

David Charles Martin

SELECTED HEAT
CONDUCTION PROBLEMS
IN THERMOMECHANICAL
TREATMENT OF STEEL

UNIVERSITY OF OULU,
FACULTY OF TECHNOLOGY,
DEPARTMENT OF MECHANICAL ENGINEERING



ACTA UNIVERSITATIS OULUENSIS
C Technica 388

DAVID CHARLES MARTIN

**SELECTED HEAT CONDUCTION
PROBLEMS IN THERMOMECHANICAL
TREATMENT OF STEEL**

Academic dissertation to be presented with the assent of
the Faculty of Technology of the University of Oulu for
public defence in OP-sali (Auditorium L10), Linnanmaa, on
30 September 2011, at 12 noon

UNIVERSITY OF OULU, OULU 2011

Copyright © 2011
Acta Univ. Oul. C 388, 2011

Supervised by
Professor Pekka Mäntylä
Professor David Porter

Reviewed by
Professor Jari Magnus
Docent Weihong Yang

ISBN 978-951-42-9518-8 (Paperback)
ISBN 978-951-42-9519-5 (PDF)
<http://herkules.oulu.fi/isbn9789514295195/>
ISSN 0355-3213 (Printed)
ISSN 1796-2226 (Online)
<http://herkules.oulu.fi/issn03553213/>

Cover Design
Raimo Ahonen

JUVENES PRINT
TAMPERE 2011

Martin, David Charles, Selected heat conduction problems in thermomechanical treatment of steel.

University of Oulu, Faculty of Technology, Department of Mechanical Engineering, P.O. Box 4200, FI-90014 University of Oulu, Finland

Acta Univ. Oul. C 388, 2011

Oulu, Finland

Abstract

This thesis considers two related problems where heat conduction is accompanied by phase transformation during the cooling of carbon steels – phenomena which are fundamental to modern thermomechanical treatment processes.

In the first problem, a non-linear heat equation formulation is described which includes linkage between irreversible diffusive and diffusionless phase transformation processes and thermophysical properties. A family of numerical schemes using finite difference methods with diagonally implicit Runge-Kutta method integrators for solving this formulation is presented. Sample calculations using these schemes are presented for a mild steel cooled under range of industrially relevant heat transfer conditions. These calculations illustrate the need for careful scheme design and solver selection when coupled heat conduction and microstructure evolution equations become non-linear and stiff.

The second section of the thesis extends this heat conduction formulation into the realm of inverse analysis. The problem of temperature reconstruction and boundary condition estimation using temperature measurements obtained from thermocouples embedded into laboratory samples undergoing phase transformation is considered. A solution method based on iterative regularization is described for solving the resulting ill-posed problem. An analysis of extremely high rate cooling experiments made on a pilot-plant descaling unit using the inverse method is presented. This analysis highlights some of the practical issues associated with embedded thermocouple temperature measurements made in this fashion.

Keywords: carbon steels, heat conduction, inverse problems, mathematical modelling, phase transformation, thermomechanical treatment, thermophysical properties

Martin, David Charles, Valikoituja teräksen termomekaanisessa käsittelyssä esiintyviä lämmön johtumisen ongelmia.

Oulun yliopisto, Teknillinen tiedekunta, Konetekniikan osasto, PL 4200, 90014 Oulun yliopisto
Acta Univ. Oul. C 388, 2011
Oulu

Tiivistelmä

Väitöstyössä tutkitaan kahta ongelmaa liittyen hiiliteräksen termomekaanisen käsittelyn numeeriseen mallintamiseen. Työssä tarkastellaan teräksessä tapahtuvaa lämmönsiirtoa ja faasimuutoksia jäähtymisen aikana, eli ensisijaisen tärkeitä ilmiöitä kehittynyttä teräksen valmistusta ja prosessointia ajatellen.

Ensimmäisenä ongelmana tutkitaan epälineaarisen energiayhtälön muodostaminen huomioiden teräksen termofysikaaliset materiaaliominaisuudet, diffuusion kontrolloimat ja diffuusiosta riippumattomat irreversiibelit faasimuutokset. Lisäksi huomioidaan näiden väliset kytkennät. Yhtälöt ratkaistaan numeerisesti ja käytetyt menetelmät esitellään differenssimenetelmien ja diagonaalisesti implisiittisten Runge-Kutta-menetelmien osalta. Väitöstyössä näiden käyttöä havainnollistetaan simuloimalla niukkaahilisen teräksen käyttäytymistä teollisia olosuhteita vastaavan jäähtymisen aikana. Laskentatulokset osoittavat ratkaisumenetelmän suunnittelun ja valinnan tärkeyden erityisesti silloin, kun lämmönjohtuminen, mikrorakenteen kehittyminen ja termofysikaaliset materiaaliominaisuudet muuttavat ratkaistavan ongelman epälineaariseksi ja jäykäksi.

Toisena ongelmana tutkitaan edellä esitetyn lämmönsiirtomallin käyttöä osanainversioanalyysiä. Laboratoriokokeiden, termopareilla suoritettujen lämpötilamittausten, työssä kehitetty lämmönsiirtomallin ja suoritettujen inversioanalyysin avulla rekonstruoidaan teräskappaleiden lämpötila- ja faasimuutoskäyttäytymistä sekä estimoidaan mallin reunaehtoja. Näin luotu inversio-ongelma on kuitenkin matemaattisesti tarkasteltuna ns. huonosti asetettu ongelma, ja sen ratkaisemiseksi käytetään iteratiiviseen säännöllistämiseen perustuvaa menetelmää. Väitöstyön inversioanalyysiä havainnollistetaan pilot-mittakaavan kokeiden avulla, joissa hilsepesuria käyttäen koekappaleen jäähtymisnopeus saadaan erittäin korkeaksi. Tulokset nostavat esille inversioanalyysiin liittyviä ongelmia ja rajoitteita nopeaan jäähtymiseen sekä lämpötilamittauksiin liittyen.

Asiasanat: faasimuutos, hiiliteräs, inversio-ongelma, lämmön siirtyminen, numeerinen mallintaminen, termofysikaaliset materiaaliominaisuudet, termomekaaninen käsittely

To my mother, Barbara.

Acknowledgements

This thesis contains work done in two periods spanning 2000-2003 and 2007-2009 in the Materials Engineering Laboratory at the University of Oulu. Parts of this work were performed under Research Programme of the Research Fund for Coal and Steel contracts RFSR-CT-2005-00016 and RFSR-CT-2008-00023. Funding and support for this work was also provided by Rautaruukki Oyj during 2000-2003.

First and foremost I would like to thank my supervisor, Professor Pekka Mäntylä, who firstly invited me to come to Finland, then helpfully suggested I focus on thermal problems and indulged me the freedom to follow the subject in slightly unexpected directions, and finally demonstrated saintly patience while waiting for the work to reach its conclusion and this thesis to emerge from it.

I was also like to take this opportunity to extend my thanks to my colleagues at the Materials Engineering Laboratory, past and present, for all the assistance they afforded me during my time there, particularly Jussi Paavola for almost a decade of fruitful scientific collaboration and personal friendship, and Pekka Hautamäki and Juha Pyykkönen for their help and the many discussions which played a significant role in shaping the work contained in this thesis.

I would also like to thank John Niska, Annika Nilsson and Anders Rensgard from Swerea-MEFOS for their considerable assistance with conducting the pilot plant descaling trials.

Finally, I would like to thank all my family and friends, both in Australia and Finland, for their support and understanding throughout this overly long journey. Most importantly, I would like to thank Marjukka for her unconditional love and unwavering belief in me, Lasse and Tuula for all their support, my sister Clarissa, who showed how things are supposed to be done, and most of all my mother, Barbara.

Turku, July 2010.

Abbreviations

Unless explicitly noted otherwise, expressions in this monograph aim to conform to Householder's notation (1, §1), so that upper case letters are used to denote matrices, lower case Roman letters are used to denote vectors, and lower case Greek letters are used to denote scalars. Two further alleviate any ambiguity, vectors and matrices have been written using a bold typeface. The various matrix and vector operators are denoted by:

$\mathbf{x}\mathbf{y}$: vector-vector product

$\mathbf{x} \cdot \mathbf{y}$: vector dot product

$\langle \mathbf{x}; \mathbf{y} \rangle$: vector inner product

$\|\mathbf{x}\|$: vector norm

$\mathbf{A}\mathbf{x}$: matrix-vector product

$\mathbf{A}\mathbf{B}$: matrix-matrix product

$\mathbf{A} \otimes \mathbf{B}$: Kronecker matrix product

$\mathbf{A} * \mathbf{B}$: Hadamard (element wise) matrix product

$\mathbf{x}^T, \mathbf{A}^T$: vector and matrix transpose operator

$\det \mathbf{A}$: matrix determinant

$\text{vec } \mathbf{A}$: matrix column major order vectorization operator

$\text{diag } \mathbf{b}$: diagonal matrix formation from a vector

For vector and matrix differentiation, we use ∇ to denote the gradient operator and \mathcal{D} to denote the directional derivative. Standard notation for scalar differentiation is generally used, however chapter 3 uses a shorthand notation whereby the scalar derivative is indicated by subscripts:

$$\begin{aligned}
 u_x &\equiv \frac{\partial u}{\partial x} \\
 u_{xx} &\equiv \frac{\partial}{\partial x} \left(\frac{\partial u}{\partial x} \right) \\
 u_{xy} &\equiv \frac{\partial}{\partial y} \left(\frac{\partial u}{\partial x} \right)
 \end{aligned}$$

which serves to reduce the length and typographical complexity of some of the Lagrangians used in the derivation of the inverse method. In addition, chapters 2 and 3 adopt a corruption of Newton's dot notation as a shorthand for the first and second derivatives of a number of scalar quantities with respect to temperature. Using this convention, if λ denotes the thermal conductivity and u then temperature, then

$$\begin{aligned}
 \dot{\lambda} &\equiv \frac{\partial \lambda}{\partial u} \\
 \ddot{\lambda} &\equiv \frac{\partial}{\partial u} \left(\frac{\partial \lambda}{\partial u} \right)
 \end{aligned}$$

represent the first and second derivatives of the thermal conductivity with respect to temperature. Again this has been done for the sake of lexical brevity.

Acronyms

BC boundary condition.

BCC body centred cubic lattice structure.

BDF backward differentiation formula.

CFL Courant–Friedrichs–Lewy condition.

CGM conjugate gradient method.

DIRK diagonally implicit Runge-Kutta.

FCC face centred cubic lattice structure.

FSAL First-same-as-last property of certain Runge-Kutta schemes.

IHCP Inverse heat conduction problem.

LHS left hand side.

ODE ordinary differential equation.

PDE partial differential equation.

RHS right hand side.

SHE sideways heat equation.

TMT thermomechanical treatment.

List of Physical Symbols

chi (χ) solid state phase volume fraction [-].

gamma (γ) volumetric heat capacity [J/kg m³].

kappa (κ) specific heat capacity [J/kg K].

lambda (λ) thermal conductivity [W/m K].

mu (μ) enthalpy of transformation [J/kg].

rho (ρ) mass density [kg/m³].

temperature (u) continuum temperature [K].

Contents

Abstract

Tiivistelmä

Acknowledgements 9

Abbreviations 11

Contents 15

1 Introduction and overview 17

2 Heat conduction during cooling of hypoeutectoid steels. 21

2.1 Introduction 21

2.2 The heat conduction equation 22

2.3 Latent heat of transformation 25

2.4 Boundary Conditions 25

2.5 Numerical approximation 27

2.5.1 The numerical method of lines 27

2.5.2 Finite Differences 27

2.5.3 Integration 31

2.6 Thermophysical Properties 37

2.6.1 Thermal conductivity 37

2.6.2 Mass density 39

2.6.3 Specific Heat 39

2.6.4 Transformation enthalpy 41

2.6.5 Microstructure evolution 44

2.6.6 Differentiability of coefficients 53

2.7 Coupling of conduction and microstructure 57

2.8 Computational examples 60

2.8.1 Cooling in still air 63

2.8.2 Spray cooling 70

2.8.3 Immersion quenching in water 77

2.9 Discussion 86

3 Inverse heat conduction problems in TMT processes 91

3.1 Introduction 91

3.2	A definition by the inverse problem by example	93
3.2.1	Ill-posedness of the SHE	94
3.2.2	A computational example	95
3.3	Overview of regularisation methods	99
3.4	Iterative Regularisation	102
3.4.1	SHE objective functionals	103
3.4.2	Extremum principles applied to the SHE	105
3.4.3	The sensitivity equation	106
3.4.4	The first order adjoint equation	107
3.4.5	Optimal depth of descent calculations	109
3.4.6	The complete iterative regularisation algorithm for the SHE	110
3.4.7	Stoppage criteria	112
3.5	The computational example revisited	114
3.6	An analysis of pilot plant hydraulic descaling	117
3.6.1	An overview of the hydraulic descaling process	117
3.6.2	Experimental procedure	118
3.6.3	Input data preparation	123
3.6.4	Inverse analysis	123
3.6.5	Results and discussion	127
3.7	Conclusions	138
4	Discussion and final conclusions	141
	References	145
	Appendices	149

1 Introduction and overview

Thermomechanical treatment of steel – the application of combinations of controlled heating, deformation, and cooling to steel to develop desirable microstructural features and obtain the mechanical properties those features provide – has been at the centre of a quiet revolution in the physical metallurgy and production technology which forms the basis of many of the new types of high performance steels which have been developed over the last three decades.

The history of steels for high pressure oil and gas pipelines provides an excellent illustration of the impact of modern TMT processes have had on the properties of high performance steels generally. Linepipe steels have fairly unique set of mechanical property, fabrication and service requirements that are particularly challenging from a metallurgical perspective. Not only do linepipes require very high levels of strength for pressure containment, but also the steel used must be ductile to permit forming into pipes from skelp or plate, possess good weldability to permit fabrication and joining of pipe sections, and exhibit excellent low temperature fracture toughness for burst resistance and safe service in the undersea and sub-arctic environments where modern gas and oil pipelines are often deployed.

In 1970, the state-of-the-art production linepipe steel was API-5L X52, which specifies a minimum *pipe* yield strength of 52 ksi (kilopound force per square inch) (2). By the mid-1970s, the first thermomechanically rolled X65 (65 ksi yield strength) steels enter production, and by 1980 thermomechanically treated X70 (70 ksi yield strength) steels became available (3).

During the 1980s, X80 grades (80 ksi yield strength) produced by thermomechanical rolling and accelerated cooling were developed and entered production, with the first large scale pipeline deployments occurring in the early 1990s. During that decade, direct quenched X100 grades (100 ksi yield strength) were under active development, with the first X100 pipeline constructed in 2002 (4). Commercialisation of next generation X120 (120 ksi yield strength) linepipe grades is currently under way (5). The net result of the introduction of modern thermomechanical treatment processes and steel alloy designs to linepipe has been a more than twofold increase in pipe strength in just over four decades. This evolution of thermomechanically treated, advanced high strength steels has had a profound influence on pipeline capital efficiency, energy efficiency and life cycle costs.

The metallurgical and processing technology progress seen in linepipe as a result of the introduction of modern thermomechanical treatment processes have been mirrored in steels for other demanding sectors, particular in automotive and structural applications. As a result of these developments, modern advanced, high strength steel production processes require increasingly complex and precise temperature controlled rolling and cooling schedules, often featuring the use of high rate accelerated cooling or direct quenching after finish rolling in order to achieve the desired microstructure and mechanical properties. Current trends in the literature suggest that future advanced high strength steel will require thermomechanical treatment schedules which are even more complex and require even more stringent control of temperature and deformation than the current state-of-the-art.

The ability to accurately *a priori* predict temperature during thermomechanical treatment, particularly cooling, and to correctly interpret measured data *post hoc* is fundamental both in commercial scale production and laboratory scale research and development. To do so requires a thorough description of the interactions between external heat transfer processes and changes in internal energy due to phenomena such as solid-state phase transformations. The thermophysical properties of the evolving microstructures of modern high strength steels vary greatly, and are strongly dependent on temperature, chemical composition and constituent phases. As a result, critical thermophysical parameters such as thermal conductivity and latent heat of transformation cannot be accurately estimated without appropriate linkage to both phase transformation kinetics and temperature fields. It is these heat conduction mechanisms which form the core problems considered in this research.

In this monograph we discuss two distinct but related classes of heat conduction problems which arise in accelerated cooling of steels during thermomechanical treatment, and propose numerical simulation and analysis methods which have application at both the laboratory and industrial scale.

Heat conduction and phase transformation in hypoeutectoid steels

The first problem is connected with computing the temperature of hypoeutectoid steel during cooling from an initial austenitic state, which an almost universal feature of modern thermomechanical treatment processes involving hot working. This is by no

means a new problem, with a vast amount of literature covering the problem as it arises in hot rolling, hot forging and welding dating at least from the work of Hollander (6) published in 1970, and probably stretching back well into the 1960s. It transpires that the thermophysical properties of these types steels are non-linear and interact to a very large degree with the solid state phase transformation reactions which occur as the austenitic microstructure cools. The austenite decomposition processes themselves are exothermic and are accompanied by latent heat of transformation. It is this interaction between temperature, thermophysical properties, and microstructural evolution which is the source of considerable complexity in what should otherwise be a relatively straightforward analysis.

In the present analysis of heat conduction during cooling of hypoeutectoid steels, we describe a non-linear heat equation for materials with strongly non-linear thermophysical properties, along with a law of mixtures scheme for coupling models of austenite decomposition and thermophysical properties to this heat equation. A method of lines based numerical implementation of the model is presented which uses second or fourth order accurate finite difference operators for spatial discretisation on uniform grids, and second or third order accurate diagonally implicit Runge-Kutta methods for time integration. The practical aspects of computer implementation, stability, accuracy and computational efficiency of this approach are discussed in the context of the problem of cooling a plain carbon steel section under different heat transfer conditions using boundary condition models developed from literature sources.

Inverse heat conduction problems in thermomechanical treatment processes

The second problem considered in this monograph is a class of inverse heat conduction problems which arise in connection laboratory and pilot plant physical simulation, where a source of measured data from a sensor such as thermocouple or pyrometer is available continuously through the thermomechanical treatment process, but otherwise boundary conditions and temperature distribution within the stock are unknown. These Cauchy problems are ill-posed by the definition of Hadamard, and cannot be solved in the usual fashion because their solution does not continuously depend on the measured data.

In this section of the work, an iterative regularisation method is described for the problem of reconstructing temperature fields and boundary from internal thermocouple

temperatures measurements. The iterative regularization technique was selected over other approaches for its superior measurement noise rejection properties and the amenability of the method to work with non-linear conduction problems where a suitable integral solution is not available. We discuss the problem of stoppage criterion for the method when knowledge of the noise present in the measured data is not known *a priori*.

To demonstrate the inverse analysis in practice, an analysis of static descaling tests made using pilot plant hydraulic descaling is presented. Although not a thermomechanical treatment process by the usual definition, these descaling experiments were of interest because it was found that phase transformation played an important role in determining a physically consistent solution to the inverse problem. The results of this analysis were compared to other published work dealing with heat transfer of high flow rate sprays, and the role of surface oxide scale in influencing the results of the inverse analysis.

2 Heat conduction during cooling of hypoeutectoid steels.

2.1 Introduction

One of the key developments in modern hot rolled, thermomechanically treated carbon steels has been the use of high rate cooling during and after deformation to promote the formation microstructural features which result in high strength from steels with low carbon equivalents and total alloying element content – typically finer ferrite grain sizes and beneficial grain morphologies (7), and higher dislocation density second phases such as bainite and martensite in place of pearlite (8).

As the tolerances on microstructure, dimensions and properties of low carbon, high strength steels tighten in response to market demand, steel mills are increasingly required to produce these steels using sophisticated processing routes involving extremely well controlled rolling passes and accelerated cooling schedules. The ability to predict and control temperature during rolling and cooling is paramount to achieving repeatable and high quality commercial production of these steels.

In response to this need, we consider the problem of predicting temperature of hypoeutectoid steels cooling from an initial austenitic state, which is the normal condition of the steel at the end of the final rolling passes. As the austenitic microstructure is cooled, it will decompose into different phases through the action of one of several possible solid state phase transformation reactions. The austenite and its' transformation products have dissimilar thermophysical properties, meaning that they conduct heat at different rates. Further, the transformation reactions themselves depend on temperature, as does the rate at which they proceed, and the amount of exothermic heat of reaction which will be liberated with them. The interaction of these mechanisms adds considerable complexity to the problem of calculating the conduction of heat through steel during cooling when accompanying by phase transformation.

The concept of latent heat of transformation and its effects on cooling behaviour of hot rolled products has traditionally been considered to be important only in the realm only of high carbon eutectoid and hypereutectoid steels, where pearlite recalescence phenomena is known increase the complexity of temperature calculation during cooling (9, 10). The relatively recent development of modern ultra high strength, low alloy

content steels which feature complex phase microstructures has meant that hypoeutectoid hot rolled steels are now routinely being cooled at very high rates to low temperatures (11). This technique has the effect of introducing austenite decomposition reactions at much lower temperatures than the typical proeutectoid ferrite and pearlite transformations. As discussed by Browne (12), latent heat of transformation associated with the ferromagnetic transformation in low carbon steels can significantly change the heat balance when austenite decomposes at temperatures which are significantly below the Curie temperature of iron, 1043 K. This implies that greater attention to treatments of the interaction of microstructure evolution and thermophysical properties may be necessary in simulating heat conduction in these types of TMT processes for hypoeutectoid steels.

The problem of numerical simulation of coupled heat conduction and microstructure evolution by no means a new one, and the subject is well represented in the literature dating back to at least 1981 and the paper of Agarwal and Brimacombe (13), with important contributions since that time by Fernandes *et al* (14), Watt *et al* (15), and Hömberg (16), amongst many others. In this work we seek to build on this literature by considering the problem of continuously cooling a hypoeutectoid steel which undergoes solid state phase transformations from an initial austenitic state, with particular emphasis on the linkage between microstructure and thermophysical properties and the interaction of heat conduction processes and latent heat of phase transformation.

Starting with the derivation of a non-linear heat equation and descriptions of the thermophysical properties and phase transformation behaviour of hypoeutectoid steels taken from literature, the following chapter addresses this problem by proposing a mathematical formulation which couples equations describing heat conduction, microstructure evolution and thermophysical properties together via a law-of-mixtures based conservation rule. Numerical methods for solving the resulting system of equations based on a method of lines approach using implicit finite difference methods are described, and we then present several computational examples which illustrate some of the practical complexities of predicting temperature under heat transfer regimes which are found both in the laboratory and industrial scale TMT processes.

2.2 The heat conduction equation

The conduction of heat due to temperature differences through a continuum at a length scale which is large with respect to phonon interactions can be represented by the

Fourier law, Equation (1). This states that the heat flux q across an infinitesimally thin unit area of the continuum is proportional to the temperature gradient across that unit area, ∇u , and the thermal conductivity of the unit area, λ . Thermodynamic constraints dictate that the thermal conductivity is a strictly positive, non zero quantity (17, §1)

$$-\lambda \nabla u = q, \lambda > 0 \forall u. \quad (1)$$

A standard differential form of the conservation law at constant pressure for the heat in the continuum, e , can be written according to Equation (2), where ρ represent the mass density and s the net rate of internal heat change due to the action of sources or sinks

$$\rho \frac{\partial e}{\partial \tau} + \nabla q = s. \quad (2)$$

The total specific heat capacity at constant pressure for a material which undergo diffusionless solid state phase transformations can be defined as

$$\frac{de}{du} = \frac{\partial e}{\partial u} + \sum_i \frac{\partial e}{\partial m_i} \frac{\partial m_i}{\partial u}, \quad (3)$$

where m_i denotes the molar fraction of the i^{th} diffusionless phase transformation product. Substituting standard physical quantities into Equation (3) leads to

$$\frac{de}{du} = \kappa - \sum_i \mu_i \frac{\partial m_i}{\partial u}, \quad (4)$$

where $\kappa = \partial e / \partial u$ denotes the specific heat capacity at constant volume, and μ_i the transformation enthalphy of the i^{th} diffusionless phase transformation process. We note that in the absence of diffusionless transformations, Equation (4) reduces to

$$\frac{de}{du} = \kappa.$$

The Fourier law can be combined with Equation (2) and this definition of the total specific heat capacity at constant pressure, to yield

$$\rho \kappa \frac{\partial u}{\partial \tau} + \nabla q = s + \frac{\partial u}{\partial \tau} \sum_i \mu_i \frac{\partial m_i}{\partial u}, \quad (5)$$

which can be simplified into a recognisable form of the standard transient heat conduction equation, Equation (6). Again, thermodynamic considerations dictate that $\kappa > 0$ for all $u > 0$

$$\rho\kappa\frac{\partial u}{\partial\tau} - \nabla \cdot (\lambda\nabla u) = \sigma. \quad (6)$$

In TMT processes, the heat capacity and thermal conductivity may be both thermally and spatially dependent, which leads to a non-linear heat equation:

$$\rho\kappa\frac{\partial u}{\partial\tau} - \lambda\nabla^2 u - \nabla\lambda\nabla u = \sigma. \quad (7)$$

In cases where the thermal conductivity is strongly dependent on temperature, the spatial dependence of the thermal conductivity can be replaced with a temperature dependency. Traditionally this is done using a variable change via the Kirchhoff transformation (17, §1):

$$\varpi = \frac{1}{\lambda_0} \int_{u_0}^u \lambda \, du, \quad (8)$$

which eliminates the spatial dependency of the thermal conduction and yields a linear ODE:

$$\frac{\gamma}{\lambda} \frac{\partial \varpi}{\partial \tau} - \nabla^2 \varpi = \frac{\sigma}{\lambda_0}. \quad (9)$$

While this approach is convenient, it is not universally applicable. There remains a number of classes of problems, such as those involving changes of phase or state, or large temperature ranges, for which it is infeasible to accurately evaluate the transformation integral, because the exact form of the thermal conductivity over a useful normalising temperature range cannot be estimated *a priori*.

The alternative approach adopted in this work is assume that variation in thermal conductivity is most due to temperature related effects induced by solid state phase transformation processes. We explicitly introduce a temperature sensitivity term via the chain rule substitution $\nabla\lambda = \partial\lambda/\partial u\nabla u$, yielding a non-linear heat equation of the form shown in Equation (10):

$$\gamma\frac{\partial u}{\partial\tau} - \lambda\nabla^2 u - \frac{\partial\lambda}{\partial u}(\nabla u)^2 = \sigma. \quad (10)$$

For the transient conditions considered in the present work, we generally expect that

$$\left\| \lambda\nabla^2 u \right\| > \left\| \frac{\partial\lambda}{\partial u}(\nabla u)^2 \right\| \quad (11)$$

meaning that the problem continues to be diffusion dominated, albeit with some convective characteristics. We note that Rincon *et al* (18) describe the derivation of

essentially the same heat equation for materials with strongly temperature dependant thermal conductivity, including formal proofs of existence and uniqueness for one- and two- dimensional cases.

While this non-linearity makes numerical approximation somewhat more complicated, this approach is still considered more appropriate for TMT problems than the aforementioned Kirchhoff transformation. The solution of the non-linear heat conduction equation given by Equation (10) and its application to TMT processes will be the main focus of the remainder of this chapter.

2.3 Sources and sinks in thermomechanical treatment processes

In the current treatment of thermomechanical processing, we restrict ourselves to treatment of metals in the solid state. Within this domain, we identify two major sources of internal heat sources or sinks which must be accounted for in any realistic heat balance:

- Changes in sensible heat due to internal work, such as that done by plastic deformation and the Joule heating action of electric currents in resistance or induction heating
- The evolution of latent heat due to solid-state phase transformations

We assume that the action of each of these sources or sinks are cumulative, so that for a control volume, the net heat change is the sum of the individual sources and sinks due to these mechanisms, satisfying

$$\sigma = \sum_i \frac{\partial w_i}{\partial \tau} - \sum_j \frac{\partial}{\partial \tau} [(\rho_j \mu_j) \chi_j], \quad (12)$$

where w_i denotes a component of volumetric work, μ_j denotes the enthalpy of a given solid-state phase transformation, and χ_j denotes the evolution rate of the product phase associated with the transformation.

2.4 Boundary condition formulations for thermomechanical treatment processes

If the surface enclosing the continuum is denoted as Γ , and the surface normal as n , then the usual Dirichlet (Type I), Neumann (Type II), and Robin (Type III) boundary conditions can be expressed on $\mathbf{z} \in \Gamma$ as:

$$u(\mathbf{z}) = b(\mathbf{z}) \quad (13a)$$

$$\frac{\partial}{\partial n} u(\mathbf{z}) = a(\mathbf{z}) \quad (13b)$$

$$\frac{\partial}{\partial n} u(\mathbf{z}) = a(\mathbf{z}) \cdot u(\mathbf{z}) + b(\mathbf{z}). \quad (13c)$$

From these three basic boundary condition forms, it is possible to include the complete range of surface heat transfer phenomenon expected during TMT processing - contact conduction, forced and natural convection, and Stefan-Boltzmann radiation. Defining the temperature of the medium surrounding the continuum as u_∞ , the fluxes corresponding to these heat transfer mechanisms can be defined by

$$q = \frac{u - u_\infty}{r_s} \quad (14a)$$

$$q = h_c (u - u_\infty) \quad (14b)$$

$$q = \varepsilon \zeta (u^4 - u_\infty^4), \quad (14c)$$

where r_s is the surface or interface thermal resistivity, h_c is the surface or interface heat transfer coefficient, ε is the surface emissivity, and ζ is the Stefan-Boltzmann constant.

From the definition of these fluxes, a standardised, non-linear Robin boundary condition can be formed with a interfacial heat transfer coefficient, ω_c , which represents the combined effects of convection, surface thermal resistivity and Stefan-Boltzmann radiation, and takes the form:

$$-\lambda \frac{\partial}{\partial \mathbf{n}} u(\mathbf{z}) = \omega_c (u - u_\infty), \mathbf{z} \in \Gamma \quad (15a)$$

with

$$\omega_c = \frac{1}{r_s + [h_c + \varepsilon \zeta (u^2 + u_\infty^2) (u + u_\infty)]^{-1}}. \quad (15b)$$

To enforce no-flux conditions or geometrical symmetry on the continuum boundary, a Neumann condition, Equation (13b), can be used, and isothermality can be imposed on the continuum boundary via a Dirichlet condition, Equation (13a). Each of these three cases can be encapsulated in a single boundary condition formulation of the form:

$$\alpha u(\mathbf{z}) + (1 - \alpha) \frac{\partial}{\partial \mathbf{n}} u(\mathbf{z}) = \beta, \quad (16)$$

where $\alpha = 1$ yields a Dirichlet condition, $\alpha = 0$ yields a Neumann condition, and

$$\alpha = \frac{\omega_c}{\omega_c + \lambda} \quad (17a)$$

$$\beta = \frac{\omega_c u_\infty}{\omega_c + \lambda} \quad (17b)$$

yields a Robin condition equivalent to Equation (15a).

2.5 Numerical approximation of the non-linear heat conduction equation

2.5.1 The numerical method of lines

All of the solution procedures used in the present work are based on the numerical method of lines (19, §1). Temporal and spatial dimensions are discretised independently such that the heat conduction equation can be recast as an ordinary differential equation of the form:

$$\frac{\partial \mathbf{u}}{\partial \tau} = f(\mathbf{u}, \nabla \mathbf{u}, \nabla^2 \mathbf{u}, \tau, \mathbf{z}), \quad \mathbf{z} \in \Omega, \quad \tau > 0, \quad (18)$$

where f is some vector valued operator which discretises the spatial terms in the original, well posed initial value partial differential equation on the continuum Ω . For the non-linear heat conduction equation under consideration, we take f to be

$$f = \frac{1}{\gamma} \left(\lambda \nabla^2 u + \frac{\partial \lambda}{\partial u} (\nabla u)^2 + \sigma \right). \quad (19)$$

This ordinary differential equation is then integrated in time to yield the solution on both spatial and temporal domains. Spatial discretisation and temporal integration are discussed separately in the following sections.

