to avoid instability, we need to set the step-size as

$$\delta t = \frac{\left(X_{i-1} - X_{i+1}\right)^2}{40}$$

Results

We see from table 6.1 that the method gives results very close to the analytic solutions, with errors less than one percent, and for the isotherm with temperature -8, the absolute values are so small, that the errors appear to be magnified although in fact they agree to 10^{-3}. This is confirmed by figure 6.2 which shows that the calculated values are extremely close to the analytic solutions. This suggests that the method is suitable for solving simple two-dimensional problems.

Furthermore, the example may be solved in a symmetrical situation, where the boundaries on $x = 0$ and $x = 1$ are insulated and the boundary $y = 0$ is held at -10, and this gives the same results; the freezing front reaches $y = 1$ in a time of 8.1 and the intermediate values for all the isotherms are essentially the same as before.

Table 6.1: The positions of the freezing front and isotherm with temperature $u = -8$ in example 6.1

	Freezing front			Isotherm -8		
Time	IMM	Analytic	% Error	IMM	Analytic	% Error
0.1	0.111	0.111	0.00	0.022	0.022	0.00
0.5	0.248	0.248	0.00	0.049	0.049	0.00
1.0	0.351	0.350	0.29	0.070	0.069	0.81
2.0	0.497	0.495	0.40	0.098	0.098	0.00
3.0	0.609	0.607	0.33	0.121	0.120	0.83
4.0	0.703	0.700	0.43	0.139	0.139	0.00
5.0	0.786	0.783	0.38	0.156	0.155	0.65
6.0	0.861	0.858	0.35	0.170	0.170	0.00
7.0	0.930	0.927	0.32	0.184	0.183	0.55
7.6	0.968	0.966	0.26	0.192	0.191	0.42
8.1	1.000	1.000	0.00	0.198	0.197	0.51

6.4 The solution using the Laplace transform

Example 6.2

We now solve the problem using the Laplace transform, and we use Ste-

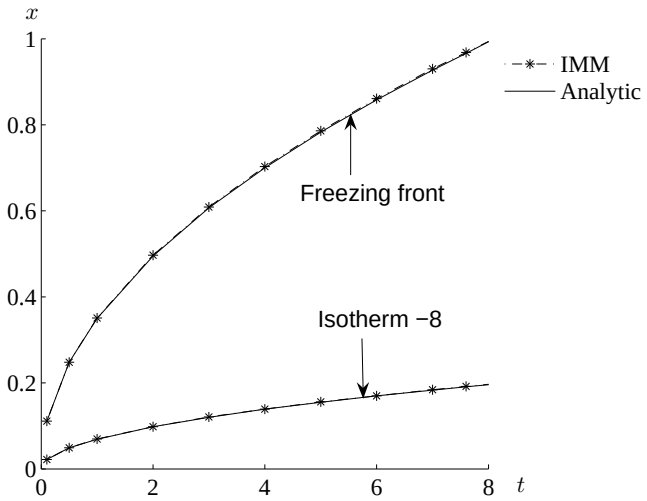

Figure 6.2: The position of the melting front and isotherm with temperature $u = -8$ with increasing time in example 6.1

hfest's numerical method to map to the Laplace space and invert, together with a Gauss-Seidel solver. Since both equations (6.7) and (6.9) are time dependent, we first need to find the corresponding expressions in Laplace space.

Equation (6.7) is non-linear and so we use direct iteration as before to make it linear before applying the Laplace transform.

Therefore

$$\left(\frac{\partial x}{\partial t}\right)^{(n)} = \frac{1}{s}\left\{1 + \left(\left(\frac{\partial x}{\partial y}\right)^2\right)^{(n-1)}\right\}\left(\left(\frac{\partial x}{\partial y}\right)^{-1}\right)^{(n-1)} \tag{6.10}$$

where $\left(\left(\frac{\partial x}{\partial y}\right)^2\right)^{(n-1)}$ and $\left(\left(\frac{\partial x}{\partial y}\right)^{-1}\right)^{(n-1)}$ are the numerical values calculated at the previous pass. In this way, the right-hand side of equation (6.10) is a numerical constant.

Under the Laplace transform this becomes

$$\lambda\bar{x}^{(n)}(\lambda) - {x_0}^{(n)}(t_0) = \frac{1}{s\lambda}\left\{1 + \left(\left(\frac{\partial x}{\partial y}\right)^2\right)^{(n-1)}\right\}\left(\left(\frac{\partial x}{\partial y}\right)^{-1}\right)^{(n-1)}$$

where t_0 is the initial time at which we start the problem. That is

$$\bar{x}^{(n)}(\lambda) = \frac{x_0^{(n)}(t_0)}{\lambda} + \frac{1}{s\lambda^2}\left\{1 + \left(\left(\frac{\partial x}{\partial y}\right)^2\right)^{(n-1)}\right\}\left(\left(\frac{\partial x}{\partial y}\right)^{-1}\right)^{(n-1)} \tag{6.11}$$

In a similar manner equation (6.9), under the Laplace transform becomes

$$\bar{x}^{(n)}(\lambda) - x_0^{(n)}(t_0) = -\frac{1}{\lambda}\left(\frac{\partial x}{\partial u}\right)^{(n-1)}\left(\frac{\partial^2 u}{\partial y^2}\right)^{(n-1)} + \frac{\partial^2 \bar{x}}{\partial u^2}\left(\left(\frac{\partial x}{\partial u}\right)^{-2}\right)^{(n-1)} \tag{6.12}$$

We write $\frac{\partial^2 \bar{x}}{\partial u^2}$ as

$$\frac{\bar{X}_{i+1} - 2\bar{X}_i + \bar{X}_{i-1}}{(\delta u)^2}$$

where $\bar{X}_i$ is the approximation to $\bar{x}_i$, the position of the isotherm with temperature u_i.

Then equation (6.12) may be expressed as

$$\bar{X}_i^{(n)} = \left[\frac{1}{\lambda + \frac{2}{\partial u^2}\left(\left(\frac{\partial x}{\partial u}\right)^{-2}\right)^{(n-1)}}\right]$$

$$\times\left\{x_0 - \frac{1}{\lambda}\left(\frac{\partial x}{\partial u}\right)^{(n-1)}\left(\frac{\partial^2 u}{\partial y^2}\right)^{(n-1)} + \left(\frac{\bar{X}_{i+1} + \bar{X}_{i-1}}{\partial u^2}\right)\left(\left(\frac{\partial x}{\partial u}\right)^{-2}\right)^{(n-1)}\right\}$$

We need to find the numerical values of the terms $\left(\frac{\partial x}{\partial u}\right)^{(n-1)}$,$\left(\frac{\partial^2 u}{\partial y^2}\right)^{(n-1)}$ and $\left(\left(\frac{\partial x}{\partial u}\right)^{-2}\right)^{(n-1)}$ before entering the Stehfest loop.

To calculate the second of these terms, we need to map back into x,y-space, and perform an interpolation.

We use the same formulae for interpolation as in example 6.1.

Results

We show the results in table 6.2 and figure 6.3. We see that using the Laplace transform gives solutions very close to those obtained using both the analytic and finite difference methods. This is useful, because we noted that in using the finite difference method, we need to take very small time steps because of problems with stability, and in using the Laplace transform

we do not have this constraint and can progress in larger steps. However, when using the Stehfest numerical method, it must be remembered that several calculations are involved in converting to Laplace space and then inverting, and we discussed in chapter 5 how using the Laplace transform with Stehfest inversion increased the number of calculations by a factor of 17 approximately, in the one-dimensional case. In this case, a counter inserted into the code shows that the isotherm migration method with the finite difference method requires 1,827,402 operations compared with 6,765,658 operations for the Laplace transform isotherm migration method. Although the number of operations in the second method is greater by a factor of approximately 3.7, this is significantly lower than in the one-dimensional case. This is likely to be due to the need for a much smaller time step needed for stability in the finite difference method, than was needed in the one-dimensional case. We hope that we shall be able to demonstrate that we can take advantage of a parallel environment to speed up the calculation with the Laplace transform isotherm migration method.

Table 6.2: The positions of the freezing front and isotherm with temperature $u = -8$ in example 6.2

	Freezing front			Isotherm -8		
Time	FDM	LTIMM	Analytic	FDM	LTIMM	Analytic
0.1	0.111	0.111	0.111	0.022	0.022	0.022
0.5	0.248	0.240	0.248	0.049	0.047	0.049
1	0.351	0.342	0.350	0.070	0.068	0.069
2	0.497	0.488	0.495	0.098	0.097	0.098
3	0.609	0.599	0.607	0.121	0.119	0.120
4	0.703	0.693	0.700	0.139	0.137	0.139
5	0.786	0.776	0.783	0.156	0.154	0.155
6	0.861	0.850	0.858	0.170	0.168	0.170
7	0.930	0.918	0.927	0.184	0.182	0.183
7.6	0.968	0.957	0.966	0.192	0.190	0.191
8.1	1.000	0.988	1.000	0.198	0.196	0.197

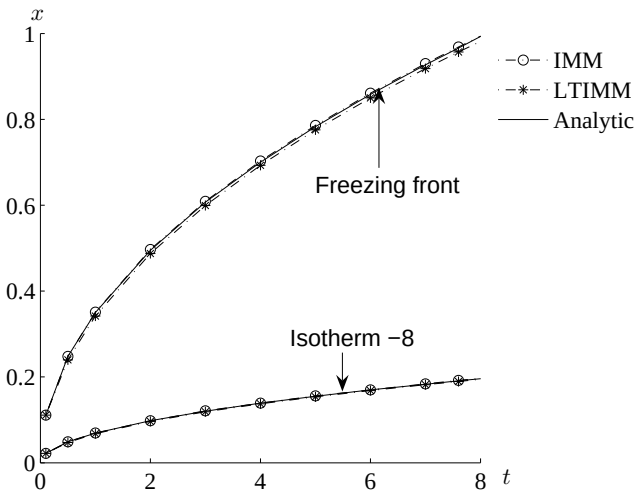

Figure 6.3: The position of the melting front and isotherm with temperature $u = -8$ with increasing time in example 6.2

6.5 The freezing problem when $y = f(x, u, t)$

Finally we ensure that there is no bias in our example, by re-writing the equations in examples 6.1 and 6.2 so that $y = f(x, u, t)$ and recalculating. We find that the numerical results are precisely the same with x and y interchanged. We do not list the results here as they are identical to those obtained in examples 6.1 and 6.2 but with x and y interchanged.

6.6 Summary of Chapter 6

In example 6.1 we have taken the one-dimensional freezing problem described in chapter 5 and used it to solve a simple two-dimensional problem, showing how to carry out an isotherm migration mapping in two dimensions. This results in an expression in which one of the space co-ordinates is a function of the other space co-ordinate, temperature and time.

We have seen that this method increases in complexity as it involves a degree of interpolation, both at the fixed boundaries and for a new term

arising in the finite difference approximation.

We have shown that to avoid instability in this finite difference method, it is necessary to make the time-steps very much smaller than in the one-dimensional case. We have demonstrated that the isotherm migration method gives a good approximation to the analytical solution with the modifications described and we have tested the method further, by looking at a symmetrical problem using the other space co-ordinate. We have not presented the numerical solution as it is precisely the same as that with the space co-ordinates interchanged.

In example 6.2 we have solved the problem described in example 6.1, using the Laplace transform isotherm migration method and we noted that the method gives very good results and that it allows larger time-steps to be taken. Although the method still requires more calculations than the isotherm migration method, for the two-dimensional case the increase in number is significantly lower than for the one-dimensional case.

6.6.1 Contribution

We have demonstrated that the Laplace transform isotherm migration method performs well in simple two-dimensional problems with phase change and the results compare favourably with those obtained from the analytic solution and the isotherm migration method. Although the method becomes more complex we have addressed these issues so that accurate results were obtained.

Chapter 7

The Laplace transform isotherm migration method for a two-dimensional solidification problem in a prism

In Chapter 6, we considered the method of Crank and Gupta (1975) in which they showed how to effect an isotherm mapping for the two-dimensional heat equation and we solved a simple two-dimensional freezing problem with it. In their work Crank and Gupta (1975) used it to solve a problem of solidification of a square prism of fluid initially at constant temperature throughout.

We now solve this problem first without and then with the use of the Laplace transform. We begin by revisiting the mapping process and we make y the dependent variable, as did Crank and Gupta (1975), so that we may have some basis for comparison of our results.

We consider some of the difficulties which could be encountered and how we may overcome these.

7.1 The mapping of the equations

We refer to chapter 6 and equations (6.1), (6.2), (6.3) and (6.4) for the usual way of writing the heat conduction equation in two dimensions in non-dimensional form. In order to use equation (6.4) to solve a problem we work on a u,x-grid and attempt to find a value of y for each grid-point at the required time. Equation (6.4) may be solved using a finite difference scheme and we shall describe later how to do this. However, we will not be able to use this method to solve every problem and there are some difficulties which we need to be aware of before choosing an example.

We consider the following situation for a square prism with boundary conditions

$$\begin{aligned} u = 0, \quad y = 0, \quad 0 \leqslant x \leqslant 1, \quad t > 0, \\ u = 1, \quad y = 1, \quad 0 \leqslant x \leqslant 1, \quad t > 0, \\ u = y^2, \quad x = 0, \quad 0 \leqslant y \leqslant 1, \quad t > 0, \\ u = y, \quad x = 1, \quad 0 \leqslant y \leqslant 1, \quad t > 0, \end{aligned}$$

and initial condition

$$u = \frac{1}{2}(x + y), \quad 0 < x < 1, \quad 0 < y < 1, \quad t = 0$$

described by Crank and Gupta (1975). To solve this, we would choose a u,x-grid such that $u_i = u_0 + i\delta u, \quad i = 1, 2, ..., N$ and $x_j = x_0 + j\delta x$, $j = 1, 2, ..., M$.

The first of the boundary conditions implies that $y = 0$ for all points x_j on the u_0 grid line, and the secondary boundary condition that $y = 1$ for all points on the x_j on the u_N grid line for all $t > 0$. The third condition states that $y = u_i^{\frac{1}{2}}$ for all points on the $x = x_0 = 0$ grid line for all $t > 0$ and the final boundary condition means that $y = u_i$ for all points on the

grid line $x = x_M = 1$ for all $t > 0$. From the initial condition when $t = 0$ we can deduce that $y_{i,j} = 2u_i - x_j$ at all the internal grid points, (u_i, x_j). We also note that $u_0 + r\delta u$ is the isotherm on which the temperature is u_r in the x,y-plane.

We are now able to proceed with evaluating numerically the solutions to this problem using a finite difference method. However we shall not develop the solution fully, we shall use the ideas to illustrate difficulties which may be encountered. We notice that in this case the boundary and initial conditions are such that y is a single-valued function of u and x, which means that there is only one value of y for each point on the u,x-grid. However if the third boundary condition had been

$$u = 0, \quad x = 0, \quad 0 \leqslant y \leqslant 1, \quad t > 0$$

then y would be multi-valued for $x = 0$, $u = 0$ on the u,x-grid and could take any value in the range $0 \leqslant y \leqslant 1$. This is not to say that we cannot overcome this, and we may be able to use interpolation or extrapolation procedures using other points on the grid.

We also know, that no isotherms other than $u = 0$ exist at $t = 0$ and this means that no values of y will be available at any internal grid points at $t = 0$. We have to find a method for generating starting values at some small time. Since we do not have any analytical solution available we shall use a method involving approximate methods of boundary layer theory, which we shall describe later.

Although the temperature u is always a single-valued function in the x,y-plane, we have seen in the one-dimensional case that when u is an independent variable, multi-valued functions arise in certain situations, as in example 3.2, the case of a rod with the boundaries held at constant temperatures, we saw that it was possible that two different points in the region could have the same temperature, and this situation could equally arise in a square region initially at zero temperature with the boundaries held at con-

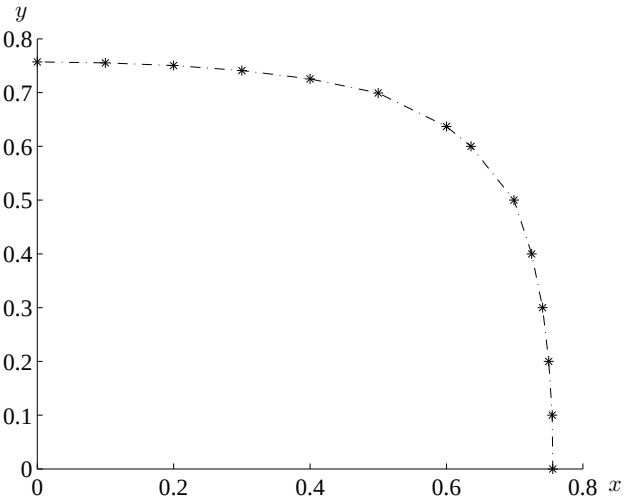

Figure 7.1: Diagram showing an isotherm which does not exist at all x grid points

stant non-zero temperatures. Furthermore, there may be situations where an isotherm does not exist at a given u, x-grid point.