2.5.2 Spatial discretisation by finite difference method

For the simplest case, discretising a single spatial dimension in rectilinear coordinates, the function f is assumed to be vector valued with a $\mathfrak{R}^n \rightarrow \mathfrak{R}^n$ mapping, represented as

$$f(\mathbf{u}) = \mathbf{f} = \begin{bmatrix} \phi_1 \\ \vdots \\ \phi_N \end{bmatrix}. \quad (20)$$

A given value of ϕ_i approximates the one dimensional form of equation (19):

$$\phi_i = \frac{1}{\gamma_i} \left(\frac{\partial \lambda_i}{\partial u} \left(\frac{\partial u_i}{\partial x} \right)^2 + \lambda_i \frac{\partial^2 u_i}{\partial x^2} + \sigma_i \right), \quad i \in \{2, \dots, N-1\}, \quad (21)$$

where finite differences are used to approximate the derivative terms. In the present work, we adopt either second-order or fourth-order accurate central difference approximations assuming a uniform grid. In the second-order accurate case, we use a three point stencil for both gradient and Laplacian approximations:

$$\begin{aligned} \frac{du}{dx} &= \frac{1}{2\delta} [-u_{i-1} + u_{i+1}] + \frac{\delta^3}{3} \frac{d^3u}{dx^3} + \mathcal{O}(\delta^4) \\ \frac{d^2u}{dx^2} &= \frac{1}{\delta^2} [u_{i-1} - 2u_i + u_{i+1}] + \frac{\delta^4}{12} \frac{d^4u}{dx^4} + \mathcal{O}(\delta^6). \end{aligned}$$

The second-order accurate case approximation of Equation (21) can then be expressed using a compact matrix-vector notation as:

$$\phi_i(\hat{\mathbf{u}}_i) = \frac{1}{\gamma_i} \left[\dot{\lambda}_i \hat{\mathbf{u}}_i^T \mathbf{Q}_i + \lambda_i \mathbf{k}_i \right] \hat{\mathbf{u}}_i + \left(\frac{\sigma_0}{\gamma} \right)_i \quad (22)$$

with

$$\begin{aligned} 4\delta^2 \mathbf{Q}_i &= \begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix} \\ \delta^2 \mathbf{k}_i &= \begin{bmatrix} 1 & -2 & 1 \end{bmatrix} \\ \hat{\mathbf{u}}_i &= \begin{bmatrix} u_{i-1} \\ u_i \\ u_{i+1} \end{bmatrix}. \end{aligned} \quad (23)$$

The coefficient row entries of the equivalent linear approximation $\mathbf{A}\mathbf{u} + \mathbf{b}$ can be formed in the same fashion. The spatial approximation accuracy can be improved by replacing the three point stencils used in Equation (22) with larger stencils. A fourth order accurate central scheme can be obtained using five point stencils:

$$\begin{aligned}\frac{du}{dx} &\approx \frac{1}{12\delta} [u_{i-2} - 8u_{i-1} + 8u_{i+1} - u_{i+2}] + \frac{2\delta^5}{5} \frac{d^5u}{dx^5} + \mathcal{O}(\delta^6) \\ \frac{d^2u}{dx^2} &\approx \frac{1}{12\delta^2} [-u_{i-2} + 16u_{i-1} - 30u_i + 16u_{i+1} - u_{i+2}] - \frac{2\delta^6}{15} \frac{d^6u}{dx^6} + \mathcal{O}(\delta^7),\end{aligned}$$

which leads to modified coefficient matrices and vectors:

$$\begin{aligned}144\delta^2\mathbf{Q}_i &= \begin{bmatrix} 1 & -8 & 0 & -8 & 1 \\ -8 & 64 & 0 & 64 & -8 \\ 0 & 0 & 0 & 0 & 0 \\ -8 & 64 & 0 & 64 & -8 \\ 1 & -8 & 0 & -8 & 1 \end{bmatrix} \\ 12\delta^2\mathbf{k}_i &= \begin{bmatrix} -1 & 16 & -30 & 16 & -1 \end{bmatrix} \\ \hat{\mathbf{u}}_i &= \begin{bmatrix} u_{i-2} \\ u_{i-1} \\ u_i \\ u_{i+1} \\ u_{i+1} \end{bmatrix}.\end{aligned}$$

Similarly, higher dimensional discretisation can be achieved by modified coefficient matrices and stencils. Complete details of the finite difference formulations for one-, two- and three- dimensional cases are provided in Appendix 1.2.

Numerical approximation of boundary conditions

Imposing the generalized Robin condition defined by equation (16):

$$\alpha u + (1 - \alpha) \frac{\partial u}{\partial n} = \beta \quad (24)$$

on the matrix-vector form outlined in the previous section only requires modified values of \mathbf{Q} , \mathbf{k} and σ to be applied to Equation (22).

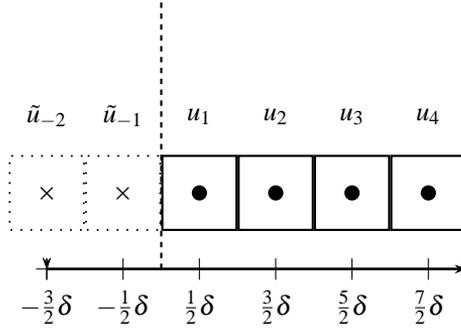


Fig 1. Ghost data arrangement on domain boundaries for cell grids.

For the Robin condition on one-dimensional, cell centred data [see Figure 1] the second order "left" boundary coefficients become

$$4\delta^2 \mathbf{Q}_0 = \begin{bmatrix} 9 & -12 & 3 \\ -12 & 16 & -4 \\ 3 & -4 & 1 \end{bmatrix}$$

$$\delta^2 \mathbf{k}_0 = \left[-\frac{9\alpha_0\delta}{\eta_0} - 1 \quad \frac{\alpha_0\delta}{\eta_0} + 1 \quad 0 \right]$$

$$\sigma_0 = \bar{\sigma}_0 + \left(\frac{8}{\delta^2} \right) \left(\frac{\beta_0}{\eta_0} \right)$$

$$\eta_0 = 3\alpha_0 - 8\alpha_0 + 8$$

$$\hat{\mathbf{u}}_0 = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

and the equivalent right boundary coefficients become

$$\begin{aligned}
4\delta^2\mathbf{Q}_N &= \begin{bmatrix} 1 & -4 & 3 \\ -4 & 16 & -12 \\ 3 & -12 & 9 \end{bmatrix} \\
\delta^2\mathbf{k}_N &= \begin{bmatrix} 0 & \frac{\alpha_N\delta}{\eta_N} + 1 & -\frac{9\alpha_N\delta}{\eta_N} - 1 \end{bmatrix} \\
\sigma_N &= \bar{\sigma}_N + \left(\frac{8}{\delta^2}\right) \left(\frac{\beta_N}{\eta_N}\right) \\
\eta_N &= 3\alpha_N - 8\alpha_N + 8 \\
\hat{\mathbf{u}}_N &= \begin{bmatrix} u_{N-2} \\ u_{N-1} \\ u_N \end{bmatrix}.
\end{aligned}$$

Complete details of the boundary condition formulations for second and fourth order accurate versions of the one-, two- and three- dimensional cases are provided in Appendix 1.3.

2.5.3 Temporal integration

To integrate Equation (18) in time, the problem time domain is firstly divided into a set of discrete time intervals $\tau : \tau \in \{\tau^{(1)}, \tau^{(2)}, \dots, \tau^{(P)}\}$, and a series of discrete temporal integration steps is used to solve the corresponding temperature values at each time, $\mathbf{u} \in \{\mathbf{u}^{(1)}, \mathbf{u}^{(2)}, \dots, \mathbf{u}^{(P)}\}$. A given integration step from $\tau = \tau^{(m)}$ to $\tau = \tau^{(m+1)}$ can be then written as

$$\int_{\tau^m}^{\tau^{m+1}} \frac{\partial \mathbf{u}}{\partial \tau} d\tau = \int_{\tau^m}^{\tau^{m+1}} f(\mathbf{u}, \nabla \mathbf{u}, \nabla^2 \mathbf{u}, \tau, \mathbf{z}) d\tau \quad (25)$$

for the *integration time step* $\Delta\tau^{(m)} = \tau^{(m+1)} - \tau^{(m)}$, which may or may not be constant. Solving Equation (25) becomes an exercise in quadrature, where the choice of scheme is dictated by the need to balance stability, accuracy, and computational complexity and expense. In this review of temporal integration, only a selected number of techniques and their implementation are minimally described. For formal analysis and proof of accuracy and stability, readers are directed to the cited literature and references.

Stiffness and numerical stability analysis for model problems

The standard stability criteria for the heat equation is derived from the classic von-Neumann analysis of finite difference approximations of the one-dimensional model linear PDE

$$\frac{\partial u}{\partial t} = \frac{\lambda}{\gamma} \nabla^2 u \approx \frac{\lambda}{\gamma \delta^2} (u_{j-1} - 2u_j + u_{j+1}) \quad (26)$$

with characteristic single step exponential solution

$$u_j = g \exp(ikj\delta). \quad (27)$$

Typically this leads to two classic single step test schemes – the explicit Euler case

$$\frac{1}{\Delta\tau} (u_j^{(m+1)} - u_j^{(m)}) = \frac{\lambda}{\gamma^2} (u_{j-1}^{(m)} - 2u_j^{(m)} + u_{j+1}^{(m)}) \quad (28)$$

with single step amplification factor

$$g = 1 + 2 \frac{\lambda \Delta\tau}{\gamma \delta^2} (\cos k\delta - 1) \quad (29)$$

and the implicit Euler case

$$\frac{1}{\Delta\tau} (u_j^{(m+1)} - u_j^{(m)}) = \frac{\lambda}{\gamma \delta^2} (u_{j-1}^{(m+1)} - 2u_j^{(m+1)} + u_{j+1}^{(m+1)}) \quad (30)$$

with single step amplification factor

$$g = \left(1 + 2 \frac{\lambda \Delta\tau}{\gamma \delta^2} (1 - \cos k\delta) \right)^{-1}. \quad (31)$$

The constant $(\lambda/\gamma)(\Delta\tau/\delta^2)$ is usually referred to as the Fourier number in the context of heat conduction, denoted here as ζ_0 . For any scheme to be deemed stable, we require the magnitude of the amplification factor to be less than unity (20, §6). For the explicit case, the amplification factor (Equation (29)) implies that stability is only attainable when

$$|g| = |1 - 4f_0| \leq 1 \quad (32)$$

or

$$|g| \leq 1 \equiv f_o \leq \frac{1}{2}, \quad (33)$$

which is to say that the scheme is stable only for a critical range of Fourier numbers. For the implicit scheme, Equation (31) implies that

$$|g| = \left| \frac{1}{1+4f_o} \right| \leq 1 \quad (34)$$

or

$$|g| \leq 1 \forall f_o, \quad (35)$$

which is to say that the implicit test equation is unconditionally stable for any positive Fourier number. This property is often referred to a A-stability (21).

We note that the Fourier number condition defined by Equation (33) is valid only for the one dimensional, second order central spatial discretisation. Using the same procedure shown in Equations (28) and (29), it can be shown that the equivalent stability condition for analogous two dimensional, second order central problem is given by

$$|g| \leq 1 \equiv f_o \leq \frac{1}{4} \quad (36)$$

and the condition for three dimensional second order problem as

$$|g| \leq 1 \equiv f_o \leq \frac{1}{6}. \quad (37)$$

Higher order spatial discretisation at any given dimension will also result in stricter bounds on the Fourier number than the second order central scheme results given here.

As is discussed in much greater detail in the next sections, the “thermal diffusion coefficient” for typical carbon steels, (λ/γ) , is of the order of 5 – 15 mm²/s over conditions of interest. In steel TMT processes, thermal length scales of interest are often less than 1 mm, and typical grid or mesh sizes might be at least an order of magnitude lower still. This implies stability limiting time steps for the explicit scheme of the order of 100 μs, which we would argue is considerably less than the minimum time scale of interest for typical heat transfer processes associated with thermomechanical treatment. On this basis alone, we suggest that these heat conduction problems can be regarded as stiff, because the stability requirements of the explicit scheme imposes time step sizes which are considerably less than the resolution of analysis requires.

Implicit integration schemes

The unconditional stability of implicit Euler scheme described in the previous section, Equation (30), would appear to make it an obvious candidate for solving stiff equations of the type under investigation here. Under this scheme, the method lines ODE given by Equation (18) becomes

$$\mathbf{u}^{(m+1)} - \mathbf{u}^{(m)} = \Delta\tau^{(m)} f\left(\tau^{(m)} + \Delta\tau^{(m)}, \mathbf{u}^{(m+1)}\right), \quad (38)$$

which can be either transformed into a linear system and solved directly, or as a non-linear problem and solved iteratively using a Newton method. This yields an unconditionally stable solution at any integration step size. This A-stable property does not, however, say anything about the accuracy of the scheme at a give step size, nor provide any obvious methodology for selecting optimal step sizes for a desired accuracy. So paradoxically, while explicit Euler type integrators have stability requirements which make them effectively “too accurate” for stiff systems by requiring infeasibly small integration step sizes, the most stable scheme is also potentially the least accurate, because is restricted to first order truncation accuracy without any useful guidance about step size selection. We therefore seek an integrator which addresses both of these shortcomings – better than first order truncation accuracy, with some sort of step size control metric which allows the integration step size to be sufficiently large to minimize computation time, while capturing all of the characteristics of the ODE being integrated with sufficient resolution.

One alternative are the family of multi-step implicit methods collectively known as backward differentiation formulas (BDF). These have been very widely applied to the integration of stiff ODEs (22, §3) , and usually take the form

$$\sum_{j=0}^s a_j \mathbf{u}^{(m+1-j)} = \Delta\tau^{(m)} f\left(\tau^{(m)} + \Delta\tau^{(m)}, \mathbf{u}^{(m+1)}\right) \quad (39)$$

and effectively improves the accuracy order of the solution by using a higher order truncation accuracy approximation for the LHS derivative term, while retaining the basic computational structure of the implicit Euler step. The simplest second and third order BDF schemes are given by Equations (40) and (41) respectively:

$$\frac{3}{2}\mathbf{u}^{(m+1)} - 2\mathbf{u}^{(m)} + \frac{1}{2}\mathbf{u}^{(m-1)} = \Delta\tau^{(m)} f\left(\tau^{(m)} + \Delta\tau^{(m)}, \mathbf{u}^{(m+1)}\right) \quad (40)$$

$$\frac{11}{6}\mathbf{u}^{(m+1)} - 3\mathbf{u}^{(m)} + \frac{3}{2}\mathbf{u}^{(m-1)} - \frac{1}{3}\mathbf{u}^{(m-2)} = \Delta\tau^{(m)} f\left(\tau^{(m)} + \Delta\tau^{(m)}, \mathbf{u}^{(m+1)}\right). \quad (41)$$

Several features of these BDF schemes are obvious, particularly that they rely on solution values from earlier time steps, which complicates starting the schemes from a single initial condition, and the truncation order accuracy of the BDF formula is predicated on a constant time step, making the construction of variable step schemes using these formulas less than straightforward.

In the present work, we adopt a different type of integrator, which relies on using higher order accurate RHS function approximations to improve integrator accuracy while maintaining acceptable stability. These are known as implicit Runge-Kutta methods, and are computed using the generalised formula:

$$\mathbf{u}^{(m+1)} = \mathbf{u}^{(m)} + \Delta\tau^{(m)} \sum_{i=1}^s b_i \mathbf{f}_i \quad (42)$$

with

$$\mathbf{f}_i = f\left(\tau^{(m)} + c_i \Delta\tau^{(m)}, \mathbf{u}^{(m)} + \Delta\tau \sum_{j=1}^s a_{ij} \mathbf{f}_j\right). \quad (43)$$

The set of coefficients $\{a_{ij}\}$, weights $\{b_i\}$, and quadrature points $\{c_i\}$ can be assembled into a convenient device usually referred to as a Butcher tableau (23, §23), which can be written for a scheme of s stages as

$$\begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b} \end{array} = \begin{array}{c|ccc} c_1 & a_{11} & \dots & a_{1s} \\ \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & \dots & a_{ss} \\ \hline & b_1 & \dots & b_s \end{array}.$$

We note that the form of Equations (42) and (43) are no different from the familiar explicit Runge-Kutta schemes used for non-stiff ODE integration. The difference lies only in the form of the coefficient matrix \mathbf{A} , with explicit schemes requiring that \mathbf{A} be strictly lower triangular, while implicit schemes feature non-zero coefficients on the diagonal or in the upper triangular portion of \mathbf{A} .

All of the time integration schemes used in the current work have Butcher tableaux where the coefficient matrix \mathbf{A} is lower triangular with non-zero diagonal entries. Schemes with this property are most commonly referred to as diagonally implicit Runge-Kutta schemes, and have been widely applied to integrating stiff ordinary differential equations (24, 21, 25). These schemes have the distinct advantage of being multi-stage repetitions of the implicit Euler method, and their stages can either be solved sequentially, as with an explicit Runge-Kutta scheme, or solved simultaneously via Newton-Raphson iteration. Compared with BDF formulas, these diagonally implicit schemes have several advantages, especially that they are “self starting”, requiring no solution values from previous steps, and they are readily adaptable to use in variable time step schemes because changes in step size has not effect on truncation order accuracy.

Throughout the rest of this monograph, we will use the nomenclature proposed by Alexander (21), so that a DIRK(s, p) scheme implies a diagonally implicit Runge-Kutta scheme of s stages, with truncation order accuracy of p . All of the temporal integration performed in the present worth uses one of three DIRK schemes – the two stage, second order DIRK(2, 2) scheme:

0	0
1	0.5 0.5
	0.5 0.5

the three stage, second order DIRK(3, 2) scheme:

0	0		
0.58578644	0.29289322	0.29289322	
1	0.35355339	0.35355339	0.29289322
	0.35355339	0.35355339	0.29289322

and the four stage, third order DIRK(4, 3) scheme:

0	0			
0.87173304	0.43586652	0.43586652		
0.46823874	0.14073777	-0.10836555	0.43586652	
1	0.1023994	-0.37687845	0.83861253	0.43586652
	0.1023994	-0.37687845	0.83861253	0.43586652

A more complete discussion of the integration schemes used in the balance of the monograph is presented in Appendix 2.

2.6 Thermophysical properties of steels

In the previous section, a non-linear variant of the heat conduction equation was derived with only minimal reference to the values of key coefficients - thermal conductivity, specific heat capacity, and mass density. The implicit justification for the additional mathematical complexity of this non-linear form is that the materials under consideration in the present work have non-constant, temperature dependent thermophysical coefficients. Here, we briefly review the thermophysical behaviour of ferrous alloys over the temperature range of interest in thermomechanical treatment of steels.

2.6.1 Thermal conductivity

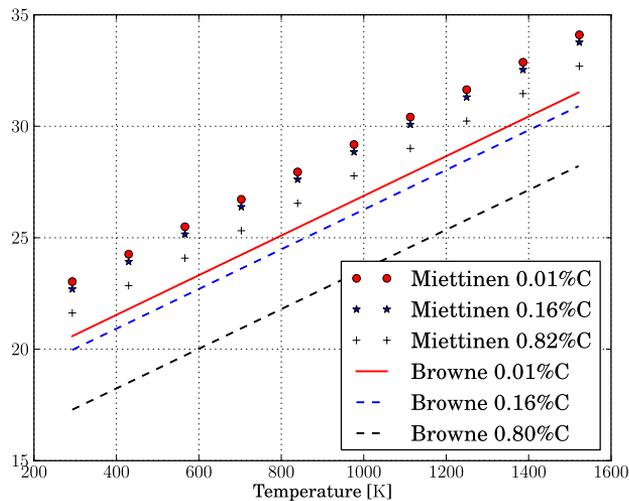


Fig 2. Thermal conductivity of γ -Fe for a composition corresponding to C-0.6Mn-0.2Si over the temperature range 298 K–1500 K. Model predictions after Browne (26) and Miettinen (27).

Browne (26) and Miettinen (27) present largely similar linear models for the thermal conductivity of austenite, as shown in Figure 2. While Browne’s calculations only

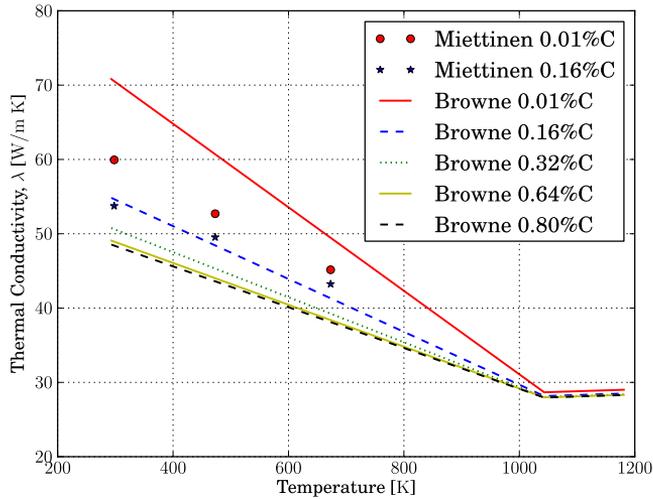


Fig 3. Thermal conductivity of α -Fe + cementite for a composition corresponding to C-0.6Mn-0.2Si over the temperature range 298 K–1184 K. Model predictions after Browne (26) and Miettinen (27).

consider the effects of carbon on austenite thermal conductivity, Miettinen's analysis includes the effects of other solute species, which results in a higher net thermal conductivity by approximately ten percent for typical mild steel compositions. Both authors predict that increasing carbon content will reduce thermal conductivity, such that the difference between pure iron and a eutectoid composition (0.82%C) is approximately 20%. In both cases the gradient of the thermal conductivity with respect to temperature is very similar, with thermal conductivity predicted to monotonically decrease with temperature.

Browne also presents a linear model for ferrite-cementite mixtures, while Miettinen provides limited calculations for ferrite-cementite at three temperatures. As for the austenite case, there is reasonable agreement between the two analyses, although the Browne predictions are somewhat higher. What is worth remarking upon is the negative sign of the thermal conductivity gradient with respect to temperature gradient below the Curie temperature (1042 K), which is the opposite of that austenite and paramagnetic ferrite. As with austenite, higher carbon content reduces the overall thermal conductivity, although this is the result of the presence of cementite, rather than the effect of carbon in solid solution.

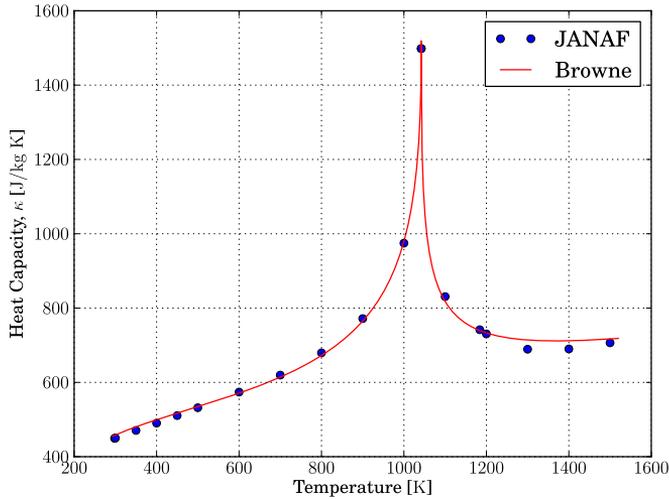


Fig 4. Specific heat capacity of pure α -Fe over the temperature range 298 K–1184 K. Experimental data taken from the NIST-JANAF thermochemical tables (28, p.1222), model predictions based on a corrected version of the model proposed by Browne (12) (Full details are given in Appendix 3).

2.6.2 Mass density

Most available literature points to the use of simple linear models for mass density of carbon steels in the solid state. In the present work we use the relationships proposed by Miettinen (27), which uses a set of influence factors to account for the mass density effects of common alloying elements. The density of ferrite is predicted to be approximately two percent lower than austenite at the same temperature over the temperature range of interest in thermomechanical treatment processes. The mass density of a reference AISI 1016 composition in the austenitic and ferrite states over the temperature range 298 – 1500 K are shown in Figure 6.

2.6.3 Specific heat capacity

The NIST-JANAF (28) tabulated specific heat capacity data for α -Fe and γ -Fe are shown plotted against temperature in Figures 4 and 5, along with model predictions based on a corrected version of the semi-empirical model proposed by Browne (12). Full details of this calculation are given in Appendix 3.

We note that the Debye temperatures of austenite and ferrite are rather similar

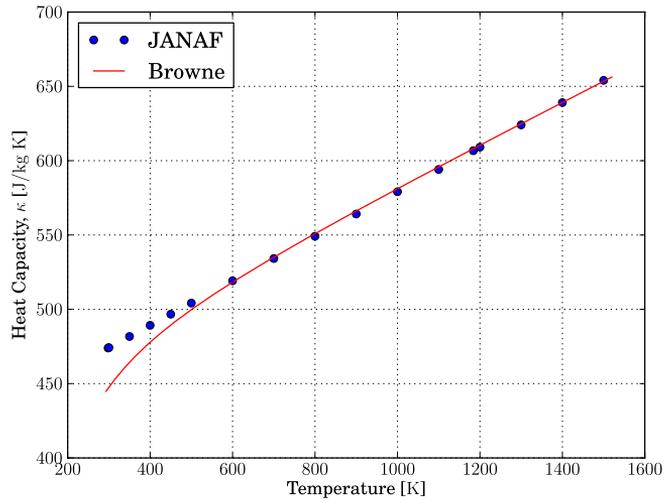


Fig 5. Specific heat capacity of pure γ -Fe over the temperature range 298 K–1500 K. Experimental data taken from the NIST-JANAF thermochemical tables (28, p.1222), model predictions based on the model proposed by Browne (12).

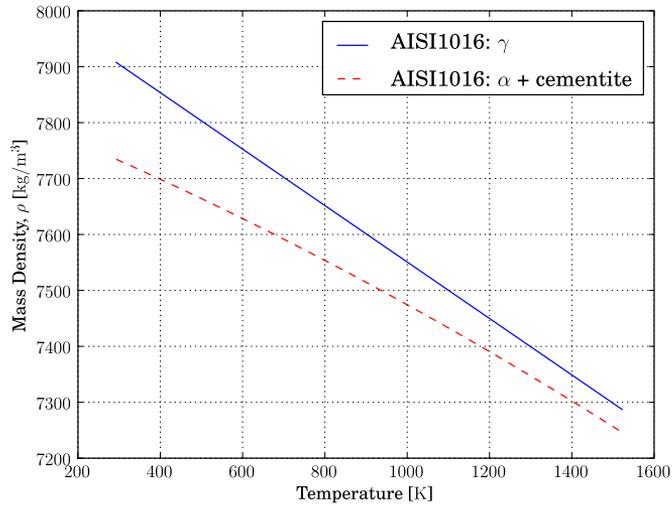


Fig 6. Mass density for a composition corresponding to AISI 1020 (0.2C-0.6Mn-0.2Si) in the austenitic and ferritic states over the temperature range 298 K–1500 K. Model predictions after Miettinen (27).

– 425 K for ferrite versus 400 K, which should imply rather similar heat capacity at constant volume for both phases. Comparing Figures 4 and 5, however, shows a very large difference in specific heat capacity around the Curie temperature, attributable to the paramagnetic-ferromagnetic transition in ferrite (29). At 1042 K, the specific heat capacity of ferrite is over two times larger than when the magnetic contribution is excluded, and almost three times that of austenite at the same temperature. The magnetic specific heat capacity component of ferrite, and by extension other ferromagnetic phases of iron, has significant thermal implications when phase transformation occurs at temperatures significantly below the Curie Temperature. This phenomena is discussed in more detail in the following section.

2.6.4 Transformation enthalpy

Solid-state transformations in steels are often accompanied by enthalpy changes, and these should be accounted for in any thermal model of processes featuring phase transformation. In the present work, the calculation methodology largely follows those proposed by Suehiro *et al* (9, 10) and Browne (12). All quantities are taken as kJ/kg and temperatures in K unless explicitly noted otherwise.

We assume that the thermophysical behaviour of any proeutectoid steel can be adequately approximated using an analogous Iron-Carbon binary alloy. While this assumption is probably inadequate for highly alloyed steels, it should be a reasonable simplification for simple plain carbon and low-alloy steels. Using the thermodynamic approach of Zener (30), we assume that any microstructure formed by decomposition of austenite containing less than the eutectoid carbon content will contain only some combination of the following constituents:

1. Ferrite
2. Cementite
3. Carbon supersaturated ferrite

Both bainites and martensite are approximated as carbon supersaturated ferrites, with transformation enthalpies adjusted for the presence or absence of cementite and internal strain energy. The effective combined transformation enthalpy is presumed to be the result of three separate energy sources

1. Iron lattice transformation enthalpy
2. Magnetic specific heat enthalpy
3. Cementite formation enthalpy

which sum according to

$$\bar{\mu} = \mu_0 + \mu_1 + \mu_2. \quad (44)$$

The lattice transformation enthalpy of ferritic phases is taken as a constant $\mu_0 = -16.7$ kJ/kg (9). For martensite, the lattice transformation enthalpy is taken as the same as that of ferrite, corrected for the presence of internal strain energy in the BCT martensite lattice. Zener suggests that the lattice strain energy associated with martensite formation is a constant 21.7 kJ/kg of transformed iron (30, p.570) (although he notes that the final strain energy of martensites formed at different temperatures will differ because of differing degrees of post transformation relaxation). Assuming that this change in free energy of formation compared to that of ferrite and bainite is reflected in the lattice transformation enthalpy of martensite, the net lattice transformation energy of martensite is taken as the sum of the ferrite lattice transformation enthalpy and the lattice stored strain energy. Because these two quantities are of approximately the same magnitude, this is simplified to a net zero martensite lattice transformation energy, ie. $\mu_0 = 0$ kJ/kg.

Magnetic transformation enthalpy is calculated by integrating the magnetic heat capacity function of ferromagnetic phases below the Curie temperature of iron, such that

$$\mu_1 = - \int_{u_0}^{u_{Curie}} \kappa_{\text{magnetic}} du. \quad (45)$$

Following the approximations of Browne (12) and Chuang *et al* (29), we propose an empirical approximation of this integral:

$$-\mu_1 = \begin{cases} 6360.1 - 5.8425 [u - 260.50] \exp\left(-4 \left[1 - \frac{u}{u_c}\right]\right) & u \leq u_c \\ -340.10 + 1.9027 [u + 79.747] \exp\left(13.066 \left[1 - \frac{u}{u_c}\right]\right) & u_c < u \leq u_{\text{Ae}3} \\ 0 & \text{otherwise} \end{cases} \quad (46)$$

where u_c denotes the Curie temperature of iron, and $u_{\text{Ae}3}$ denotes the Ae3 temperature of the steel in question, both in K, and μ_1 is taken in J/kg. A fuller description of the derivation and fitting of this expression can be found in Appendix 3.2.

The Cementite formation enthalpy is calculated using the relationship proposed by Chipan (31):

$$-\mu_2 = 586.9 \left(\frac{n_C}{n_{Fe}} \right) + 1323 \left(\frac{n_C}{n_{Fe}} \right)^2, \quad (47)$$

where the ratio n_C/n_{Fe} is the molar ratio of carbon to iron in the transforming austenite. We note that cementite transformation enthalpy is only expected in pearlitic and bainite phases where the carbide is observed to form. For proeutectoid ferrites and martensites, the cementite transformation heat is taken as zero at all temperatures.

Figure 7 shows these components of the transformation enthalpy plotted as functions of temperature. As discussed by Browne (12), magnetic latent enthalpy is by far the largest single component of the combined transformation enthalpy, approaching an order of magnitude larger than either the lattice transformation or cementite formation enthalpy when transformation temperatures are significantly below the Curie temperature. Browne and Suehiro *et al* (9) also concur that much of the recalescence phenomena observed in pearlitic steels during cooling from the austenitic state is due to the combined effects of low transformation start temperatures from the carbon content and the resulting magnetic latent enthalpy which evolves during the transformation at those low transformation temperatures.

The net rate of transformation latent heat evolution for a given phase is then computed as

$$\sigma_i = -\frac{\partial}{\partial \tau} \left[(\rho_i \bar{\mu}_i) \chi_i \right], \quad (48)$$

where χ_i is the volume fraction of the product phase. This can be simplified using the assumption that all enthalpy components can be regarded time independent to

$$\sigma_i = -\rho_i \bar{\mu}_i \frac{\partial \chi_i}{\partial \tau}. \quad (49)$$

This net latent heat evolution rate expression can be substituted into the heat equation as a volumetric heat source, as discussed in Section 2.3.