Figure 7.1 shows a possible isotherm position. We can see that this isotherm does not cross the gridlines $x = 0.8$ and $x = 0.9$ at this particular time, and therefore we would be unable to find a y-value for points on the u, x-grid where $x = 0.8$ and $x = 0.9$.

It may be possible to resolve these difficulties using the symmetry of the problem, if it exists.

We have already shown in chapter 6, how to formulate the boundary condition on the solid-liquid interface from

$$K_{sol}\frac{\partial u_{sol}}{\partial n} - K_{liq}\frac{\partial u_{liq}}{\partial n} = L\rho\frac{dx}{dt} \tag{7.1}$$

with the y-dependence given by

$$\frac{\partial y}{\partial t} = \frac{1}{s}\left\{1 + \left(\frac{\partial y}{\partial x}\right)^2\right\}\left\{K_{sol}\left(\frac{\partial y}{\partial u_{sol}}\right)^{-1} - K_{liq}\left(\frac{\partial y}{\partial u_{liq}}\right)^{-1}\right\} \tag{7.2}$$

where s is the Stefan constant which depends on the thermal properties of the material.

7.2 Solidification of a square prism of fluid

Example 7.1

We consider the problem described by Crank and Gupta (1975) of an infinitely long prism initially filled with a fluid at the fusion temperature $u = 1$. In the present example the temperature on the surface of the liquid is subsequently maintained constant at $u = 0$ below the fusion temperature so that solidification occurs from the surface inwards. We assume that the prism extends between $-1 \leqslant x \leqslant 1$ and $-1 \leqslant y \leqslant 1$. In all of our previous examples, we have used water as the material being frozen or melted and therefore a fusion temperature of zero, but we choose the fusion temperature in this example to be $u = 1$ because we wish to compare our results with those of Crank and Gupta (1975). Following their model we use the non-dimensional form of the heat equation

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \tag{7.3}$$

and the boundary conditions are $u = 0$ on all four sides and on the solidification front

$$u = 1, \quad u_F(x, y, t) = 0, \quad t > 0 \tag{7.4}$$

where $u_F(x, y, t) = 0$ is the contour on the freezing front. This is analogous to the parameter $\tilde{x}_0$ which we used for the moving boundary in one dimension.

Initially we have $u_F(x, y, 0) = 0, \quad t = 0$ because solidification has not yet started.

We are able to simplify equation (7.1) to

$$\frac{\partial u}{\partial n} = -s\frac{dx}{dt}, \quad u_F(x, y, t) = 0 \tag{7.5}$$

where n is the outward normal to $u_F(x, y, t) = 0$, the direction of heat flow being in the direction normal to the interface contour. This simplification

may be made, because the liquid phase is always at the uniform temperature $u = 1$ and consequently there is no temperature gradient in the liquid phase.

We are therefore able to use the shorter form of equation (7.2)

$$\frac{\partial y}{\partial t} = \frac{1}{s}\left\{1 + \left(\frac{\partial y}{\partial x}\right)^2\right\}\left(\frac{\partial y}{\partial u}\right)^{-1} \tag{7.6}$$

This example has symmetry about the axes and so we need only to consider one quadrant of the prism enclosed by the axes and $x = 1$ and $y = 1$ and we know that inside the region across the axes the flux will be zero so that we have further boundary conditions

$$\frac{\partial u}{\partial x} = 0, \quad x = 0$$

and

$$\frac{\partial u}{\partial y} = 0, \quad y = 0$$

together with

$$\frac{\partial y}{\partial x} = 0, \quad x = 0$$

by symmetry.

7.3 The finite difference form.

We evaluate the numerical solution on a u,x-grid, choosing δu and δx such that

$$u_i = u_0 + i\delta u, \quad i = 1, 2, ..., N \quad (u_0 = 0, \quad u_N = 1)$$

and

$$x_j = x_0 + j\delta x, \quad j = 1, 2, ..., M \quad (x_0 = 0, \quad x_M = 1)$$

We need to calculate the values of y on this grid for successive values of δt. We represent equation (7.6) by the finite difference form

$$\frac{Y_{N,j}^{(n+1)} - Y_{N,j}^{(n)}}{\delta t} = \frac{1}{s}\left\{1 + \left(\frac{Y_{N,j}^{(n)} - Y_{N,j-1}^{(n)}}{\delta y}\right)^2\right\}\frac{\delta u}{Y_{N,j}^{(n)} - Y_{N-1,j}^{(n)}} \tag{7.7}$$

which is our previous equation (6.8) with X replaced by Y which gives us $Y_{N,j}^{(n+1)}$ for $j = 1, 2, ..., M$. We use the backward difference for $\left(\frac{\partial y}{\partial u}\right)^{-1}$ because at the interface we need to refer back to the previous isotherm.

We now consider equation (6.4), in which x and y have been interchanged, which may be simplified to

$$\frac{\partial y}{\partial t} = -\left(\frac{\partial y}{\partial u}\right)\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 y}{\partial u^2}\left(\frac{\partial y}{\partial u}\right)^{-2} \tag{7.8}$$

and is represented by

$$\frac{Y_{i,j}^{(n+1)} - Y_{i,j}^{(n)}}{\delta t} = -\left(\frac{Y_{i,j}^{(n)} - Y_{i-1,j}^{(n)}}{\delta u}\right)\frac{\partial^2 u}{\partial x^2} + \frac{Y_{i-1,j}^{(n)} - 2Y_{i,j}^{(n)} + Y_{i+1,j}^{(n)}}{\left(Y_{i,j}^{(n)} - Y_{i-1,j}^{(n)}\right)^2}$$

To remain consistent we use a backward difference for the term $\frac{\partial y}{\partial u}$.

In addition, we have a term $\frac{\partial^2 u}{\partial x^2}$ and we deal with this by interpolating or extrapolating linearly the values of u corresponding to $y_{i,j}$ at $x_{j-1}$and x_{j+1}. The formulae we use are for u at x_{j-1}

$$U = \frac{U_{i+1}\left(Y_{i,j-1} - Y_{i,j}\right) - U_i\left(Y_{i+1,j-1} - Y_{i,j}\right)}{Y_{i,j-1} - Y_{i+1,j-1}} \tag{7.9}$$

and for u at x_{j+1}

$$U = \frac{U_{i-1}\left(Y_{i,j+1} - Y_{i,j}\right) - U_i\left(Y_{i-1,j+1} - Y_{i,j}\right)}{Y_{i,j+1} - Y_{i-1,j+1}} \tag{7.10}$$

and we can then apply a central difference formula to find a value for $\frac{\partial^2 u}{\partial x^2}$ at the grid point we are interested in.

We now encounter some of the difficulties discussed in section 7.1. On the y-axis equation (7.8) breaks down for the following reason. We know that

$$\frac{\partial y}{\partial x} = 0$$

and

$$\frac{\partial u}{\partial x} = 0$$

But

$$\frac{\partial y}{\partial x} = \frac{\partial y}{\partial u}\frac{\partial u}{\partial x}$$

so that

$$\frac{\partial y}{\partial u}$$

is not determined.

We therefore fit a quadratic function through the points (x_1, y_1) and (x_2, y_2), remembering that $\frac{\partial y}{\partial x} = 0$ on the y-axis, allowing us to calculate the intercept on the y-axis, which is the position where the isotherm crosses the axis. Another symmetry of the problem is the diagonal $y = x$ so we need only consider those points for which $y \geqslant x$. Figure 7.2 shows a possible situation of a typical isotherm at times t and $t+\delta t$. For the position $y = k\delta t$ we can see that as the isotherm passes through the line $y = x$ it will cut the grid line at x_{r+1}. We know that the solidification front moves along the direction normal to the front as described in equation (7.1). We also know that the line $y = x$ is an axis of symmetry and therefore we follow the work of Crank and Gupta (1975) and find the point R by fitting a circle, centred on $y = x$, and passing through the points P and Q. We need to know when this method will be appropriate, so we need to know whether the isotherm will cut the grid line at x_{r+1} at the next step. This will happen if the y value of the point Q is greater than $x + \delta x$ and we can see that this step will be appropriate when this condition is met. On the other hand, if the condition is not met, then we have the situation shown in figure 7.2 for the isotherm at $(k + 1)\delta t$ where W is less than $x + \delta x$ and that after passing through the line $y = x$, the isotherm will not cut the grid line at x_{r+1} grid line. In this case, we use the circle centred on the line $y = x$ and passing through the points S and T to find the point W on the grid line at x_r. By choosing the condition $y > x + \delta x$ we ensure that the circle fitted through the two neighbouring points will cut the next grid line parallel to the y-axis because of the symmetry about the diagonal $y = x$.

Using the symmetry of this problem will ensure that we avoid having double-valued functions as well as reducing the number of calculations to

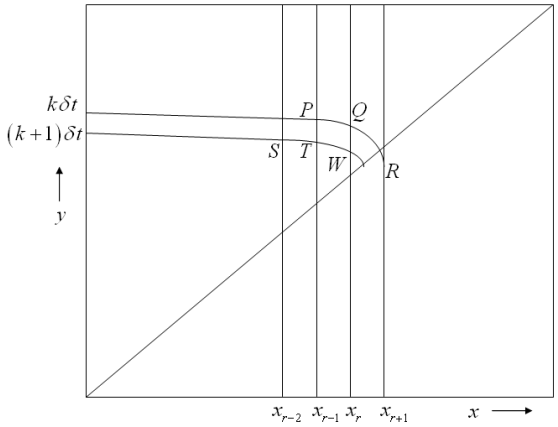

Figure 7.2: Diagram showing possible positions of an isotherm at times t and $t + \delta t$

be performed. However extra calculations are needed for the fitting of the quadratic and circle on the axes and the line $y = x$ respectively.

7.4 Starting values for the problem

As we have seen previously, we need to find some values for the positions of the isotherms a short time after $t = 0$ in order to start the problem. We use the one-parameter method of Poots (1962), as suggested by Crank and Gupta (1975).

Poots solved the problem of the two-dimensional solidification of a liquid in a prism using approximate integral methods for solving boundary-layer equations in fluid dynamics. The method, which we follow, is the Kármán-Polhausen method, which reduces the problem of finding the two-dimensional solidification front to the numerical integration of an ordinary first-order differential equation. As we have previously stated, the contour of the solidification front is represented by $u_F(x, y, t) = 0$. At $t = 0$, the position of the solidification front is at the edges of the prism, and so u_F is

satisfied if

$$\left(x^2-1\right)\left(y^2-1\right)=0$$

From observations of the freezing of a liquid in a prism, after a small time, the freezing front takes the shape of a square having round corners, and near the end of solidification period, the shape becomes circular, that is of the form

$$x^2+y^2-f(t)=0$$

At the end of solidification, the front lies on the axes $x=0$ or $y=0$. Therefore it is reasonable to assume the shape of the solidification front satisfies

$$\left(x^2-1\right)\left(y^2-1\right)-\varepsilon\left(t\right)=0 \tag{7.11}$$

where $\varepsilon=0$ for $t=0$ and $\varepsilon=1$ at the time of complete solidification.

The heat balance equation is found by integrating both sides of equation (7.3) over the solidified phase bounded by the edge of the prism and the moving boundary contour. Using Green's theorem and the divergence theorem, an integral is then obtained which is used to find the value $\varepsilon(t)$. To satisfy both the boundary conditions and equation (7.11), we assume the form

$$u=\frac{\left(x^2-1\right)\left(y^2-1\right)}{\varepsilon} \tag{7.12}$$

for the temperature distribution in the solidified phase. This expression is substituted into the heat balance integral and the resulting first order differential equation is solved, leading to an expression

$$t=\int_0^{\varepsilon}\left\{\frac{3}{8}sA_0\left(\varepsilon\right)+A_1\left(\varepsilon\right)\right\}d\varepsilon \tag{7.13}$$

where

$$A_0=-\varepsilon\frac{dI_0}{d\varepsilon}$$

and

$$A_1=\frac{1}{24}\left(\frac{4-3I_1}{\varepsilon}\right)$$

Poots (1962) defines a function

$$R\left(x,\varepsilon\right)=\left\{\frac{1-\varepsilon-x^2}{1-x^2}\right\}$$

and says that the required integrals are then

$$I_0=\int_0^{\sqrt{(1-\varepsilon)}}R^{\frac{1}{2}}dx$$

and

$$I_1=\int_0^{\sqrt{(1-\varepsilon)}}\left(2-2x^2+\varepsilon\right)R^{\frac{1}{2}}dx$$

which must be expressed as standard elliptic functions of the first and second kind.

Poots provides tables with values of $A_0(\varepsilon)$ and $A_1(\varepsilon)$ for values of ε from 0 to 1.0 in steps of 0.4, and so for a particular value of ε we can find the time t from equation (7.13) and the shape of the contour of each isotherm of temperature u at that time, from equation (7.12).

In order that we may compare our results with Crank and Gupta, we follow their suggestion and use a starting value of $t=0.0461$ and $\varepsilon=0.32$ to find the positions of the isotherms and mesh sizes $\delta u=\delta x=0.1$, while s has a value 1.561, and the time step is $\delta t=0.0001$ remembering that taking larger time steps may cause instability.

Table 7.1: Values of the y co-ordinate on the solid-liquid interface for fixed values of x at various times

t	0	0.1	0.2	0.3	0.4	0.5	0.6
0.05	0.8121	0.8102	0.8044	0.7937	0.7761	0.7474	0.6902
0.10	0.6950	0.6936	0.6892	0.6809	0.6657	0.6367	0.5571
0.15	0.6124	0.6108	0.6061	0.5966	0.5774	0.5156	
0.20	0.5438	0.5418	0.5358	0.5230	0.4743		
0.25	0.4828	0.4800	0.4715	0.4521	0.3830		
0.30	0.4263	0.4223	0.4102	0.3598			
0.35	0.3725	0.3665	0.3485	0.2790			
0.40	0.3291	0.3109	0.2565				
0.45	0.2885	0.2529	0.1753				

Table 7.2: Values of the y co-ordinate on the solid-liquid interface for fixed values of x at various times quoted by Crank and Gupta for comparison

t	0	0.1	0.2	0.3	0.4	0.5	0.6
0.050	0.8125	0.8106	0.8048	0.7940	0.7764	0.7476	0.6904
0.100	0.6979	0.6965	0.6821	0.6836	0.6683	0.6392	0.5606
0.150	0.6157	0.6141	0.6095	0.6000	0.5810	0.5201	
0.200	0.5473	0.5453	0.5394	0.5268	0.4789		
0.250	0.4865	0.4838	0.4755	0.4567	0.3894		
0.300	0.4302	0.4263	0.4146	0.3654			
0.350	0.3766	0.3708	0.3534	0.2859			
0.400	0.3337	0.3158	0.2623				
0.450	0.2816	0.2585	0.1893				
0.495	0.2376	0.2056	0.1097				

We show the values found for y on the solid/liquid interface for fixed values of x at various times in table 7.1 and we show the results found by Crank and Gupta in table 7.2 for comparison. We note they are very similar. Because there is symmetry about the line $y = x$, only the values of y above the diagonal are tabulated. The computations are stopped at $t = 0.45$ because after this, there are only two values of y left corresponding to the grid points in the x direction on the solid-liquid interface and so we cannot proceed any further with this method.

Figure 7.4, shows the contours of the interface at several different times and the diagram shows how the freezing front is similar to a square with rounded corners at the smaller times and as solidification progresses, it becomes more circular in shape. Figure 7.4 shows the final positions of all the isotherms at $t = 0.45$. Our results are comparable with those of Crank and Gupta (1975) and since we do not have an analytic solution, we have no other means of comparing our solutions to the example.

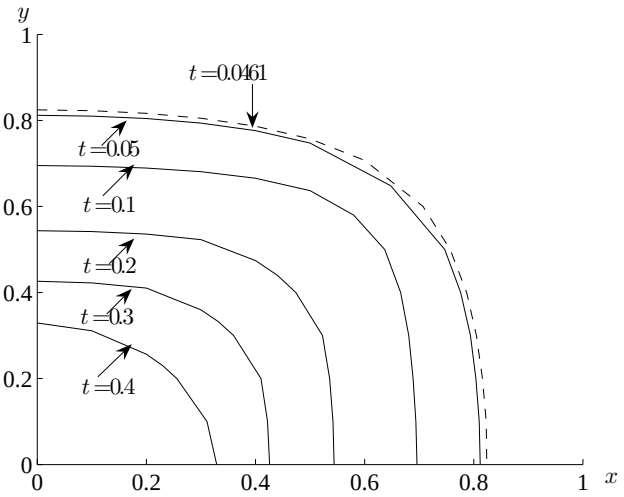

Figure 7.3: Diagram showing the positions of the interface at various times. The dotted line shows the position at $t = 0.0461$ obtained from the Poots one-parameter method

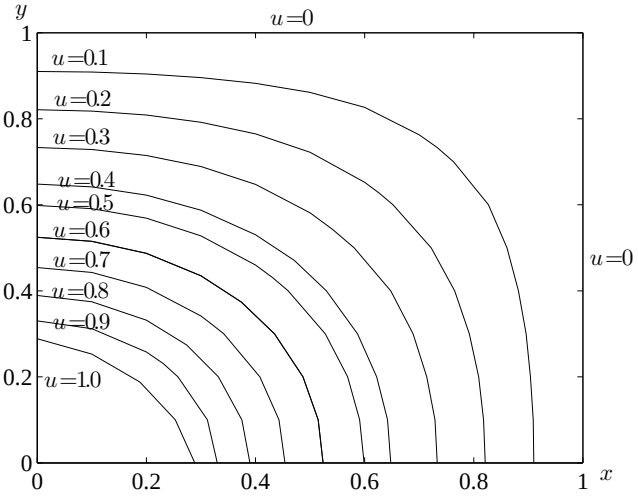

Figure 7.4: Diagram showing the final positions of the isotherms at $t = 0.45$

7.5 The Laplace transform isotherm migration method

Example 7.2

We now solve example 7.1, using the Laplace transform method. We notice that the transformed heat equation, equation (7.8), and the equation used to calculate the position of the freezing front, equation (7.6), are non-linear, and we use direct iteration, as we have previously described, to linearise them.

We write equation (7.8) as

$$\left(\frac{\partial y}{\partial t}\right)^{(n)} = -\left(\frac{\partial y}{\partial u}\right)^{(n-1)}\left(\frac{\partial^2 u}{\partial x^2}\right)^{(n-1)} + \left(\left(\frac{\partial y}{\partial u}\right)^{-2}\right)^{(n-1)}\left(\frac{\partial^2 y}{\partial u^2}\right)^{(n)} \tag{7.14}$$

where the expressions with superscript $(n-1)$ are the numerical results for those terms calculated at the previous step. We continue the iterative process until a satisfactory convergence is achieved.

We take the Laplace transform of equation (7.14) to get

$$\lambda\bar{y}^{(n)} - y\left(t_0\right) = -\frac{1}{\lambda}\left(\frac{\partial y}{\partial u}\right)^{(n-1)}\left(\frac{\partial^2 u}{\partial x^2}\right)^{(n-1)} + \left(\left(\frac{\partial y}{\partial u}\right)^{-2}\right)^{(n-1)}\left(\frac{\partial^2 \bar{y}}{\partial u^2}\right)^{(n)}$$

where t_0 is the chosen time at which to start the calculation. This equation leads to

$$\bar{y}^{(n)} = \frac{y\left(t_0\right)}{\lambda} - \frac{1}{\lambda^2}\left(\frac{\partial y}{\partial u}\right)^{(n-1)}\left(\frac{\partial^2 u}{\partial x^2}\right)^{(n-1)} + \frac{1}{\lambda}\left(\left(\frac{\partial y}{\partial u}\right)^{-2}\right)^{(n-1)}\left(\frac{\partial^2 \bar{y}}{\partial u^2}\right)^{(n)} \tag{7.15}$$

Similarly, we write equation (7.6) as

$$\left(\frac{\partial y}{\partial t}\right)^{(n)} = \frac{1}{s}\left\{1 + \left(\left(\frac{\partial y}{\partial x}\right)^2\right)^{(n-1)}\right\}\left(\left(\frac{\partial y}{\partial u}\right)^{-1}\right)^{(n-1)}$$

Taking the Laplace transform of this gives

$$\lambda\bar{y}^{(n)} - y\left(t_0\right) = \frac{1}{\lambda}\frac{1}{s}\left\{1 + \left(\left(\frac{\partial y}{\partial x}\right)^2\right)^{(n-1)}\right\}\left(\left(\frac{\partial y}{\partial u}\right)^{-1}\right)^{(n-1)}$$

so that we have

$$\bar{y}^{(n)} = \frac{y\left(t_0\right)}{\lambda} + \frac{1}{\lambda^2}\frac{1}{s}\left\{1 + \left(\left(\frac{\partial y}{\partial x}\right)^2\right)^{(n-1)}\right\}\left(\left(\frac{\partial y}{\partial u}\right)^{-1}\right)^{(n-1)} \tag{7.16}$$

As before, we use a Gauss Seidel solver to solve equations (7.15) and (7.16).

The process is similar to that using the finite difference method, but we note the following modifications. We calculate all the non-linear terms before entering Laplace space. This includes the calculation of $\frac{\partial^2 u}{\partial x^2}$ which has to be done in x, y-space. As before, we calculate the values on the y-axis by fitting a quadratic function and we fit the circle to the line $y = x$ in x, y-space. Clearly this cannot be done within the Laplace space and so we have to use Stehfest inversion at each stage to be able to fit these points.

We notice that our trial problem has only a total time span of 0.45, and we know problems can arise if we use very small values of T in the Stehfest method, because of the term $\lambda_j = j\frac{\ln 2}{T}$. However, using a value of $T = 0.1$ in this problem means we are taking a first step equivalent to about one quarter of the total time and that such a large step might cause difficulties. Indeed we find this to be the case; the process breaks down with this value of T and this is probably because when $t = 0.1$ some of the isotherms have moved a significant distance and may no longer exist at the larger values of the x grid-points. By trying different values of T we find we are able to obtain results with a value of T of 0.03.

A further problem arises in calculating the interpolation for $\frac{\partial^2 u}{\partial x^2}$. We find that in certain circumstances equation (7.10) does not return a value because both $y_{i,j+1}$ and $y_{i-1,j+1}$ are undefined, as the isotherm involved does not cut the x_{j+1} gridline. This difficulty does not appear to arise in the finite difference method because the time steps are much smaller and this means that the isotherms are progressing more slowly than in the Laplace transform isotherm migration method, in which the time steps are necessarily larger. Clearly there exists some temperature value here but our

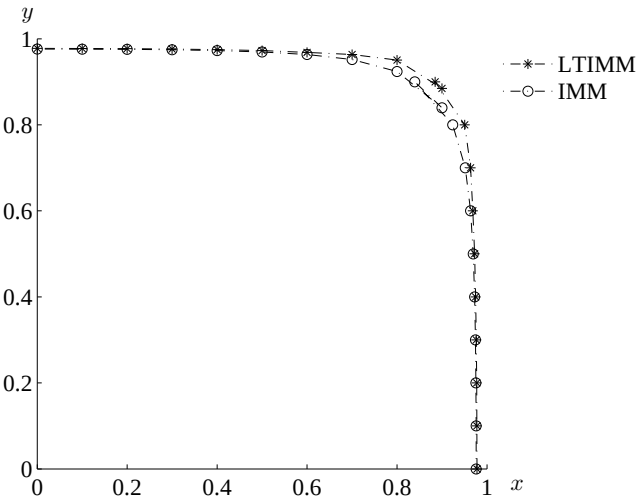

Figure 7.5: Diagram showing the position of the isotherm with temperature $u = 0.1$ at time $t = 0.0761$ in example 7.2

method of interpolation cannot be used. When this arises, we make a guess at the value, using the result found for equation (7.9). We write

$$u(x_{j+1}) = u(x_j) - \omega(u(x_{j-1}) - u(x_j))$$

where

$$1 \leqslant \omega \leqslant 2$$

and investigate the solutions with a variety of values of ω. We find that a value $\omega = 1.0$ gives solutions which are of the order expected. Higher values of ω give the same results, so provided ω is positive, other values could be used, but there is no need for this.

7.5.1 Results

We show the results obtained for three isotherms at three different times. Considering figures 7.5, 7.6 and 7.7, which show the positions of isotherms with temperatures 0.1, 0.5 and the freezing front after a small elapsed time of 0.0761, we see that the positions of all three are similar to those obtained

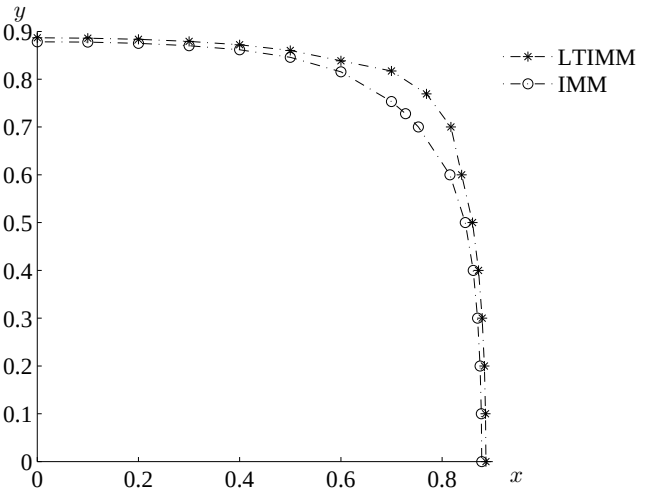

Figure 7.6: Diagram showing the position of the isotherm with temperature $u = 0.5$ at time $t = 0.0761$ in example 7.2

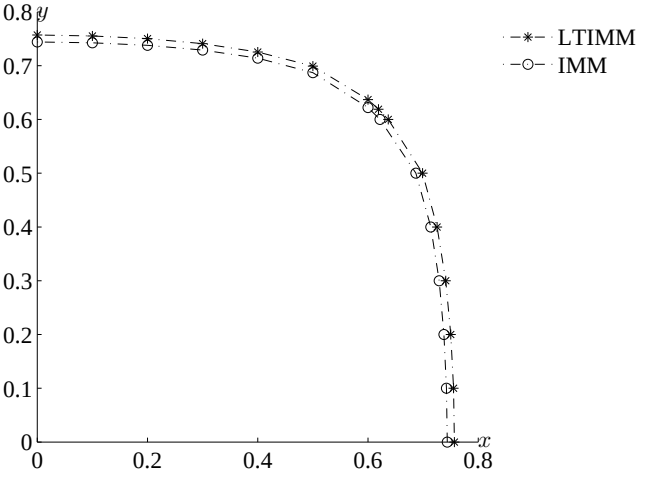

Figure 7.7: Diagram showing the position of the freezing front with temperature $u = 1$ at time $t = 0.0761$ in example 7.2

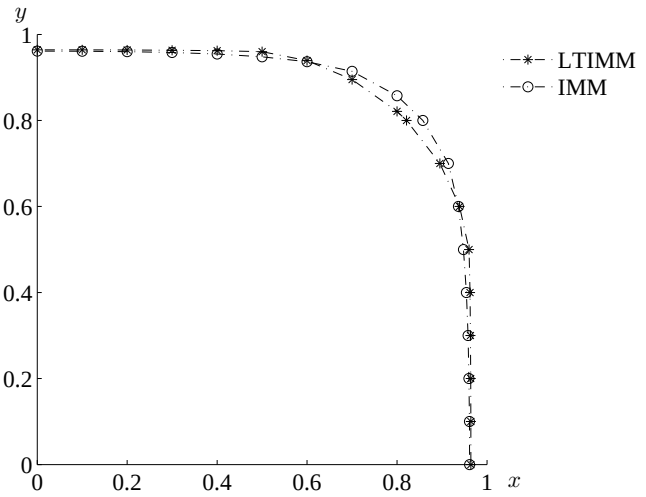

Figure 7.8: Diagram showing the position of the isotherm with temperature $u = 0.1$ at time $t = 0.1661$ in example 7.2

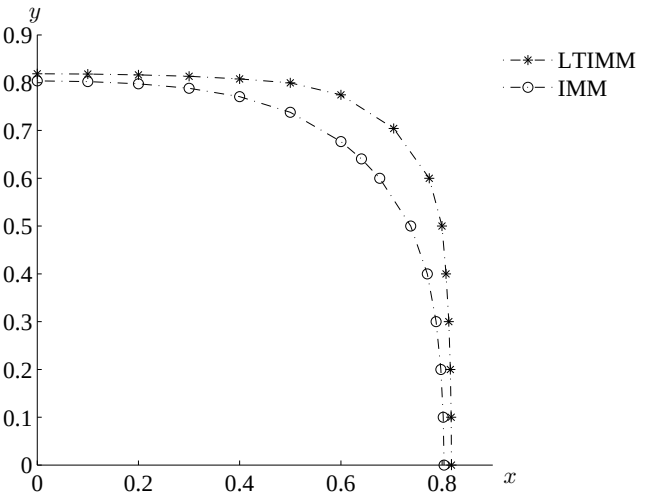

Figure 7.9: Diagram showing the position of the isotherm with temperature $u = 0.5$ at time $t = 0.1661$ in example 7.2

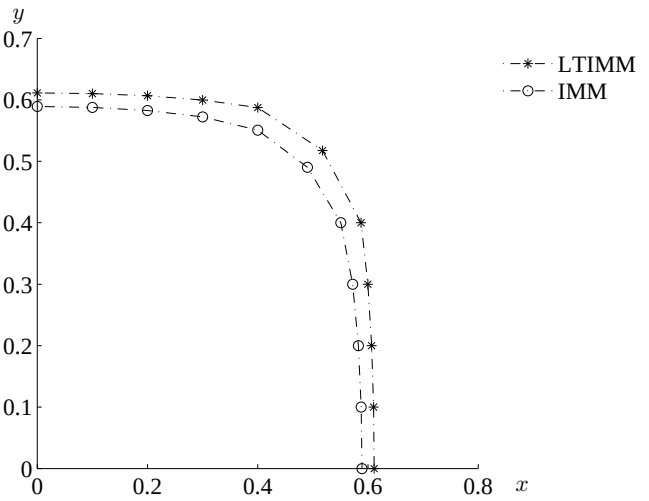

Figure 7.10: Diagram showing the position of the freezing front with temperature $u = 1$ at time $t = 0.1661$ in example 7.2

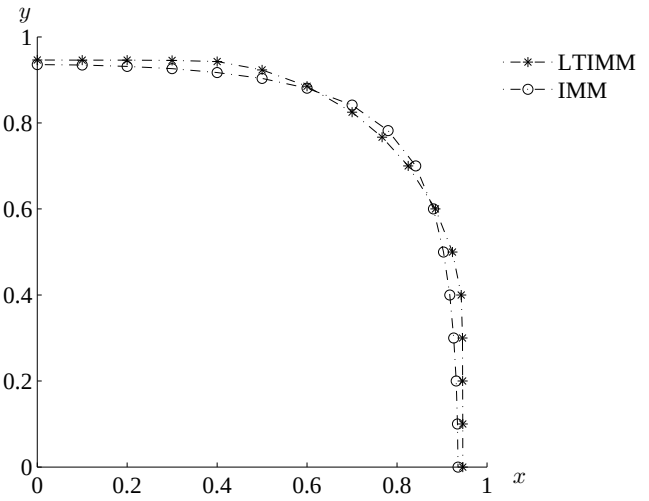

Figure 7.11: Diagram showing the position of the isotherm with temperature $u = 0.1$ at time $t = 0.3461$ in example 7.2

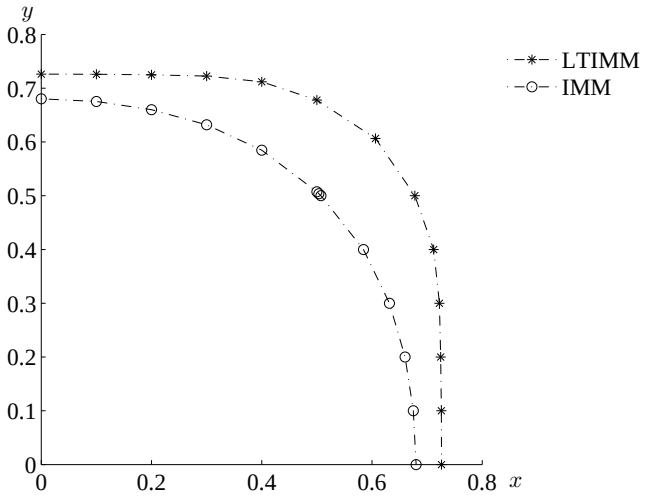

Figure 7.12: Diagram showing the position of the isotherm with temperature $u = 0.5$ at time $t = 0.3461$ in example 7.2

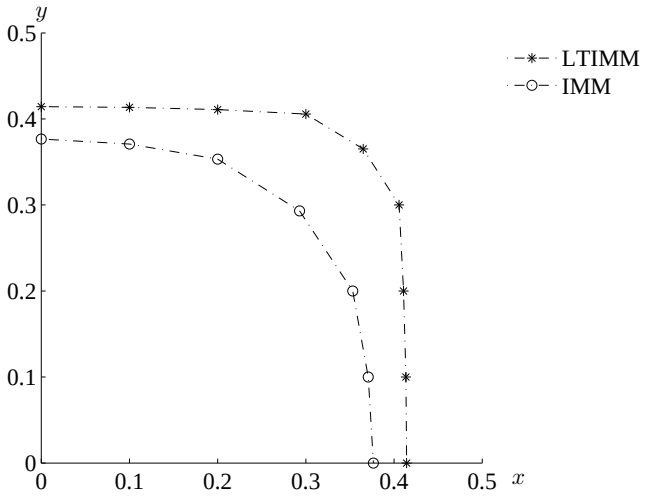

Figure 7.13: Diagram showing the position of the freezing front with temperature $u = 1$ at time $t = 0.3461$ in example 7.2

Table 7.3: Values of the y co-ordinate on the solid-liquid interface for fixed values of x at various times calculated using the Laplace transform in example 7.2

t	0	0.1	0.2	0.3	0.4	0.5	0.6
0.076	0.7569	0.7552	0.7503	0.7410	0.7254	0.6990	0.6369
0.106	0.7016	0.7002	0.6959	0.6878	0.6739	0.6495	0.5767
0.136	0.6539	0.6526	0.6488	0.6415	0.6288	0.6057	
0.196	0.5727	0.5716	0.5684	0.5621	0.5509		
0.256	0.5036	0.5027	0.4998	0.4943	0.4454		
0.286	0.4723	0.4714	0.4687	0.4633	0.4064		
0.346	0.4143	0.4135	0.4110	0.4061			
0.376	0.3873	0.3865	0.3841	0.3350			
0.406	0.3614	0.3606	0.3583	0.3016			
0.436	0.3364	0.3357	0.3335				
0.466	0.3124	0.3116	0.3095				
0.496	0.2873	0.2884	0.2426				
0.526	0.2806	0.2653	0.2028				

using the the finite difference method with the greatest difference being around the axis of symmetry $y = x$.