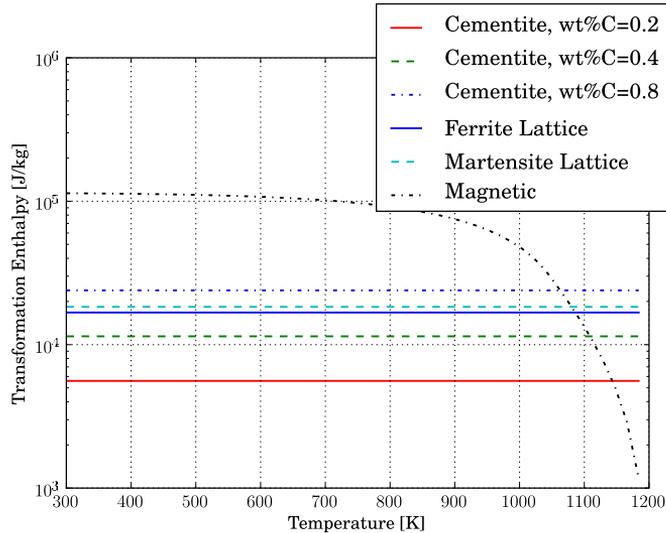


Fig 7. Predicted transformation enthalpy components associated with lattice transformation, magnetic specific heat and cementite formation at temperatures below 1184 K. Model predictions made using the models described in Appendix 3.2. Note logarithmic scale for enthalpies.

2.6.5 The computational implications of microstructure evolution

From the previous sections, we believe the following is a reasonable summary of the thermophysical properties of carbon steels as they apply to the present analysis:

- All of the important thermophysical properties of steels are, at least to some degree, temperature dependent.
- Quantities with some relationship to specific heat capacity can be highly non-linear, particular for “ferrite like” transformation products which exhibit ferromagnetic behaviour.
- Key thermophysical properties of austenite and ferrite (and by extension “ferrite like” transformation products) are considerably dissimilar.
- Carbon and carbides play an important role in determining several important thermophysical quantities, particularly thermal conductivity and transformation related latent heat.

and we suggest that it is, therefore, unrealistic to treat the thermophysical properties of steels undergoing solid-state transformations without reference to the underlying

transformation processes themselves.

In light of this, we represent the effect of different transformation products on net thermophysical coefficients using a simple law of mixtures rule, based the assumption that individual transformation product characteristic length scales are small relative to the dimensions of the smallest control volume in the spatial discretisation of the continuum. For the purposes of the current analysis, we classify the possible range of transformation products into four possible categories:

- proeutectoid ferrite
- pearlite
- bainites
- martensites

and we propose the following governing volume conservation equation

$$1 - \chi_a - \chi_f - \chi_p - \chi_b - \chi_m = 0, \quad (50)$$

where χ denotes the volume fraction of a given transformation product, and the subscripts a, f, p, b, and m identify the quantity as being associated with untransformed austenite, pro-eutectoid ferrite, pearlite, bainites and martensites, respectively. This scheme allows the complete range of transformation products to be represented using four degrees of freedom per control volume, with the volume fraction of austenite implicitly derived from sum of the volume fractions of these four transformation products. Any given thermophysical coefficient can then be calculated using an inner product of a coefficient vector with a vector of volume fractions. For the example of thermal conductivity, this approach yields:

$$\lambda = \lambda_a + \begin{bmatrix} \Delta\lambda_f & \Delta\lambda_p & \Delta\lambda_b & \Delta\lambda_m \end{bmatrix} \begin{bmatrix} \chi_f \\ \chi_p \\ \chi_b \\ \chi_m \end{bmatrix} \quad (51)$$

with

$$\begin{aligned}\Delta\lambda_f &= \lambda_f - \lambda_a \\ \Delta\lambda_p &= \lambda_p - \lambda_a \\ \Delta\lambda_b &= \lambda_b - \lambda_a \\ \Delta\lambda_m &= \lambda_m - \lambda_a.\end{aligned}$$

This equation can be used for any time independent, extensive quantity. For time independent intensive properties, such as mass density, a reciprocal form is required:

$$\frac{1}{\rho} = \frac{1}{\rho_a} + \begin{bmatrix} \frac{1}{\Delta\rho_f} & \frac{1}{\Delta\rho_p} & \frac{1}{\Delta\rho_b} & \frac{1}{\Delta\rho_m} \end{bmatrix} \begin{bmatrix} \chi_f \\ \chi_p \\ \chi_b \\ \chi_m \end{bmatrix} \quad (52)$$

while for the case of latent heat of transformation, the volume fractions are replaced with volumetric evolution rates, such that

$$\bar{\sigma} = - \begin{bmatrix} (\rho\bar{\mu})_f & (\rho\bar{\mu})_p & (\rho\bar{\mu})_b & (\rho\bar{\mu})_m \end{bmatrix} \begin{bmatrix} \partial\chi_f/\partial\tau \\ \partial\chi_p/\partial\tau \\ \partial\chi_b/\partial\tau \\ \partial\chi_m/\partial\tau \end{bmatrix}. \quad (53)$$

The changes in microstructure during cooling due to diffusive transformations are presumed to be described by the theory proposed by Kirkaldy and Venugopalan (32). Following the clarified version of this theory proposed by (15), In this approach, the volume fraction of transformation product at any given time is assumed to be calculable as

$$\chi_j(\tau) = \int_0^\tau \frac{\partial\chi_j}{\partial\tau} d\tau = \int_0^\tau g_j(\chi_j, \chi_j^{\max}) d\tau, \quad (54)$$

where χ_j^{\max} denotes the equilibrium or non-equilibrium maximum volume fraction of the j transformation product. In this theory, all reactions are presumed to proceed to completion. For the pro-eutectoid ferrite reaction, the maximum volume fraction formed is computed for equilibrium conditions auxiliary function $d_f(u)$ which approximates the hypo-eutectoid portion of the iron carbon diagram to estimate the equilibrium volume

fraction of ferrite at any temperature. The maximum amount of pearlite which can form under equilibrium conditions is then $1 - d_f(u)$. For bainite, the volume conservation statement given by Equation (50) is used to compute the maximum feasible volume fractions, such that the phase fraction limits for the three diffusive transformations are computed as

$$\chi_f^{\max} = d_f(u) \quad (55)$$

$$\chi_p^{\max} = 1 - \chi_f^{\max} \quad (56)$$

$$\chi_b^{\max} = 1 - \chi_f - \chi_p. \quad (57)$$

This approach is permissible because each feasible reaction is presumed to proceed sequentially as temperature decreases during cooling. The overall rate equation for a given transformation product is calculated as

$$\frac{\partial \chi_j}{\partial \tau} = g_j(u, \xi) = \beta_j(u, \xi)(u) \left[\xi_j^{\alpha(1-\xi)_j} (1 - \xi_j)^{\alpha \xi_j} \right], \quad (58)$$

where ξ_j is a normalised fraction reacted, and β_j is a reaction specific function of temperature, chemistry and microstructure. For the pro-eutectoid ferrite and pearlite reactions, the volume fraction reacted is taken as

$$\xi_j = \frac{\chi_j}{\chi_j^{\max}}. \quad (59)$$

When bainite forms after the commencement of the pearlite formation, the reaction is presumed to continue from the same austenite-pearlite interface, and the resulting fraction reacted is calculated as

$$\xi_b = \frac{\chi_p + \chi_b}{\chi_p^{\max}}. \quad (60)$$

For the martensite transformation, the modified Koistinen-Marburger relationship proposed by Brokate and Sprekels (33, §8) is used:

$$\chi_m = \chi_m^{\max} [1 - \exp(-\eta(u - u_{ms}))], \quad (61)$$

where u_{ms} denotes the martensite start temperature. This relationship is transformed into a time dependent ODE given by

$$g_m(u) = -\eta \chi_m^{\max} \exp[-\eta \Delta u_{ms}] \widehat{\frac{\partial u}{\partial \tau}}, \quad (62)$$

where

$$\chi_m^{\max} = 1 - \chi_f - \chi_p - \chi_b \quad (63)$$

$$\widehat{\frac{\partial u}{\partial \tau}} = \mathcal{H} \left(-\Delta u_{ms} \frac{\partial u}{\partial \tau} \right) \frac{\partial u}{\partial \tau} \quad (64)$$

$$\Delta u_{ms} = u_{ms} - u. \quad (65)$$

Here \mathcal{H} denotes a smooth approximation of the Heaviside function

$$\mathcal{H}(v) = \frac{1}{2} \lim_{\varepsilon \rightarrow 0} \left[1 + \tanh \left(\frac{v}{\varepsilon} \right) \right], \quad (66)$$

which enforces irreversibility by both preventing martensite formation above the martensite start temperature and any reversion of the reaction in the case of temperature increase below the martensite start temperature.

The complete set of four transformation reaction ODEs can be assembled as

$$\frac{\partial}{\partial \tau} \begin{bmatrix} \chi_f \\ \chi_p \\ \chi_b \\ \chi_m \end{bmatrix} = \begin{bmatrix} \delta_f(1 - \delta_m)(1 - \delta_b)(1 - \delta_p)g_f(\chi_f, \chi_f^{\max}, u) \\ \delta_p(1 - \delta_m)(1 - \delta_b)g_p(\chi_p, \chi_p^{\max}, u) \\ \delta_b(1 - \delta_m)g_b(\chi_b, \chi_b^{\max}, u) \\ g_m(\chi_m^{\max}, u, \partial u / \partial \tau) \end{bmatrix}, \quad (67)$$

where δ_j is a numerical device introduced to enforce irreversibility about the reaction start temperature, defined as

$$\delta_j(u) = \begin{cases} 1 & u \leq u_{js} \\ 0 & \text{otherwise} \end{cases} \quad (68)$$

and u_{js} is the start temperature of the transformation for phase j . In the numerical implementation this is approximated using a smoothed Heaviside function

$$\delta_j \approx \mathcal{H}(u_{js} - u). \quad (69)$$

Further implementation details of these calculations are provided in Appendix 3.

To illustrate the implications of this law of mixtures approach to thermophysical coefficient calculate, a number of calculations have been made for a reference mild steel

(0.2C-0.6Mn-0.2Si, corresponding to a standard silicon killed AISI 1020 or ASTM A1011 SS grade 36 structural grade) which is continuously cooled at one of three different cooling rates. The changes microstructure during cooling from an assumed austenitic initial state have been predicted using this theory by integrating the ODE system defined by Equation (67) using the DIRK(3, 2) integrator.

Three different constant cooling rates have been assumed for the purposes of this numerical experiment - 8 K/s, 80 K/s and 160 K/s. These cooling rates were selected purely on the basis that they permit the microstructure evolution theory to simulate the formation of three distinct types of microstructures. We do not mean to suggest that these cooling rates are necessarily industrially relevant or even technically feasible, or that such a simple mild steel would be heat treated in this fashion. The starting microstructure of the steel was assumed to be fully austenitic, with a constant austenite ASTM grain size number of 7 (approximately 40 μm) in each case. The steel is assumed to cool linearly from the starting temperature of 1500 K. We stress that no heat conduction calculations were performed in these continuous cooling examples, with the temperature externally prescribed by the starting temperature and constant cooling rate. The temperature of the reference volume is assumed to be externally governed to achieve the desired cooling rate in each case. This is approximately analogous to the conditions achieved in a continuous cooling experiment performed on a laboratory dilatometer.

The 8 K/s case microstructure predictions are shown in Figure 8. In this slow cooling case, the austenite is predicted to decompose into proeutectoid ferrite initially, and then more slowly into pearlite, with a final microstructure of approximately 77% proeutectoid ferrite by volume, with the balance pearlite.

The 80 K/s case microstructure predictions are shown in Figure 9. In this fast cooling case, the austenite is predicted to decompose rather rapidly into proeutectoid ferrite, with the cooling rate sufficiently high to permit the remaining austenite to be cooled below the bainite start temperature, leading to a final microstructure of approximately 68% proeutectoid ferrite by volume, with the balance bainite.

The 160 K/s case microstructure predictions are shown in Figure 10. In this ultra-fast cooling case, the austenite is predicted to commence decomposing into proeutectoid ferrite, with pearlite and bainite formation avoided, permitting the untransformed austenite to be cooled below the martensite start temperature, resulting in a final microstructure of approximately 45% proeutectoid ferrite and 55% martensite by volume.

Figure 12 shows the predicted austenite carbon content as a function of temperature for the three cooling rates. In the 8 K/s example, the largest volume fraction of

proeutectoid ferrite forms, and at the highest temperatures. This leads to a relatively small volume fraction of residual austenite, but with greatly elevated carbon concentration. The resulting pearlite which forms from this austenite is rather close to the equilibrium eutectoid carbon concentration. For the two faster cooling rates, the volume fraction of proeutectoid ferrite which forms is considerably lower, producing a less pronounced partitioning of carbon into the residual austenite at the end of the proeutectoid ferrite reaction, and the resulting bainites and martensites which form from that austenite have lower carbon contents as a result.

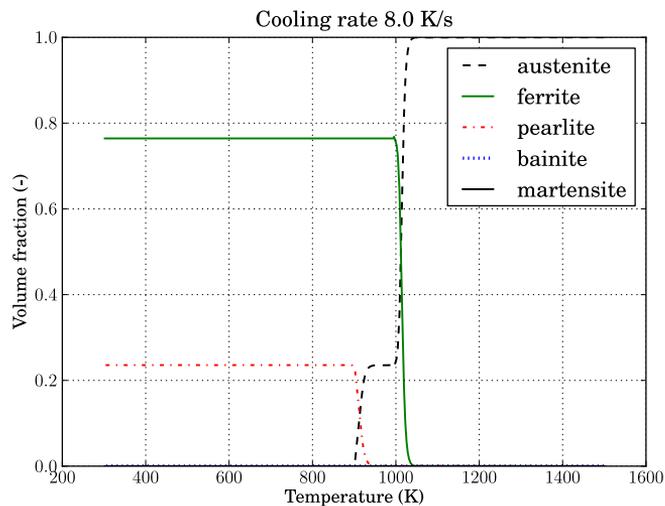


Fig 8. Predicted microstructure evolution for an AISI 1020 (0.2C-0.6Mn-0.2Si) steel during continuous cooling at 8 K/s from 1500 K assuming a constant austenite ASTM grain size number of 7. Full details of the calculations are given in Appendix 3.

These three sets of microstructure evolution data were then used to predict the corresponding law of mixtures thermal-physical coefficients using the formulation just described.

Figure 13 shows the thermal conductivity for the three cooling rates. Each case initially follows the same austenite thermal conductivity behaviour until proeutectoid ferrite begins to form, after which the cases deviate, depending on the amount of proeutectoid ferrite which forms. This occurs both because of the ferrite volume itself, but also due to the effect of carbon partitioning between the newly formed ferrite and

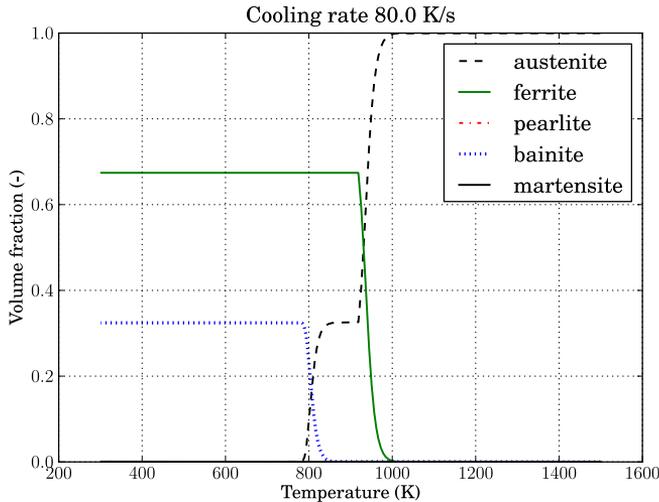


Fig 9. Predicted microstructure evolution for an AISI 1020 (0.2C-0.6Mn-0.2Si) steel during continuous cooling at 80 K/s from 1500 K assuming a constant austenite ASTM grain size number of 7. Full details of the calculations are given in Appendix 3.

the untransformed austenite, which has the effect of increasing the carbon content and lowering the thermal conductivity of the residual austenite (refer to Figures 2 and 12). The deviation between the three cases at the cessation of austenite decomposition is mostly due to the effect of the relative proeutectoid ferrite and cementite volume fractions, the latter being present the ferrite-pearlite and ferrite-bainite mixtures, and absent where martensite is formed.

Figure 14 shows the specific heat capacity of the same transforming mixtures. The three cases show variation in heat capacity of almost 30%, which is mostly caused by the volume fraction of total ferrite in the microstructure and the temperature at which it forms relative to the Curie temperature. The double peaks visible in the 8 K/s and 80 K/s cases come from the two stages of austenite decomposition reactions, initially proeutectoid ferrite, and then either pearlite or martensite.

Figure 11 shows the mass density as a function of temperature for the three cooling rates. In each case, the law of mixtures calculations is effectively making a transition from the austenite to ferrite density functions following the net rate of austenite decomposition. We note that the density of each of the austenite decomposition products – proeutectoid ferrite, pearlite and martensite – are assumed to be the same.

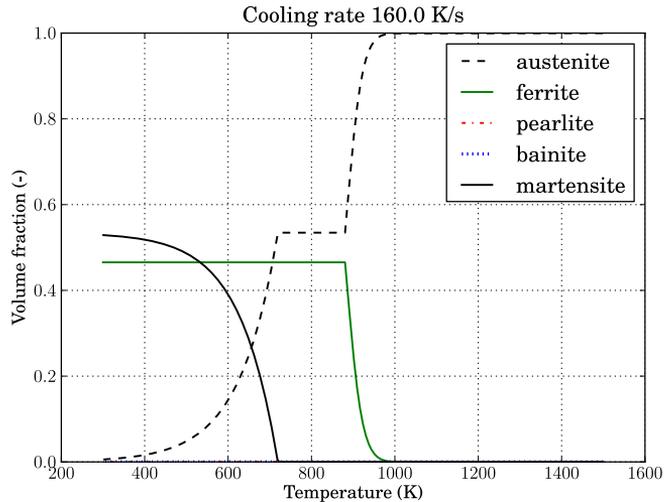


Fig 10. Predicted microstructure evolution for an AISI 1020 (0.2C-0.6Mn-0.2Si) steel during continuous cooling at 160 K/s from 1500 K assuming a constant austenite ASTM grain size number of 7. Full details of the calculations are given in Appendix 3.

The transformation latent heat for the three cooling rates is shown in Figure 15. We note there is a marked difference in latent heat of transformation between the slowest cooling rate and the two quenching cases. This is mostly due to a large fraction of the total proeutectoid ferrite forming relatively slowly at temperatures above the Curie temperature, so that the magnetic transformation enthalpy is relatively low compared to the faster cooling cases and the rate of heat dissipation is also lower. For the 80 K/s and 160 K/s cases, a much larger fraction of the proeutectoid ferrite forms below the Curie temperature and at faster rates, which greatly increases the net latent heat.

For the faster cooling rates, where bainite and martensite forms, we see large secondary latent heat peaks at lower temperatures. The martensite peak is somewhat lower in peak magnitude than the Bainite peak despite the higher lattice enthalpy, mostly because it is assumed that no cementite forms along with martensite, while the opposite is true for the bainitic transformation.

Finally, Figure 16 shows the equivalent thermal diffusivity calculated as the ratio λ/γ using the thermal conductivity, specific heat capacity and density values computed for each of the three cooling cases. While there is only about 8% variation in the final thermal conductivity at low temperatures, in the region of peak transformation activity between 800 K and 1050 K, the differences are considerably larger.

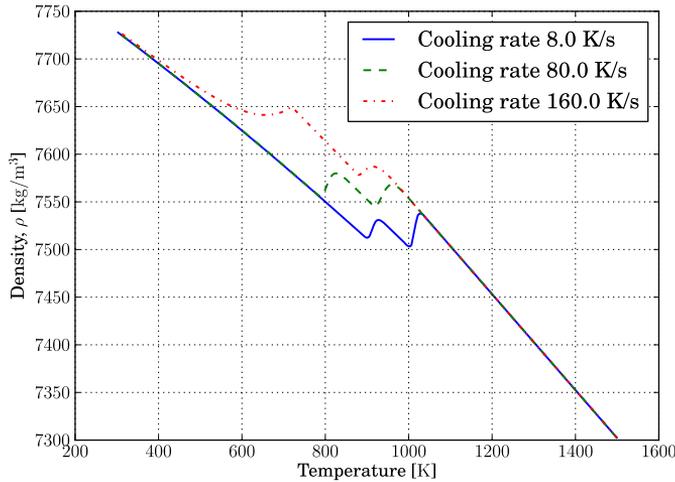


Fig 11. Mass density for a composition corresponding to AISI 1020 (0.2C-0.6Mn-0.2Si) undergoing three different continuous cooling treatments in the range 298 K–1500 K (illustrated in Figures 8, 9, and 10). Model predictions after Miittinen (27).

On the basis of these results, we suggest that there is an inextricable linkage between microstructure evolution, thermophysical properties and heat conduction. (12) suggests that the accuracy of any simulation temperature in steel thermomechanical treatment processes should be less than 10 K, or 1%, if the simulation is to contribute no appreciable variation to either process or final product. We opine that this degree of accuracy is simply not achievable without considering both microstructure evolution and heat conduction processes as a fully linked system.

2.6.6 Differentiability of thermophysical coefficients

The method of lines discretisation of non-linear heat conduction equation being considered in the present work, Equation (19), requires the computation of a thermal conductivity temperature derivative, while the Jacobian matrix formulation (refer to Section 1.5 for full details) also requires temperature derivatives of the heat capacity, latent heat of transformation, and the second temperature derivative of the thermal conductivity. As has been outlined in the previous sections, apart from significant non-linearity of some of these quantities, the mathematical functions used to describe them are not always smooth or continuous functions, and thus not differentiable at all

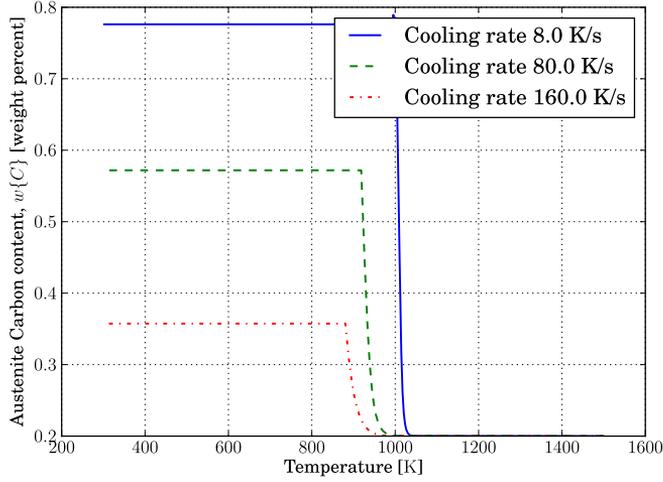


Fig 12. Austenite carbon content during phase transformation for a composition corresponding to AISI 1020 (0.2C-0.6Mn-0.2Si) undergoing three different continuous cooling treatments in the range 298 K–1500 K (illustrated in Figures 8, 9, and 10).

points on the feasible temperature domain. Therefore, some care is required in the calculation or approximation of the derivatives of these coefficients.

As for the spatial discretisation of the heat conduction terms, finite difference approximations are used to compute the coefficient derivatives, but with some modifications compared to the spatial differences already described. For a given coefficient described by the function $y(u)$ on a reference volume, we begin by evaluating the function about the volume centroid temperature u_0 to yield three coefficient values:

$$y^+ = y(u_0 + \Delta u) \quad (70)$$

$$y^0 = y(u_0) \quad (71)$$

$$y^- = y(u_0 - \Delta u) \quad (72)$$

and then compute the forward and backward divided differences:

$$\Delta y^+ = y^+ - y^0 \quad (73)$$

$$\Delta y^- = y^0 - y^-. \quad (74)$$

The values of these divided differences are then used to determine how a given

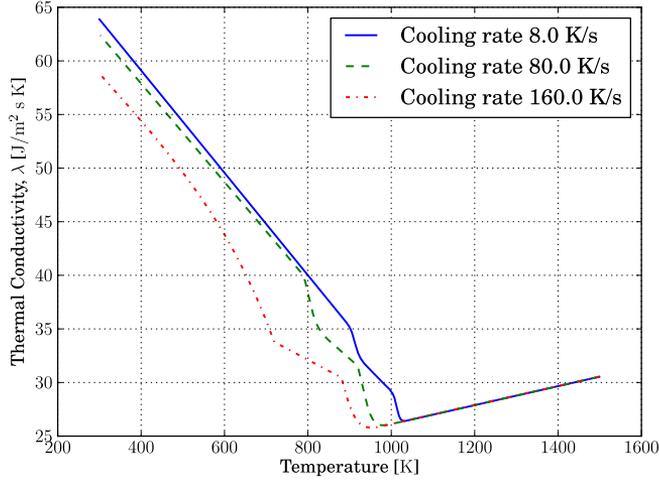


Fig 13. Thermal conductivity for a composition corresponding to AISI 1020 (0.2C-0.6Mn-0.2Si) undergoing three different continuous cooling treatments in the temperature range 298 K–1500 K (illustrated in Figures 8, 9, and 10). Model predictions after Browne (26).

coefficient derivative should be approximated, using one of two possible cases

Case 1. $\Delta y^+ \Delta y^- > 0$

$$\left. \frac{\partial y}{\partial u} \right|_{u=u_0} = \frac{1}{\Delta u} \text{absmin}(\Delta y^+, \Delta y^-) \quad (75)$$

$$\left. \frac{\partial^2 y}{\partial u^2} \right|_{u=u_0} = \frac{1}{\Delta u^2} (\Delta y^+ - \Delta y^-) \quad (76)$$

with

$$\text{absmin}(x, y) = \begin{cases} x & |x| < |y| \\ y & \text{otherwise} \end{cases} \quad (77)$$

Case 2. $\Delta y^+ \Delta y^- \leq 0$

$$\left. \frac{\partial y}{\partial u} \right|_{u=u_0} = 0 \quad (78)$$

$$\left. \frac{\partial^2 y}{\partial u^2} \right|_{u=u_0} = 0 \quad (79)$$

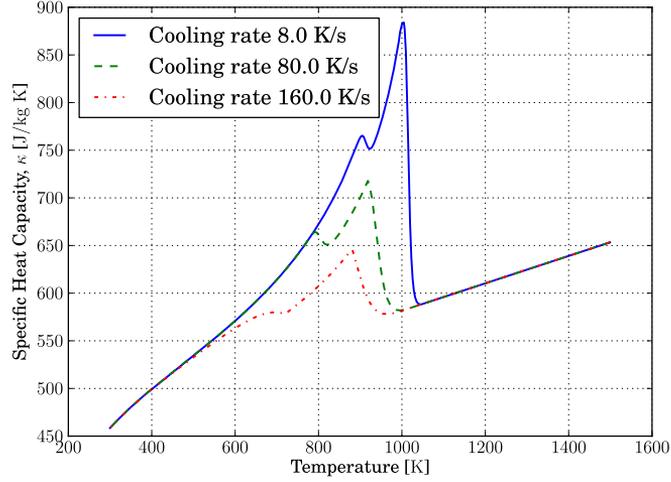


Fig 14. Specific heat capacity for a composition corresponding to AISI 1020 (0.2C-0.6Mn-0.2Si) undergoing three different continuous cooling treatments in the temperature range 298 K–1500 K (illustrated in Figures 8, 9, and 10). Model predictions based on a corrected version of the model proposed by Browne (12) (Full details are given in Appendix 3).

which is to say the function is judged to be safely twice differentiable when monotone on a compact interval about u_0 , with zero derivatives otherwise. We further choose the minimum magnitude divided difference to compute the first derivative as a way of introducing a form of slope limiter into the formulation. This has been observed to improve the robustness of the numerical implementation.

For coefficients which are derived from combinations of other intrinsic coefficient functions, in this case the volumetric heat capacity and the latent heat of transformation, it is preferred to compute the derivatives via the product rule rather than use finite differences directly, such that

$$\dot{\gamma} = \rho \dot{\kappa} + \dot{\rho} \kappa \quad (80)$$

$$\dot{\sigma} = (\dot{\rho} \bar{\mu} + \rho \dot{\bar{\mu}}) \frac{\partial \chi}{\partial \tau} + \rho \bar{\mu} \frac{\partial}{\partial \tau} \left(\frac{\partial \chi}{\partial \tau} \right). \quad (81)$$

By definition, the law of mixtures formulation of the thermophysical coefficients calculated by Equation (51) are simple linear functions of phase composition. Again, using the example of thermal conductivity, the derivative respect to phase fractions taken

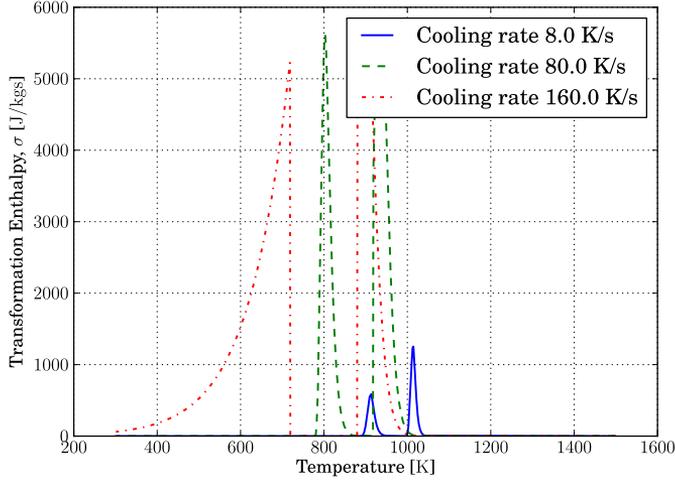


Fig 15. Transformation latent heat for composition corresponding to AISI 1020 (0.2C-0.6Mn-0.2Si) undergoing three different continuous cooling treatments in the range 298 K–1500 K.

directly as

$$\frac{\partial \lambda}{\partial} \begin{bmatrix} \chi_f \\ \chi_p \\ \chi_b \\ \chi_m \end{bmatrix} = \begin{bmatrix} \Delta \lambda_f & \Delta \lambda_p & \Delta \lambda_b & \Delta \lambda_m \end{bmatrix}. \quad (82)$$

The equivalent derivative for other thermophysical parameters is computed in the same fashion.

2.7 Coupling of heat conduction and microstructure evolution

In the diffusive transformation theory of Kirkaldy and Venugopalan, coupling of the microstructure evolution equations and the non-linear heat conduction formulation is relatively straightforward because the only interaction between the transformation rates and heat conduction rate is in the latent heat of transformation, via a source term in the heat conduction ODE. The transformation rates depend only on solution values (temperature and volume fraction transformed), therefore interaction between the transformation ODEs and heat conduction ODEs within a non-linear implicit Euler step can be resolved iteratively using a suitable Jacobian matrix.

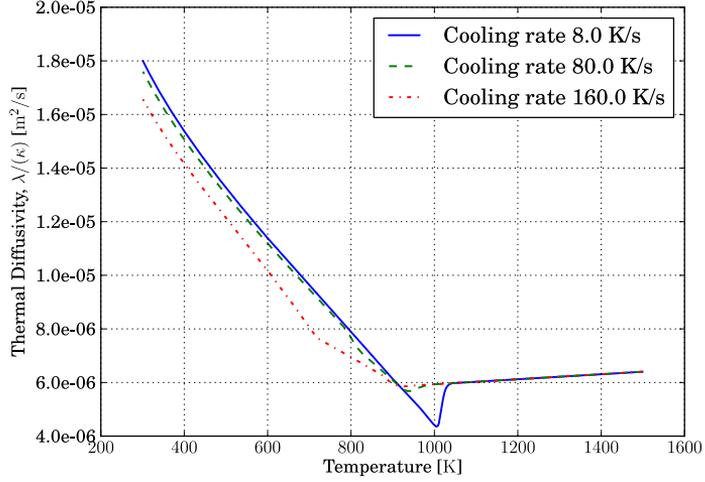


Fig 16. Thermal diffusion coefficient for a composition corresponding to AISI 1020 (0.2C-0.6Mn-0.2Si) undergoing three different continuous cooling treatments in the range 298 K–1500 K (illustrated in Figures 8, 9, and 10). Model predictions after Miettinen (27) and Browne (26, 12).

The coupling of the martensite reaction to the heat conduction calculations, on the other hand, is more complex because the reaction rate depends on both solution values (temperature), and function values (cooling rate). This is particularly problematic both because the martensite reaction rate implied by the Koistinen-Marburger relationship can be very high at temperatures just below the martensite start temperature, and because the martensite start temperature itself is usually well below the Curie temperature, implying the reaction should be accompanied by considerable magnetic latent heat of transformation. This can result in dramatic variations in both cooling rate and transformation rate values if the two function values are not correctly coupled.