Considering figures 7.8, 7.9 and 7.10, which show the positions of the same isotherms at a time of 0.1661, we see the same pattern for isotherms at temperatures 0.1 and 0.5, and the freezing front still follows the same shape, but we see a lag beginning to develop. The isotherm positions calculated using the Laplace transform appear to be moving more slowly than those calculated using the finite difference method. Although we are updating the initial values at each time step as described in chapter 5, it appears that the constraints on the size of the time step mean we are forced to use values which do not relate closely enough to those at the new time we are interested in.

Finally looking at figures 7.11, 7.12 and 7.13 which show the positions of the isotherms at a time of 0.3461, that is close to the time where we can no longer continue the calculations, we see that the position of isotherm with temperature 0.1 is still fairly close to that obtained using the finite

difference method, while the positions of isotherm with temperature 0.5 and the freezing front are lagging behind and they seem to be retaining the shape of a square with rounded corners rather than becoming circular. We also present table 7.5, which shows that there is indeed a time lag, and that we can continue calculations up to a time of $t = 0.526$.

7.6 A re-calculation of the problem using a time-step of 0.001

Example 7.3

We know that when using the Stehfest numerical inversion method we must take care not to use a value of T which is too small. Crann (2005) suggests that the lower limit for T should be 0.1. This is because calculation of the Stehfest parameters and the inversion involves division by T, which could lead to very large numbers if T is numerically small. This in turn may cause errors, as the Stehfest weights used in the inversion have a very wide range as shown in table 4.1. However the optimum value of T is subjective and very much depends on the circumstances of the individual problem. In order to improve our results, we wish to take a smaller value of T, which will allow solutions at intermediate times. In this section we show the results when the example is re-calculated using a time of $T = 0.001$.

Figures 7.14, 7.15, and 7.16 should be compared with figures 7.5, 7.6 and 7.7, figures 7.17, 7.18 and 7.19 with 7.8, 7.9 and 7.10 and figures 7.20, 7.21 and 7.22 with figures 7.11, 7.12 and 7.13.

In all cases there is a noticeable improvement, in that the curves more closely follow those obtained using the finite difference method. Table 7.6 shows the values of the y co-ordinate on the solid-liquid interface for fixed values of x at various times using the Laplace transform isotherm migration method. This table should be compared with table 7.5, and we see that

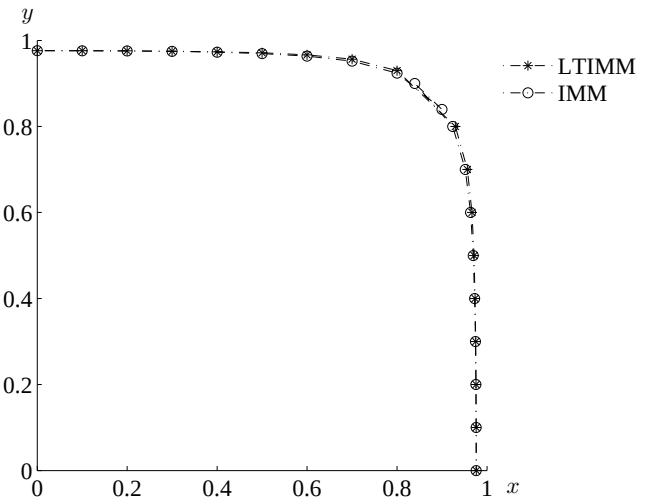

Figure 7.14: Diagram showing the position of the isotherm with temperature $u = 0.1$ at time $t = 0.0761$ using a time-step of 0.001 in example 7.3

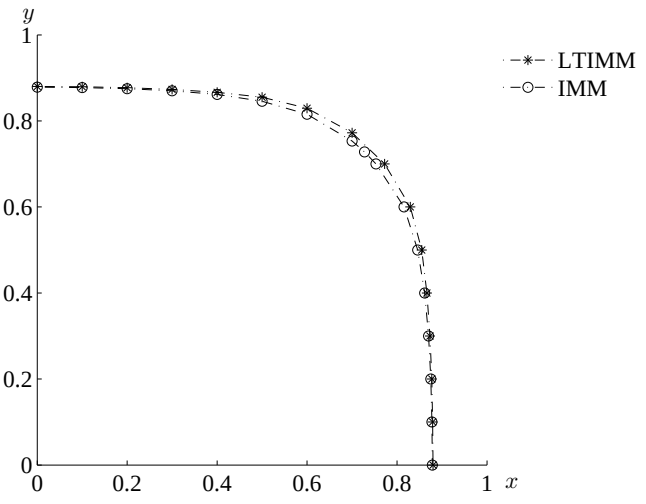

Figure 7.15: Diagram showing the position of the isotherm with temperature $u = 0.5$ at time $t = 0.0761$ using a time-step of 0.001 in example 7.3

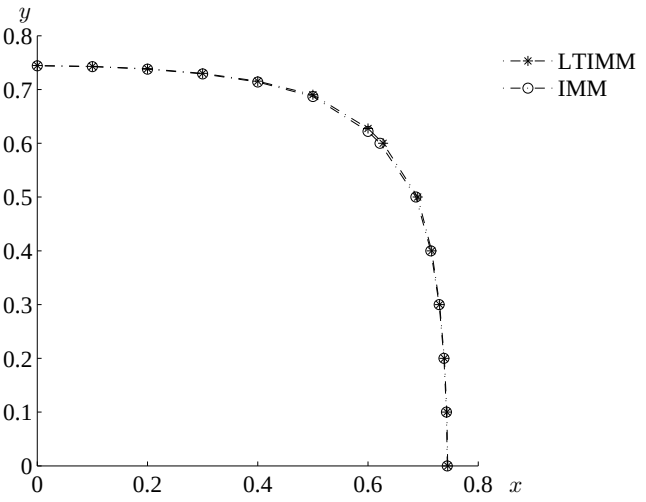

Figure 7.16: Diagram showing the position of the freezing front with temperature $u = 1$ at time $t = 0.0761$ using a time-step of 0.001 in example 7.3

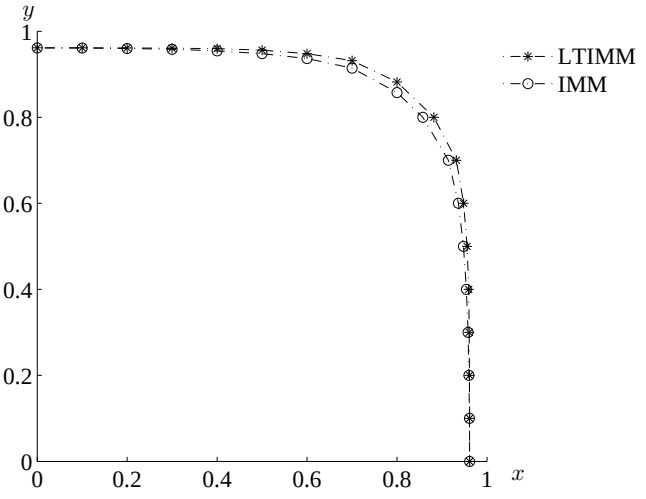

Figure 7.17: Diagram showing the position of the isotherm with temperature $u = 0.1$ at time $t = 0.1661$ using a time-step of 0.001 in example 7.3

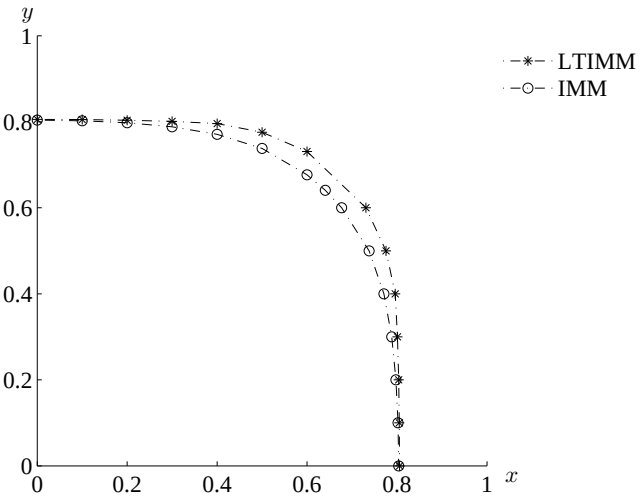

Figure 7.18: Diagram showing the position of the isotherm with temperature $u = 0.5$ at time $t = 0.16614$ using a time-step of 0.001 in example 7.3

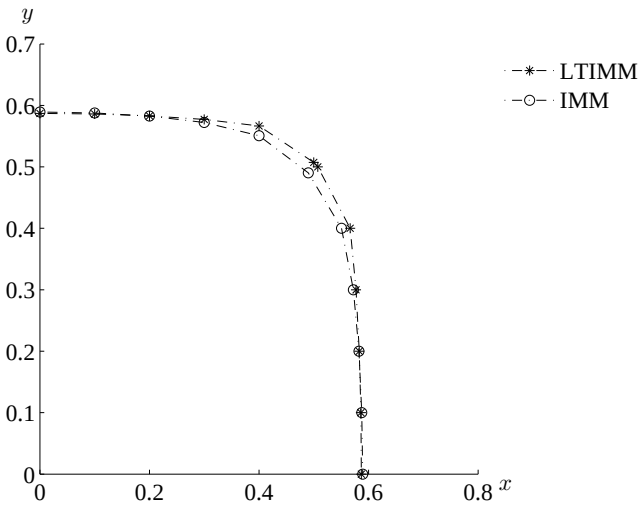

Figure 7.19: Diagram showing the position of the freezing front with temperature $u = 1$ at time $t = 0.1661$ using a time-step of 0.001 in example 7.3

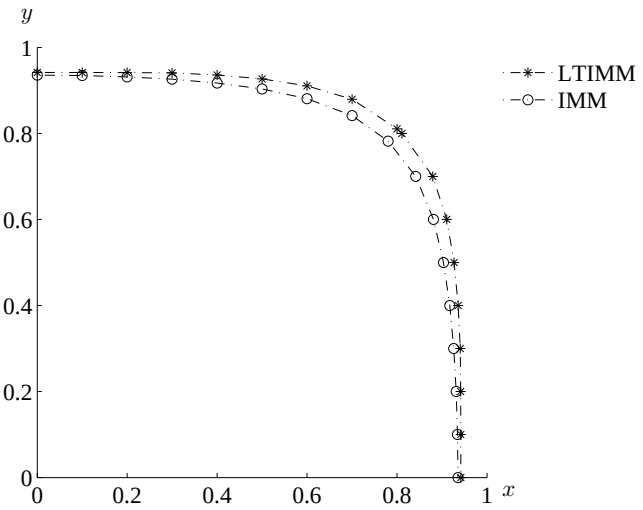

Figure 7.20: Diagram showing the position of the isotherm with temperature $u = 0.1$ at time $t = 0.3461$ using a time-step of 0.001 in example 7.3

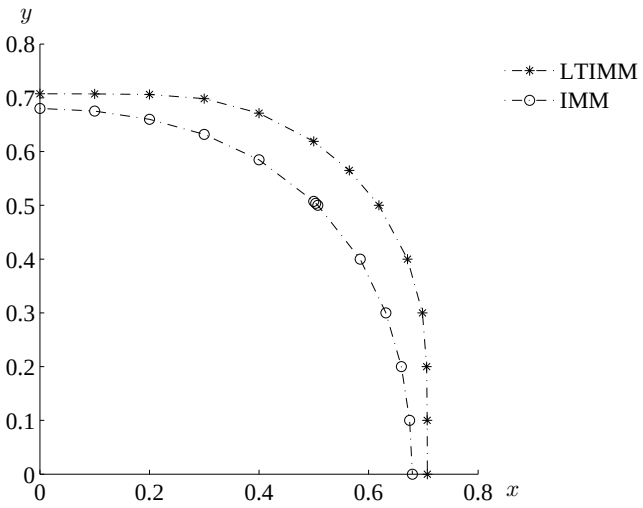

Figure 7.21: Diagram showing the position of the isotherm with temperature $u = 0.5$ at time $t = 0.3461$ using a time-step of 0.001 in example 7.3

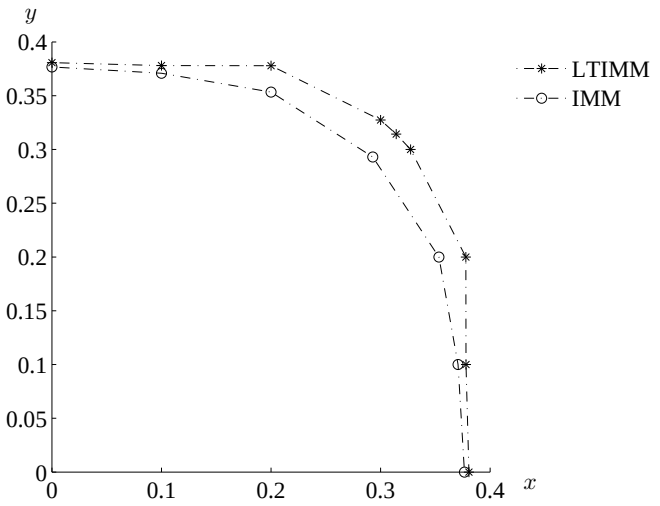

Figure 7.22: Diagram showing the position of the freezing front with temperature $u = 1$ at time $t = 0.3461$ using a time-step of 0.001 in example 7.3

although there is still some lag in the movement of the freezing front, it is not as marked as that found when a time step of 0.03 was used.

We have also attempted to use a finer mesh, by doubling the number of x grid-points and/or doubling the number of isotherms. We found that this gave less accurate results. Using a small mesh with a time step of 0.03, we find that almost immediately we have a run-time error during the calculation of the position of the circle on $y = x$.

We conclude that the Laplace transform isotherm migration method is useful for calculating the positions of the isotherms at small times. The results show that where we have an isotherm moving relatively slowly, for example, in this case isotherm with temperature 0.1, the calculated position seems to be acceptable at all times, but where the isotherms are moving more rapidly, discrepancies soon arise, and in the case of the freezing front this is the most marked. The greatest discrepancies in the results appear close to the $y = x$ symmetry line and this may be due to inaccurancies in the curve-fitting process across this line. We believe that, because the total

Table 7.4: Values of the y co-ordinate on the solid-liquid interface for fixed values of x at various times calculated using the Laplace transform and a time step of 0.001 in example 7.3

t	0	0.1	0.2	0.3	0.4	0.5	0.6
0.05	0.8120	0.8101	0.8043	0.7936	0.7759	0.7472	0.6899
0.10	0.6948	0.6935	0.6897	0.6822	0.6693	0.6460	0.5727
0.15	0.6104	0.6094	0.6063	0.6002	0.5891	0.5354	
0.20	0.5409	0.5400	0.5373	0.5320	0.5221		
0.25	0.4805	0.4797	0.4772	0.4724	0.4185		
0.30	0.4263	0.4263	0.4234	0.4189			
0.35	0.3769	0.3762	0.3741	0.3226			
0.40	0.3312	0.3305	0.3286				
0.45	0.3074	0.2875	0.2274				
0.500	0.3133	0.2350	0.1388				
0.526	0.2758	0.2068	0.1082				

time for solving this problem problem is 0.45, the time step we are restricted to taking, means that the values used in the iteration loop are too remote from the true values, but as discussed previously, we are limited in choosing the time step, by the Stehfest numerical method.

It may also be that the model of a square with rounded corners may not be the best shape for the freezing front. Crank and Crowley (1978) describe a method to solve the same problem, using a cylindrical co-ordinate system using an orthogonal grid instead of a u, x-grid together with a locally one-dimensional isotherm migration method form of the radial heat equation, and this may represent the shape of the freezing front more accurately. We know that heat flow is normal to the isotherms everywhere, and this method maintains an orthogonal grid system of isotherms and flow lines, and geometry allows for the changing shape and orientation of the grid system. However, we note that we would still need to deal with the points on the y-axis and the line $y = x$ in a geometrical manner, by fitting circles to pass through these lines. Other difficulties are also present in this method, and Crank and Crowley say that if the points on a given isotherm move too close together while calculating the local co-ordinates, they must be

respaced using a given algorithm.