To address this complexity, we revisit the discrete heat conduction equation defined by Equation (22), which can be parameterised as

$$\frac{\partial u_i}{\partial \tau} = \phi_i = f(u_i, \nabla u_i, \nabla^2 u_i) + (g_m(u, \partial u / \partial \tau, \chi_m) \rho_m \mu_m)_i \quad (83)$$

so as to separate the heat conduction terms from the latent heat associated with the martensite reaction. Substituting the martensite reaction kinetics described earlier in the chapter, Equation (62), we have a non-linear expression for the cooling rate term

$$\phi_i - f(u_i, \nabla u_i, \nabla^2 u_i) + (\rho_m \mu_m \chi_m^{\max})_i \left(\eta \exp[-\eta(\Delta u_{ms})_i] \mathcal{H}(-(\Delta u_{ms})_i \phi_i) \phi_i \right) = 0. \quad (84)$$

Equation (84) can be solved iteratively for ϕ_i using Newton-Raphson iterations with a diagonal inverse Jacobian matrix computed directly as

$$\mathbf{J}^{-1} = \text{diag} \left[\left(1 + (\rho_m \mu_m \chi_m^{\max})_i \left(\eta \exp[-\eta(\Delta u_{ms})_i] \mathcal{H}(-(\Delta u_{ms})_i \phi_i) \right) \right)^{-1} \right]. \quad (85)$$

Once the Newton-Raphson iteration has converged to a suitable tolerance, the function value can be used to compute the martensite transformation rate, in the process ensuring agreement between the martensite transformation ODE and the heat conduction ODE within a given solution step or stage.

The final stage in formulating the completed heat conduction formulation for steel thermomechanical treatment processes is to couple the microstructure evolution and heat conduction processes together into a set of coupled ODEs:

$$\frac{\partial}{\partial \tau} \begin{bmatrix} \mathbf{u} \\ \mathbf{x}_f \\ \mathbf{x}_p \\ \mathbf{x}_b \\ \mathbf{x}_m \end{bmatrix} = \begin{bmatrix} f(\mathbf{u}, \nabla \mathbf{u}, \nabla^2 \mathbf{u}, \frac{\partial \mathbf{x}_f}{\partial \tau}, \frac{\partial \mathbf{x}_p}{\partial \tau}, \frac{\partial \mathbf{x}_b}{\partial \tau}, \frac{\partial \mathbf{x}_m}{\partial \tau}) \\ g_f(\mathbf{u}, \mathbf{x}_f) \\ g_p(\mathbf{u}, \mathbf{x}_p) \\ g_b(\mathbf{u}, \mathbf{x}_b) \\ g_m(\mathbf{u}, \frac{\partial \mathbf{u}}{\partial \tau}, \mathbf{x}_m) \end{bmatrix}, \quad (86)$$

where \mathbf{x}_j holds volume fractions of each transformation product and the function $g(u, x)$ encapsulates the applicable rate equations. This system of equations can be expanded into a non-linear implicit Euler step

$$\mathbf{r}^{(i+1)} = \begin{bmatrix} \mathbf{u} \\ \mathbf{x}_f \\ \mathbf{x}_p \\ \mathbf{x}_b \\ \mathbf{x}_m \end{bmatrix}^{(i+1)} - \begin{bmatrix} \mathbf{u} \\ \mathbf{x}_f \\ \mathbf{x}_p \\ \mathbf{x}_b \\ \mathbf{x}_m \end{bmatrix}^{(i)} - \Delta \tau^{(i)} \begin{bmatrix} \mathbf{f} \\ \mathbf{g}_f \\ \mathbf{g}_p \\ \mathbf{g}_b \\ \mathbf{g}_m \end{bmatrix}^{(i+1)} \quad (87)$$

with a block jacobian matrix calculated as

$$\frac{\partial \mathbf{r}}{\partial} \begin{bmatrix} \mathbf{u} \\ \mathbf{x}_f \\ \mathbf{x}_p \\ \mathbf{x}_b \\ \mathbf{x}_m \end{bmatrix} = \mathbf{I} - \Delta\tau \begin{bmatrix} \left[\frac{\partial f}{\partial \mathbf{u}} \right] & \left[\frac{\partial f}{\partial \mathbf{x}_f} \right] & \left[\frac{\partial f}{\partial \mathbf{x}_p} \right] & \left[\frac{\partial f}{\partial \mathbf{x}_b} \right] & \left[\frac{\partial f}{\partial \mathbf{x}_m} \right] \\ \left[\frac{\partial g_f}{\partial \mathbf{u}} \right] & \left[\frac{\partial g_f}{\partial \mathbf{x}_f} \right] & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \left[\frac{\partial g_p}{\partial \mathbf{u}} \right] & \mathbf{0} & \left[\frac{\partial g_p}{\partial \mathbf{x}_p} \right] & \mathbf{0} & \mathbf{0} \\ \left[\frac{\partial g_b}{\partial \mathbf{u}} \right] & \mathbf{0} & \mathbf{0} & \left[\frac{\partial g_b}{\partial \mathbf{x}_b} \right] & \mathbf{0} \\ \left[\frac{\partial g_m}{\partial \mathbf{u}} \right] & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad (88)$$

whose component sub-matrices are computed using the Jacobian formulations described in Appendices 1 and 3.

To evaluate the ODE function within the implicit step, we apply the following procedure

1. Compute initial estimates of microstructure evolution rates using step estimates of $\mathbf{u}^{(i+1)}$ and $\mathbf{x}^{(i+1)}$, yielding the set of $\mathbf{g}_j^{(i+1)}$ function values. For the martensite reaction, which requires a cooling rate, we estimate it as

$$\frac{\partial \mathbf{u}}{\partial \tau} \approx \frac{1}{\Delta\tau} \left(\mathbf{u}^{(i+1)} - \mathbf{u}^{(i)} \right)$$

2. Compute the thermophysical coefficients and transformation enthalpies heats using the estimates of $\mathbf{u}^{(i+1)}$ and $\mathbf{x}^{(i+1)}$
3. Compute the transformation latent heat using the thermophysical coefficients and $\mathbf{g}_j^{(i+1)}$
4. Use Newton-Raphson iterations to compute a converged value of the heat conduction function $\mathbf{f}^{(i+1)}$
5. Compute final estimates of microstructure evolution rates using stage values and taking the cooling rate to be the calculated value of $\mathbf{f}^{(i+1)}$.

This procedure for a single non-linear implicit Euler step can be used directly as part of the implicit Runge-Kutta integration schemes discussed earlier in this Chapter, and described in more detail in Appendix 2.

2.8 Computational examples for thermomechanical treatment processes

In this example, cooling of a 16 mm square bar of an AISI-1045 (0.45%C-0.75%Mn-0.25%Si) structural steel is simulated in quarter plane symmetry using a 32×32 cell centred uniform grid, as illustrated in Figure 17. The bar is continuously cooled from

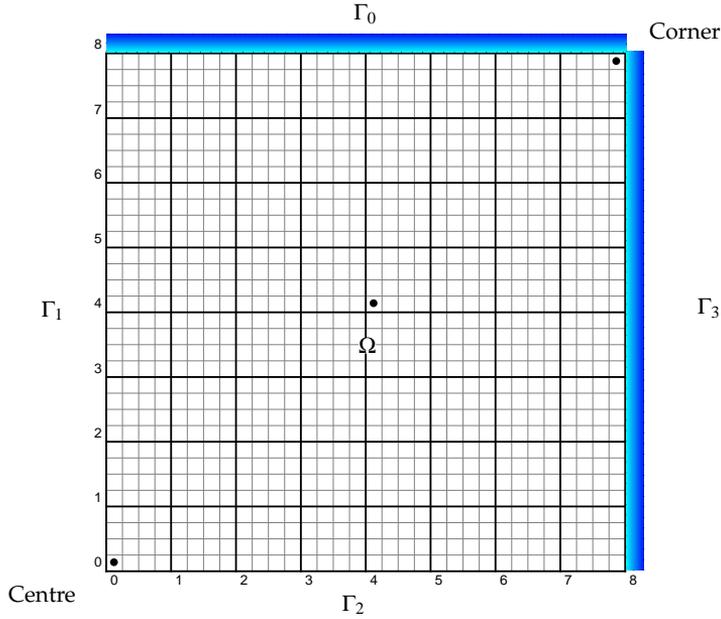


Fig 17. Cross-section of a 16x16 mm square bar modelled in quarter plane symmetry using a 32x32 cell uniform grid. Outer surfaces denoted in blue, centre, mid-thickness and corner cells marked with dots.

an initial austenitic state. Heat flow within the bar is presumed to be governed by the two-dimensional conduction equation

$$\frac{\partial u}{\partial \tau} = \frac{1}{\gamma} \left[\lambda \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + \dot{\lambda} \left(\frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} \right)^2 + \sigma \right] \{x, y\} \in \Omega, \tau > 0 \quad (89)$$

with boundary conditions

$$\frac{\partial u}{\partial n} = 0 \quad \{x, y\} \in \Gamma_1 \cup \Gamma_2, \tau > 0 \quad (90)$$

$$\alpha u + (1 - \alpha) \frac{\partial u}{\partial n} = \beta \quad \{x, y\} \in \Gamma_0 \cup \Gamma_3, \tau > 0 \quad (91)$$

and initial conditions

$$u = 1250 - \frac{50}{8\sqrt{2}}\sqrt{x^2 + y^2} \text{ K} \quad \{x, y\} \in \Omega, \tau = 0 \quad (92)$$

$$\chi_f = 0 \quad \{x, y\} \in \Omega, \tau = 0 \quad (93)$$

$$\chi_p = 0 \quad \{x, y\} \in \Omega, \tau = 0 \quad (94)$$

$$\chi_b = 0 \quad \{x, y\} \in \Omega, \tau = 0 \quad (95)$$

$$\chi_m = 0 \quad \{x, y\} \in \Omega, \tau = 0, \quad (96)$$

which is, in turn, coupled to the system microstructure evolution equations discussed in Section 2.7, resulting in a total of five degrees of freedom per cell. For the 32×32 grid used in the analysis, this implies a total of 5120 coupled equations per implicit Euler stage.

We consider three different heat transfer regimes

1. Cooling in still air by Stefan-Boltzmann radiation and convection
2. Cooling by combined Stefan-Boltzmann radiation and convection from a low pressure hydraulic spray
3. Cooling by combined Stefan-Boltzmann radiation and immersion quenching in still water

using a different Robin condition for each case, combining a heat transfer coefficient function with a surface thermal resistance from the presence of a uniform wustite scale layer of $10 \mu\text{m}$ thickness with thermal conductivity given by the model proposed by (34), and described in Appendix 3.3. These components of the net surface thermal conductance are combined to form a pair of Robin coefficient functions using Equation (17a). Details of each boundary condition function is discussed in the following sections.

For each heat transfer regime, solutions were made using four combinations of finite difference spatial discretising schemes and integrators:

- The second order spatial discretisation with the DIRK(3, 2) integrator, hereafter referred to as the FD2-DIRK(3, 2) scheme.
- The second order spatial discretisation with the DIRK(4, 3) integrator, hereafter referred to as the FD2-DIRK(4, 3) scheme.
- The fourth order spatial discretisation with the DIRK(3, 2) integrator, hereafter referred to as the FD4-DIRK(3, 2) scheme.
- The fourth order spatial discretisation with the DIRK(4, 3) integrator, hereafter referred to as the FD4-DIRK(4, 3) scheme.

In each case, the computations have been made in double precision floating point arithmetic using code written in the Python programming language making extensive use of the scipy library (35). The various order spatial discretising operators, boundary condition formulations and time integration schemes were coded as a hierarchy of native Python modules and classes from which a complete simulation can be assembled using only a few tens of lines of code. An example of one of the simulation cases described in this section is included in Appendix 4.

After a solution for a given case was obtained, average cooling rate values were calculated by central differences. Solution values were also used to compute the Jacobian matrix of the combined system of ODEs at selected intervals through the problem integration time, and the 1024 largest eigenvalues of each Jacobian matrix computed using the ARPACK package (36).

2.8.1 Cooling in still air

To simulate air cooling, convection and Stefan-Boltzmann radiation are presumed to operate over the surface of the bar corresponding to $\{x,y\} \in \Gamma_0 \cup \Gamma_3$. The combined surface heat transfer coefficient is calculated as

$$\omega_c = \omega_0 + \varepsilon \zeta_b (u + u_\infty) (u^2 + u_\infty^2) \quad (97a)$$

with

$$\omega_0 = 10 \text{W/m}^2 \text{K} \quad (97b)$$

$$u_\infty = 299 \text{K} \quad (97c)$$

$$\varepsilon = 0.75 \quad (97d)$$

$$\zeta_b = 5.6704 \times 10^{-8} \text{W/m}^2 \text{K}^4. \quad (97e)$$

The problem was integrated for 600 s using each of the four spatial discretization/integrator combinations, which was sufficient for all phase transformation reactions to reach completion and for the temperature to reach 600 K from the initial temperature of 1250 – 1200 K. In each case, 400 output solution values were requested, implying an integrator output frequency of 0.667 Hz. As will be discussed in more detail below, this meant that some solution values were produced by interpolation rather than integration.

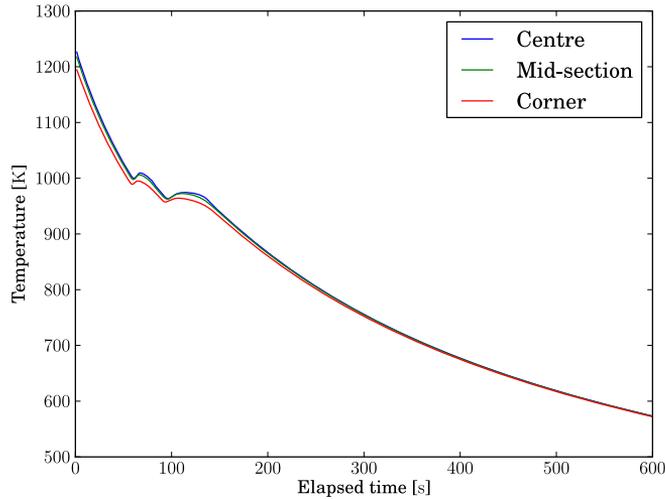


Fig 18. Predicted temperature distribution through the cross section of a 16 mm by 16 mm AISI 1045 bar during air cooling from an initially austenitic state. Computed using the FD4-Dirk(4,3) scheme.

Relative and absolute error tolerances for the DIRK integrator were taken as 10^{-4} in each case.

Figure 18 shows the predicted temperatures during air cooling made using the FD4-Dirk(4,3) scheme at three spatial positions through the cross-section of the bar – the bar centre cell, the outer corner cell, and the cell mid way between the corner and centre cell. These positions are annotated in Figure 17. From the initial difference of 30 K, temperatures at these three positions remain within 15 K during transformation, and to less than 2 K after 600 s. The maximum observed cooling rate was slightly greater than 6 K/s, with the minimum slightly more than -2 K/s during the peak rate of the proeutectoid ferrite reaction. This is shown in Figure 19. Even for this slow cooling case, the effects of latent heat of transformation during both the pro-eutectoid ferrite and pearlite transformations are obvious in the temperature and cooling rate predictions, with typical recalescence behaviour producing temperature rises of up to 15 K from the onset of transformation to the peak temperature reached during transformation. In this example, the effect of latent heat of transformation due to the pro-eutectoid ferrite reaction is actually slightly larger than that for the pearlite reaction. This may seem slightly counterintuitive at first inspection, as the thermophysical models predict that the combined enthalpy associated with pearlite should be larger due to the effects of cementite formation and the lower transformation temperature. In this case, however,

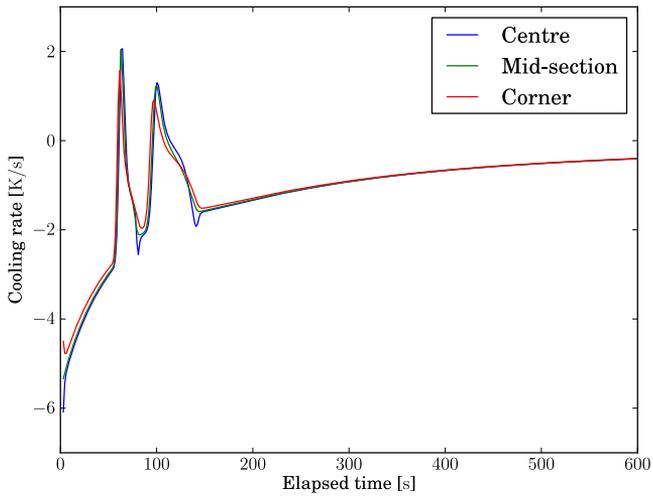


Fig 19. Predicted cooling rate at different positions through the cross section of a 16 mm by 16 mm AISI 1045 bar during air cooling from an initially austenitic state. Computed using the FD4-DIRK(4,3) scheme.

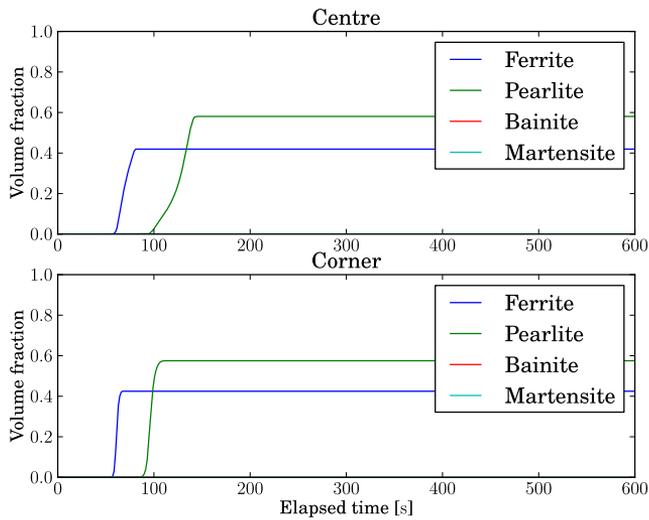


Fig 20. Predicted austenite decomposition reactions through the cross section of a 16 mm by 16 mm AISI 1045 bar during air cooling from an initial austenitic state at 1200 – 1250 K. Computed using the FD4-DIRK(4,3) scheme.

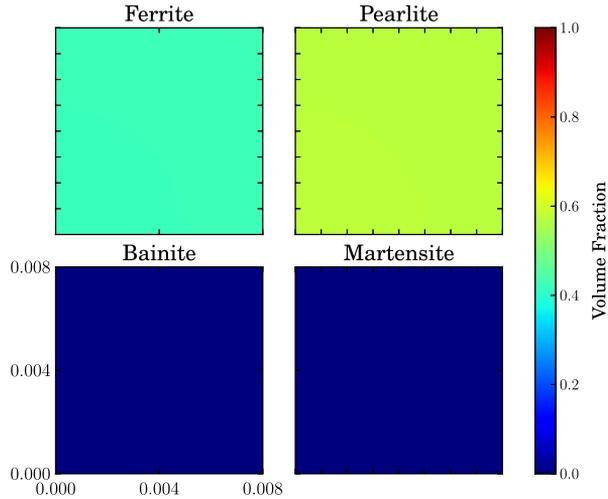


Fig 21. Predicted final distribution of phases through the cross section of a 16 mm by 16 mm AISI 1045 bar during air cooling from an initial austenitic state at 1200 – 1250 K. Computed using the FD4-DIRK(4,3) scheme.

Table 1. Solution scheme performance for the AISI 1045 air cooling case.

Scheme	Steps	Failures	Iterations	Relative time
FD2-DIRK(3,2)	1411	3594	7970	1
FD4-DIRK(3,2)	908	1972	5207	0.615
FD2-DIRK(4,3)	417	495	4957	0.508
FD4-DIRK(4,3)	406	476	4552	0.544

the transformation rate of the pearlite reaction is considerably slower than the preceding proeutectoid ferrite reaction. As a result, the net latent heat evolution rate is somewhat slower for pearlite, even though the enthalpy is higher.

The predicted transformation-time histories at the centre and corner positions of the bar and the distribution of phases across the cross-section are shown in Figures 20 and 21, respectively. As might be expected from a slow cooled plain Carbon steel, the predicted microstructure at the end of cooling is a proeutectoid ferrite-pearlite mixture with approximately 42% volume fraction proeutectoid ferrite and 58% volume fraction pearlite. The relatively uniform temperature distribution during transformation meant variation in predicted phase fractions across the cross-section are minimal.

Turning from the features of the solution itself to the performance of the four solution

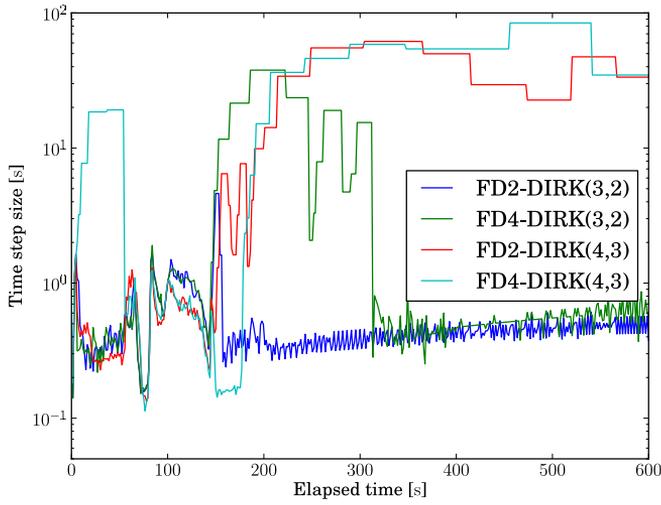


Fig 22. Comparison of average integrator step sizes for the air cooling case solved with the fourth-order spatial discretisation and the DIRK(3,2) and DIRK(4,3) integrators.

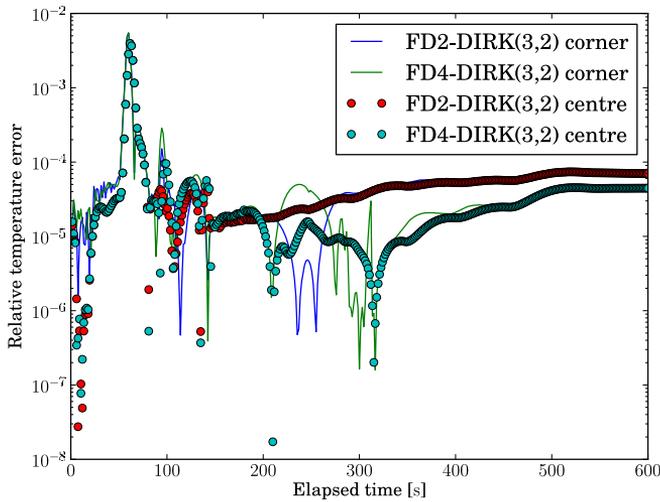


Fig 23. Comparison of relative temperature error for the air cooling case solved with the fourth order FD4-DIRK(4,3) scheme and the FD2-DIRK(3,2) and FD4-DIRK(3,2) schemes.

schemes tested, we see some interesting differences between the behaviour of each scheme. Table 1 summarises the performance of the four schemes for the air-cooling case. What is immediately clear is that the DIRK(4,3) scheme is considerably more efficient overall than the DIRK(3,2) scheme, despite requiring an additional stage evaluation per step, which increases the total number of unknowns in the iterative DIRK scheme solver from 15360 to 20480, with accompanying superlinear increases in the complexity of the $\mathcal{O}(N^2)$ and $\mathcal{O}(N^3)$ matrix operations associated with the solver. In particular, we observe that, at least at the tolerances used in these experiments, the DIRK(3,2) scheme not only requires many more steps to integrate the 600 s of the problem, but also yields a considerably inferior ratio of successful to failed steps – which is approximately 1:3 for the DIRK(3,2) scheme as opposed to approximately 4:5 for the DIRK(4,3).

Figure 22 illustrates the average step size used by each scheme as a function of the total integrated time. This largely reflects the results in Table 1, emphasising the much larger step sizes admitted by the cases using the DIRK(4,3) integrator over much of the duration of the air cooling problem. We note that during the period of the problem when the proeutectoid ferrite and pearlite transformation reactions are active (between 75 and 150 seconds), all four schemes require almost identical step sizes, which is a result consistent with the results discussed in Appendix 3.4 for implicit solutions of the Kirkaldy-Venugopalan transformation model. The fourth order spatial discretisation also appears to be more efficient than the second order version, with the FD4-DIRK(3,2) requiring less time to solve, admitting larger timesteps outside of the transformation region of the solution and incurring fewer failed trial steps and iterations per step to converge than its FD2-DIRK(3,2) counterpart. This efficiency advantage is despite the penalty of approximately twice the computational expense of computing both the heat conduction function and the Jacobian matrix compared to the second order scheme.

Finally, Figure 23 shows the relative temperature error between the FD4-DIRK(4,3) scheme and the other three schemes at the centre and corner cells at functions of time. Overall there is little difference in solution values between any of the four schemes, with the only major differences lying in the region corresponding to the onset of transformation, which, as will be discussed in greater detail below, coincides with the region of peak ODE instability, and is where the FD4-DIRK(4,3) scheme appears somewhat more robust than the lower order schemes.

Figure 24 shows the computed maximum and dominant eigenvalues of the Jacobian matrix of the fourth order version of the air-cooling problem. The most obvious feature of the eigenvalue analysis is the presence of positive maximum eigenvalues at the onset

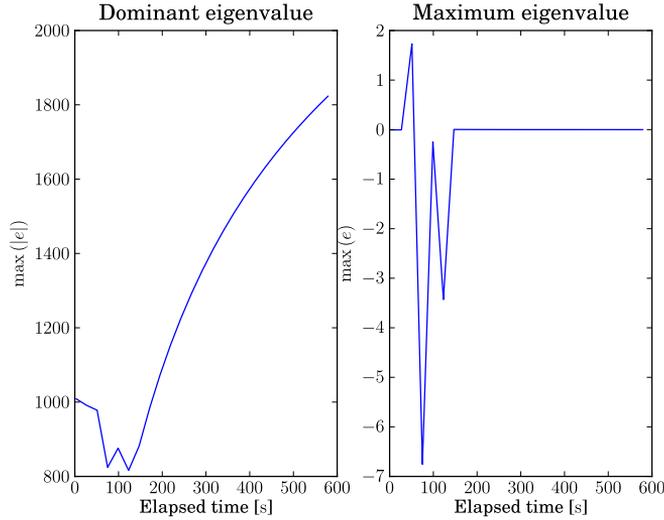


Fig 24. Dominant (absolute maximum) and maximum eigenvalues of the coupled FD4 scheme Jacobian matrix for the air cooling case. Calculated from the FD4-DIRK(4, 3) solution shown in Figure 18.

of phase transformation. This is usually indicative of ODE instability, and we observe dramatic reductions in the step sizes required by all of the schemes to converge at around 70 s into the solution (refer to Figure 22), accompanied by occasional singular or near singular Jacobian matrices and numerical blow-up of trial DIRK steps. This was something of a surprise given that the air cooling case has neither large temperature gradients, highly non-linear boundary conditions or rapid transformation processes, and should be relatively benign compared to the more complex cases considered in the current work. We opine that it is most likely caused by a rapid increase in the magnitude of convective component of the heat conduction equation as phase transformation begins, almost coincident with the Curie temperature, where non-linearity of the thermo-physical coefficients is at a maximum. The dominant eigenvalue is also observed to increase in magnitude after the completion of the phase transformation processes. This is most likely due to the increase in thermal conductivity in both ferrite and pearlite as the temperature decreases, which should imply faster heat conduction rates and potentially reduced maximum admissible step sizes for integrator stability.

With the DIRK integrators used in the current work, the only remedy is to use the UMFPACK LU factorization routine diagnostics to check for near singular conditions, rejecting the step and reducing the step size until a step size is found where convergence

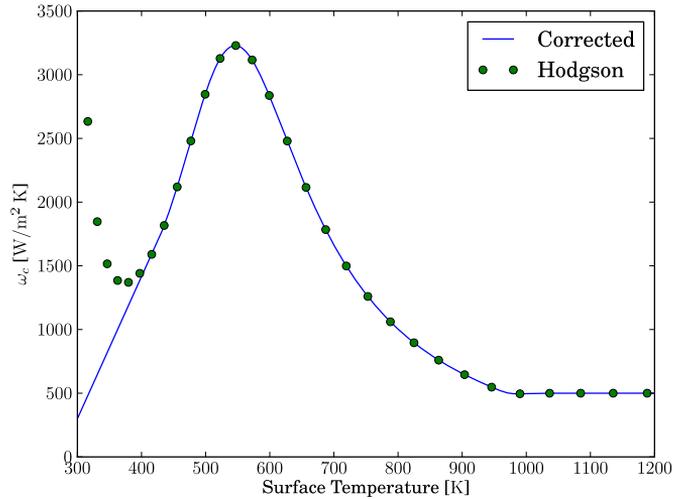


Fig 25. Predicted spray cooling transfer coefficient for 300 K water using a modification of the model described by Hodgson *et al* (37).

can be achieved. This approach was found to work satisfactorily in all cases, but it does impose a considerable computational overhead which might be avoided if the integrator were to perform some sort of approximate eigenvalue analysis to detect stiffness, such as that discussed by Hairer *et al* (22, §IV.2). Certainly an eigenvector analysis of the type used to produce Figure 24 is impractical, as the computational cost of even this reduced analysis was approaching an order of magnitude larger than time required to generate the solution itself.

2.8.2 Spray cooling

For simulating heat transfer due to spray cooling, a modification of the empirical relationship proposed by Hodgson *et al* (37) is used. A spray or sprays with water flux $\dot{\theta}$ l/m² s is presumed to operate over the surface of the bar corresponding to $x, y \in \Gamma_0 \cup \Gamma_3$, such that the effective heat transfer coefficient in the direct impingement zone of the spray is calculated as

$$\omega_c = 3.16 \times 10^9 \zeta_0 \zeta_1 ((u - u_0) - \zeta_2 (u - u_3))^{-2.455} \dot{\theta}^{0.616} \quad (98)$$

$$\zeta_0 = \begin{cases} 0 & u < u_0 \\ \frac{u_1 - u}{u_1 - u_0} & u \in [u_0, u_1] \\ 1 & u > u_1 \end{cases}, \quad (99)$$

where

$$\zeta_1 = \left(1 - \frac{1}{1 + \exp\left(\frac{u - u_2}{40}\right)} \right) \quad (100)$$

$$\zeta_2 = \left(1 - \frac{1}{1 + \exp\left(\frac{u - u_3}{10}\right)} \right) \quad (101)$$

with $u_0 = 273\text{K}$, $u_1 = 425\text{K}$, $u_2 = 573\text{K}$, and $u_3 = 973\text{K}$. The model is intended to encapsulate the three major heat transfer modes associated with spray cooling, so that at high surface temperatures, a stable vapour layer forms and the effective heat transfer coefficient is relatively constant. Below the Leidenfrost temperature (taken as 973 K in this model), nucleation boiling commences, which produces a large increase in the effective heat transfer coefficient, reaching a pronounced peak at approximately 550 K. Below this temperature, the rate of nucleation boiling rapidly decreases and convection becomes the dominant heat transfer mechanism.

The form of Equation (98) differs from the original published model in one important respect. In the original version, ζ_0 was taken as unity at all surface temperatures. This leads to non-physical behaviour at temperatures below 400 K, where insufficient superheat for nucleation boiling should cause convection to be the dominant heat transfer mechanism. Under these conditions, the overall heat transfer coefficient should be low and decreasing with reductions in surface temperature and the degree of water superheat (38, §8). To correct this, ζ_0 is added to scale the heat transfer coefficient at 425 K linearly down to zero at 273 K. These modifications and their effects are illustrated in Figure 25.

In the current calculations, the effective spray water flux is taken as $2.01/\text{m}^2 \text{ s}$, with the same surface radiative properties that were used in the air cooling case. The problem was integrated for a total of 75 s using each of the four solution schemes, with an output frequency of 5.333 Hz. Relative and absolute tolerances for the DIRK integrators were taken as 5×10^{-5} and 10^{-4} , respectively.

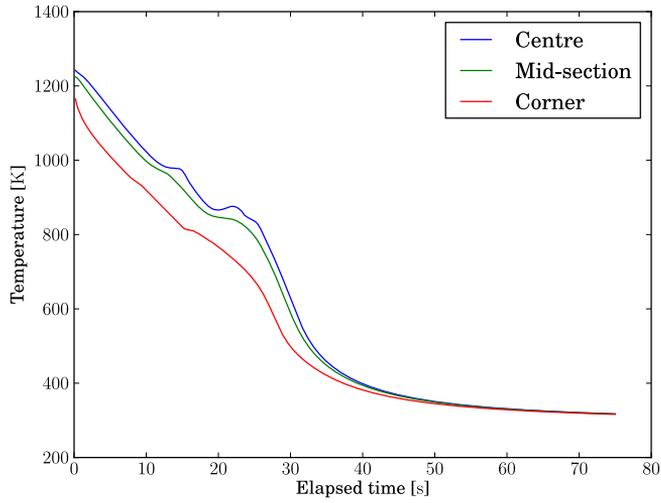


Fig 26. Predicted temperature distribution through the cross section of a 16 mm by 16 mm AISI 1045 bar during spray cooling from an initially austenitic state. Computed using the FD4- $\text{DIRK}(4, 3)$ scheme.

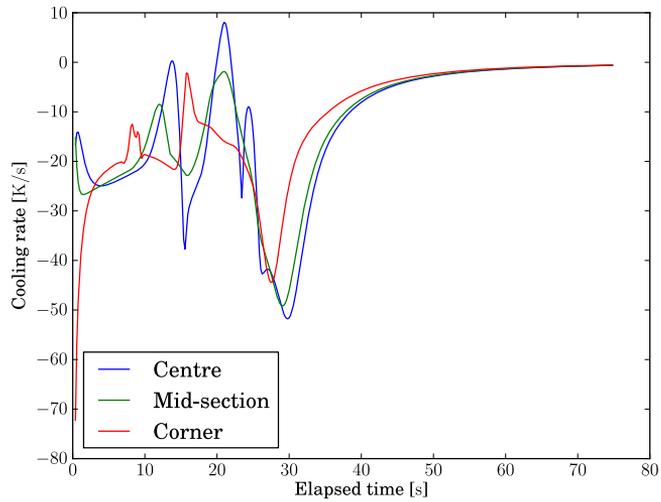


Fig 27. Predicted cooling rate at different positions through the cross section of a 16 mm by 16 mm AISI 1045 bar during spray cooling from an initially austenitic state. Computed using the FD4- $\text{DIRK}(4, 3)$ scheme.

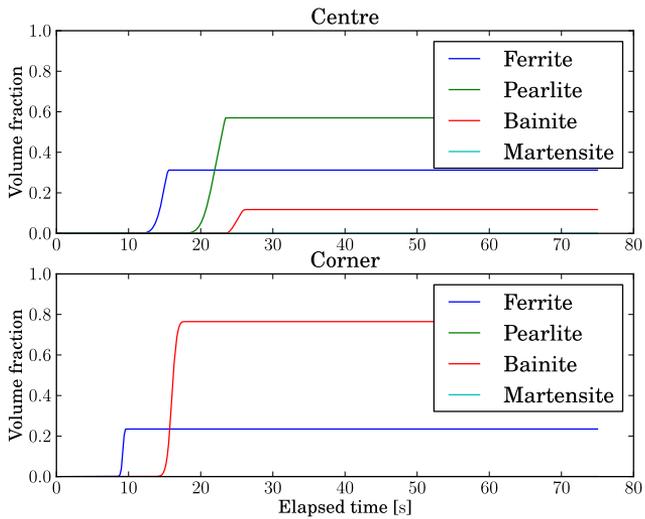


Fig 28. Predicted austenite decomposition reactions through the cross section of a 16 mm by 16 mm AISI 1045 bar during spray cooling from an initial austenitic state at 1200 – 1250 K. Computed using the FD4- $DIRK(4,3)$ scheme.

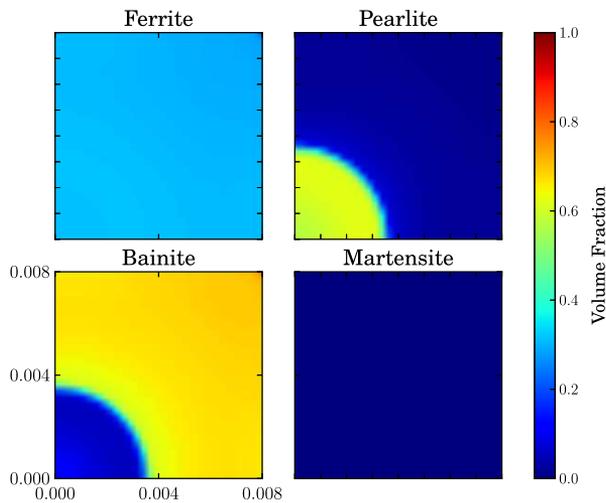


Fig 29. Predicted final distribution of phases through the cross section of a 16 mm by 16 mm AISI 1045 bar during spray cooling from an initial austenitic state at 1200 – 1250 K. Computed using the FD4- $DIRK(4,3)$ scheme.

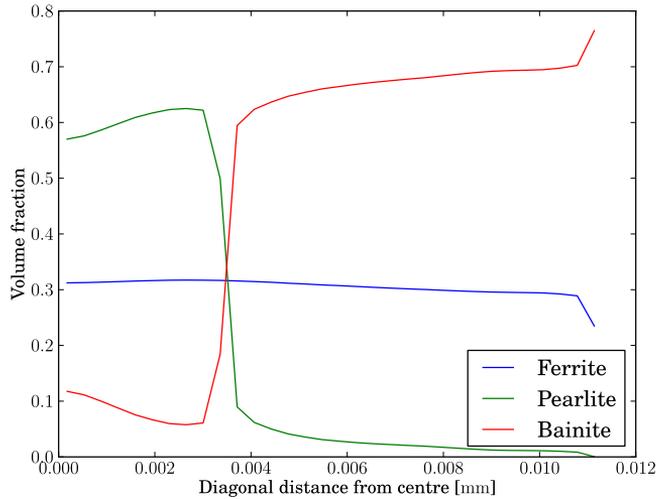


Fig 30. Predicted final distribution of phases along the centre-corner diagonal of a 16 mm by 16 mm AISI 1045 bar during spray cooling from an initial austenitic state at 1200 – 1250 K. Computed using the FD4-DIRK(4,3) scheme.

Figure 26 shows the predicted temperatures at the three reference positions within the cross section made using the FD4-DIRK(4,3) scheme, while Figure 27 shows the corresponding cooling rates. During the first 10 seconds of cooling, a centre to corner temperature difference of approximately 150 K quickly develops, with a relatively uniform cooling rate away from the edge as the sample cools to the onset of phase transformation. After approximately 10 seconds we see considerable differences between the three reference positions, with the effects of transformation heat far more obvious in the centre of the bar than at the surface. At a corner temperature of slightly more than 725 K we see the effects of nucleation boiling begin to become apparent, and the cooling rate rises rapidly to a peak of 45 – 50 K/s through the cross section between 26 and 28 seconds elapsed time. As nucleation boiling gives way to convection at lower surface temperatures, and phase transformation rates approach zero, the bar temperature distribution becomes considerably more uniform.

The kinetics of austenite decomposition in the centre and corner reference positions, and the predicted distribution of phases after 75 seconds of spray cooling are shown in Figures 28 and 29, respectively. Compared to the earlier aircooling case, we see marked differences in the predicted microstructure through the cross section. The edge of the bar is predicted to exhibit a primarily bainitic microstructure, with 78% volume fraction bainite and the balance pro-eutectoid ferrite, while at the centre, the bainite has been

Table 2. Solution scheme performance for the AISI 1045 spray cooling case.

Scheme	Steps	Failures	Iterations	Relative time
FD2- $\text{DIRK}(3, 2)$	1074	695	3194	1
FD4- $\text{DIRK}(3, 2)$	579	526	2520	0.728
FD2- $\text{DIRK}(4, 3)$	248	321	1817	0.623
FD4- $\text{DIRK}(4, 3)$	204	303	2019	0.644

mostly replaced by pearlite, with the volume fraction of pro-euctectoid ferrite remaining relatively constant over the whole cross-section.

Closer inspection of the microstructure predictions reveal some interesting features in the predicted final phase fractions, which are more easily observed in Figure 30 than Figure 29. This shows the volume fraction bainite progressively decreases moving from corner to centre, reaching 62% volume fraction bainite at the beginning of a transition zone approximately 4 mm from the centre along the corner-centre diagonal, with the volume fraction of pearlite progressively increasing. In this approximately 1 mm wide transition zone, the volume fraction bainite falls rapidly to approximately 6%, while the volume fraction of pearlite rises correspondingly from approximately 6% to 62%. We also observe that the volume fraction of bainite begins to increase moving away from the transition zone and towards the centre of the bar. We attribute this to the higher residual austenite content at the bainite start temperature due to the higher average temperatures and resulting slower reaction rates during the pearlite transformation close to the centre of the bar.

The overall performance of the four schemes is summarized in Table 2, with average step sizes shown in Figure 31, relative errors between the FD4- $\text{DIRK}(4, 3)$ scheme and the other three schemes shown in Figure 32, and the results of the FD4- $\text{DIRK}(4, 3)$ eigenvalue analysis shown in Figure 33. Most of the remarks made about the relative differences in performance of the different solver schemes for the aircooling case equally apply to the spray cooling case and shall not be repeated. Overall, the $\text{DIRK}(4, 3)$ integrator is considerably more efficient than the $\text{DIRK}(3, 2)$ integrator at the tolerances used in the spray cooling case, without major differences in the quality of the solution between any of the schemes tested. The eigenvalue analysis also follows the pattern observed in the aircooling case, with positive maximum eigenvalues observed, and the same sort of near singular or near blow-up conditions detected at the onset of transformation.

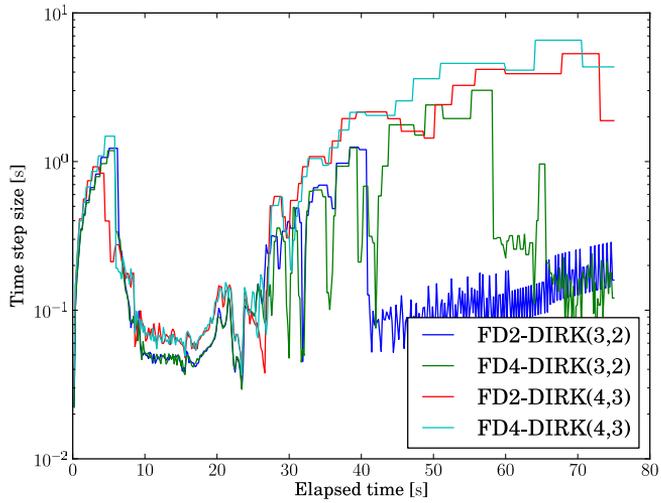


Fig 31. Comparison of average integrator step sizes for the spray cooling case solved with the fourth-order spatial discretisation and the DIRK(3,2) and DIRK(4,3) integrators.

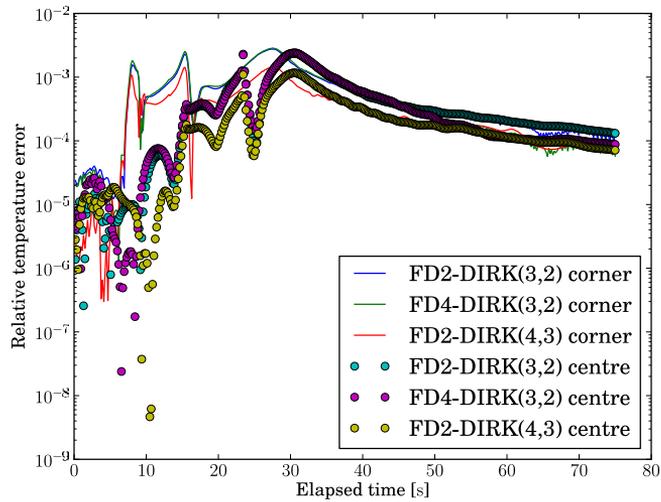


Fig 32. Comparison of relative temperature error for the spray cooling case solved with the FD4-DIRK(4,33) scheme and the FD2-DIRK(3,2), FD4-DIRK(3,2), and FD2-DIRK(4,3) schemes.

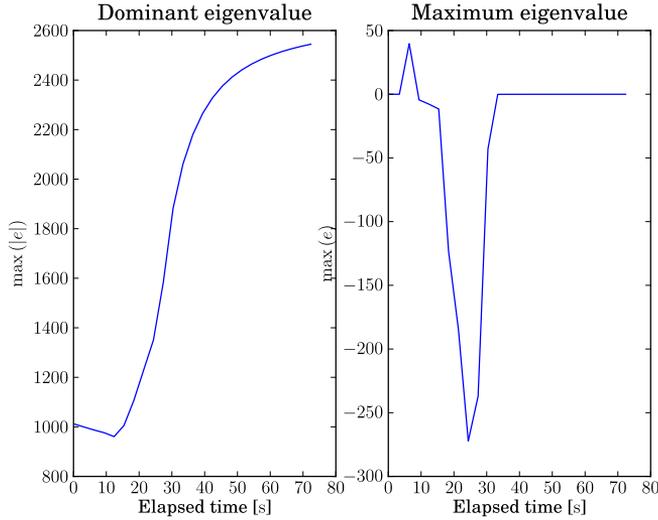


Fig 33. Dominant (absolute maximum) and maximum eigenvalues of the coupled FD4 scheme Jacobian matrix for the spray cooling case. Calculated from the FD4-DIRK(4, 3) solution shown in Figure 26.

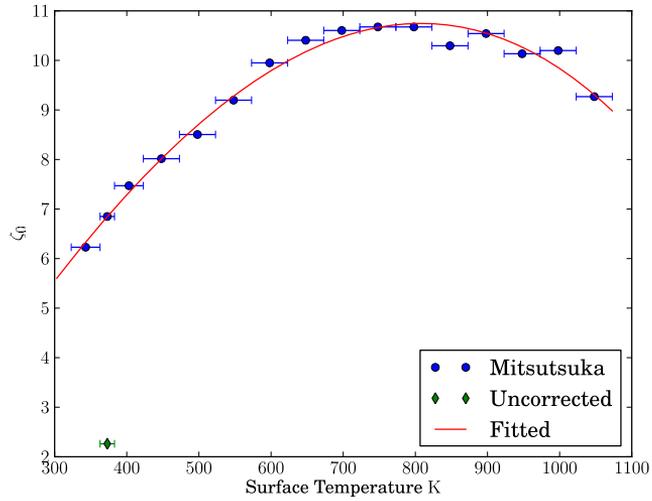
2.8.3 Immersion quenching in water

For the immersion quenching case, we apply an adaption of the correlations proposed by Mitsutsuka and Fukuda (39) for still water immersion quenching. In this approach, the heat transfer coefficient is calculated using an equation of the form:

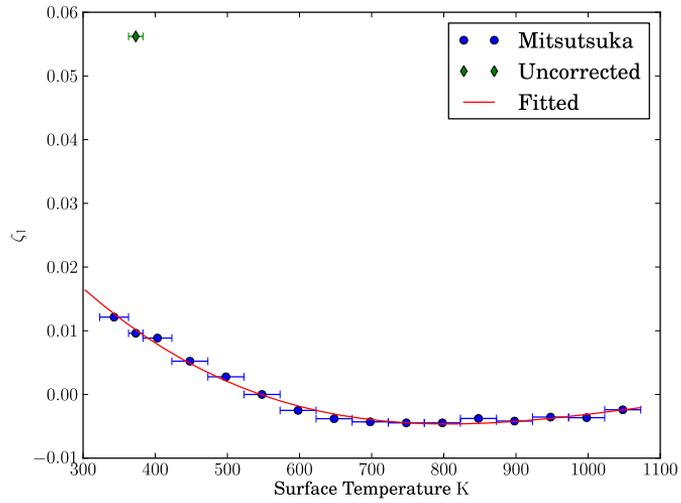
$$\omega_c = \exp(\zeta_0 + \zeta_1 u) (1 - \zeta_2 (u_\infty - u_{\text{ref}})), \quad (102)$$

where ζ_0 , ζ_1 , ζ_2 are surface temperature dependent parameters, u_∞ is the temperature of the quench water, and $u_{\text{ref}} = 299 \text{ K}$ is a reference temperature. In original work, these parameters are presented as a tabulated set of fitted values valid over narrow surface temperature ranges between $50 \text{ }^\circ\text{C}$ and $800 \text{ }^\circ\text{C}$. To make the calculations more amenable to implementation in a numerical model, these tabulated results have been used to fit smoothed interpolating cubic spline curves for each parameter, and at the same time the model has been converted into SI units and reformulated to use base e rather than base 10. The recomputed parameters are given in Table 3, and Figures show the fitted spline curves and One anomalous data point adjusted to improve the smoothness of the ζ_0 and ζ_1 curves (refer to the note in Table 3).

The spline curve fitting was performed using the automatic procedure from the



(a) Spline fit for ζ_0



(b) Spline fit for ζ_1

Fig 34. Interpolating spline functions for the discrete parameter sets presented by Mitsutsuka and Fukuda (39) and summarised in Table 3.

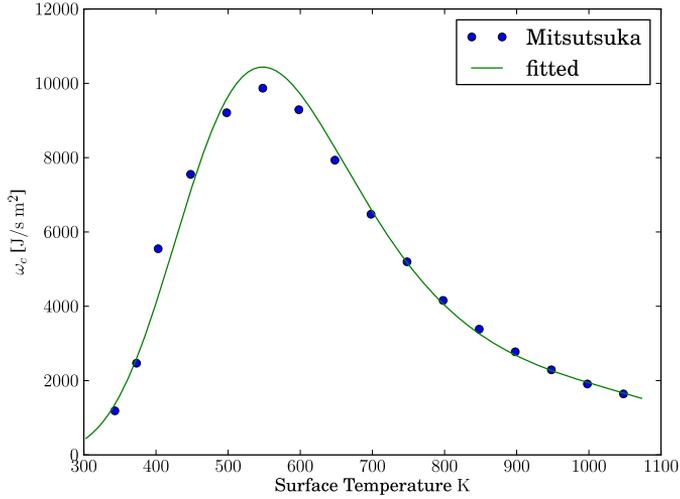


Fig 35. Predicted quenching heat transfer coefficient for 298 K water using the modified model based on the work of Mitsutsuka and Fukuda (39).

fitpack package (40, §13). The resulting knot vectors \mathbf{z}_j and coefficient vectors \mathbf{c}_j for ζ_0 and ζ_1 are given by Equations (103) and (104), respectively. As a simplifying assumption, we assume a quench media temperature of 299 K, allowing ζ_2 to be neglected from the calculation. The final heat transfer coefficient function is plotted as a function of surface temperature in Figure 35.

$$\mathbf{z}_0 = \{343, 343, 343, 343, 1048, 1048, 1048, 1048\} \quad (103a)$$

$$\mathbf{c}_0 = \{2.87879, 12.63682, 13.82092, 9.97269, 0, 0, 0, 0\} \quad (103b)$$

$$\mathbf{z}_1 = \{343, 343, 343, 343, 1048, 1048, 1048, 1048\} \quad (104a)$$

$$\mathbf{c}_1 = \{0.01265, -0.00793, -0.00590, -0.00244, 0, 0, 0, 0\} \quad (104b)$$

The quenching problem was integrated using the same procedure as for the air-cooling and spray-cooling cases, with a total integration time of 15 seconds. The integrator was requested to output 400 solution values uniformly spaced in time, which corresponds to an output frequency of 26.67 Hz. Relative and absolute tolerances for the DIRK integrators were taken as 5×10^{-5} and 10^{-4} , respectively.

The FD4-Dirk(4, 3) predicted temperatures and cooling rates for the water quenching case are shown in Figures 36 and 37.

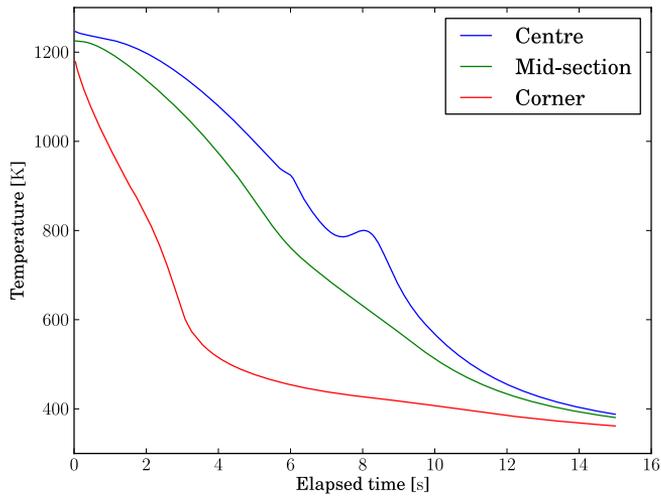


Fig 36. Predicted temperature distribution through the cross section of a 16 mm by 16 mm AISI 1045 bar during water quenching from an initially austenitic state. Computed using the FD4-DIRK(4, 3) scheme.

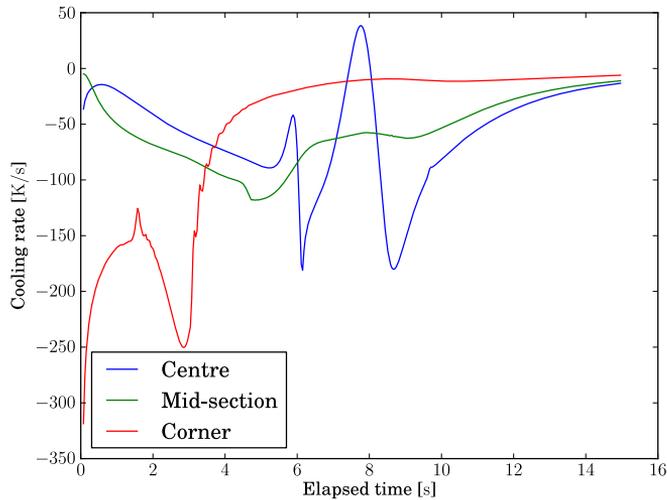


Fig 37. Predicted cooling rate at different positions through the cross section of a 16 mm by 16 mm AISI 1045 bar during immersion quenching in water from an initial austenitic state. Computed using the FD4-DIRK(4, 3) scheme.

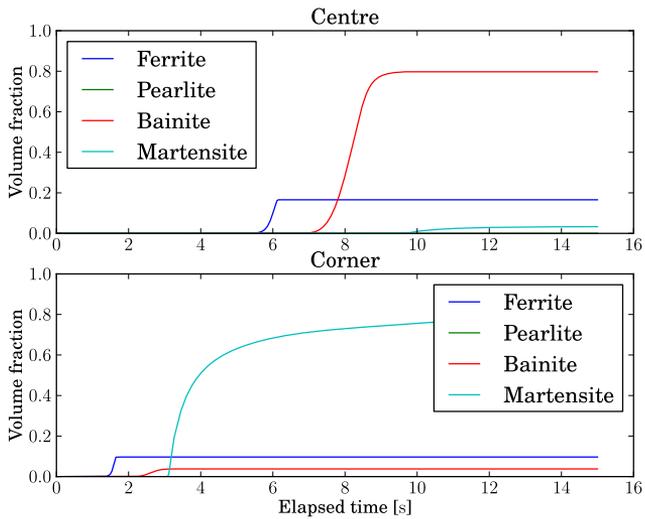


Fig 38. Predicted austenite decomposition reactions through the cross section of a 16 mm by 16 mm AISI 1045 bar during water quenching from an initial austenitic state at 1200 – 1250 K. Computed using the FD4-DIRK(4,3) scheme.

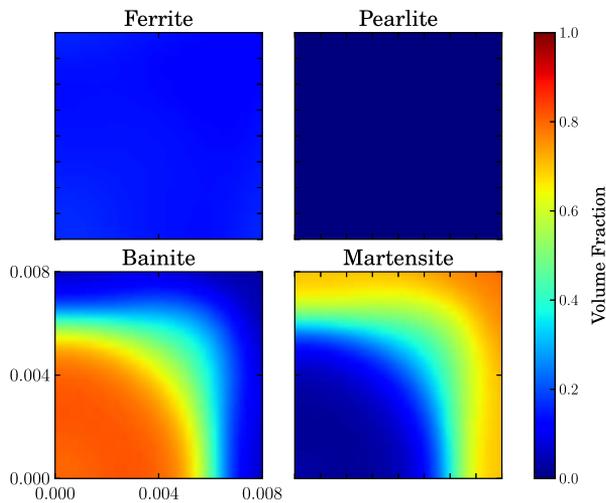


Fig 39. Predicted final distribution of phases through the cross section of a 16 mm by 16 mm AISI 1045 bar during water quenching from an initial austenitic state at 1200 – 1250 K. Computed using the FD4-DIRK(4,3) scheme.

Table 3. Recomputed coefficients for the quenching heat transfer coefficient model proposed by Mitsutsuka and Fukuda (39)

Temp K	ζ_0	ζ_1	ζ_2
343	6.22916	0.01214	0.01000
373	6.84970 ¹	0.00961 ¹	0.01000
403	7.47025	0.00885	0.01000
448	8.01596	0.00522	0.00920
498	8.50411	0.00277	0.00900
548	9.19719	0.00000	0.00900
598	9.95013	-0.00250	0.00920
648	10.40605	-0.00381	0.01000
698	10.60407	-0.00430	0.01090
748	10.67545	-0.00446	0.01170
798	10.67545	-0.00446	0.01210
848	10.29552	-0.00377	0.01230
898	10.54190	-0.00418	0.01240
948	10.13434	-0.00355	0.01250
998	10.19881	-0.00365	0.01250
1048	9.26857	-0.00241	0.01250

¹ This coefficient set was refitted from the original base 10 values of $\eta_0 = 0.917$ and $\eta_1 = 24.421 \times 10^{-3}$ to improve the smoothness of the interpolating splines

The heat transfer to the quenching medium is sufficiently high that the corner cell is predicted to cool at over 200 K/s over the first 500 ms of contact, reaching the Ae3 temperature (1038 K) of the steel in 0.75 seconds after the start of cooling, the bainite start temperature (858 K) in less than 2 seconds, and martensite start temperature (595 K) in slightly more than 3 seconds. After 4 seconds the cooling rate of the corner cells begins to rapidly decrease and for the balance of the cooling time, is only 5 – 15 K/s, which is somewhat lower than the corner cooling rates predicted in the spray cooling rate example at comparable surface temperatures.

In the interior of the bar, the temperature-time profile is greatly dissimilar to that of corner. Both the mid section and centre cells cool with an almost linearly increasing cooling rate for the first 5 seconds of quenching before signs of the onset of phase transformation appear. During this period, the mid-section cooling rate ranges from

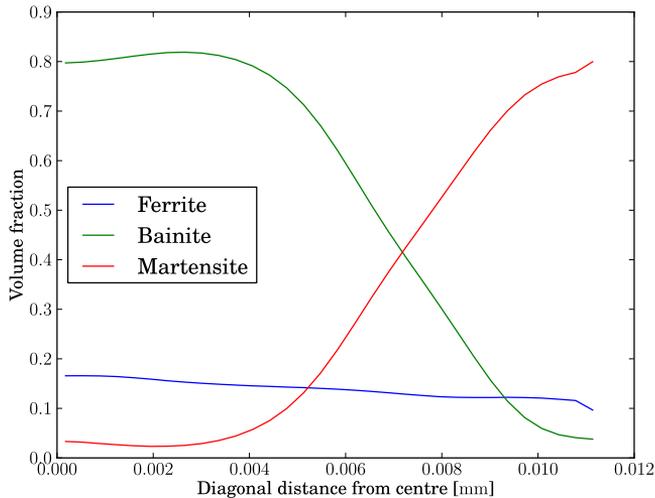


Fig 40. Predicted final distribution of phases along the centre-corner diagonal of a 16 mm by 16 mm AISI 1045 bar during water quenching from an initial austenitic state at 1200 – 1250 K. Computed using the FD4-DIRK(4, 3) scheme.

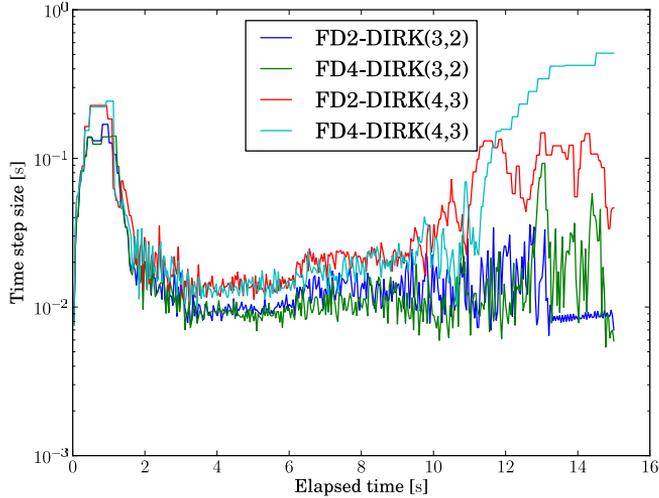
approximately 50 – 100 K/s, the centre is somewhat lower, ranging from 25 – 75 K/s. Coinciding with this, an extremely large edge to corner temperature gradient develops, peaking at 565 K after 3.6 seconds of cooling. This leads to the interesting situation where the bar surface has been supercooled to such a degree that the cooling rate of the interior of the bar quickly becomes governed almost entirely by the thermal gradients within the bar, rather than any direct effect of contemporaneous heat transfer processes operating at the bar surface.

The effects of phase transformation process on the cooling behaviour of the bar are best illustrated in the cooling rates, as shown in Figure 37. Each of the three reference positions show large changes in cooling rate during transformation, both due to latent heat of transformation and the changes in thermal-physical properties associated with austenite decomposition.

The predicted microstructure evolution for the corner and centre positions is shown in Figure 20, with the final microstructure after 15 seconds of quenching visualized in two different fashions in Figures 39 and 40. We see that the predicted microstructure is approximately 80% volume fraction martensite, 12% proeutectoid ferrite and the balance retained austenite at the corner, and approximately 80% bainite, 18% proeutectoid ferrite and the balance retained austenite at the centre, with a 6 mm wide transition zone containing all three phases starting from approximately 2 mm from

Table 4. Solution scheme performance for the AISI 1045 quenching case.

Scheme	Steps	Failures	Iterations	Relative time
FD2- $\text{DIRK}(3,2)$	1499	1195	6453	1
FD4- $\text{DIRK}(3,2)$	1244	1225	6295	0.957
FD2- $\text{DIRK}(4,3)$	454	574	4775	0.843
FD4- $\text{DIRK}(4,3)$	436	601	4807	0.879

**Fig 41. Comparison of average integrator step sizes for the water quenching case solved with the fourth-order spatial discretisation and the $\text{DIRK}(3,2)$ and $\text{DIRK}(4,3)$ integrators.**

the corner. We note the presence of retained austenite throughout the microstructure is due to the temperature at the end of the quenching simulation being above the Koistinen-Marburger theory martensite stop temperature.

The relative performance metrics for the four solver schemes in the quenching example are summarized in Table 4, with the average step sizes used by the four schemes are shown in Figure 41. While the $\text{DIRK}(4,3)$ integrator based schemes clearly require fewer total timesteps than the $\text{DIRK}(3,2)$ versions, the computational efficiency advantages of the four stage integrator is considerably less than was the case for the air cooling and spray cooling examples. This is mostly due to the fact that both integrators require approximately the same step sizes during phases transformation, and in this case, phase transformation occurs for almost the complete duration of the problem, which

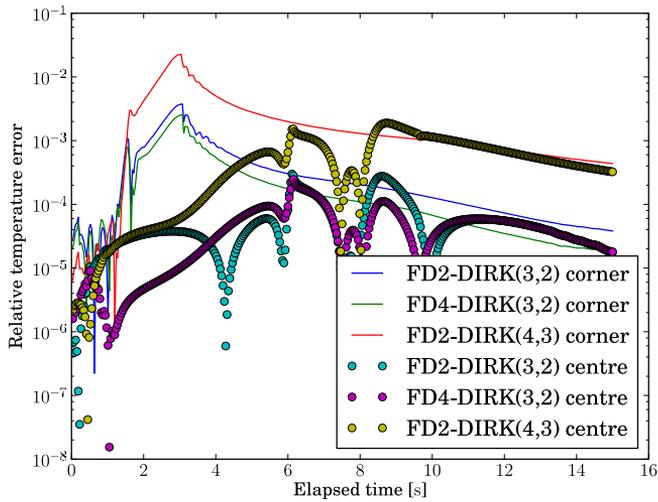


Fig 42. Comparison of relative temperature error for the water quenching case solved with the fourth order FD4-DIRK(4,3) scheme and the FD2-DIRK(3,2) and FD4-DIRK(3,2) schemes.