We see that while the Laplace transform isotherm migration method is simple to use and gives good results in one dimension, the two dimensional case requires a different geometrical approach to account for cases where the isotherms may not exist or where there is not a unique solution to the problem. The method becomes longer and more complicated because we have to invert from Laplace space to perform the geometrical curve fitting at the axes of symmetry, before being able to continue in Laplace space.

7.7 Summary of Chapter 7.

We have discussed the general isotherm mapping in two dimensions and described the manner in which it is different from the one-dimensional case, and considered properties of the mapping which might cause difficulties in solving the two dimensional case and ways in which we might overcome this.

We have solved the problem discussed by Crank and Gupta (1975) and given details of the method proposed by Poots (1962) to find the starting values and the solution to the problem has been repeated using the Laplace transform isotherm migration method, a new method, and a comparison has been made with the results from the finite difference method.

The problem exhibits instability and we have attempted to overcome this.

We showed that it is possible under certain circumstances to use a fairly small time-step in the Stehfest inversion method.

7.7.1 Contribution

The Laplace transform isotherm migration method has been applied to the example of solidification in a prism and it was apparent that this problem posed many difficulties. It particular, the need to choose a Laplace time

parameter which was not too small meant that positions of isotherms moving rapidly were not able to be accurately predicted. However we were able to demonstrate that the method is useful for isotherms moving slowly and there may be other problems in which solidification is slower which would be amenable to solution by this method.

We were also able to demonstrate that it is possible to use a small value for the Laplace time parameter in appropriate circumstances and that our results were improved in doing this.

Chapter 8

The use of multiple processors to solve diffusion problems using the Laplace transform isotherm migration method

8.1 Background

Traditionally software has been written for serial computation. This means that it is designed to run on a single computer with a single Central Processing Unit (CPU), the problem is broken down into a discrete set of single instructions which are executed one after another with only one instruction being executed at any moment in time. As time has evolved, problem solving has become increasingly sophisticated and complex, so that this model may well be considered to be inefficient. Walker and Houstis (2005) wrote in an editorial '... not only are problems becoming more complex in terms

of their expression and formulation, the techniques and resources needed to solve them are also becoming increasingly complex.' It was in response to this need that parallel computing was developed, although this happened more than twenty years before the remarks of Walker and Houstis (2005). In the simplest sense, parallel computing is the simultaneous use of multiple computer resources to solve a computational problem. This means that the problem is run using multiple CPUs, it may be broken into discrete parts that can be solved concurrently and each part is further broken down to a series of instructions which are executed simultaneously on different CPUs. In doing this we share the workload and the results then become available much more quickly.

In the early days, the ability to do this was limited to high-performance computers (supercomputers) which were huge, expensive and located in computer centres, thereby limiting their use to a few people. Today the low cost of personal computers means that it is possible for everyone to own his own facility, and by networking these together and logging in to other computers and sharing resources much greater problem-solving capacity is available than by the use of a single processor.

The first computers were developed in the 1940s and the mathematician von Neumann (1945) wrote a report describing a general purpose stored-program computing machine (the EDVAC). At the simplest level, the memory is used to store both program and data instructions and a CPU gets instructions and/or data from memory, decodes the instructions and then performs them sequentially. This system has the disadvantage that it can lead to a 'bottleneck', which seriously limits the effective processing speed when the CPU is continuously forced to wait for vital data to be transferred to or from memory. Although this has been overcome to some extent by new developments it is still recognised that a sequential mode of operation is not necessarily the most efficient.

In considering parallel computing methods, we have a number of different options which are available, and these are very clearly described by the taxonomy attributed to Flynn (1972), and we show these in figure 8.1.
The four classifications defined by Flynn are based upon the number of concurrent instructions (or control) and data streams available in the architecture:

1. Single Instruction, Single Data stream (SISD)
   This is a sequential computer which exploits no parallelism in either the instruction or data streams. Examples of SISD architecture are the traditional uniprocessor machines like PCs or old mainframes.

2. Multiple Instruction, Single Data stream (MISD)
   This is unusual due to the fact that multiple instruction streams generally require multiple data streams to be effective. However, this type is used for example on aeroplanes, which need to have several backup systems in case one fails. Some theoretical computer architectures have also been proposed which make use of MISD, but none have entered mass production.

3. Single Instruction, Multiple Data streams (SIMD)
   This is a computer which uses multiple data streams with a single instruction stream to perform operations which may be naturally parallelised. One drawback is that because the processors are executing instructions simultaneously, some processors may be idle for long periods of time.

4. Multiple Instruction, Multiple Data streams (MIMD)
   This is a system where multiple autonomous processors simultaneously execute different instructions on different data. Distributed systems are generally recognized to be MIMD architectures, either exploiting a single shared memory space or a distributed memory space. In this

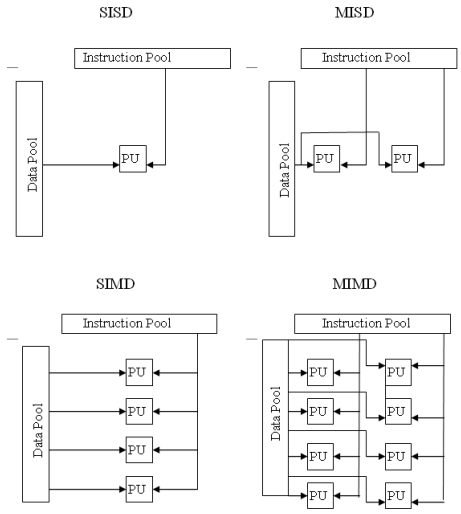

Figure 8.1: Diagram showing Flynn's Taxonomy where PU represents a processing unit

system, there is no global clock and the processors will not synchronise with each other unless specifically programmed to do so (Pacheco 1997).

It is the MIMD system which is used by most of the world's known powerful computers according the TOP500 website. The TOP500 project was started in 1993 to provide a reliable basis for tracking and detecting trends in high-performance computing. Twice a year, a list of the sites operating the 500 most powerful computer systems is assembled and released. The list contains a variety of information including the system specifications and its major application areas.

The MIMD is usually divided into two groups: shared memory and distributed memory. Shared memory refers to a block of memory that can be accessed by several different processors and is relatively easy to program since all processors share a single view of data and the communication between processors can be as fast as memory accesses to a same location. This is shown diagramatically in figure 8.2.

The issue with shared memory systems is that many CPUs need fast access to memory and will likely cache memory, a cache being a temporary storage area where frequently accessed data can be stored for rapid access. This has two complications:
CPU-to-memory connection becomes a bottleneck and cache coherence. Whenever one cache is updated with information that may be used by other processors, the change needs to be reflected to the other processors, otherwise some processors will be working with old data. Such coherence protocols can, when they work well, provide extremely high performance access to shared information between multiple processors. On the other hand they can sometimes become overloaded and become a bottleneck to performance. These systems are subdivided into Uniform Memory Access (UMA) and Non-Uniform Memory Access (NUMA). The UMA system is most commonly represented by Symmetric Multiprocessor (SMP) machines, which have identical processors and equal access and access times to memory. These are sometimes called CC-UMA - Cache Coherent UMA. Cache coherence is accomplished at the hardware level. The NUMA system is often made by physically linking two or more SMPs which may have direct access to each other's memory. Not all processors have equal access time to all memories and memory access across link is slower. If cache coherency is maintained, then this may also be called CC-NUMA - Cache Coherent NUMA

A distributed memory shown in figure 8.3, refers to a multiple-processor computer system in which each processor has its own private memory. This requires computational tasks to be distributed to the different processors for processing, after which the data must be reassembled. Memory addresses in one processor do not map to another processor. Because each processor has its own local memory, it operates independently and changes made to its local memory have no effect on the memory of other processors. Hence, the

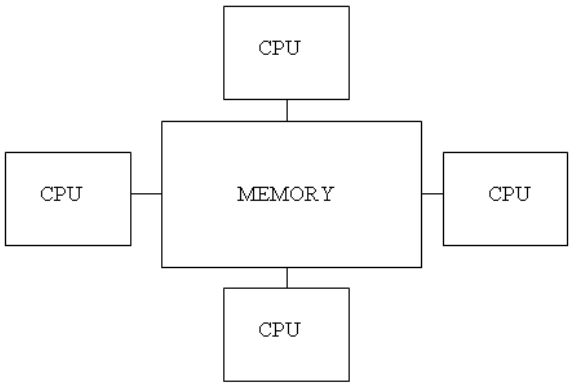

Figure 8.2: Diagram showing a shared memory system

concept of cache coherency does not apply. When a processor needs access to data in another processor, it is usually the task of the programmer to explicitly define how and when data is communicated. Synchronization between tasks is likewise the programmer's responsibility. Distributed memory systems fall into two classes:

The first of these is the Massively Parallel Processor (MPP) class, where the network and infrastructure are 'tightly coupled and specialised' for use in a parallel computer. These systems may contain many thousands of processors according to Mattson *et al.* (2004), but they are very expensive.

The second class is the cluster class. They are cost effective being composed of off-the-shelf computers being connected by off-the-shelf networks and are being used as a cost effective alternative to MPP.

We also have hybrid systems which are a combination of shared and distributed memory systems. The shared memory component is usually a cache coherent SMP machine. Processors on a given SMP can address that machine's memory as global. The distributed memory component is the networking of multiple SMPs which know only about their own memory,

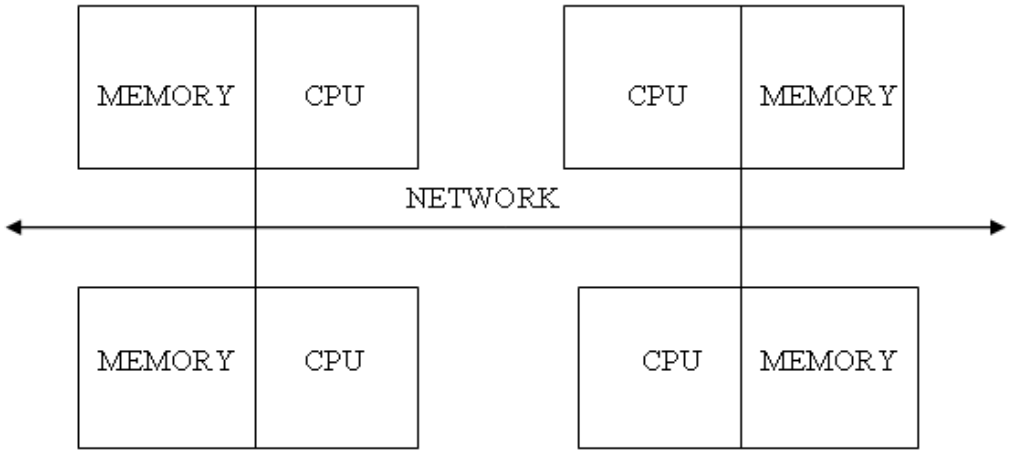

Figure 8.3: Diagram showing a distributed memory system

not the memory on another SMP. Therefore, network communications are required to move data from one SMP to another. These systems are being used increasingly.

8.1.1 Parallel Programming Models

Because of the variety of possible parallel architectures available, there is no one model which is suitable for every case. Therefore an appropriate programming method has to be selected which is most suitable to the requirements of the system. There are two recognised parallel programming models.

The first is a directives-based parallel programming language. High Performance Fortran (HPF) is one example, but OpenMP is the most widely used. Directives, which appear as comments in the serial code, tell the processor how to distribute the data and work across the processors. This method is for implementation on shared memory architectures.

The second method is Message Passing Interface (MPI), in which messages are passed to send and receive data between processes and each process

has its own local variables. It can be used on either shared or distributed memory architectures.

Each method has advantages and disadvantages. OpenMP is easier to program and debug than MPI and the directives can be added incrementally so that the program is gradually parallelised while it can still run as a serial code. The serial code statements usually do not need modification and code is easier to understand and maybe more easily maintained, but OpenM can be run only in shared memory computers and requires a compiler that supports OpenMP. It is mostly used for loop parallelisation. On the other hand MPI runs on either shared or distributed memory architectures and can be used on a wider range of problems than OpenMP. Each process has its own local variables. Probably of most importance, distributed memory computers are less expensive than large shared memory computers. However, it requires more programming changes to go from a serial to a parallel version, can be harder to debug and its performance is limited by the communication network between the nodes.

8.1.2 MPI

For our work we have access to a cluster of Unix machines at Greenwich University, which is a distributed memory system and implements MPI as the programming model.

MPI was designed by a broad group of parallel computer users, vendors, and software writers and the standardisation process began with the Workshop on Standards for Message Passing in a Distributed Memory Environment, in April 1992, in Williamsburg, Virginia, (Walker 1992). MPI 1 followed from 1993-1995 meetings; MPI 2 from a 1997 forum. Gropp, Lusk and Skjellum (1994) were among those at the forefront of development. The MPI standard defines a set of library routines that are called from within programs written in sequential language such as C, C++ or Fortran. There

are several MPI implementations available, but the two most commonly used are Local Area Multicomputer (LAM)/MPI and MPICH. These can be downloaded and implemented free of charge. For our work we use NT-MPICH in conjunction with Compaq Visual Fortran. The MPI interface consists of nearly two hundred functions but in general most codes use only a small subset of the functions. Full details of using MPI, together with examples of how to employ the pre-defined functions in MPI are given by Pacheco (1997).

8.1.3 Designing Parallel Programs

When developing a parallel program, if we are starting from a sequential code, the first step is to consider whether it is suitable to run in parallel. We need to look for blocks of code which are being repeatedly run sequentially, especially where the progress is being held up by the need to wait until a particular set of data has been processed before being able to proceed.

One of the first steps in designing a parallel program is to break the problem into discrete "chunks" of work that can be distributed to multiple tasks. This is known as decomposition or partitioning.

There are two ways to partition computational work among parallel tasks: domain decomposition and functional decomposition. In domain decomposition the data associated with a problem is decomposed. Each parallel task then works on a portion of the data. An example of this is in solving a differential equation using the Laplace transform. We might require the solution at several different times and these could be grouped so that each processor calculates the solution for a set of times simultaneously. Davies and Crann (2007) describe this process. In functional decomposition the focus is on the computation to be performed rather than on the data manipulated by the computation. The problem is decomposed according to the work that must be done, as a collection of independent computational

tasks that can be executed concurrently. Ideally the data required for the tasks should not be dependent on much data from other tasks, as communication between processors will slow the calculation down. Andrade *et al.* (2002) provide an example of a process involving functional decomposition.

8.1.4 Communication

The issue of communication is important when designing parallel programs, because whenever communciation between processors is needed, the running time for the program will be slower. Therefore we need to minimise the amount of communication between processors so that the speed of performance is not compromised. Some problems can be decomposed and executed in parallel with virtually no need for tasks to share data. These types of problems are often called embarrassingly parallel because they are so straight-forward. Very little inter-task communication is required. However, most programs do need message passing between processors to some degree and communications frequently require some type of synchronization between tasks, which can result in tasks spending time 'waiting' instead of doing work.

8.1.5 Measuring performance

Having designed our parallel program, we then would like to measure the increase in speed compared with using a sequential program. In our work, we use the parameter *speed-up*, which is the ratio of the sequential runtime to the parallel runtime. If t_1 is the time to run the program on one processor and t_p is the time for the last of the p processors to complete, then the speed-up S_p is given by

$$S_p = \frac{t_1}{t_p} \tag{8.1}$$

For values of $p > 1$ usually $0 < S_p < p$. If $S_p = p$ then the program is said to have linear speed-up. This is rare, because the overhead from

communication in the program adds extra time to the computation which does not occur in a sequential code.

8.2 Application to the Laplace transform isotherm migration method

8.2.1 An earlier problem

The implementation of the Laplace transform in time on a distributed memory architecture has been discussed previously by Davies *et al.* (1997). They applied the Laplace transform to the diffusion problem, thereby reducing the problem to a modified Helmholz equation in the transform space which they solved in a parallel environment using five different methods. The method was illustrated by solving the two-dimensional heat conduction problem

$$\nabla^2 u = \frac{1}{\alpha}\frac{\partial u}{\partial t} \quad -1 < x < 1, \quad -1 < y < 1 \tag{8.2}$$

subject to the boundary conditions

$$u(-1, y, t) = u(x, -1, t) = u(1, y, t) = u(x, 1, t) = 1$$

and the initial condition

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

The numerical Laplace transform method was used to solve the problem with a value of $M = 6$ for the number of transform parameters, as suggested by Crann (1966) and Moridis and Reddell (1991c).
The solution methods were:

1. The finite difference method (FDM) which has been described in chapter 2, and in which a uniform grid on the square region was defined and the usual five-point formula for the Laplacian (Smith, 1978) was used to define a Gauss-Seidel formulation.

2. The finite element method (FEM) also previously mentioned in chapter 2.

3. The boundary element method (BEM) using 68 linear elements with eight-point Gauss quadrature to develop the system matrices. The singular integrals were managed using Telles's transformation method (Telles 1987 and Telles and Oliveira 1994).