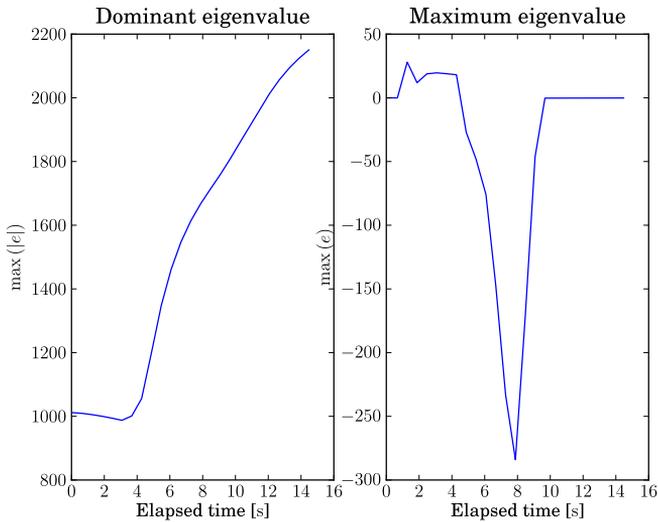


Fig 43. Dominant (absolute maximum) and maximum eigenvalues of the coupled FD4 scheme Jacobian matrix for the quenching case. Calculated from the FD4-DIRK(4,3) solution shown in Figure 36.

reduces the advantage of the DIRK(4, 3) versions compared to other cases. As was the case in the air cooling and spray cooling examples, the eigenvalue analysis shows positive maximum eigenvalues during the onset of the diffusive phase transformation reactions, and the same near blow-up or near singular DIRK scheme Jacobian matrices requiring very small step sizes to resolve.

The relative error analysis for the quenching example, illustrated in Figure 42, is somewhat different in character to the air cooling and spray cooling examples. We note that the corner position relative temperature difference between the FD4-DIRK(4, 3) scheme and the others rises above 10^{-2} at around 3 seconds into the cooling, which translates into maximum absolute corner temperature differences of approximately 12 K. Further investigation suggests that the relative differences between solutions exist between the second and fourth order discretizations rather than between the three and four stage DIRK integrators. We hypothesise that this is due to limitations in accuracy of the second order boundary cell formulation for the Robin condition, when compared to the third order interpolant used in the fourth order discretisation. The transient temperature distribution at this point in the solution is probably beyond the useful approximation accuracy of the second order version. This is probably the only example in these computational experiments where a case could be made to favour the fourth order version of the finite difference approximation over the second order version on the basis of the accuracy of the solution produced. Otherwise, the arguments for the higher order schemes are purely on the basis of computational efficiency.

2.9 Discussion and conclusions

This chapter presented a fairly straightforward analysis of the coupled heat conduction-microstructure evolution problem as we interpret it based on representative literature describing theories of austenite decomposition and the thermophysical properties for hypoeutectoid steels. The approach we have described contains five main elements:

- A non-linear transient heat equation for materials with strongly temperature dependent thermal conductivity taking the form

$$\gamma \frac{\partial u}{\partial \tau} = \lambda \nabla^2 u + \frac{\partial \lambda}{\partial u} (\nabla u)^2 + \sigma.$$

- Irreversible forms of the transformation theories of Kirkaldy and Venugopalan (32) for diffusive transformations and Koistinen and Marburger (41) for the martensite

transformation. The resulting rate equations have been implemented for the hypoeutectoid portion of the Iron-Carbon phase diagram and coupled via a source term to the heat conduction equation.

- A law of mixtures based set of thermophysical properties models for hypoeutectoid steels which express the governing thermophysical coefficients as functions of phase and chemical composition and temperature.
- A method of lines approach for the solution of this system coupled equations, using second or fourth order, central finite difference operators for spatial discretisation which include resolution of the interdependency of heat conduction and transformation rate for the evolution equation which implements the irreversible martensite transformation reaction.
- A step adaptive integrator based on second or third order accurate diagonally implicit Runge-Kutta schemes, solved using a Newton-Raphson method with algebraically calculated Jacobian matrix factorized by LU decomposition.

Through computational examples covering the range of typically cooling conditions associated with thermomechanical treatment of hypoeutectoid steel, we have demonstrated the efficacy of this approach in resolving the considerable non-linearity which is associated with the problem. The sample calculations showed that there are considerable computational efficiency gains to be had from using higher order truncations accuracy integrators combined with the algebraically calculated Jacobian matrix formulations presented for the conduction and microstructure evolution models. There is no avoiding the fact that solution of these fully coupled conduction-microstructure problems is a computationally intensive task, but we believe that the per-step costs of the higher order diagonally implicit Runge-Kutta schemes can be successfully offset when larger step sizes can be used at a give tolerance and a computationally inexpensive Jacobian matrix is available to the solver.

The potential problems associated with ODE stiffness and stability were able to be dealt with competently in this approach, albeit with noticeable computational penalties in the worst cases. Eigenvalue analysis indicated that the interaction of the non-linear conduction equation, thermophysical coefficients, and phase transformations produced Jacobian matrices with real, positive Eigenvalues in some cases. This required very small step sizes to resolve adequately, greatly taxing the step adaptation features of the implicit integrator and increasing the total solution time considerably. A better designed integrator which performs some sort of approximate eigenvalue analysis during as part

of the analysis may have performed better in this situation by detecting stiffness and adjusting the time step accordingly, rather than relying purely on step size adaptation to progressively reduce step sizes until a suitable step size was found. Solutions may be obtained with less total computational expense if such a concept proved feasible.

From a physical viewpoint, the results confirm the large effect that magnetic specific heat can have on the heat balance of proeutectoid steels when austenite decomposition occurs at temperatures well below the Curie Temperature. Evaluating the heat of transformation requires numerical integration of the magnetic heat capacity function with respect to temperature. To reduce the computational cost of computing the transformation enthalpy, we provide an empirical approximation of the magnetic heat capacity integral which can be used directly for this purpose.

We propose that the effect of magnetic latent heat applies as equally to the martensite transformation as to other transformation reactions. There is a school of thought which suggests that the net transformation enthalpy of the steel martensite reaction should be low compared to that of diffusive transformation products because of the high dislocation density and lattice stored energy compared to ferrite rich phases (42). While it is the case that absence of carbide formation accompanying the martensite transformation should lower the total latent heat associated with the lattice transformation, there is still a paramagnetic-ferromagnetic transformation in steel martensite. Because the martensite transformation usually occurs at the low temperatures compared to the Curie temperature, there should be a large amount of magnetic latent heat released during austenite decomposition. The analysis of Browne (12) notes that magnetic latent heat represents at least 80% of the total latent heat of transformation in proeutectoid steels during ferrite and pearlite transformation, and we expect that the situation for martensite will be similar.

The treatment of magnetic specific heat in martensite presented here should still be regarded as a rough approximation, because we have assumed that the magnetic specific heat data for BCC iron can be applied identically in the case of steel martensite. There is something of a dearth of information in the literature regarding the thermodynamic and thermophysical properties of steel martensite, and it is for this reason we have resorted to using this approximation. As martensite is becoming an increasingly important microstructural feature of many classes of advanced high strength, low carbon steels, we suggest that further investigation of the thermophysical behaviour of low carbon steel martensite would be a valuable asset in future study of thermomechanical treatment of such steels.

The other area in which our analysis should be regarded as deficient is the interaction of stress fields with the evolution of microstructure and heat conduction. This has been recognised as an important feature of steel thermomechanical treatment in the literature going back at least to the work of Denis *et al* (43), and while analysis of heat conduction, stress and microstructure are common in the context of distortion of quenched casting and forgings, and heat effected zones in welding, there appears to have been far less attention paid to the subject in the context of thermomechanically treated hot rolled steels. As hot rolled steel microstructures are become more complex, and bainite and martensite are becoming more prominent microstructural features, it would appear that coupling stress fields to heat conduction and microstructure evolution calculations may become a necessity in the near future of the subject.

3 Inverse heat conduction problems in thermomechanical treatment processes

3.1 Introduction

In the first section of this monograph, we described a treatment of the coupled heat conduction-microstructure evolution-thermophysical coefficient problem for hypoeutectoid carbon steels which has application in the simulation of cooling operations which arise in thermomechanical treatment of such steels. In this chapter, we introduce a related but distinct family of problems which also arise in connection with industrial and laboratory scale thermomechanical treatment processes – inverse heat conduction problems.

In general terms, the system of differential equations discussed in Chapter 2 can be solved to yield a unique, stable solution over the complete spatial and temporal domains of the problem if the conditions at the beginning of cooling are known, and the boundary conditions acting at the surface of the workpiece being cooled can fully described *a priori*. There are many practical cases where both of these conditions cannot be satisfied sufficiently to allow prediction of temperature and microstructure, for example calculation of the temperature on the complete spatial and temporal domains in any of the following common scenarios:

- Laboratory simulation of continuous cooling using a deformation dilatometer, where the specimen cools by unknown conduction to the machine jaws or anvils, with continuous surface temperature measurements available only at a single spatial location from a pyrometer or contact thermocouple.
- Laboratory scale hot rolling and cooling, where an embedded thermocouple provides continuous internal temperature measurements at a single spatial location throughout the duration of the thermomechanical treatment schedule.

These two cases represent the techniques that are normally used for temperature measurement during physical simulation in the Materials Engineering Laboratory at the University of Oulu, and it has become customary to use temperature data acquired in this fashion without necessarily paying a great deal of attention to what such measurements are actually indicating, or how they relate to the temperature at spatial locations away

from the measurement point. Although laboratory scale samples tend to have modest cross sections compared to industrial scale product, there can still be a considerable degree of temperature inhomogeneity during rapid heating and cooling cycles.

In both of these examples, insufficient information is available to perform the calculations described in Chapter 2 and quantify the temperature field within the sample during thermomechanical processing. However, in each case, the availability of measured data offers the hope that some sort of useful numerical solution might be obtainable if the measured data could be combined with the differential equations in some fashion. For the purposes of the current discussion, we will refer to this class of problem as *inverse heat conduction problems* or IHCPs, which we will define with more mathematical precision in the next section.



Fig 44. Typical laboratory scale plate with internal thermocouple installed for temperature measurement during thermomechanical treatment experiments.

3.2 A definition of the inverse heat conduction conduction problem by example

To provide a definition of the inverse conduction problems at hand in this Chapter, we consider the problem of computing the temperature within a laboratory scale plate of thickness $2d$ being cooled by a spray whose exact heat transfer characteristics are unknown *a priori*. An internal thermocouple located in the thickness centreline of the plate is connected to data acquisition equipment provides temperature measurements throughout the duration of the cooling process. A typical example type of this type of experiment is shown in Figure 44.

Assuming heat transfer occurs mostly on the top surface of the plate, with uniform heat transfer characteristics across that surface, a one-dimensional approximation of the conduction of heat through the thickness plate can be used. In one spatial dimension, the heat conduction equation analogous to the classes of problem discussed in Chapter 2 is defined as follows

$$\gamma u_\tau = [\lambda u_x]_x + \sigma(u, x), \quad (x, \tau) \in (-d_0, d_1) \times (0, t] \quad (105a)$$

$$u(x, \tau = 0) = w(x), \quad x \in [-d_0, d_1] \quad (105b)$$

$$u(x = d_1, \tau) = v(\tau) \quad \tau \in [0, t] \quad (105c)$$

$$u_x(x = -d_0, \tau) = 0, \quad \tau \in [0, t] \quad (105d)$$

where $w(x)$ defines the initial conditions and $v(\tau)$ is the unknown function which describes the temperature of the surface under the influence of the spray. In this form, Equation (105) cannot be solved to provide a unique solution because of the unknown boundary condition function $v(\tau)$ which acts at $x = d_1$.

If the measured data from the thermocouple is encapsulated in the time dependant, differentiable function $f(\tau)$, Equation (105) can be reformulated as a pair of coupled conduction problems with different characteristics. The first is a Cauchy problem on $(x, \tau) \in [0, d_1] \times [0, t]$ with Cauchy data in the form of conductive heat flux and the thermocouple measured data at $x = 0$:

$$\gamma u_\tau = [\lambda u_x]_x + \sigma(u, x), \quad (x, \tau) \in (0, d_1) \times (0, t) \quad (106a)$$

$$u(x, \tau = 0) = w(x), \quad x \in [0, d_1] \quad (106b)$$

$$u(x = 0, \tau) = f(\tau) \quad \tau \in (0, t] \quad (106c)$$

$$u_x(x = 0, \tau) = k(\tau), \quad \tau \in (0, t] \quad (106d)$$

with the unknown boundary condition function $v(\tau)$ acting at $x = d_1$. As is the case for Equation (105), this problem cannot be solved in the conventional sense because of the unknown boundary condition. We remark that this is occasionally referred to in the literature as the *sideways heat equation*, or SHE (44, 45), and we adopt that nomenclature here to differentiate it from other types of inverse heat conduction problems.

The second problem is a similar Cauchy problem on $(x, \tau) \in [-d_0, 0] \times [0, t]$, with the same Cauchy data at $x = 0$ and a known boundary condition at $x = -d_0$:

$$\gamma u_\tau = [\lambda u_x]_x + \sigma(u, x), \quad (x, \tau) \in (-d_0, 0) \times (0, t] \quad (107a)$$

$$u(x, \tau = 0) = w(x), \quad x \in [-d_0, 0] \quad (107b)$$

$$u(x = 0, \tau) = f(\tau) \quad \tau \in [0, t] \quad (107c)$$

$$u_x(x = 0, \tau) = k(\tau), \quad \tau \in (0, t] \quad (107d)$$

$$u_x(x = -d_0, \tau) = 0, \quad \tau \in (0, t]. \quad (107e)$$

This problem, which we will refer to as the *direct problem*, can be solved directly in the same fashion as those problems discussed in Chapter 2 because it has fully defined boundary conditions, unlike Equation (106).

3.2.1 *Ill-posedness of the SHE*

For the linear operator system which approximates the SHE, Equation (106),

$$\mathbf{A}\mathbf{v} = \mathbf{f}, \quad \mathbf{A} : \mathbb{H}_1 \rightarrow \mathbb{H}_2 \quad (108)$$

with the operator \mathbf{A} bounded between Hilbert spaces \mathbb{H}_1 and \mathbb{H}_2 , \mathbf{v} representing the unknown boundary solution and \mathbf{f} containing the measured data, we define the problem of solving Equation (108) to be *well posed* by the postulates of Hadamard if it satisfies

1. A solution exists for any measured data \mathbf{f} in the data space
2. The resulting solution \mathbf{v} is unique in the solution space
3. The resulting solution \mathbf{v} depends continuously on the data \mathbf{f} .

Conversely, we define any problem which cannot satisfy these three conditions as being *ill-posed*. In practice, the first two requirements imply that to solve Equation (108) in the usual algebraic sense:

$$\mathbf{v} = \mathbf{A}^{-1}\mathbf{f}$$

the inverse of the operator \mathbf{A}^{-1} must be well defined. The third condition implies that the inverse of the operator must be bounded, and it is this requirement which is most often not satisfied in inverse heat conduction problems, leading them to be regarded as ill-posed problems (46). As discussed by Berntsson (45), the amplification factors of typical solutions are positive, and in most cases even small disturbances in the high frequency components of the measured data can lead to catastrophic blow up of the solution. This has severe practical implications for typical sources of measured data which can contain both random noise and systematic errors.

3.2.2 A computational example

The combined effects of ill-posedness of the SHE and measurement noise can easily illustrated by considering a naïve finite difference solution to a greatly simplified, linear version of the problems given by Equation (105):

$$u_{\tau} = (\lambda/\gamma)u_{xx}, \quad (x, \tau) \in (0, 10h) \times [0, 10] \quad (109a)$$

$$u_x = v(\tau), \quad x = 10h, \tau \in [0, 10] \quad (109b)$$

$$u_x = 0, \quad x = 0, \tau \in [0, 10] \quad (109c)$$

$$u = f(\tau), \quad x = 5h, \tau \in [0, 10] \quad (109d)$$

with $\lambda/\gamma = 5 \text{ mm}^2/\text{s}$, and $h = 0.4517 \text{ mm}$. For the purposes of this computational experiment, we solved a well-posed version of Equation (109) with the boundary condition function $v(\tau)$ taken as a time periodic flux given by

$$v(\tau) = \min \left\{ 0, 8.5 \times 10^5 \sin(1 - \tau) \right\},$$

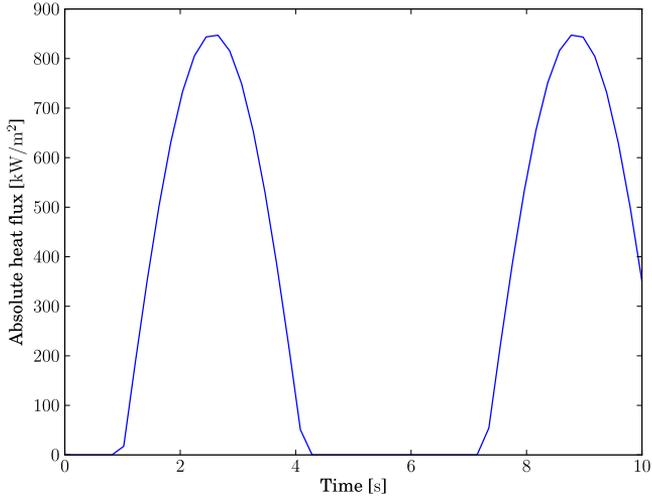


Fig 45. Periodic boundary condition used in the well-posed solution to Equation (109).

by Equation (110), yielding a linear scheme similar to those proposed by Alifanov (47, §5) and Shidfar and Pourgholi (48):

$$\mathbf{K}_2 \bar{\mathbf{u}} = \mathbf{b}$$

with

$$\mathbf{K}_2 = \begin{bmatrix} -\zeta_o & (1+2\zeta_o) & -\zeta_o & & & & \\ & -\zeta_o & (1+2\zeta_o) & -\zeta_o & & & \\ & & -\zeta_o & (1+2\zeta_o) & -\zeta_o & & \\ & & & -\zeta_o & (1+2\zeta_o) & -\zeta_o & \\ & & & & -\zeta_o & (1+2\zeta_o) & -\zeta_o \\ & & & & & -\zeta_o & (1+2\zeta_o) \end{bmatrix}$$

$$\bar{\mathbf{u}}^T = [u_0^{(j+1)}, u_1^{(j+1)}, u_2^{(j+1)}, u_3^{(j+1)}, u_4^{(j+1)}]$$

$$\mathbf{b}^T = [u_1^{(j)}, u_2^{(j)}, u_3^{(j)}, u_4^{(j)} + \zeta_o f^{(j)}, f^{(j)} - (1+2\zeta_o)f^{(j+1)} + \bar{u}_6^{(j+1)}]$$

and

$$f^{(j)} = \bar{u}(x = 5h, \tau = j\Delta\tau) \equiv \bar{u}_5^{(j)}.$$

Here we denote the temperatures computed from the direct problem as \bar{u} to differentiate them from the solution of the SHE.

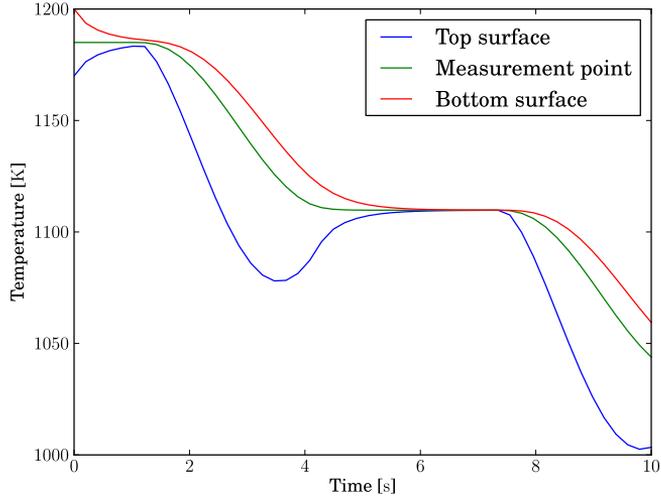


Fig 46. Exact solution and simulated measured data for the well-posed problem given by Equation (109) with the boundary condition shown in Figure 45.

The unknown heat flux $v(\tau)$ can then be estimated from the solution to the ill-posed problem using a second order accurate heat balance at the boundary node:

$$v^{j+1/2} = \frac{\gamma h}{2\Delta\tau} \left[u_0^{(j+1)} - u_0^{(j+1)} \right] - \frac{\lambda}{2h} \left[-u_0^{(j)} + u_1^{(j)} - u_0^{(j+1)} + u_1^{(j+1)} \right], \quad (112)$$

which yields a second order accurate flux estimate centred in both space and time.

The results of these calculations are summarised in Figure 47, which shows the inverse estimated temperature at $x = 10h$ from the simulated measured data without noise, and with normally distributed random noise of zero mean and standard deviations of 1, 2 and 4 K respectively. The corresponding unknown heat flux estimates are shown in Figure 48. These two figures show how rapidly the ill-posed solution deteriorates as measurement noise is introduced into the problem. Unless the measured data used to compute such a solution is effectively noise free, or sufficient *a priori* information about the characteristics of any noise which is present so that precise filtering can be achieved, this type of direct inverse solution method is of little practical use in solving the types of SHE which are of interest in the present work.

As a result of this ill-posedness and the demonstrated effect that even small amounts of measurement noise can on any solution obtain by direct solution methods, some alternative approach is required to obtain a useful solution to the SHE when using

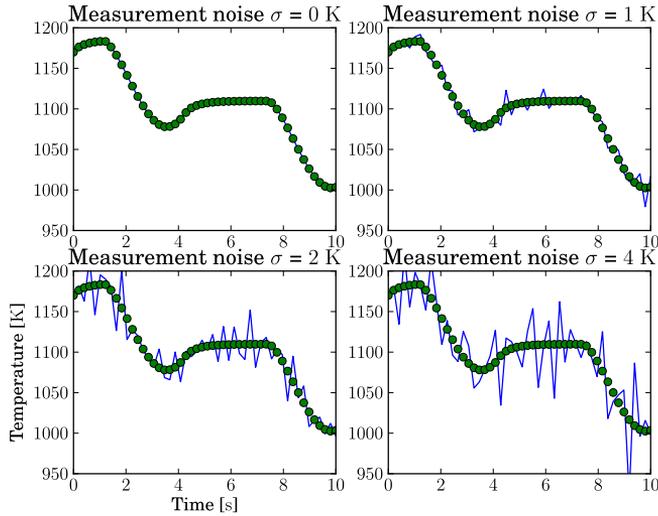


Fig 47. Inverse estimated temperatures at $x = 10h$ for the example SHE given by Equation (109) with simulated random, Gaussian measurement noise. Exact solution shown with points, inverse solution with lines.

measured data containing noise. These alternatives usually involves introducing some kind of additional *a priori* information into the problem to stabilise it sufficiently to yield a useful solution where methods such as the one outlined above fail. This process is usually referred to as *regularisation*.

3.3 Overview of regularisation methods for the Inverse Heat Conduction Problem.

There are, in general, two widely reported approaches to solving problems of the ill-posed problems of the type defined by Equation (108). The first is what will be termed *direct regularisation* methods for the purposes of the current discussion, where the solution procedures require direct access to the entries of the operator matrix \mathbf{A} to effect regularisation on the problem.

One of the earliest forms of direct regularisation was the procedure proposed by Tikhonov (49), whereby an alternative problem is formulated by introducing a penalty term $\chi^2 \|\mathbf{G}\mathbf{v}\|^2$ to explicitly stabilise the ill-conditioned system of linear equations:

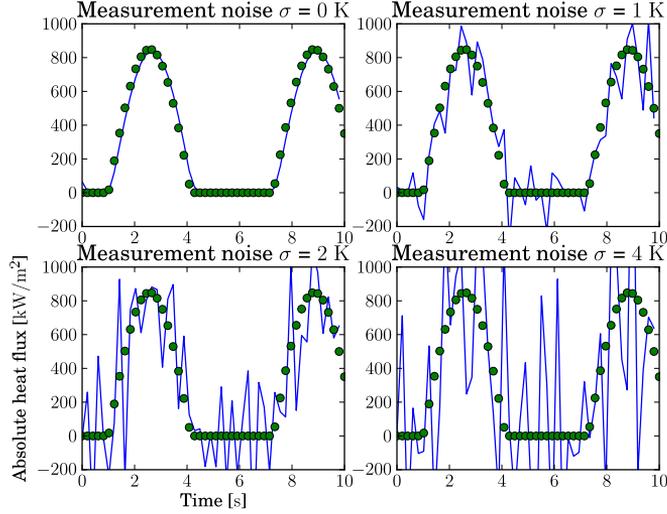


Fig 48. Inverse estimated boundary heat flux at $x = 10h$ for the example SHE given by Equation (109) with simulated random, Gaussian measurement noise. Exact solution shown with points, inverse solution with lines

$$\min_{\mathbf{v}} \|\mathbf{A}\mathbf{v} - \mathbf{f}\|^2 + \chi^2 \|\mathbf{G}\mathbf{v}\|^2, \quad (113)$$

where $\|\bullet\|$ denotes the norm on the applicable Hilbert space. The regularisation parameter χ used to control the degree of smoothness of the solution and the matrix \mathbf{G} is typically some form of differentiation operator. Solving the SHE then becomes a matter of finding a compromise between an acceptably smooth solution and a minimum deviation from the Cauchy data. In the original method described by Tikhonov, \mathbf{A}_Δ is formed as a lower triangular matrix, allowing a least squares method solution via a Q-R factorisation of

$$\min_{\mathbf{v}} \left\| \begin{bmatrix} \mathbf{A}_\Delta \\ \chi \mathbf{G} \end{bmatrix} \mathbf{v} - \begin{bmatrix} \mathbf{f} \\ 0 \end{bmatrix} \right\|. \quad (114)$$

The biggest drawback with this approach is the regularisation parameter and smoothing operator cannot, as a rule, be determined *a priori*, and trial and error is required to find a solution which balances smoothness and accuracy (47, §7). This makes the task of designing general purpose, automatic Tikhonov regularisation procedures rather complex, and has led to a range of other approaches to solving ill-posed systems similar to that defined by Equation (106).

There is a vast amount of literature addressing other types of direct regularisation techniques for Cauchy problems for non-characteristic parabolic systems and poorly conditioned least squares analysis which arise in connection with many common problems in the physical sciences, engineering, and computational finance. The scope of the available literature is so broad that it would be futile to attempt to discuss them all in the context of the present work. Instead, readers are directed to one of the many reviews of literature in the field, such as the excellent ones by Berntsson (50), which covers many of the direct regularisation techniques reported in the literature for Cauchy problems for parabolic and elliptical systems, including single value decomposition methods, sequential methods using integral equations, spectral approximation using Fourier transformations, and mollification techniques based on wavelet functions.

One feature all direct regularisation methods share is the need to directly access the entries of the linear operator matrix \mathbf{A} in Equation (108). For complex, non-linear heat-conduction problems such as the one described in Chapter 2 for cooling of transforming hypoeuctectoid steels, there is no guarantee that the operator matrix, or equivalent integral kernel, is known. For this reason alone, we have sought other types of solution methods to the SHE problem which avoids the necessity of forming and directly manipulating the operator matrix.

The solution method adopted for solving inverse and ill-posed problems in thermo-mechanical treatment process is drawn from the family gradient search based procedures collectively referred to as *iterative regularisation* methods. This approach is described in the literature by a number of authors, including Alifanov (47, §6 & §8) and Jarny *et al* (51). In this method, regularisation of the SHE is effected by formulating a problem based on an extremum statement for the Cauchy problem given by Equation (106) that can be incrementally regularised by an iterative method. The iterative solution method is selected to impart regularisation on the ill-posed problem without the need to directly access the operator matrix, as is the case in Tikhonov methods and other direct regularisation procedures. A detailed description of one such iterative regularisation procedure, as it has been applied to thermo-mechanical treatment problems, is presented in the following sections of the present work.

3.4 Iterative regularisation methods for solution of boundary inverse conduction problems

For the linear operator equation defined by Equation (108), a standard residual function can be expanded into a quadratic problem:

$$\begin{aligned}
 e : e(\hat{\mathbf{v}}) &= \min_{\mathbf{v}} \|\mathbf{A} \cdot \mathbf{v} - \mathbf{f}\|^2 \\
 &= (\mathbf{A}\mathbf{v} - \mathbf{f})^T (\mathbf{A}\mathbf{v} - \mathbf{f}) \\
 &= \frac{1}{2} \mathbf{v}^T \mathbf{A}^T \mathbf{A} \mathbf{v} - \mathbf{v}^T \mathbf{A}^T \mathbf{f} + \frac{1}{2} \mathbf{f}^T \mathbf{f}.
 \end{aligned} \tag{115}$$

This quadratic form can be shown to minimise a system of linear equations given by

$$\mathbf{A}^T \mathbf{A} \mathbf{v} = \mathbf{A}^T \mathbf{f}. \tag{116}$$

Forming $(\mathbf{A}^T \mathbf{A})^{-1}$ and directly solving $\mathbf{v} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{f}$ is, in all likelihood, infeasible because the operator matrix \mathbf{A} is not defined for the non-linear problems considered in the current work. Given $\mathbf{A}^T \mathbf{A}$ is symmetric positive definite and the residual function $e(\tau)$ is Fréchet differentiable with respect to \mathbf{v} , gradient based multi-dimensional optimisation procedures such as the steepest descent and conjugate gradient methods become feasible for solving Equation (116) using only implicit access of the entries in the matrix $\mathbf{A}^T \mathbf{A}$ via a function which returns the equivalent matrix-vector product (52). To this end, we seek to form an objective functional

$$e : e(\mathbf{v}) = (\mathbf{A}^T \mathbf{A})^{-1} - \mathbf{A}^T \mathbf{f}, \tag{117}$$

which can be minimised by iterative regularisation without the need to explicitly form and manipulate the operator matrix in Equation (116).

A general iterative gradient method can be written for a given search direction $\mathbf{g}^{(k)}$ accordingly to Equation (118). In the simplest case $\mathbf{g}^{(k)}$ is simply taken directly from the residual function gradient ∇e , yielding the steepest descent method. Using the Polak-Ribiere variant of the conjugate gradient method described by Press *et al* (53), the search direction is calculated according to Equations (119) and (120):

$$\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} - \beta^{(k)} \cdot \mathbf{g}^{(k)} \tag{118}$$

$$\mathbf{g}^{(k)} = \nabla e^{(k)} + \phi^{(k)} \cdot \mathbf{g}^{(k-1)} \quad (119)$$

$$\phi^{(k)} = \max \left[0, \frac{(\nabla e^{(k+1)} - \nabla e^{(k)}) \cdot \nabla e^{(k+1)}}{|\nabla e^{(k)}|^2} \right]. \quad (120)$$

The depth of descent, β , is calculated so as to minimise the objective functional for the current search direction according to:

$$\beta^{(k)} : e(\mathbf{v}^{(k+1)}) = \min_{\beta \geq 0} e(\mathbf{v}^{(k)} - \beta^{(k)} \cdot \mathbf{g}^{(k)}). \quad (121)$$

Successive iterations of the system of equations leads to refinement of an initial estimate of the unknown heat flux, $\mathbf{v}^{(0)}$. For the initial iteration a steepest descent step with $\phi^{(0)} = 0$ and $\mathbf{g}^{(0)} = \nabla e^{(0)}$ is made.

3.4.1 SHE objective functionals

From the preceding discussion, the primary objectives of formulating the inverse problem using gradient methods shifts to finding both a suitable residual function e and its gradient ∇e that can be used to satisfy Equation (115), and solve the SHE in the least squared sense. Alifanov (47) and Jarny (51) proposes a family of integral objective functions for the SHE. The square of the error between the measured temperature at a given spatial location, $f(\tau)$, and $u(x=0, \tau)$ and the corresponding solution obtained with an estimated value of the unknown heat flux $v(\tau)$, is integrated over the temporal domain to give a final scalar value for the objective cost of the problem to be minimised. In the simplest form, this can be represented by Equation (122).

$$e(v(\tau)) = \int_{\tau=0}^t [u(v(\tau), 0, \tau) - f(\tau)]^2 dt. \quad (122)$$

When multiple temperature measurements are available within the spatial domain, a more general form may be written for k measurements as Equation (123):

$$e_1 = \int_{t=0}^t \sum_{i=1}^k \omega_i [u(v(t), x_i, t) - f_i(t)]^2 dt, \quad (123)$$

where $\{\omega_0, \dots, \omega_k\}$ denotes the set of weight values for the sensors located at spatial locations $\{0, \dots, x_k\}$ through the one-dimensional spatial domain. We note that even with

multiple sensors, as long as there is a sensor at $x = 0$, the problem remains fundamentally a Cauchy problem of the type defined in Section 3.2.

As noted by Alifanov (47, §8), an alternative objective functional exploits the existence of the direct problem defined by Equation (107), by evaluating the error over the directly recoverable temperature field, \bar{u} , on $x \in [-d_0, 0]$ rather than just at the boundary Cauchy data point $x = 0$. This is achieved by integrating the error over not only the temporal domain but also in the spatial domain $[-d_0, 0]$, according to Equation (124):

$$e_2 = \int_{\tau=0}^t \int_{x=-d_0}^0 [u(x, \tau) - \bar{u}(x, \tau)]^2 dx d\tau. \quad (124)$$

In the case where multiple temperature measurements are available, for example $f_1(x = x_1, \tau)$ and $f_2(x = x_2, \tau)$, the concept can be expanded by evaluating Equation (124) over $x \in [-d_0, x_1]$ and $x \in [x_1, x_2]$, with the SHE then defined on $x \in [x_2, d_1]$, yielding Equation (125):

$$e_3 = \int_{\tau=0}^t \int_{x=-d_0}^{x_1} [u(x, \tau) - \bar{u}_1(x, \tau)]^2 dx d\tau + \int_{\tau=0}^t \int_{x=x_1}^{x_2} [u(x, \tau) - \bar{u}_2(x, \tau)]^2 dx d\tau. \quad (125)$$

An extension to these objective functional kernels is the use of penalty terms as a way of introducing a priori information about the solution into the inverse problem, and at the same time promote stability and smoothness during the regularisation process. This leads to an objective function of the general form

$$e_4 = \int_{\tau=0}^t [u(0, \tau) - f(\tau)]^2 d\tau + \chi_0 \int_{\tau=0}^t v(\tau)^2 d\tau + \chi_1 \int_{\tau=0}^t \left(\frac{\partial v}{\partial \tau} \right)^2 d\tau + \chi_2 \int_{\tau=0}^t \left(\frac{\partial^2 v}{\partial \tau^2} \right)^2 d\tau. \quad (126)$$

where zeroth-, first- and second-order penalty terms are added with a set of Lagrange multipliers, $\{\chi_0, \chi_1, \chi_2\}$ whose magnitude is selected to control the degree and mode of smoothing (magnitude, gradient or curvature) in the final solution.

For cases where no *a priori* information as to the nature of the unknown heat flux, the search space for v is infinite dimensional and conventional differential calculus cannot be used to deduce ∇e . For this infinite dimensional case, a method based

on extremum principles is presented. In some specific cases, there may be *a priori* information available about the unknown heat flux, in which case a set of basis functions, ω , encapsulating this information may be written and the unknown heat flux expressed in P -dimensional space as

$$\mathbf{v} = \sum_{i=1}^P v_{(i)} \cdot \omega(\tau)_{(i)}. \quad (127)$$

Depending on the nature of the basis functions, there is the possibility of directly solving for \mathbf{v} using the method of least squares or a gradient method with ∇e deduced analytically. Where ∇e cannot be deduced analytically, a variant of the same basic extremum principles used for the infinite dimensional case can be applied to this finite dimensional case.

3.4.2 Extremum principles applied to the SHE

A Lagrangian identity \mathcal{L} can be written for the one-dimensional inverse problem with a e_1 objective functional by introducing a Lagrange multiplier, $\psi(x, \tau)$, and combining Equations (106) with and (122) to yield Equation (128):

$$\begin{aligned} \mathcal{L}(u, v, \psi) = & \int [u(0, \tau) - f(\tau)]^2 d\tau + \iint \psi(x, \tau) \gamma u_\tau dx d\tau \\ & - \iint \psi(x, \tau) [\lambda u_x]_x dx d\tau - \iint \psi(x, \tau) \sigma(u, x) dx d\tau. \quad (128) \end{aligned}$$

For a given weak solution to Equation (106) on $(x, \tau) \in [0, d_1] \times [0, \tau]$, all terms other than

$$\int [u(0, \tau) - f(\tau)]^2 d\tau$$

will vanish and the Lagrangian will be equal to the value of the objective functional. It follows that some suitable minimisation of the Lagrangian should represent a solution to the ill-posed problem defined by Equation (106).

In the following sections we largely follow the definitions of Alekseev and Navon (54, 55) to derive an adjoint problem for computing the gradient of the regularization problem using this Lagrangian.

3.4.3 The sensitivity equation

If the vector function containing the unknown boundary condition, \mathbf{v} , is perturbed by an incremental change $\Delta\mathbf{v}$, we expect to see a response in stock temperature and thermophysical properties, and in the objective functional e :

$$\begin{aligned}\mathbf{v} &\rightarrow \mathbf{v} + \Delta\mathbf{v} \\ u &\rightarrow u(\mathbf{v} + \Delta\mathbf{v}) \equiv u(\mathbf{v}) + \delta(\Delta\mathbf{v}) \\ e &\rightarrow e(\mathbf{v} + \Delta\mathbf{v}) \equiv e(\mathbf{v}) + \Delta e(\Delta\mathbf{v})\end{aligned}$$

with the Fréchet differentiability of the objective function implying that directional derivative of the objective functional along $\Delta\mathbf{v}$ is

$$D_{\Delta\mathbf{v}}[e(\mathbf{v})] = \lim_{\eta \rightarrow 0} \frac{e(\mathbf{v} + \eta\Delta\mathbf{v}) - e(\mathbf{v})}{\eta}. \quad (129)$$

Adopting the objective functional defined by Equation (122) so that $e = e_1$, the response of the objective function, Δe , is given by

$$\begin{aligned}\Delta e &= e(\mathbf{v} + \Delta\mathbf{v}) - e(\mathbf{v}) \\ &= \int [u(v(\tau), 0, \tau) + \delta(\Delta v(\tau), 0, \tau) - f(\tau)]^2 d\tau \\ &\quad - \int [u(v(\tau), 0, \tau) - f(0, \tau)]^2 d\tau \\ &= \int 2\delta(\Delta v(\tau), 0, \tau) \varepsilon(\tau) d\tau + \int [\delta(\Delta v(\tau), 0, \tau)]^2 d\tau, \quad (130)\end{aligned}$$

where $\varepsilon(\tau) = u(0, \tau) - f(\tau)$. From the definition of the directional derivative, Equation (129), it follows that the objective functional response approximates the directional derivative, i.e.,

$$\Delta e \approx D_{\Delta\mathbf{v}}[e(\mathbf{v})]. \quad (131)$$

The one-dimensional heat conduction equation defined by Equation (106) can be rewritten in terms of the perturbed temperature response to the heat flux increment $\Delta\mathbf{v}$, represented by $u + \delta$, yielding Equation (132):

$$\gamma(u + \delta)_\tau - [\lambda(u + \delta)_x]_x - \sigma(u + \delta, x) = 0. \quad (132)$$

If the unknown boundary condition is assumed to take a Neumann form (Dirichlet and Robin forms are equally admissible (56)), the perturbed boundary conditions become:

$$-\lambda (u + \delta)_x = v(\tau) + \Delta v(\tau) \quad x = d_1, \tau \in [0, \tau] \quad (133a)$$

$$-\lambda (u + \delta)_x = r(\tau) \quad x = 0, \tau \in [0, t]. \quad (133b)$$

The unperturbed components of Equation (132) can be eliminated by subtracting Equation (106). Considering only the resultant linear terms, this yields Equation (134). In the context of the current discussion, this PDE for the temperature increment will be referred to as the *sensitivity equation*:

$$\gamma \delta_\tau - [\lambda \delta_x]_x - \sigma_u \delta = 0 \quad (x, \tau) \in (0, d_1) \times (0, t) \quad (134a)$$

$$-\lambda \delta_x(x = d_1, \tau) = \Delta v(\tau) \quad \tau \in (0, t) \quad (134b)$$

$$-\lambda \delta_x(x = 0, \tau) = 0 \quad \tau \in (0, t) \quad (134c)$$

$$\delta(x, 0) = 0 \quad x \in [0, d]. \quad (134d)$$

By combining the sensitivity problem and the objective functional response, the the perturbed Lagrangian, $\Delta \mathcal{L}$, can be written as:

$$\begin{aligned} \Delta \mathcal{L} = & \int 2\delta \varepsilon(\tau) d\tau + \int \delta^2 d\tau + \iint \psi \gamma \delta_\tau dx d\tau \\ & - \iint \psi [\lambda \delta_x]_x dx d\tau - \iint \psi \sigma_u \delta dx d\tau. \quad (135) \end{aligned}$$

As was the case for the Lagrangian, Equation (128), the properties of the perturbed Lagrangian imply that when $\delta(x, \tau)$ is a weak solution the PDE given by Equation (134) on $(x, \tau) \in [0, d_1] \times [0, \tau]$, the terms from the sensitivity equation will vanish and $\Delta \mathcal{L}$ will take the value of Equation (130), and thus approximate the directional derivative of the objective functional.

3.4.4 The first order adjoint equation

The analysis used to deduce the sensitivity equation in the previous section allows an expression for the increment in the objective functional, and by extension the perturbed

Lagrangian. The Fréchet differentiability of the objective function on a square space (47, §6) allows the directional derivative of e to be given by

$$D_{\Delta \mathbf{v}} [e(\mathbf{v})] = \int \Delta \mathbf{v} \nabla e d\tau, \quad (136)$$

where ∇e denotes the gradient of e . With this form in mind, the Lagrangian for the variational conduction problem, Equation (135), can be rearranged by integrating by parts to yield the *adjoint* of $\Delta \mathcal{L}$:

$$\begin{aligned} \Delta \mathcal{L}^\dagger = & \int 2\delta(d_1, \tau) \varepsilon(\tau) d\tau + \int [\delta \gamma \psi]_{\tau=0}^t dx - \iint \delta [\gamma \psi]_\tau d\tau dx \\ & - \int [\lambda_x \psi \delta_x]_{x=0}^{d_1} d\tau + \int [\lambda \psi_x \delta]_{x=0}^{d_1} d\tau \\ & - \iint \delta [\lambda \psi_x]_x dx d\tau - \iint \delta \sigma_u \psi dx d\tau, \quad (137) \end{aligned}$$

which has the property that if the Lagrangian has a mapping between two Hilbert spaces $\Delta \mathcal{L} : \mathbb{H}_1 \rightarrow \mathbb{H}_2$, the adjoint has the reverse mapping $\Delta \mathcal{L}^\dagger : \mathbb{H}_2 \rightarrow \mathbb{H}_1$. From Equation (137), we can deduce the following set of equalities:

$$\begin{aligned} - \iint \delta [\gamma \psi]_\tau d\tau dx = & \iint \delta [\lambda \psi_x]_x d\tau dx \\ & + \iint \delta \sigma_u \psi d\tau dx \quad (138a) \end{aligned}$$

$$\int [\delta \lambda \psi_x]_{x=0}^{d_1} d\tau = \int \delta(d_1, \tau) 2\varepsilon(\tau) d\tau \quad (138b)$$

$$\int [\delta \gamma \psi]_{\tau=0}^t d\tau = 0 \quad (138c)$$

$$\Delta \mathcal{L}^\dagger = - \int [\psi \lambda \delta_x]_{x=0}^{d_1} d\tau. \quad (138d)$$

From these inequalities, a parabolic, final value problem can be constructed: (138):

$$- [\gamma \psi]_\tau = [\lambda \psi_x]_x + \sigma_u \psi \quad (x, \tau) \in (0, d_1) \times (0, t) \quad (139a)$$

$$\lambda \psi_x = 2\varepsilon \quad x = d_1, \tau \in (0, t) \quad (139b)$$

$$\lambda \psi_x = 0 \quad x = 0, \tau \in (0, t) \quad (139c)$$

$$\psi = 0 \quad x \in [0, d_1], \tau = t, \quad (139d)$$

which we will refer to as the *first-order adjoint problem*. Being a final value problem implies that integration should be performed *backwards* in time from the final conditions.

Because Equation (139) has a time independent RHS, a simple variable substitution can be used to transform the problem to an initial value one which can be integrated using conventional ODE integration techniques.

Finally, when $\psi(x, \tau)$ satisfies the governing equations for the first-order adjoint problem, Equation (139), the increment in the Lagrangian identity $\Delta\mathcal{L}$, can be written as

$$\begin{aligned}\Delta\mathcal{L}^\dagger &= - \int [\psi\lambda\delta_x]_{x=0}^{d_1} d\tau \\ &= - \int \{\psi(x=d_1, \tau)\lambda\delta_x - \psi(x=0, \tau)\lambda\delta_x\} d\tau \\ &= - \int \psi(d_1, \tau)\Delta v(\tau) d\tau.\end{aligned}\tag{140}$$

Recalling that $\Delta e = \int (\Delta v)(\nabla e) d\tau$, the final expression for the can be deduced from Equation (140) as

$$\nabla e = -\psi(h, \tau).\tag{141}$$

The derivation for the adjoint problem presented above is only one of a number of different adjoint equations which can be formulated, depending on the temperature sensor or sensors position and the form of objective functional used.

3.4.5 Optimal depth of descent calculations

With the gradient of the inverse problem determined from a solution to the first order adjoint equation, the final task to complete a solution step of the iterative regularisation process is to determine the optimum depth of descent for a given search direction. From the optimisation statement defined by Equation (121), and the objective functional form shown in Equation (122), a first order accurate Taylor series expansion:

$$u(\mathbf{v}^{(k)} - \beta^{(k)}\mathbf{g}^{(k)}) = u(\mathbf{v}^{(k)}) - \beta^{(k)}\mathbf{g}^{(k)}\nabla_v u + \mathcal{O}([\beta\mathbf{g}]^2)\tag{142}$$

can be used to rewrite the objective functional as a parabolic function in β

$$e(\mathbf{v} - \beta\mathbf{g}) = e(\mathbf{v}) - \beta \int_{\tau=0}^t 2\varepsilon\mathbf{g}\nabla_v u d\tau + \beta^2 \int_{\tau=0}^t [\mathbf{g}\nabla_v u]^2 d\tau\tag{143}$$

This expression is minimised with respect to β when $\partial e/\partial\beta = 0$, implying

$$2\beta \int_{\tau=0}^t (\mathbf{g}\nabla_{\mathbf{v}}u)^2 d\tau - 2 \int_{\tau=0}^t \boldsymbol{\varepsilon}\mathbf{g}\nabla_{\mathbf{v}}u d\tau = 0 \quad (144)$$

leading to the expression for β :

$$\beta = \frac{\int_{\tau=0}^t \boldsymbol{\varepsilon}\mathbf{g}\nabla_{\mathbf{v}}u d\tau}{\int_{\tau=0}^t (\mathbf{g}\nabla_{\mathbf{v}}u)^2 d\tau}. \quad (145)$$

From the definition of the Taylor series expansion given in Equation (142) and the definition of the objective functional increment, Equation (130), it follows that

$$\mathbf{g}\nabla_{\mathbf{v}}u = \delta(\mathbf{g}, t) \quad (146)$$

and

$$\beta = \frac{\int_{\tau=0}^t \boldsymbol{\varepsilon}\delta(\mathbf{g}, t) d\tau}{\int_{\tau=0}^t [\delta(\mathbf{g}, t)]^2 d\tau}. \quad (147)$$

The optimal depth of descent for a given search direction can, therefore, be computed via a solution to the sensitivity problem, Equation (134), with the boundary condition taken as \mathbf{g} , and Equation (147).

3.4.6 *The complete iterative regularisation algorithm for the SHE*

The process required to solve a given conjugate gradient iteration for the iterative regularisation method can be summarized as follows:

1. A solution to the SHE given by Equation (106) is made on $(x, t) \in [0, d_1] \times (0, t]$ with the current best estimate of the unknown boundary condition

$$\mathbf{u}^{(k)} = u(x, t, \mathbf{v}^{(k)}). \quad (148)$$

2. The residual vector \mathbf{r}^k is computed from the current solution to the SHE as

$$\mathbf{r}^{(k)} = f - \mathbf{u}^{(k)}(0, t). \quad (149)$$

3. The objective function value for the current iteration is calculated using the residual data \mathbf{r}

$$e^{(k)} = \left\| \mathbf{r}^{(k)} \right\|^2. \quad (150)$$

4. The adjoint problem is solved using backward time integration with the current temperature error at the measurement point taken as a boundary condition

$$\mathbf{p}^{(k)} = \psi \left(f - \mathbf{u}^{(k)}(0, t), x, t \right). \quad (151)$$

5. The gradient is calculated from the solution to the adjoint problem

$$\nabla e^{(k)} = -\mathbf{p}^{(k)}(h, t). \quad (152)$$

6. The search direction for the current CGM iteration is calculated from the objective functional directional derivative.

$$\gamma^{(k)} = \begin{cases} 0 & k = 1 \\ \max \left[0, \frac{(\nabla \mathbf{e}^{(k)} - \nabla \mathbf{e}^{(k-1)}) \cdot \nabla \mathbf{e}^{(k)}}{|\nabla \mathbf{e}^{(k-1)}|^2} \right] & \text{otherwise} \end{cases} \quad (153)$$

$$\mathbf{g}^{(k)} = \nabla \mathbf{e}^{(k)} + \gamma^{(k)} \cdot \mathbf{g}^{(k-1)}. \quad (154)$$

7. The sensitivity problem is then solved using the current search direction as a boundary condition

$$\mathbf{d}^{(k)} = \delta \left(\mathbf{g}^{(k)}, x, t \right). \quad (155)$$

8. The optimal depth of descent is calculated using the results of the sensitivity problem

$$\beta = \frac{\left\langle \left(f - \mathbf{u}^{(k)}(0, t) \right); \delta(\mathbf{g}, t) \right\rangle}{\|\delta(\mathbf{g}, t)\|^2}, \quad (156)$$

where $\langle \bullet; \bullet \rangle$ denotes the inner product on the \mathbb{L}^2 space.

9. Finally, the heat flux estimate is updated according to

$$\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} - \beta^{(k)} \cdot \mathbf{g}^{(k)}. \quad (157)$$

Each solution iteration requires sequential solution to three separate systems of PDEs. We apply the same method of lines approach to the solution of these PDEs that was used for the non-linear heat conduction equation described in detail in Chapter 2, using finite differences for the spatial discretisation and one of the DIRK integrators for time integration. Following the suggestion of Schewchuk (52), we reset the conjugate search direction after every 20th solution iteration to improve convergence.

3.4.7 Stoppage criteria

The final question in constructing a complete iterative regularization scheme for the SHE is that of when to stop iterating and declare a final solution. Alifanov (47, §6) proposes that iteration should be stopped when the objective function satisfies

$$e^{(k)} \approx \int_0^t s^2(\tau) d\tau, \quad (158)$$

where $s^2(\tau)$ is the variance of the noise in the measured data. The reason for this is straightforward – in the early stages of iteration, the regularization process is mostly minimizing objective functional error due to the low frequency components of the ill-posed solution, which are generally those associated with the true solution to the SHE. As the iterations continue, the regularization begins to be influenced by higher frequency components of the ill-posed solution. At some point the ill-posed solution will begin to follow the noise and non-physical components of the measured data and move away from the true solution to the SHE, implying the existence of some usefully smooth, “best” solution at which the iterative regularization process should be stopped, even when such a solution may not be the locally minimal solution in the least squares sense.

This behaviour is demonstrated in Figure 49, which shows the convergence rates of the four example cases discussed in Section 3.2.2, solved with the iterative regularization method described above (more complete details of these solutions is provided in Section 3.5). In each of the four cases, the same initial estimate of the unknown heat flux was used, implying that any differences in the rate of convergence and the magnitude of the objective functional at a given solution iteration are attributable only to differences in the noise added to the simulated measured data. It is clear from Figure 49 that the convergence rate and objective function values of each of the four cases is almost identical over the first few iterations of the solution, until the case using the “noisiest” measured data ($\sigma=4$ K) begins to diverge. As the number of iterations is increased, each of the noise containing cases eventually diverge from the noise free case and then slowly approach some limiting objective functional value.

It seems logical that the “best” solution will coincide with an iteration number close to the point where the convergence behaviour begins to diverge from the noise free solution, and that this should be describable by the condition stated by Equation (158). Of course, implicit in the use of this stoppage criterion is the availability of *a priori* information about the characteristics of noise in the measured data. We suggest that this

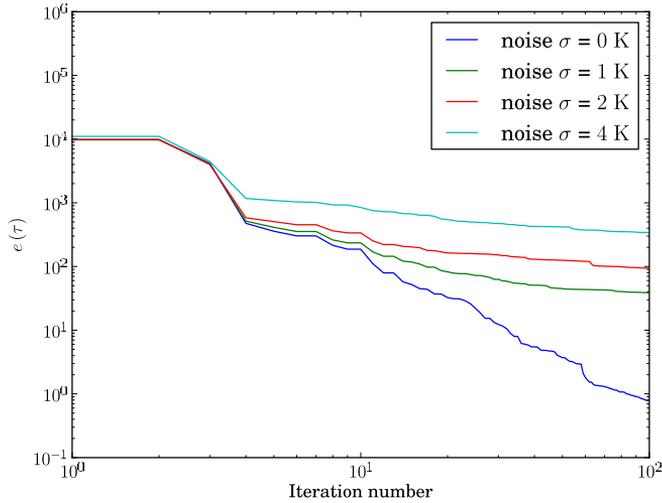


Fig 49. Convergence behaviour of the problem given by Equation (109), solved by the iterative regularization algorithm for simulated measured data with differing amounts of random Gaussian noise.

is more the exception than the rule for most measured data.

It is for this reason that we have elected to omit an automatic stoppage criteria of this kind in the current implementation. Instead, the software implementation has been designed to be re-entrant, and the iteration scheme is provided with two conditions, either of which will cause early stoppage of the regularisation iterations:

1. A small, fixed number of iterations have been executed during this call to the iterative regularization code
2. The objective functional $e(\tau)$ reaches an arbitrary value

Using this approach, the solution is permitted to iterate to a given point, after which the solution can be inspected and visualised. If the current solution is judged to be visually acceptable, i.e. sufficiently smooth with tolerably small deviation from the measured data, we can accept the solution and no further iteration is required. Otherwise the process can be repeated until such a solution is obtained.

3.5 The computational example revisited

To illustrate features of the iterative regularization method described earlier in the Chapter, we revisit the ill-posed problem defined by Equation (109), a direct algebraic solution to which was already presented in Section 3.2.2. We now solve this problem again, this time using the iterative regularisation algorithm described in Section 3.4.6.

To make this solution, we define the objective functional for the ill-posed problem to be

$$e(\tau) = \sum_{j=1}^{50} [f^{(j)} - u_{(5)}^{(j)}]^2$$

with $f^{(j)}$ being the simulated thermocouple measured data taken from the true solution to the problem, shown in Figure 46 on page 98. Solving the adjoint equation on $x \in [-d_0, d_1]$ requires a slightly different adjoint formulation to that defined by (139):

$$-[\gamma\psi]_{\tau} = \lambda \psi_{xx} + 2\varepsilon \quad (x, \tau) \in (-d_0, d_1) \times (0, t] \quad (159a)$$

$$\lambda \psi_x = 0 \quad x = -d_0, \tau \in (0, t] \quad (159b)$$

$$\lambda \psi_x = 0 \quad x = d_1, \tau \in (0, t] \quad (159c)$$

$$\psi = 0 \quad x \in [-d_0, d_1], \tau = t. \quad (159d)$$

A linear version of the sensitivity problem, Equation (134) was used in the regularisation process, with both the adjoint and sensitivities problems solved using the method of lines. Node centred, second order accurate central differences were for the spatial discretisation, as was the case in the solution described in Section 3.2.2, and the resulting ODE was integrated in time using the DIRK(2, 2) integrator. The manual stoppage criteria outlined in the previous section was used to determine when the subjectively “best” looking solution had been reached, with iterations being performed in a re-entrant fashion in sets of four.

The “best” solutions for the four cases are illustrated in Figures 50 and 51. In each case the iterative regularization process was able to resolve a usefully smooth boundary heat flux function which agrees well in both general form and value with the exact periodic heat flux used to generate the measured data. These results are in stark contrast to the direct algebraic solution presented in Section 3.2.2, where the presence of random measurement noise produced catastrophic numerical blow up in the solution (see Figure 47 on page 99 and Figure 48 on page 100 for comparison). The selected stoppage iteration and objective functional values at stoppage for these solutions are given in Table 5.

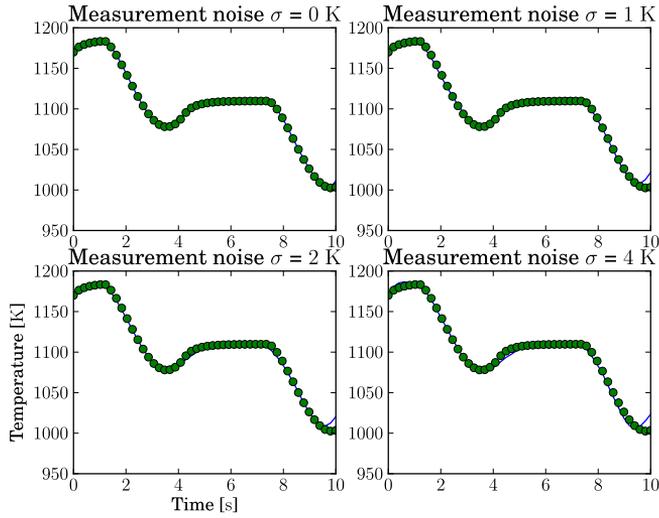


Fig 50. Unknown boundary temperature estimates for the problem given by Equation (109), solved by the iterative regularization algorithm for simulated measured data with differing amounts of random Gaussian noise.

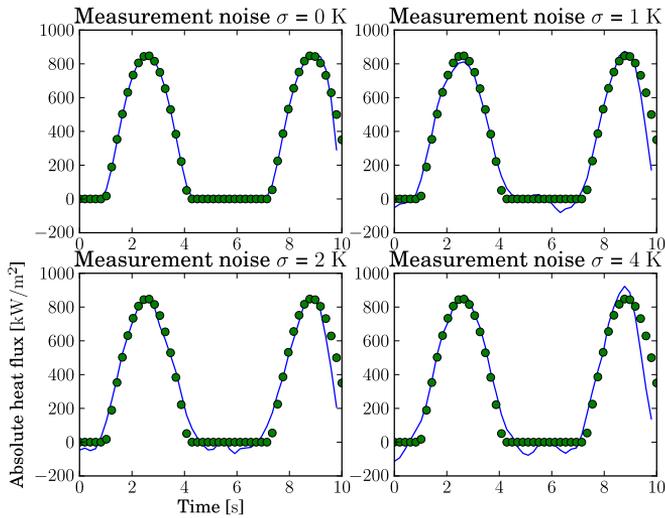


Fig 51. Unknown boundary heat flux for the problem given by Equation (109), solved to a “best” solution using the iterative regularization algorithm for simulated measured data with differing amounts of random Gaussian noise.

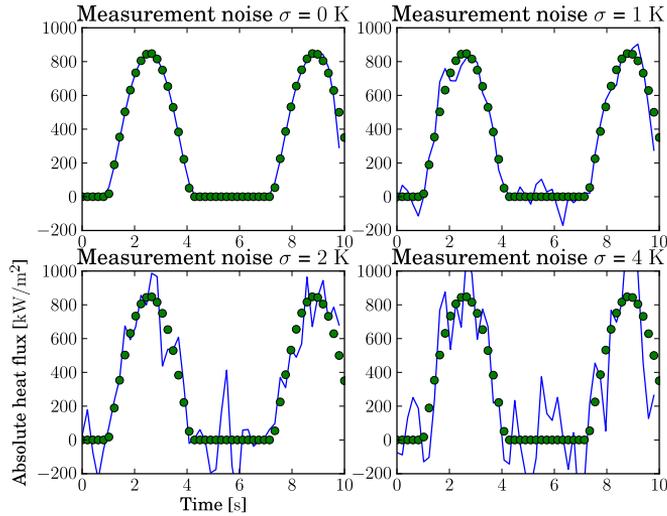


Fig 52. Unknown boundary heat flux for the problem given by Equation (109), solved using 100 CGM iterations of the iterative regularization algorithm.

Table 5. Stoppage iterations and objective functional values for the solutions to the SHE shown in Figures 50, 51 and 52.

noise	stoppage		
standard	iteration	$e(\tau)^{(k_{\text{stop}})}$	$e(\tau)^{(100)}$
deviation	k_{stop}		
0	100	0.8646	0.8646
1	24	38.199	20.335
2	16	280.53	102.49
4	12	886.16	524.75

To demonstrate the benefits of the early stoppage strategy, iteration was also continued beyond the “best” solution to a final total of 100 CGM iterations in each case. The final objective function values after 100 CGM iterations are also given in Table 5. In each of the cases where random noise was added to the simulated measured data, the “best” solution occurs at a considerable larger objective functional value than that achieved when the solution was allowed to proceed to 100 solution iterations. Figure 52 shows the boundary heat flux solutions for each of the four cases after 100 iterations, along

with the exact solution. While the effect is not as pronounced as in the direct algebraic solutions, it is clear that excessive iteration of can produce considerable deterioration in the usefulness of the solution as the iterative regularisation process becomes influenced by the noise in the measured data.

3.6 An analysis of pilot plant hydraulic descaling

We demonstrate the use of the inverse method through an analysis of a set of experiments intended to study the surface temperature behaviour and effectiveness of the hydraulic descaling process. While this process is not conventionally regraded as a thermomechanical treatment process by the strictest definition because it is usually not associated with microstructure evolution, in this case the analysis required coupled phase transformation and non-linear thermophysical properties models with many of the features of the simulation techniques discussed in Chapter 2.

We note that an alternative, simplified version of the analysis presented in this section was published in reference (57). The present work supersedes this previous work and corrects a number of shortcomings present in that previous analysis.

3.6.1 An overview of the hydraulic descaling process

Hydraulic descaling is a high pressure spray cooling process used almost universally during hot rolling of high quality carbon steel flat and long products, with the primary objective of removing surface oxide scales which form on the surface of these steels at working temperatures from oxidation reactions with atmospheric oxygen and water vapour. This removal of surface oxides is regarded as critical to producing final rolled products with the highest levels of surface quality and integrity, particularly in hot rolled flat products such as plate and strip.

The process usually takes the form of one or two banks of flat type spray nozzles fed by a very high pressure water supply, typically 100 – 500 bar. The bar or plate is passed under the banks of nozzles so that the spray impinges on the steel surface, typically at an angle of 15° from vertical. The combined effects of the impact pressure of the spray jet and the rapid cooling of the scale act to fracture and then remove the scale from the bar surface. This process typically occurs directly before entry of the steel into a rolling pass so as to ensure that the steel is free of oxide scale, and that the surface temperature isn't excessive, both countermeasures to ensure good surface quality during hot rolling.

3.6.2 Experimental procedure

The tests were conducted using samples of a commercially produced HSLA steel. Heavy coil samples, 11 mm in thickness, were cut into 75 x 250 mm pieces and prepared by drilling a pair of parallel holes in the long edge to a depth of 30 mm. Two embedded thermocouples per sample were prepared using standard fibre insulated 0.8 mm K-type thermocouple cables which are threaded through a short length of 6 mm diameter high-Ni austenitic stainless steel tubing. A mechanical junction is made between the thermocouple wires, and the junction folded against the tube, placing the hot junction approximately 10 mm behind the end of the tube. A rough diagram of the final embedded thermocouple layout is shown in Figure 53. The principle of operation of these thermocouple assemblies is that the tube is oriented inside a plate specimen such that the junction faces upwards towards the plate upper surface. The specimen is then rolled, and the resulting plastic deformation partially closes the hole containing the thermocouple, crushing the thermocouple junction between the tube and the inner surface of the specimen, and ensuring the thermocouple and tube are securely embedded in the specimen.