4. The method of fundamental solutions (MFS) which has been referred to in chapter 2 and its solution requires a knowledge of the fundamental solution for the modified Helmholz equation which is given by

$$u^* = \frac{1}{2\pi} K_0 \left(\sqrt{\frac{\lambda}{\alpha}} R \right)$$

where K_0 is the modified Bessell function of the second kind, and R is the distance between a fixed source point and a variable field point. The modified Helmholz equation has the form

$$\nabla^2 u - \gamma^2 u = f$$

and under the Laplace transform the heat equation becomes

$$\nabla^2 \bar{u} = \frac{1}{\alpha} \left(\bar{u} - \lambda u_0 \left(x \right) \right)$$

and so has the form of a modified Helmholtz equation. Referring to equation (2.19) in chapter 2, we now have a similar expression to solve with a new fundamental solution, this being effected as described in chapter 2.

5. Kansa's multiquadric method (KMM) which has some similarity to the method of fundamental solutions in that it involves interpolation with points outside the bounded region on which the equation is to be solved, but these points are on a grid system, rather than on a circle surrounding the region.

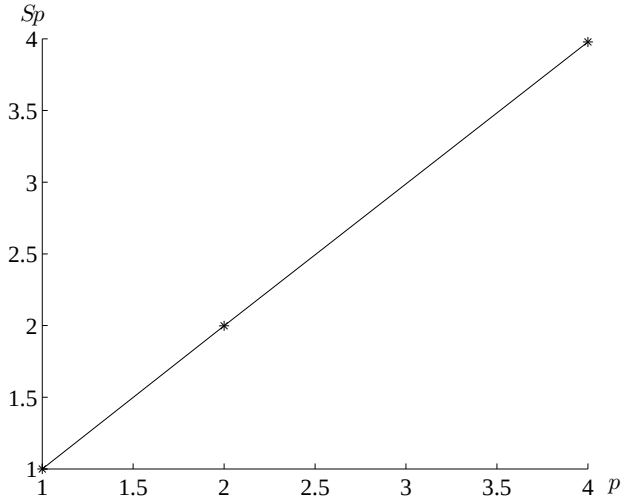

Figure 8.4: Speed-up for the method of fundamental solutions.

These problems were solved on a network of processors comprising four T800 transputers and the solution was sought at eight time values:
$t = 0.1, 0.2, 0.5, 1, 2, 5, 10, 20.$

Table 8.1: cpu times for the five different methods.

No of proc's	FDM	FEM	BEM	MFS	KMM
1	2537	2617	923	92.3	73.6
2	1269	1309	464	46.2	36.9
4	634	654	233	23.2	18.8

Table 8.2.1 shows the computing times for the five different methods. Clearly, the finite difference and finite element methods require a much greater cpu time, but in this instance, this was not under scrutiny, merely the speed-up of the individual methods. Figure 8.4 shows the speed-up for the method of fundamental solutions, a typical example since it was found that in each case the speed-up was indistinguishable from any of the others; they all exhibit almost linear speed-up.

For our work, we use the University of Greenwich parallel system, which comprises three DEC4100 servers with eighteen single alpha processors.

8.2.2 The options for parallel implementation of the Laplace transform isotherm migration method

In using the numerical Laplace transform method there are choices to be made in the division of work among the processors. Throughout our work we have used eight Stehfest weights for our Laplace inversion and we could construct the code so that the weights are divided between the different processors, and look at the speed-up this produces.

We could also approach the problem by evaluating the solution at 16 different times, which will be shared among the processors, as in the work of Davies *et al.* (1997), Crann *et al.* (1998) and Crann *et al.* (2007).

We have shown in chapter 5, that for problems involving phase change we implement the Laplace transform method in a different way, so that the initial condition is updated for each time value. This means that we cannot proceed in the manner of Davies *et al.* (1997) for such problems, because the initial condition at each stage requires knowledge of the positions of the isotherms at the previous time and so for such problems we take the first approach.

Example 8.1

We refer to example 4.1. In this example we solve the problem of a bar of unit length, initially at zero degrees, and from time t_0 the temperature of the boundary at $x = 0$ is held at a value of 10. This is a one-dimensional problem with no phase change, and we use the isotherm migration method together with the Laplace transform to solve it. Because we are interested in measuring time differences which might be quite small, we increase the number of isotherms to 21, that is, we take them at intervals of 0.5 rather than 2, to get a more accurate picture of any time differences in the calculation.

We solve the problem for 16 times from $t = 0.2$ to $t = 1.7$ in increments of 0.1, remembering that we start the calculation at a small time, $t = 0.1$. We use the UNIX clock to measure the time for each processor to finish its

work. Because the calculation involves a Gauss-Seidel iterative procedure, we cannot say that the processors have identical workloads as some calculations require more iterations to converge than others, and so we take the end point of the calculation to be when the last processor has completed its task. We also eliminate the necessity of each processor having to write its results to the output screen, to avoid the possibilty of a processor having to wait to write, which would introduce inaccuracies into the timing.

We use the processors to perform the calculation in the Laplace space subroutine in the program. As we use a Stehfest inversion with 8 parameters, we compare the speed-up using 1,2 4 or 8 processors. When using 1 one processor, all eight sets of Stehfest parameters, are used by this processor, that is, in effect the sequential case. When using two processors, we allocate four Stehfest parameters each to the two processors, and these two processors pass their results to the main processor to continue the calculation. The case using four processors means each now has two sets of Stehfest parameters and with eight processors, one set of Stehfest parameters. At the end of the calculation for each time, the main processor must broadcast the new positions of the isotherms to the other processors, as these become the new initial conditions for the next time step. With this method, just a small part of the overall calculation is being performed in parallel, and there is considerable message passing between the processors, and that the processors are not being used in an efficient way as they will be idle while the main processor collates the results, but this seems to be unavoidable for this manner of working.

We show the results in figure 8.8 and table 8.2. We see that there is an initial speed-up when using two processors, but this is less marked for four processors and the speed-up falls back when using eight processors. Although at first sight this might seem unexpected, we remember that much of the calculation is being done in a sequential manner and that the more pro-

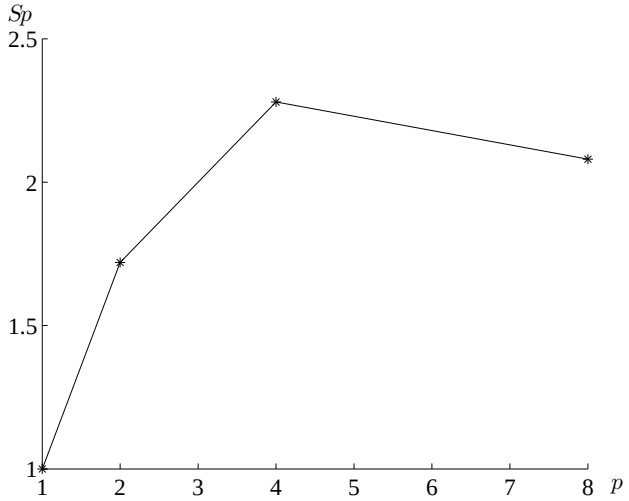

Figure 8.5: Speed-up for the Laplace transform isotherm migration method in one dimension allocating Stehfest values to varying numbers of processors using 21 isotherms in example 8.1

cessors in use, the more message passing involved, which would contribute to a time delay.

Table 8.2: Speed-up when allocating Stehfest parameters to varying numbers of processors with 21 isotherms in example 8.1

p	Sp
1	1.00
2	1.72
4	2.28
8	2.08

Example 8.2

We now repeat example 8.1, but increase the number of isotherms and grid points to 41 each.

We see that increasing the number of calculations leads to an improvement in speed-up and that increasing the number of processors reduces the calculation time. This is because the time used for message passing is small compared with the time spent on calculation. We conclude that while there is some benefit in using multiple processors for a simple problem where only

Table 8.3: Speed-up when allocating Stehfest parameters to varying numbers of processors and 41 isotherms in example 8.2

p	Sp
1	1.00
2	1.71
4	2.71
8	4.14

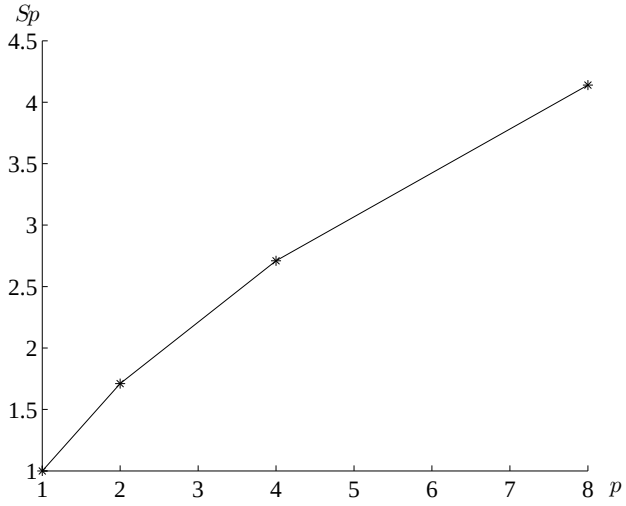

Figure 8.6: Speed-up for the Laplace transform isotherm migration method in one dimension allocating Stehfest values to varying numbers of processors and 41 isotherms in example 8.2

a small part of it can be properly calculated in parallel, the method is very effective when the problem is complex and requires many calculations to be carried out in a partially parallel way.

Example 8.3

Since we have used a problem with no phase change in examples 8.1 and 8.2, this would be amenable to solution using the method of Davies *et al.* (1997), that is, the sharing of the time values in which we are interested, rather than the Stehfest parameters. For completeness, we carry this out in this example, bearing in mind that we shall not be able to do the same for problems with phase change. The number of isotherms is the same as that

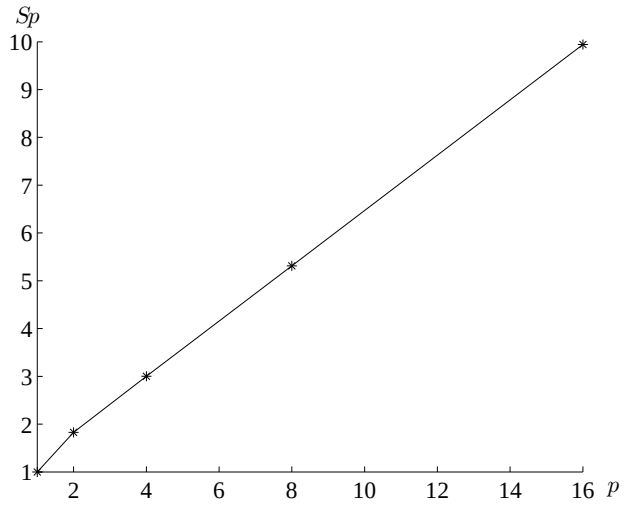

Figure 8.7: Speed-up for the Laplace transform isotherm migration method in one dimension with no phase change with 21 isotherms in example 8.3

in example 8.1. Because the processors are shared with other users and may be running several codes we find slight variations in the times for the last processor to finish its work each time we run the code. To get a reasonable estimate of calculation time we run each set of results ten times and take an average value. Our results are shown in table 8.4 and figure 8.7.

Table 8.4: Speed-up for the Laplace transform isotherm migration method in one dimension with no phase change in example 8.3

p	Sp
1	1.00
2	1.83
4	3.00
8	5.31
16	9.94

We see from figure 8.7 that between 2 to 16 processors there is a relationship between speed-up and the number of processors, although it is not linear according to our definition. When two processors are used rather than one, the time taken is not exactly halved. This may be due to the two processors each having to access data before starting their work, which does

not occur when only one process is used. This is a "small problem" with few unknowns and so interprocessor communications may be relatively important. Nevertheless, we conclude that when several processors are available, this method could well be efficient in saving time in calculation.

8.3 A Stefan problem in one dimension.

Example 8.4

We now consider a Stefan problem and use example 5.1 discussed in chapter 5. We saw that for Stefan problems we need to update the initial values at each time step, before finding the new positions of the isotherms. For this reason we cannot use a total parallel implementation as in example 8.3, by allocating specific time values to specific processors since each the calculation at each time will depend upon the positions of the isotherms found at the previous time step. However, we can make use of having several processors available, by sharing the work done in Laplace space as in examples 8.1 and 8.2. We therefore, allocate to each processor particular values of λ and the Stehfest weights, and after convergence in the Gauss-Seidel procedure has been achieved, the results are passed to one processor, which sums them and tests for convergance of the non-linear loop. We use 21 isotherms in this calculation.

Table 8.5: Speed-up for the Laplace transform isotherm migration method in one dimension for a Stefan problem in example 8.4

p	Sp
1	1.00
2	1.60
4	2.28
8	2.91

We tabulate the results for this example in table 8.5 and plot these in figure 8.8. The trend appears to be that speed-up increases as the number

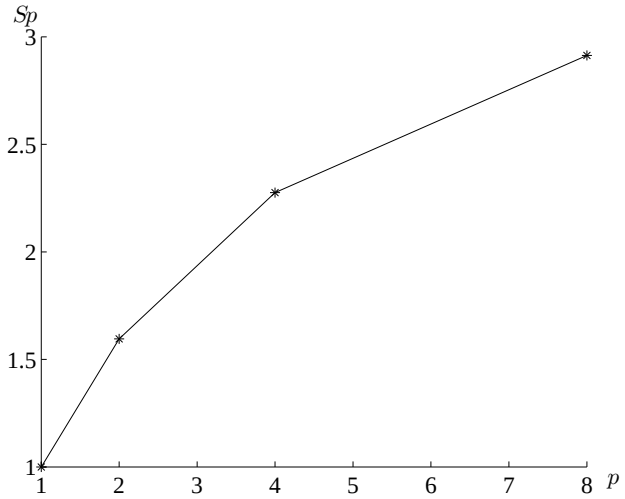

Figure 8.8: Speed-up for the Laplace transform isotherm migration method in one dimension for a Stefan problem in example 8.4

of processors increases, but it is not linear according to our definition. The time spent passing data between processors is reflected in the rate of increase of speed-up.

8.4 A Stefan problem in two dimensions

Example 8.5

We refer to example 7.3, the solidification of a square prism of liquid. We have already described how to solve the problem using the Laplace transform isotherm migration method and that a time step of $t = 0.001$ gives results which are to be similar to those using the finite difference method. As in example 8.4, because we have a Stefan condition, we need to use the Laplace transform method in a different way, by updating the initial conditions at each step. We cannot use each processor to evaluate a set of solutions at a number of times independently of the other processors, since the solutions at any time are needed as the initial conditions for the solution at the next time value.

Therefore, as in examples 8.1 and 8.2, we distribute the Stehfest parameters evenly between one, two, four or eight processors. We have seen in chapter 7 that the solution to the problem requires a significant amount of geometrical computation and the work in Laplace space is a very small portion in comparison to the overall work. We use the value S_p given by equation (8.1) as a measure of speed-up.

From the results shown in table 8.6 and figure 8.9 we see a reasonable speed-up when using two processors rather than running the program sequentially on one processor. Using more processors shows an improvement in speed-up, but the time saved becomes less. Clearly as more processors are involved, there is more message passing, and the nature of this problem, with its non-linearity and the need to fit curves at the lines of symmetry means that much of the work has to be done in a sequential way. Therefore because it is not a 'true' parallel problem, in the sense that all the processors carry out exactly the same tasks, we might expect the speed-up not to be exactly linear.

Table 8.6: Speed-up shown by using multiple processors to calculate the time to freeze a square prism of liquid in example 8.5

p	Sp
1	1.00
2	1.48
4	1.69
8	1.99

8.5 Conclusions regarding the Laplace transform isotherm migration method solution in a parallel environment

We have considered several examples in which we have used the Laplace transform isotherm migration method in a parallel environment as a tool to

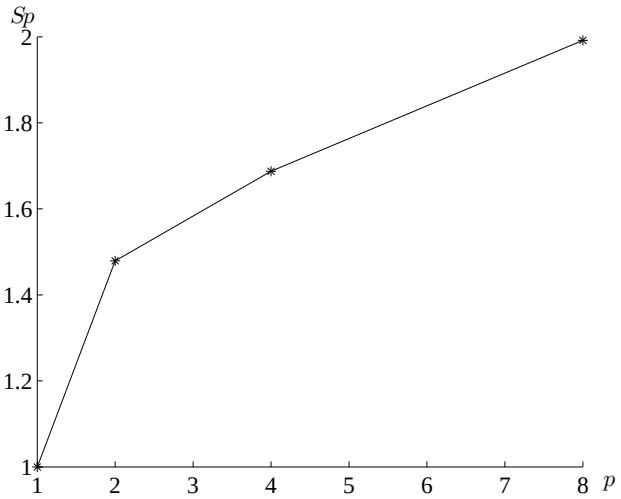

Figure 8.9: Speed-up for the Laplace transform isotherm migration method for a two-dimensional Stefan problem in example 8.5

solve both one-dimensional and two-dimensional problems. Our results indicate that there is always some benefit to be gained in a multiple-processor method.

In cases where there is no phase change, the problem may be solved independently at required times by several processors and although speed-up is demonstrated, the relationship between speed-up and number of processors in not linear. Provided the problem is of such complexity that the time spent on calculations far outweighs the time spent on message passing, speed-up is improved.

8.6 Summary of Chapter 8

This chapter begins with a discussion of how parallel computing was developed in response to both the advance in hardware technology and the need for more complex problems to be solved.

Flynn's taxonomy (Flynn 1972) describes the four classifications of systems of computing, and it is the multiple instruction, multiple data model

upon which we focus. We mention the advantages and disadvantages of the different parallel programming models, but conclude that MPI is accessible to more users than Open MP and HPF, because it is suitable for use by an individual with a PC and access to a remote cluster.

We present new work, solving five different examples with the Laplace transform isotherm migration method in a parallel environment, showing that there are choices of how the work may be shared among the processors available and that the use of multiple processors always results in some time saving, although how much depends upon the relative times spent on calculation and message passing. In problems with phase change, our choice of work-sharing by the processors is limited by the constraints of needing to update the initial conditions at each time step.

8.6.1 Contribution

We have solved several examples with the Laplace transform isotherm migration method in a parallel environment and we have shown that for problems where no phase change is involved there are different models for division of work, but when phase change is involved, the choice of how to share the workload is limited.

We have shown that benefit in the form of speed-up is achieved in a parallel environment but is dependent on the amount of time spent on message passing compared with the time spent on calculations.

We had hoped to demonstrate that our new method would be suitable for a parallel implementation and we have shown this to be the case.

Chapter 9

Conclusions and future work

9.1 Summary of thesis

In the final chapter an assessment is made as to whether the objectives described in chapter 1 have been achieved and the contribution this work has made towards the chosen topic is evaluated. Further work which might be implemented is also suggested.

Chapter 2 provided a derivation of the heat equation and showed that there are many methods to choose from to find its solution, each having its own merits but with a suitability depending on the particular problem.

The first aim was to examine the isotherm migration method described by Crank and Phahle (1973), a method appearing less frequently in recent years, to see its advantages and disadvantages when applied to simple problems. This would help in realising the limitations of the method.

The method was applied to the solution of conduction problems in one dimension at first, this in itself being a new application, since we have only found references to its use as a tool for solving problems involving phase change, that is, cases involving melting or freezing. It was noticed that in carrying out the isotherm migration mapping, the linear heat equation is exchanged for a non-linear equation. Furthermore, in considering the

heating of a rod at zero degrees, no isotherms other than that having a temperature of zero degrees exist at the initial time so that a method to generate the positions of some isotherms a short while after the initial time is needed. This presents no difficulty if an analytic solution is available for the problem, but could lead to inaccuracies if an alternative numerical method were to be used which might introduce some error. It was noted that when using the isotherm migration method with a finite difference solver, there is a constraint on the size of the time step, and this means that initially only very small times steps may be taken, although as the solution proceeds these are allowed to increase.

Example 3.2 showed a difficulty when considering the heating of a rod where each end was held at a constant temperature as some of the isotherms existed for only a finite time before disappearing. In addition, some isotherms did not have a unique position.

Problem: Isotherms may be transient and have non-unique positions.

Decision: Consider the example as two simpler cases. The isotherm migration method equation is non-linear and the solutions to the two cases may not be added directly, but the results may be used to produce a plot of each in x, t-space and since the heat equation is linear these two functions may be added together.

Problem: Positions for the isotherms after a small time are needed to start the isotherm migration method and a numerical method may introduce errors at the initial stage, which may then increase during the solution.

Decision: The effect of errors in the initial data should be considered and this was carried out in example 3.3 by introducing known errors into the analytic starting values and evaluating the outcome

It was found that the isotherm migration method was quite tolerant of errors, tending to the correct solution in the long term. This supported the belief that the method was robust.

The method was tested in examples where α, the thermal diffusivity, was not constant, and it was found that although there were no analytic solutions, the results followed the expected trends.

The Laplace transform was then applied to the isotherm migration method and examples were solved in one dimension. This was new work and led to other decisions to be made and problems to overcome.

Problem: A method for inverting the Laplace transform is required.

Decision: Use the Stehfest numerical inversion process which has been shown by Crann (2005) to be a reliable and easy method to operate. A choice of the number of Stehfest parameters needs to be made, and following the suggestion of Davies and Crann (1998), $M = 8$, where M is the number of Stehfest parameters, is taken as this is believed to give accuracy without performing more calculations than necessary.

Problem: The mapped isotherm migration equation is non-linear and the Laplace transform may only be applied to linear equations.

Decision: Use the direct iteration method suggested by Crann (2005), substituting the numerical values for the non-linear terms obtained at the previous time step. Iterations are then performed until the required accuracy is achieved.

The results obtained for the solution to simple heating examples using the Laplace transform isotherm migration method are compared with those using the isotherm migration method alone, and it may be seen that both methods produced the same accuracy. It is possible to take larger time steps with the Laplace transform isotherm migration method than with the isotherm migration method, but this has to be weighed against the increased number of calculations needed for the Stefhest numerical inversion method.

The Laplace transform isotherm migration method was then used to solve one-dimensional problems with phase change, so-called Stefan problems, and it was found that because the position of the moving boundary was

described by an equation involving a time derivative, the Laplace transform could not be applied in the normal way, that is keeping the initial condition the same throughout calculation, as a lag developed in the movement of the boundary of the phase change.

Problem: If the initial condition is fixed in the Laplace transform expression throughout the calculation which is the usual manner of proceeding, the positions of the isotherms fall behind their true values and this becomes more marked with increasing time. This is due to the derivative term in the equation at the moving boundary.

Decision: Updating the initial condition at every time required overcomes this problem.

The results using this modification compared well with the analytic solution.

At this stage an analysis was performed on the number of calculations required for the Laplace transform isotherm migration method compared with the isotherm migration method and it was found that the former increased the number of calculations by approximately a factor of seventeen. However, this was not interpreted as meaning that our method lacked efficiency as the intention was to show later on that the method would be amenable to solution in a parallel computing environment.

The method was shown to be useful in solving a phase change problem with convective boundary conditions. In this case there is a need to deal with isotherms which appear during the process of the problem and the model of Gupta and Kumar (1988) was followed in which the new temperature of the wall at which convection is taking place is estimated at each chosen time and it is decided whether any new isotherms have been introduced. These are incorporated at the next step. A value for the Biot number was required and since this can take a wide range of values, it seemed logical to take the value suggested by Gupta and Kumar (1988) for our first example. The

results in example 5.3 showed the expected trend and example 5.4 confirmed that the method was appropriate.

The next stage of the work was to examine the solution of two-dimensional problems, in the first instance following the method of Crank and Gupta (1975), involving an example where a square region of water is frozen by keeping one face at a temperature below the fusion temperature. A description is provided of how to carry out the isotherm migration mapping in two dimensions and the differences arising from this mapping to the one-dimensional case are noted. A new second derivative term appears in this mapping and, in general, this requires some form of interpolation in its evaluation. It was also noted that some form of quadratic curve fitting would be necessary to position the isotherms on the boundaries and this indicated that the two-dimensional case had complexities which might not have previously been envisaged. It was found that to avoid instability in this example the isotherm migration method required a time step some five times smaller for its solution using the finite difference method than in the one-dimensional case and this made the Laplace transform isotherm migration method increasingly attractive, since it would need less intermediate time solutions, and so the difference in the number of calculations between the methods might not be so great. The Laplace transform isotherm migration method performed well in terms of accuracy of numerical solutions and the plan was to set this in a parallel computing environment later.

Following on from this, a problem solved by Crank and Gupta (1975) was considered, in which they tracked the movement of isotherms when a prism of liquid is solidified. Their method employed the isotherm migration method with a finite difference method. A review of their work showed some of the limitations and difficulties of the method. The first problem is that following the two-dimensional mapping the resulting isotherm equation may not be single-valued. Furthermore, some isotherms may not exist at all the

grid points. An example, described by Crank and Gupta (1975), shows how these difficulties may be overcome by making use of the symmetry of the prism, so that a region bounded by $x = 0$, $y = 1$ and the line $y = x$ is used. Because the isotherms move along a normal, the contour of the isotherm crossing the line $y = x$ is completed by fitting a circle whose centre lies on $y = x$. At the axis $x = 0$ the relevant equations are undefined and it was necessary to resort to curve fitting again, this time with a quadratic function. As suggested by Crank and Gupta (1975), Poot's (1962) one-parameter method was chosen to find the necessary starting values. This method presented no difficulties, was quite easy to follow and we were able to produce similar numerical results to Crank and Gupta (1975).

The new work used the Laplace transform isotherm migration method to solve the solidification process in a prism. There was a problem with the evaluation of the interpolation for the second derivative term, which occurred because in certain circumstances the values needed were undefined as the isotherms required were not on the gridlines involved in that calculation. This had not been a difficulty when using Crank and Gupta's method and so it appears to be linked to the need to take Stehfest time values of $T = 0.1$, in equation (4.8), in keeping with the requirements for the Stehfest inversion method, but this was a large step relative to the total time for freezing.

Problem: If $T = 0.1$ is chosen in equation (4.8) the process breaks down, which is probably due to the fact that some of the isotherms have moved a significant distance and may no longer exist at the grid points needed for the calculation.

Decision: Although it has been suggested by Crann (2005), that in general, values of T less than 0.1 may give unreliable results, it is necessary to try such values in this case.

It was found that a value of $T = 0.03$ gave reasonable results while using a finer mesh failed to provide any improvement. It was decided to

try a very small time value of $T = 0.001$ and the solutions were improved, indicating that taking a very small time value might be preferable, but since the Stehfest parameter involves division by T, the choice is limited.

It appears that while the Laplace transform isotherm migration method performs well in one dimension, its value may be limited in the two-dimensional setting.

The last piece of work was to put the Laplace transform isotherm migration method into a parallel environment, using multiple processors to evaluate the solution. We saw that in the Laplace transform isotherm migration method, for problems with no phase change, the division of work could be such that either the required times could be shared among the processors, or the calculations in the Stehfest loop could be divided giving each processor a set of parameters and weights and having one processor collocate their results. Speed-up was demonstrated in both cases, but in the second method, the efficiency was not so great due to more time spent message passing. Using a finer mesh confirmed that when the time spent on calculations was relatively large compared to the time spent message passing, greater efficiency is achieved.

Problem: There is a choice to make as to how to divide the work among the available processors: share the times at which solutions are required or share the calculations performed within Laplace space.

Decision: Where possible choose the option with the least amount of message passing, in this case share the times.

When solving problems with phase change, the sharing of the Stehfest calculations is the only suitable way of dividing the work, since the initial values must be updated at each stage. We were able to show speed-up in both the one and two-dimensional cases, leading us to conclude that it is worthwhile to solve these problems using a message passing interface on multiple processors; the adjustments needed to the code are relatively

simple.

9.2 Research objectives

Our objectives described in chapter 1 may be summarised as follows:

1. To look at the isotherm migration method to establish its advantages and disadvantages.

2. To develop the method further by the use of the Laplace transform method and to test this method by solving problems in one dimension which did not involve phase change.

3. To extend the use of Laplace transform isotherm migration method by solving phase change problems both in one and two dimensions.

4. To examine the performance of the Laplace transform isotherm migration method in a parallel environment to find out whether there was a suitable and sensible way of division of work and to establish whether there was any benefit to be gained from the use of a parallel environment.

In the following subsections we consider each objective and demonstrate that it has been properly addressed.

9.2.1 To look at the isotherm migration method to establish its advantages and disadvantages

Although it is usual to see the isotherm migration method used for problems involving phase change, in chapter 3 we considered it as a tool for solving problems in which there is no phase change. We examined a variety of examples, among which we identified a difficulty in cases where isotherms may not have a unique position or may disappear and how to overcome this,

the effect of introducing errors into the initial data and the case when α, the thermal diffusivity is not constant, and we concluded that the method was simple to operate, robust and tolerant of errors.

9.2.2 To develop the method further by the use of the Laplace transform method and to test this method by solving problems in one dimension which did not involve phase change

We applied the Laplace transform to the isotherm migration method in chapter 4 and solved several examples including cases where α, the thermal diffusivity, is non-linear and found the method produced acceptable results whose accuracy compared well with those using the isotherm migration method in the usual way.

9.2.3 To extend the use of Laplace transform isotherm migration method by solving phase change problems both in one and two dimensions

It is shown in chapter 5, that due to the derivative term in the equation describing the position of the melting front, the Laplace transform method needed to be modified. The initial values must be updated at each stage. With this detail in place, the Laplace transform isotherm migration method performed as well as the isotherm migration method, although it does require a greater number of calculations and we also showed that it could be used to solve examples with convective boundary conditions. Chapters 6 and 7 showed how the method may be adapted to solve problems in two dimensions and it was here that many problems were encountered and attempts were made to overcome these. To a large extent, this was possible and other examples with a longer total freezing time may not exhibit the instabilities apparent in the chosen problem. Certainly, even without the use of the

Laplace transform, new complexities showed up in the two-dimensional case and this may indicate why the method is less frequently used now.

9.2.4 To examine the performance of the Laplace transform isotherm migration method in a parallel environment to find out whether there was a suitable and sensible way of division of work and to establish whether there was any benefit to be gained from the use of a parallel environment

In chapter 8 we considered several examples, both with and without phase change, and demonstrated that it is possible to use the Laplace transform isotherm migration method in a parallel environment. In the case of problems with phase change the way of dividing the work is restricted due to the need for updating at each stage, but we were still able to show speed-up in all cases and concluded that there is benefit in using a parallel environment and that the benefit is increased whenever time spent on message passing is small compared with that spent on calculation.

9.3 Published work

We list here our publications and briefly highlight the content referring to the relevant section.

1. Davies A J, Mushtaq J, Radford L E and Crann D (1997) The numerical Laplace transform solution method on a distributed memory architecture, *Applications of High Performance Computing V,* 245-254, Computational Mechanics Publications.
   Paper on the parallel implementation of the Laplace transform method with five different solvers.
   Subsection 8.2.1

2. Davies A J and Radford L E (2001), A method for solving diffusion-type problems using separation of variables with the finite difference method, *Int. J. Math. Educ. Sci. Technol.*, **32**, 449-456.

Paper on a solution of the heat equation.
Subsection 2.2.7

9.4 Future research work

We feel that our objectives set out in chapter 1 have been met, but on the way, we encountered several difficulties and we address these in our ideas for future work.

1. The example of the Stefan problem with a time-dependant Neumann condition described by Kutluay and Esen (2004) could be solved using the Laplace transform isotherm migration method. This should not be difficult as it is an extension to our work in one dimension and should require the time dependent boundary condition to be included in the Laplace transform.

2. An alternative to fitting a circle across the line $y = x$ in the solution of the freezing in a prism could be the fitting of a parabola. Provided the isotherms move in a normal direction any suitable curve could be chosen. In this example, for small times, the freezing front has the shape of a square with rounded corners, the shape tending towards a circle as freezing approaches completion. Therefore it would seem reasonable to compare the solutions found using a parabola.

3. A disadvantage in the two-dimensional case is the necessity to invert from Laplace space in order to perform the curve fitting at $x = 0$ and $y = x$. It is difficult to see how this could be overcome, as we found no way around this. It would seem that it would be necessary to begin the problem afresh to see if it could be solved.

4. There may be a better grid system. Crank and Crowley (1979) described a solution method for the isotherm migration method along

orthogonal flow lines. To implement this, the isotherm migration equations are reformulated in cylindrical polar co-ordinates. The isotherms themselves are divided into segments and using a local co-ordinate system, the local centre and radius of curvature of each isotherm segment is recalculated at each time interval, so that the motion of each isotherm along the normal to itself may be found. This method still depends on geometry and so if our method were used, would still involve reverting to geometric space to calculate the local co-ordinates. However the effort might be worthwhile if it gave improved accuracy in solutions.

Chapter 10

References

Abramowitz M and Stegun A (1972) *Handbook of Mathematical Functions*, Dover.

Alarcón E, Martin A and Paris F (1979) Boundary elements in potential and elasticity theory, *Computers & Structures*, **10**, 351-362.

Andrade H, Kurc T, Sussman A and Saltz J (2002) Exploiting Functional Decomposition for Efficient Parallel Processing of Multiple Data Analysis Queries, *UM Computer Science Department Technical Report*, CS-TR-4404.

Brebbia C A and Dominguez J (1977) Boundary element methods for potential problems, *Appl. Math. Modelling*, **1**, 7.

Brebbia C A and Dominguez J (1989)*Boundary elements, an introductory course,* McGraw-Hill/Computational Mechanics Publications.

Brebbia C A (1978) *The boundary element method for engineers*, Pentech Press.

Brunton I L and Pullan A J (1996) A semi-analytic boundary element method for analytic problems, *Engng. Anal. Bdry. Elem.*, **18**, 253-264.

Carslaw H S and Jaeger J C (1959) *Conduction of Heat in Solids,* 2nd ed., Oxford University Press.

Chernous'ko F L (1970) Solution of non-linear heat conduction problems in media with phase changes, *Int. Chem. Eng.,* **10**, 42-48.

Chun C K and Park S O (2000) A fixed-grid finite difference method for phase change problems, *Num. Heat Trans. Part B*, **38**, 59-73.

Courant R and Hilbert D (1953) *Methods of Mathematical Physics*, Vol 1, New York: Interscience Publishers.

Crank J (1957) Two methods for the numerical solution of moving boundary problems in heat flow, *Q. Jl. Mech. Appl. Math.*, **10**, 220-231.

Crank J (1979) *The Mathematics of Diffusion,* Oxford University Press.

Crank J (1984) *Free and moving boundary problems*, Oxford University Press.

Crank J and Crowley A B (1978) Isotherm migration method along orthogonal flow lines, *Int. J. Heat Mass Transfer*, **21**, 393-398.

Crank J and Crowley A B (1979) On an implicit scheme for the isotherm migration method along orthogonal flow lines in two dimensions, *Int. J. Heat Mass Transfer*, **22**, 1331-1337.

Crank J and Gupta R S (1972a) A Moving Boundary Problem Arising from the Diffusion of Oxygen in Absorbing Tissue, *J. Inst. Math. Applics.*, **10**, 19-33.

Crank J and Gupta R S (1972b) A Method for Solving Moving Boundary Problems in Heat Flow using Cubic Splines or Polynomials, *J. Inst. Math. Applics.*, **10**, 296-304.

Crank J and Gupta R S (1975) Isotherm migration method in two dimensions, *Int. J. Heat Mass Transfer*, **18**, 1101-1107.

Crank J and Phahle R D (1973) Melting ice by the isotherm migration method, *Bull. Inst. Math. Applics.,* **9**, 12-14.

Crank J and Nicolson P. (1947) A practical method for numerical evaluation of solutions of partial differential equations of the heat conduction type, *Proc. Camb. Phil. Soc.*, **43**, 50-67.

Crann D (1996) The Laplace transform: numerical inversion of computational methods, *University of Hertfordshire Mathematics Department Technical Report,* **21**.

Crann D (2005) *The Laplace transform boundary element method for diffusion type problems,* PhD Thesis, University of Hertfordshire.

Crann D, Davies A J, Lai C-H and Leong S H (1998) Time domain decomposition for European options in financial modelling, *Contemporary Mathematics*, **218**, 486-491.

Crann D, Davies A J and Christianson B (2005) The Laplace transform dual reciprocity boundary element method for electromagnetic heating problems, *Advances in Boundary Element Techniques VI*, EC Ltd, 229-234.

Crann D, Davies A J and Mushtaq J (1998) Parallel Laplace transform boundary element methods for diffusion problems, *Boundary Elements XX*, Computational Mechanics Publications, 259-268.

Crann D, Kane S J, Davies A J and Lai C-H (2007) A time-domain decomposition method for the parallel boundary element solution of diffusion problems, *Advances in Boundary Element Techniques VIII*, EC Ltd, 281-286.

Crowley A B (1979) On the weak solution of moving boundary problems, *J. Inst. Math. Applics.*, **24**, 43-57.

Davies A J (1986) *The finite element method: a first approach,* Oxford University Press.

Davies A J (1989) Mapping the boundary element method to the ICL DAP, *Conpar88*, Cambridge University Press, 230-237.

Davies A J (1993) Using a Spreadsheet to Investigate Models of Heat Conduction and Vibrating Strings, *Teaching Mathematics and its Applications*, **12**, 179-185.

Davies A J, Crann D and Mushtaq J (1996) A parallel implementation of the Laplace transform BEM, *Boundary Elements XVIII*, Computational Mechanics Publications, 213-222.

Davies A J and Crann D (1998) The boundary element method on a spreadsheet, *Int. J. Math. Educ. Sci. Technol.*, **29**, 851-865.

Davies A J and Crann D (1999) The solution of differential equations using numerical Laplace transforms, *Int. J. Math. Educ. Sci. Technol.*, **30**, 65-79.

Davies A J and Crann D (2007) Laplace transform time domain-decomposition for diffusion problems, *DCABES 2007*, Hubei Sci. and Tech. Press, 1-4.

Davies A J, Honnor M E, Lai C-H, Parrott A K and Rout S (2004) A distributed Laplace transform method for European options, *Computational Finance and its Applications*, 157-166.

Davies A J and Mushtaq J (1995) A multiblock solution process for the Navier-Stokes equations on a network of transputers, *Applications of High-Performance Computing in Engineering IV*, Computational Mechanics Publications, 149-156.

Davies A J and Mushtaq J (1996) The domain decomposition boundary element method on a network of transputers, *Boundary Element Technology XI*, Computational Mechanics Press, 397-406.

Davies A J and Mushtaq J (1998) Parallel implementation of the boundary element and finite volume coupled aerofoil problem on an nCUBE multiprocessor system, *Advances in Fluid Mechanics II*, Computational Mechanics Publications, 172-180.

Davies A J and Radford L E (2001) A method for solving diffusion-type problems using separation of variables with the finite difference method, *Int. J. Math. Educ. Sci. Technol.*, **32**, 449-456.

Davies A J, Mushtaq J, Radford L E and Crann D (1997) The numerical Laplace transform solution method on a distributed memory architecture, *Applications of High Performance Computing V,* Computational Mechanics Publications, 245-254.

Davies B and Martin B (1979) Numerical Inversion of the Laplace Transform: a Survey and Comparison of Methods, *Jnl. of Comp. Physics,* **33**, 1-32.

Dix R C and Cizek J (1970) The isotherm migration method for transient heat conduction analysis, *Heat Transfer*, **1**, 4th International Heat Transfer Conference, Paris, Versailles, Elsevier.

Dixon L C W and Ducksbury P G (1985) Finite element optimisation on the DAP, *Comput. Phys. Commun.*, **37**, 187-193.

Douglas J and Gallie T M (1955) Variable Time Steps in the Solution of the Heat Flow Equation by a Difference Equation, *Proc. Am. Math. Soc.*, **6**, 787-793.

Elliott C M (1980) On a Variational Inequality Formulation of an Electrochemical Machining Moving Boundary Problem and its Approximation by the Finite Element Method, *J. Inst. Maths. Applics.*, **25**, 121-131.

Elliott C M (1981) On the Finite Element Approximation of an Elliptic Variational Inequality Arising from an Implicit Time Discretization of the Stefan Problem, *IMA J. Num. Anal.*, **1**, 115-125.

Eyres N R, Hartree D R, Ingham J, Jackson R, Sarjant R J and Wagstaff S M (1946) The calculation of variable heat flow in solids, *Phil. Trans. R. Soc.*, **A240**, 1-57.

Fairweather G and Karageorghis A (1998) The method of fundamental solutions for elliptic boundary value problems, *Advances in Computational Mathematics*, **9**, 69-95.

Fairweather G, Rizzo F J, Shippy D J and Wu Y S (1979) On the numerical solution of two-dimensional potential problems by an improved boundary integral equation method, *J. Comp. Phys.*, **31**, 96-112.

Flynn M (1972) Some Computer Organizations and Their Effectiveness, *IEEE Trans. Comput.*, **C-21**, 948-960.

Furzeland R M (1977) *Department of Mathematics, Brunel University, Technical Report*, **77** (quoted by Furzeland 1980).

Furzeland R M (1980) A Comparative Study of Numerical Methods for Moving Boundary Problems, *J. Inst. Maths. Applics.*, **26**, 411-429.

Gipson G S (1985) The coupling of Monte-Carlo integration with boundary integral techniques to solve Poisson-type problems, *Engng. Anal. Bdry. Elem.*, **2**, 138-145.

Golberg M A (1995) The method of fundamental solutions for Poisson's equation, *Engineering Analysis with Boundary Elements*, **16**, 205-213.

Goodman T R (1958) The heat balance integral and its application to problems involving a change of phase, *Trans. ASME*, **80**, 335-342.

Gropp W, Lusk E and Skjellum A (1994) *Using MPI: Portable Parallel Programming with the Message Passing Interface*, MIT Press.

Gupta R S and Kumar A (1988) Isotherm migration method applied to fusion problems with convective boundary conditions, *Int. J. Num. Meth. Eng.*, **26**, 2547-2558.

Hadamard J (1923) *Lectures on Cauchy's problem in linear partial differential equations*, Dover.

Hansen P C (1992) Numerical tools for analysis and solution of Fredholm integral equations of the first kind, *Inverse Problems*, **8**, 849-872.

Honnor M E and Davies A J (2002) The Laplace transform dual reciprocity boundary element method for nonlinear transient field problems, *Boundary Elements XX*, Computational Mechanics Press, 363-372.

Honnor M E, Davies A J and Kane S J (2003) Nonlinear transient field problems with phase change using the generalized Newmark dual reciprocity boundary element method, *Boundary Elements XXV*, WIT Press, 309-318.

Jaswon M A and Symm G T (1977) *Integral Equation Methods in Potential Theory and Elastostatics*, Academic Press.

Kane S J, Davies A J and Honnor M E (2004) A dual reciprocity method for phase change problems, *Boundary Elements XXVI*, WIT Press, 173-182.

Kreider D L, Kuller R G, Ostberg D R and Perkins F W (1966) *An Introduction to Linear Analysis,* Addison-Wesley.

Kupradze V D (1964) On a method of solving approximately the limiting problems of mathematical physics, *Z.Vychisl. Mat. i. Mat. Fiz.*, **4**, 1118-1121.

Kutluay S and Esen A (2004) An isotherm migration formulation for one-phase Stefan problem with a time dependent Neumann condition, *Applied Mathematics and Computation*, Elsevier, **150**, 59-67.

Lachat J C and Combescure A (1977) Laplace transform and boundary integral equation applications to transient heat conduction problems, *Proc. First Sump. on Innovative Numerical Analysis in Applied Engineering,* CETIM, Versailles, 1.57-1.62.

Landau H G (1950) Heat conduction in a melting solid, *Q. Appl. Math*, **8**, 81-94.

Lightfoot N M H (1929) The solidification of molten steel, *Proc. Lond. Math. Soc.*, **31**, 97-116.

Liu G R (2003) *Mesh free methods: moving beyond the finite element method*, CRC Press, 9-25.

Mattson T, Sanders B and Massingill B (2004) *Patterns for Parallel Programming*, Addison Wesley Professional.

Meyer G H (1970) On a Free Interface Problem for Linear Ordinary Differential Equations and the One-Phase Stefan Problem, *Numer. Math.*, **16**, 248-267.

Mitra A K and Ingber M S (1993) A multiple-node method to resolve the difficulties in the boundary integral equation method caused by corners and discontinuous boundary conditions, *Int. J. Num. Meth. Eng.*, **36**, 1735-1746.

Moridis G J and Reddell D L (1991a) The Laplace Transform Finite Difference (LTFD) Method for Simulation of Flow Through Porous Media, *Water Resources Research,* **27**, 1873-1884.

Moridis G J and Reddell D L (1991b) The Laplace Transform Finite Element (LTFE) Numerical Method for the Solution of the Groundwater Equation, paper H22C-4, ASGU91 Spring Meeting, *EOS Trans. of the AGU,* **72**.

Moridis G J and Reddell D L (1991c) The Laplace Transform Boundary Element (LTBE) Numerical Method for the solution of diffusion-type problems, *Boundary Elements XIII,* Elsevier, 83-97.

Murray W D and Landis F (1959) Numerical and Machine Solutions of Transient Heat-Conduction Problems Involving Melting or Freezing, *J. Heat Transfer, Trans ASME*, **81**, 106-112.

Mushtaq J and Davies A J (1996) A multiblock solution process for the equations of fluid dynamics on a network of transputers, *Advances in Engineering Software*, **26**, 133-149.

Mushtaq J and Davies A J (1997a) Parallel implementation of the boundary element method using PVM and HPF on a collection of networked workstations, *High Performance Computing 97*, Computational Mechanics Press, 181-188.

Mushtaq J and Davies A J (1997b) Parallel boundary element implementation of the lifting aerofoil problem on a local area network multicomputer system using PVM and HPF, *Proceedings of the fourth Australian conference on parallel and real-time systems*, Springer-Verlag, 126-133.

Osher S and Sethian J A (1988) Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations,*J. Comp. Physics*, **79**, 12-49.

Pacheco P (1997) *Parallel Programming with MPI*, San Francisco, Morgan Kaufmann Publishers Inc.

Paris F and Cañas J (1997) *Boundary element method: Fundamentals and applications*, Oxford University Press.

Partridge P W, Brebbia C A and Wrobel L C (1992) *The Dual Reciprocity Boundary Element Method*, Computational Mechanics Publications and Elsevier Applied Science.

Patel P D (1968) Interface conditions in heat conduction problems with change of phase, *AIAA Journal*, **6**, 2454.

Philip J R (1995) Numerical solutions of equations of the diffusion type with diffusivity concentration-dependent, *Trans. Faraday Soc.*, **51**, 885-892.

Piessens R (1972) A New Numerical Method for the Inversion of the Laplace Transform, *J. Inst. Maths Applics.,* **10**, 185-192.

Poots G (1962) An approximate treatment of a heat conduction problem involving a two-dimensional solidification front, *Int. J. Heat Mass Transfer*, **5**, 339-348.

Power H and Cesares Long J J (1997) *Applications of High Performance Computing V,* Computational Mechanics Publications, 245-254.

Price R H and Slack M R (1954) The effect of latent heat on the numerical solutions of the heat flow equation, *Br. J. Appl. Phys.*, **5**, 285-287.

Rizzo F J and Shippy D J (1970) A method of certain problems of transient heat conduction, *AIAA Journal,* **8**, 2004-2009.

Rose M E (1967) On the melting of a slab, *SIAM J. Appl. Maths.,* **15**, 495-504.

Schapery R A (1962) Approximate Methods of Transforming Inversion for Viscoelastic Stress Analysis, *Proc. Fourth US National Congress on Applied Mechanics,* **2**, 1075-2085.

Shih Y-P and Tsay S-I (1971) Analytical solutions for freezing a saturated liquid inside or outside cylinders, *Chem. Eng. Sci.*, **26**, 809-816.

Smith G D (1978) *Numerical solution of partial differential equations,* Oxford University Press.

Spiegel M R (1959) *Theory and Problems of Vector Analysis and an introduction to Tensor Analysis*, Schaum Publishing Company, 126-127.

Spiegel M R (1965) *Theory and Problems of Laplace Transform*, Schaum, New York.

Stefan J (1891) Über die Theorie der Eisbildung inbesondere über die Eisbildung im Polarmeere, *Ann. Phys. U. Chem.* **42**, 269-286.

Stehfest H (1970) Numerical inversion of Laplace transforms, *Comm. ACM,* **13**, 47-49 and 624.

Tanaka M, Matsumoto T and Takakuwa S (2003) Dual reciprocity BEM based on time-stepping scheme for the solution of transient heat conduction problems, *Boundary Elements XXV*, WIT Press, 299-308.

Telles J C F (1987) A self-adaptive co-ordinate transformation for efficient numerical evaluation of general boundary integrals, *Int. J. Num. Meth. Engng.*, **24**, 959-973.

Telles J C F and Oliveira R F (1994) Third degree polynomial transformation for boundary element integrals: further improvements, *Engng. Anal. with Boundary Elements*, **13**, 135-141.

Toutip W (2001) *The dual reciprocity boundary element method for linear and non-linear problems*, PhD Thesis, University of Hertforshire.

Voller V R (1985) Implicit Finite-difference Solutions of the Enthalpy Formulation of Stefan Problems, *IMA J. Num. Anal.*, **5**, 201-214.

Voller V R, Swenson J B, Kim W and Paola C (2006) An enthalpy method for moving boundary problems on the earth's surface, *Int. J. Heat and Fluid Flow*, **16**, 641-654.

von Neumann J (1945) *First Draft of a Report on the EDVAC*, Moore School of Electrical Engineering, University of Pennsylvania, re-published in IEEE Annals of the History of Computing, (1993), **15**, 27-75.

Walker D E (1992) Standards for Message Passing in a Distributed Memory Environment, *Oak Ridge National Laboratory Technical Report*, ORNL/TM-12147.

Walker D and Houstis E (2005) Complex problem solving environments for grid computing, *Future Generation Computer Systems*, **21**, 841-842.

Weinberger H F (1965) *A First Course in Partial Differential Equations,* Xerox College Publishing.

Wood A S (1991a) An unconditionally stable implementation of the isotherm migration method, *J. Numer. Heat Transfer (B)*, **20**, 105-113.

Wood A S (1991b) A fast implementation of the isotherm migration method, *Computational modelling of free and moving boundary problems*, Southampton, **2**, 133-147.

Wood A S (1991c) A note on the use of the isotherm migration method, *J. Comput. Appl. Math.*, **36**, 371-384.

Zakian V and Littlewood R K (1973) Numerical inversion of Laplace transforms by weighted least-squares approximation, *Comp. J.*, **16**, 66-68.

Zhu S-P (1999) Time-dependent reaction-diffusion problems and the LT-DRM approach, *Boundary Integral Methods, Numerical and Mathematical Aspects*, Computational Mechanics Publications, 1-35.

Zhu S, Satravaha P and Lu X (1994) Solving linear diffusion equations with the dual reciprocity method in Laplace space, *Engineering Analysis with Boundary Elements*, **13**, 1-10.

Zienkiewicz O C and Taylor R L (2000) *The Finite Element Method*, Vols 1, 2, 3, Fifth edition, Butterworth-Heinemann.

Websites.

Top 500 computer sites, www.top500.org