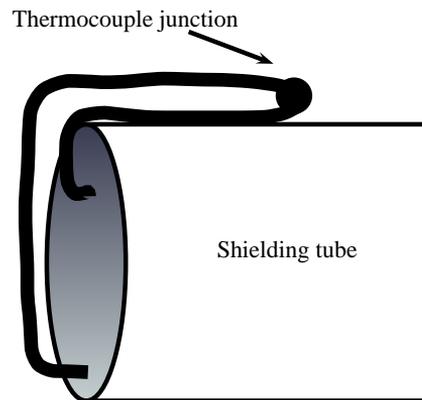


Fig 53. General construction of the shielded internal thermocouples used in the pilot descaling trials.

To prepare the test samples, each drilled plate was reheated to 1000 °C in an electric chamber furnace. After removal from the furnace, the thermocouples were inserted and oriented inside the holes of the hot samples, which were then hot rolled down to 8.5 mm thickness in a single pass using a laboratory two high mill. This reduction was done to

develop sound thermal contact between the thermocouple junction and the sample. After cooling, the samples were allowed to air cool back room temperature. During cooling, a voltmeter was connected to each thermocouple cable to confirm correct operation. Three plates were prepared in this fashion for the pilot descaling trials, one of the completed plates is shown in Figure 54.

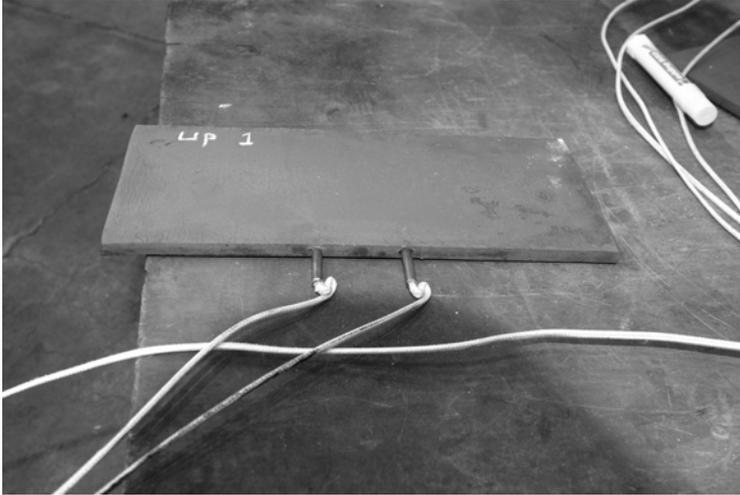


Fig 54. Hot rolled descaling test sample showing internal thermocouples.

At the Swerea-MEFOS pilot plant in Luleå, Sweden, the samples were reheated in an electric chamber furnace to 1100 °C for approximately 12 min. The samples were then removed from the furnace and placed into a sample holder constructed inside the MEFOS standalone descale box, and descaled while stationary for approximately 15 s on the top surface. Water was delivered to the sample surface by a pair Lechler type 694.767 industrial descaling nozzles, which deliver 71 l/min per nozzle angle of 30° at the test pressure of 150 bar, according to manufacturer supplied data. The nozzle to sample distance was fixed at 330 mm for the trials. The geometry of the descaler and samples meant that one thermocouple recorded temperature directly under the descale spray, and the other 10 mm to 20 mm outside the direct impingement zone of the spray. A photograph of a dummy sample of the same size as the thermocouple containing test pieces *in-situ* in the descaler is shown in Figure 55.

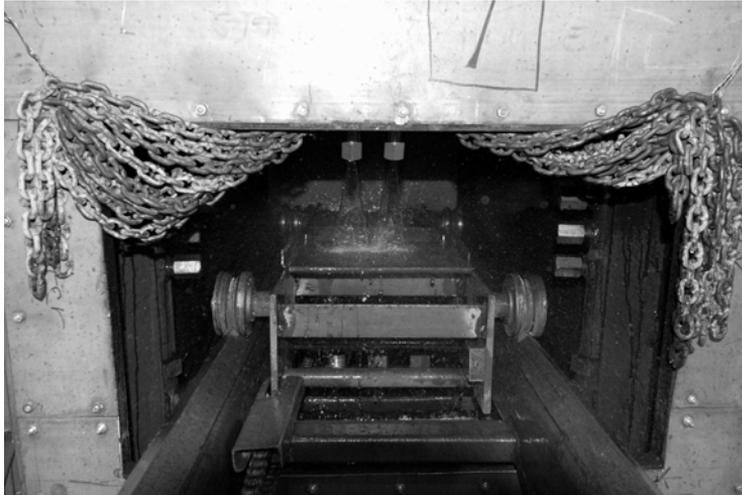


Fig 55. Dummy descaling test sample in the Swerea-MEFOS descaler. Only bypass water shown.

To record data from the thermocouples, a PC running National Instruments Labview software with a high speed data acquisition card was used. The thermocouples were connected to the data acquisition system via 10 m long compensated leads, which were, in turn, connected to the PC via a set of isolated analogue signal conditioning amplifiers which include cold junction compensation from an inbuilt thermistor. Temperature recordings were made at 200 Hz, starting from the last few tens of seconds in the furnace, until about 10 seconds after the descaling water valves were closed and cooling of the samples had ceased.

Three samples were descaled in this fashion. During the first test, some difficulties were experienced inserting the sample into the descaling box sample holder. One of the thermocouple cables snagged during this time, partly damaging the connection to the compensated leads. As a result of this problem, only the thermocouple inside of direct impingement zone recorded for the complete duration of the tests and the sample was cooled by bypass water for longer before the descaling valves were opened than on subsequent tests. After two more tests had been completed, a comparison of the results excellent data quality on all channels bar the first test. At this point it was felt little more would be gained by performing further repetitions, particularly with the complexities of surface changes between successive descale and reheating cycles, and the trials were

concluded. The raw measured temperature data from the tests is shown in Figure 56.

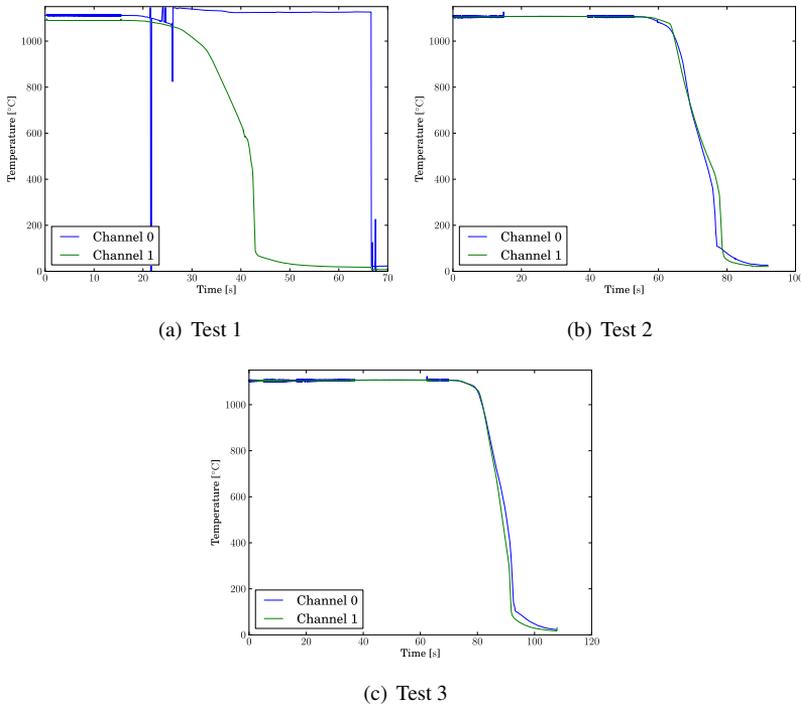


Fig 56. Raw 200 Hz measured temperature data from the three pilot plant tests. Damage to a connection on channel 0 of Test #1 resulted in loss of valid measurements.

After the descaling tests, the samples underwent physical analysis. Each sample was sectioned and ground using a coarse metallographic grinding pad until the thermocouple junctions were located and exposed. The depth of the junction below the descaled surface and the position of the junction relative to the direct impingement zone were measured using a stereo microscope. In each of the samples, the junctions were found between 20 mm and 22 mm in from the sample edge, at a depth of between 2.1 mm and 2.2 mm below the sample surface. The descaler jet direct impingement zone was physically visible on each sample (see Figure 57 for an example), and measurements showed the width to be between 14 mm and 16 mm in each case. Similarly, there was a clear demarcation between regions adjacent to the descaler jets which had been descaled, and those which had the primary scale layer largely intact. On each sample, the total

length of the descaled portion was measured to be between 155 mm and 165 mm.

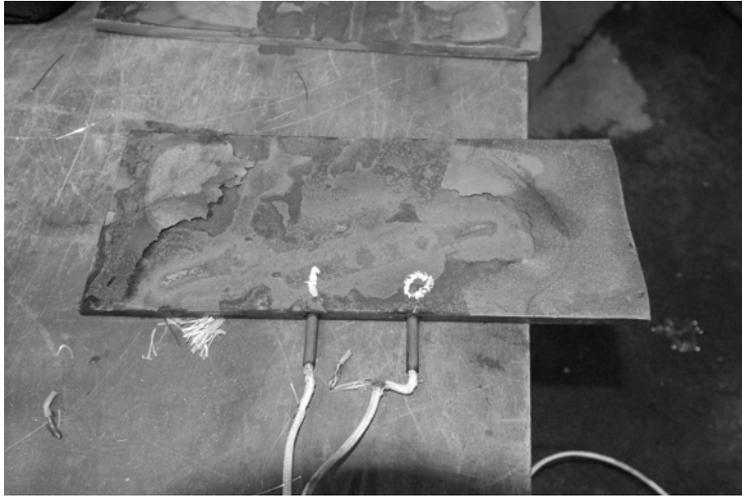


Fig 57. Cold descaling test sample after stationary descaling for 20 seconds at 150 bar water pressure.



Fig 58. Micrograph showing thermocouple wire pairs in cavity after rolling and descaling.

3.6.3 Input data preparation

The raw temperature data sets from the three tests were decimated from the 200 Hz raw signal to 20 Hz using an 8th order Chebyshev low pass filter. The decimated data was then trimmed to contain approximately the final 10 s of measurements before the commencement of descaling, and about another 10 s of data after the end of descaling. This produced a relatively noise free input data set of approximately 1200 samples, or 60 s of problem time per channel per experiment. The 5 valid sets of input data are shown in Figure 59.

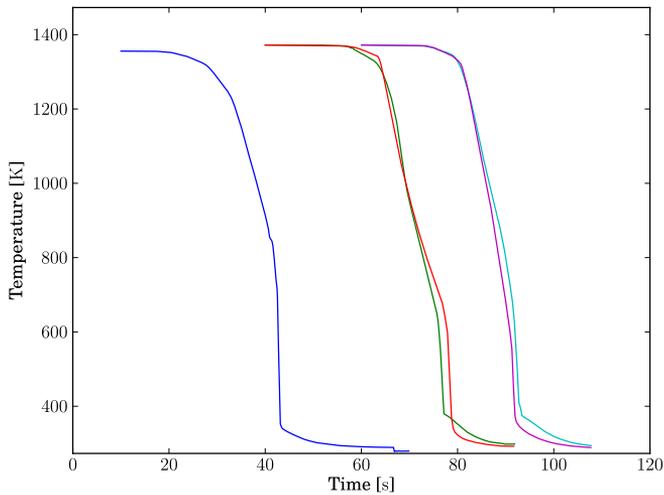


Fig 59. Decimated measured temperature datasets used for the inverse analysis. Test #1 on the left, Test #2 centre, Test #3 right.

3.6.4 Inverse analysis

In the hydraulic descaling example, both Stefan-Boltzmann radiation and convection are assumed to act on the descaled surface, while the inner surface (ie. that exposed to the thermocouple cavity and shielding tube assembly) assumed to be insulated. At first inspection, this might seem like an unreasonable assumption, however we justify its use on two grounds. Firstly, we expect the combination of the air gap within the shielding tube will have low thermal conductivity and heat capacity compared with the parent plate, meaning that conductive losses from the inner surface should be low. Secondly,

We expect that the temperature of the inner surfaces of the tube should be such that net radiative losses from the region of the tube contacting the thermocouple junction should be negligible. This interpretation of the problem is illustrated in Figure 60.

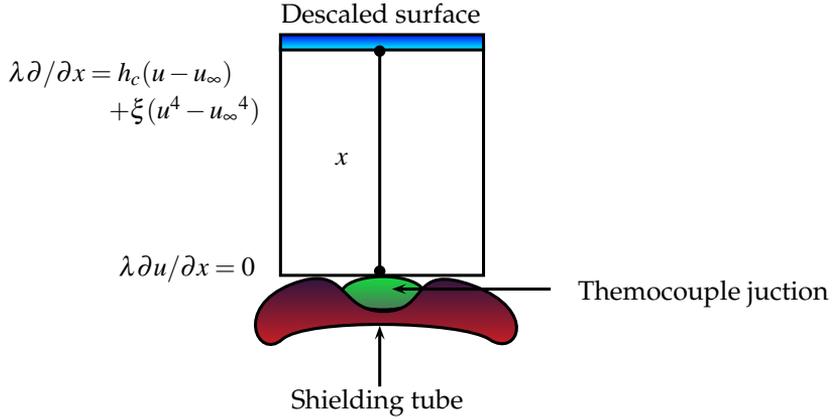


Fig 60. Assumed geometry of the test piece for the inverse analysis.

To perform inverse analysis on the measured data from the descaling trials, we consider the non-linear version of the SHE problem discussed in Section 3.2, solved by a variant of the iterative regularisation process outlined in Section 3.4. This situation requires some slight modifications to the basic solution procedure discussed so far in this Chapter, which are summarised below.

The ill-posed SHE for this situation, with combined Robin and Stefan-Boltzmann boundary conditions with unknown coefficients is taken as:

$$\gamma u_{\tau} = \lambda u_{xx} + \lambda_x u_x + \sigma \quad (x, \tau) \in (0, t) \times (0, d) \quad (160a)$$

$$-\lambda u_x = v(\tau)(u - u_{\infty}) + \xi(u^4 - u_{\infty}^4) \quad x = d, \tau \in (0, t) \quad (160b)$$

$$-\lambda u_x = 0 \quad x = 0, \tau \in (0, t) \quad (160c)$$

$$u = f(\tau) \quad x = 0, \tau \in [0, t] \quad (160d)$$

$$u = f(0) \quad x \in [0, d], \tau = 0 \quad (160e)$$

with $v(\tau)$ being the unknown boundary heat transfer coefficient, $d = 2.4$ mm, and $f(\tau)$ is the thermocouple measured data. For the surface radiative properties, we assume an

emission coefficient of $4.2528 \times 10^{-8} \text{ W/m}^2 \text{ K}^4$ throughout each experiment. Although the sample surface is oxidised, we have neglected any effect of oxide scale, hence the heat transfer coefficient $\nu(\tau)$ is an effective net heat transfer coefficient which implicitly contains the conductance and capacitance effects of the surface scale through which heat is being conducted from the parent steel to the convective cooling medium.

The measured data used for the inverse analysis contains both temperatures measured with the sample inside the reheating furnace, and in ambient temperature air. The temperature inside the reheating furnace is taken as 1373 K, while the ambient temperature of both air and descaling water was assumed to be 300 K. To allow sensible results with combined radiation and convection over such large changes in ambient temperature, the ambient temperature used in all calculations is formulated as a time dependent function

$$u_{\infty}(\tau) = \begin{cases} 1373 & \tau < \tau_{\text{extract}} \\ 300 & \text{otherwise} \end{cases}, \quad (161)$$

where τ_{extract} is the time the specimen was removed from the furnace, which is taken as 10 s after the commencement of measuring in each of the three tests.

Because of the extreme high cooling rates, the microstructure analysis was somewhat simplified compared to the fully coupled problem discussed in Chapter 2 by considering only the irreversible martensite transformation. For the steel in question, the martensite start temperature was estimated to be 754 K using the calculations described in Appendix 3, meaning that we assume an austenitic microstructure above that temperature, and transformation according to the modified Koistinen-Marburger equation given by Equation (162). Latent heat of transformation associated with the martensite reaction is introduced into the problem via the source term in the conduction equation, as outlined in Chapter 2, so that the source term is defined by Equation (163).

$$\frac{\partial \chi}{\partial \tau} = -\eta \exp[-\eta(u_{ms} - u)] \widehat{\frac{\partial u}{\partial \tau}} \quad (162)$$

with

$$\widehat{\frac{\partial u}{\partial \tau}} = \mathcal{H} \left(-(u_{ms} - u) \frac{\partial u}{\partial \tau} \right) \frac{\partial u}{\partial \tau}$$

$$\sigma = \begin{cases} 0 & u > u_{ms} \\ -(\rho\mu)_m \frac{\partial \chi}{\partial \tau} & u \leq u_{ms} \end{cases}. \quad (163)$$

The inclusion of Stefan-Boltzmann radiation and a Robin condition coefficient at the descaled surface in Equation 160 requires some modifications to the sensitivity and adjoint equations compared to the ones derived for a Neumann condition in Section 3.4. The sensitivity problem for this case is taken as

$$\gamma \delta_\tau = \lambda \delta_{xx} + \lambda_x \delta_x + \dot{\sigma} \delta \quad (x, \tau) \in (0, t) \times (0, d) \quad (164a)$$

$$-\lambda \delta_x = \Delta v (u - u_\infty) + (v + 4\xi u^3) \delta \quad x = d, \tau \in (0, t) \quad (164b)$$

$$-\lambda \delta_x = 0 \quad x = 0, \tau \in (0, t) \quad (164c)$$

$$\delta = 0 \quad x \in [0, d], \tau = 0, \quad (164d)$$

where

$$\dot{\sigma} = \frac{\partial \sigma}{\partial u}.$$

We define the objective functional for the descaling analysis to be

$$\begin{aligned} e(\tau) &= \int_0^t [f(\tau) - u(\tau)]^2 d\tau \\ &= \langle f_k - u(0, \tau_k), f_k - u(0, \tau_k) \rangle, \end{aligned} \quad (165)$$

which leads to an adjoint problem

$$\gamma \psi_\tau = \lambda \psi_{xx} + \lambda_x \psi_x + \dot{\sigma} \psi \quad (x, \tau) \in (0, t) \times (0, d) \quad (166a)$$

$$\lambda \psi_x = (v + 4\xi u^3) \psi \quad x = d, \tau \in (0, t) \quad (166b)$$

$$\lambda \psi_x = 2\varepsilon \quad x = 0, \tau \in (0, t) \quad (166c)$$

$$\psi = 0 \quad x \in [0, d], \tau = t \quad (166d)$$

and a gradient of the objective functional defined by

$$\nabla e = -\psi(d, \tau) (u(d, \tau) - u_\infty(\tau)). \quad (167)$$

The same CGM based iteration scheme described in Section 3.4.6 was used to regularise the ill-posed problem using the modified partial differential equations defined above. Each of the direct, adjoint and sensitivity problems were solved by a method of lines approach using a second order accurate one-dimensional finite difference based

spatial discretisation and the DIRK(3,2) integrator for the time marching scheme. A total of 11 nodes were in the spatial discretisation, with $\Delta x = 0.22$ mm. Coupling between the transformation equation and the sensitivity and adjoint problems was effected only via the source sensitivity term computed as part of the solution to the direct problem. All numerical analysis was performed using Python code sharing common modules with that described in Chapter 2.

3.6.5 Results and discussion

Between 50 and 100 iterations of the CGM solver were required to reach visually satisfactory solutions from an initial zero valued solution contain no *a priori* information, with total execution times for each input data set of between two and five minutes on a standard desktop PC per solution run. The evolution of a solution and the convergence rate of the method for one analysis are illustrated in Figures 61(a) and 61(b).

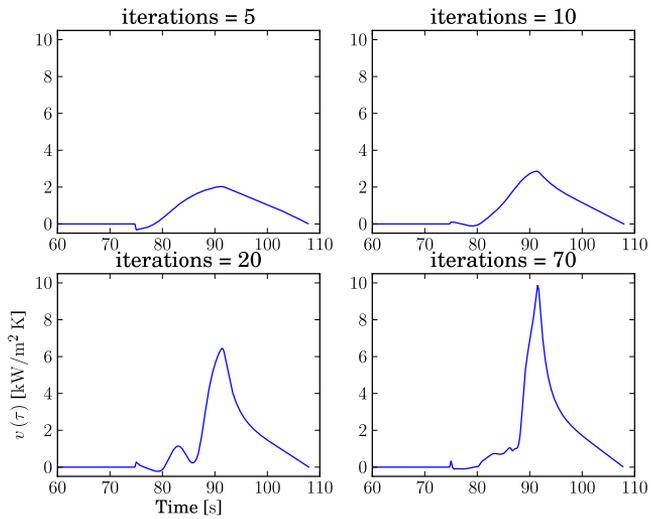
The main results from each individual inverse analysis for the five input data sets are summarised in Figures 62, 63, 64, 65, and 66. Each figure shows the estimated temperature at the measurement point compared with the measured input data, and the estimated effective boundary heat transfer coefficient against time and estimated surface temperature. We stress that these heat transfer coefficient estimates are *effective* values, with implicit surface effects related to the presence of scale and the descaling action included.

The inverse estimated temperatures have very good agreement down to surface temperatures of approximately 550 K, after which there is noticeable deviations of over to 100 K about this temperature which were not resolvable by further iteration, resulting only in distortion of the heat flux estimate. We interpret this convergence limit as meaning that there are features of the true solution which have frequencies which are larger than the maximum for which the regularisation process can resolve. As a result, it was difficult to determine the peak heat transfer coefficient with any degree of certainty. The stoppage points for each test were, therefore, selected on the basis of the smoothness of the solution alone, stopping when the well fitted portions of the solution began to follow noise in the input data. In this respect, the results from Test #3 were the best performing, while Test #1, which had the highest peak cooling rate of all of the five input data sets, was clearly the poorest.

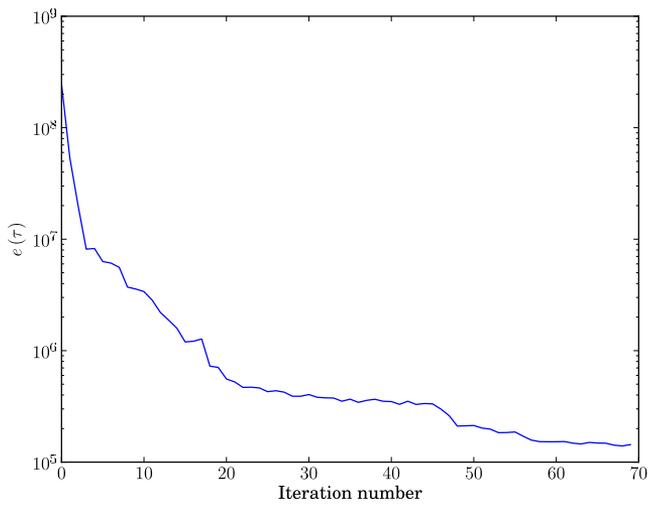
The results exhibit all the heat transfer characteristics which might be expected of hydraulic sprays, with the heat transfer coefficient being relatively independent

of surface temperature at high degrees of superheat, then a transition to much larger values with peak heat transfer coefficient values of around $8 - 10 \text{ kW/m}^2 \text{ K}$ at surface temperatures in the range of $450 - 550 \text{ K}$, or $150 - 250 \text{ K}$ superheat over the assumed ambient temperature of 300 K . The transition from the temperature independent to temperature dependent regime is somewhat broader and more diffuse than that predicted by models of pressure spray cooling, such as the one proposed by Hodgson *et al* (37), or the later work of Wendelstorf *et al* (58). We postulate this is most likely due to the dual effects of the ill-posed nature of the problem, and the influence of surface oxide scale which is not explicitly considered in the analysis.

The samples started with a compact, mill scale layer of the order of $10 \mu\text{m}$ thickness, and were then reheated in air at $1000 \text{ }^\circ\text{C}$ for 15 min before rolling. After slow cooling back to room temperature, the samples were reheated for a second time in air to $1100 \text{ }^\circ\text{C}$ for a further 12 min prior to descaling. As is clear in Figure 57, the sample surface is oxidized to a considerable degree in the undescaled portions of the sample top surface. Although the oxide scale thickness was not measured on these samples, thermobalance experiments and scale thickness measurements on similar steel compositions done as part of the same trials suggest that scale thickness should have been of the order of $100 \mu\text{m}$ at the time of furnace extraction during the descaling experiments (57, §4).

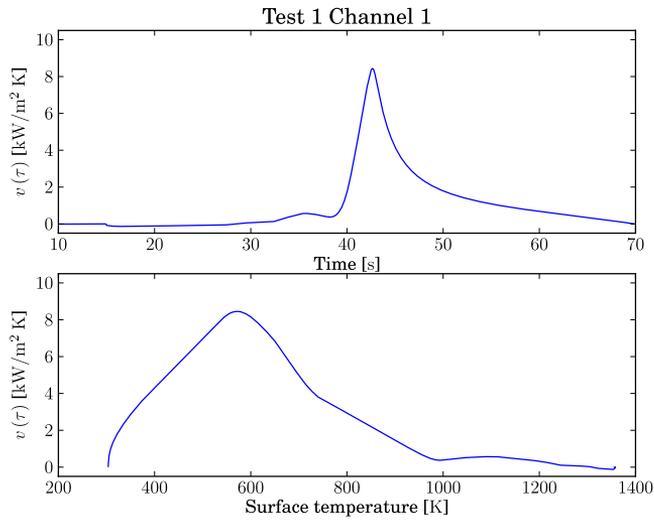


(a) Solution

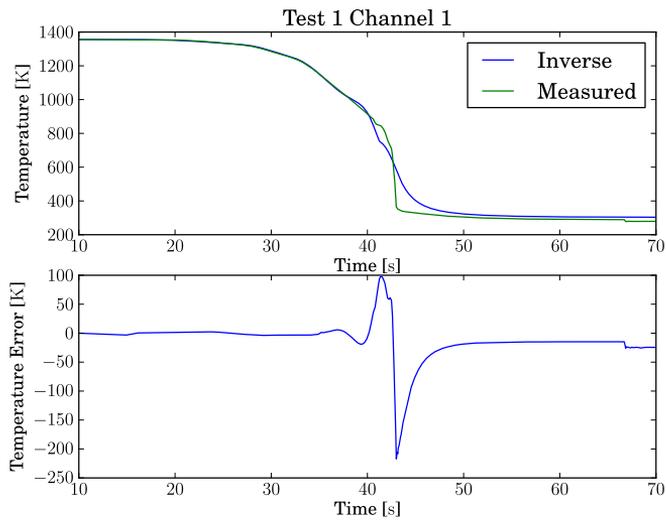


(b) Objective functional

Fig 61. Solution and objective function value after different numbers of iterations from an initial zero valued estimate for the input data set from Test #3, Channel #1.

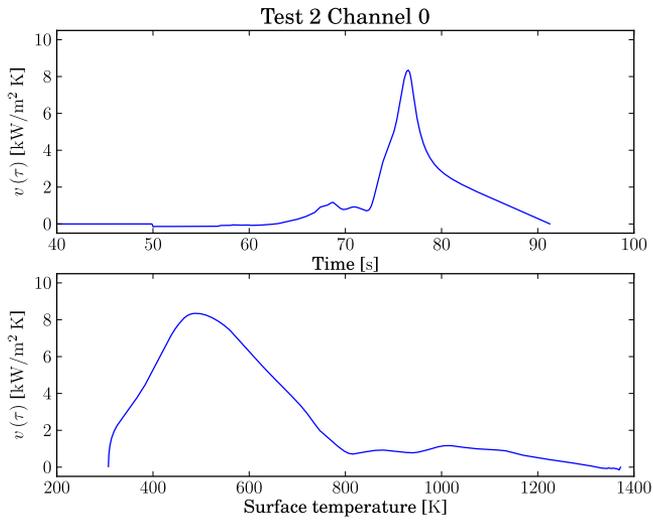


(a) Heat transfer coefficient

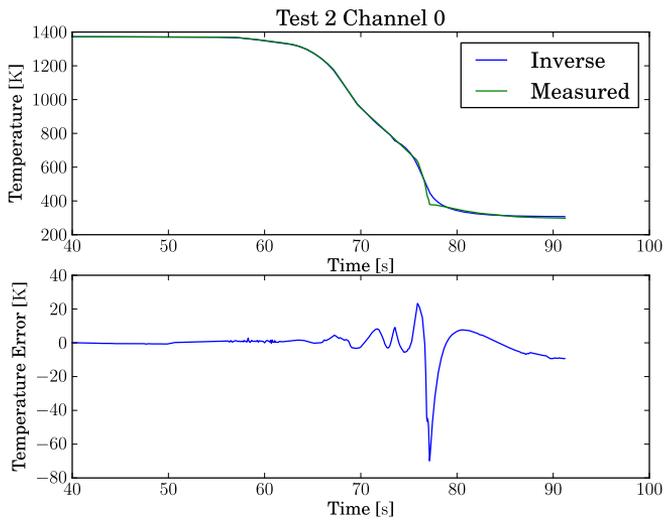


(b) Temperature

Fig 62. Inverse analysis results for Test #1, Channel 1.

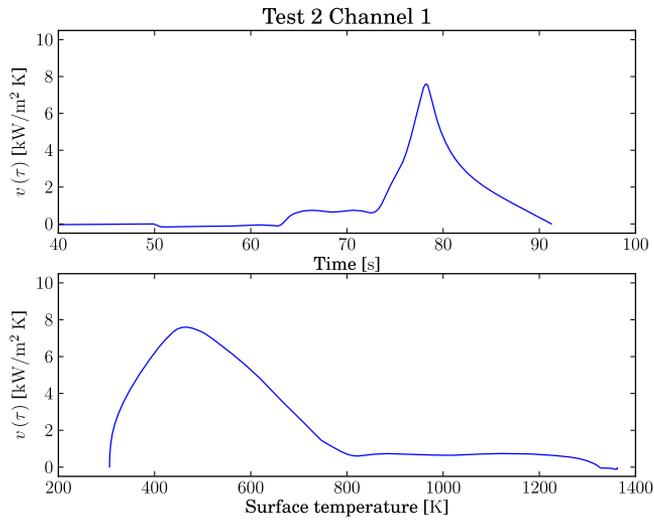


(a) Heat transfer coefficient

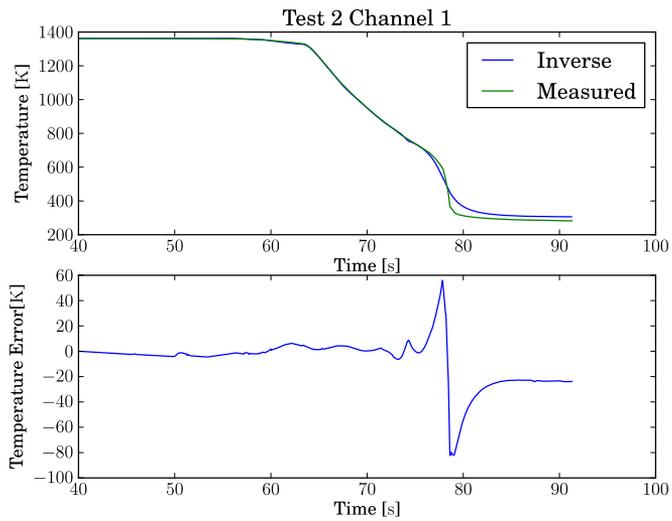


(b) Temperature

Fig 63. Inverse analysis results for Test #2, Channel 0.

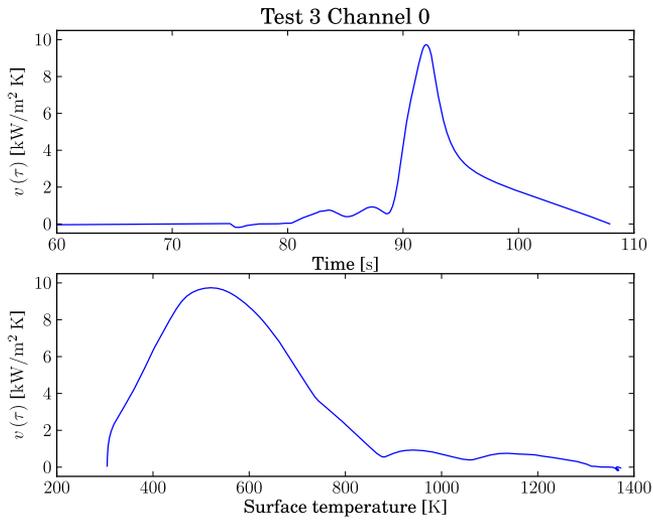


(a) Heat transfer coefficient

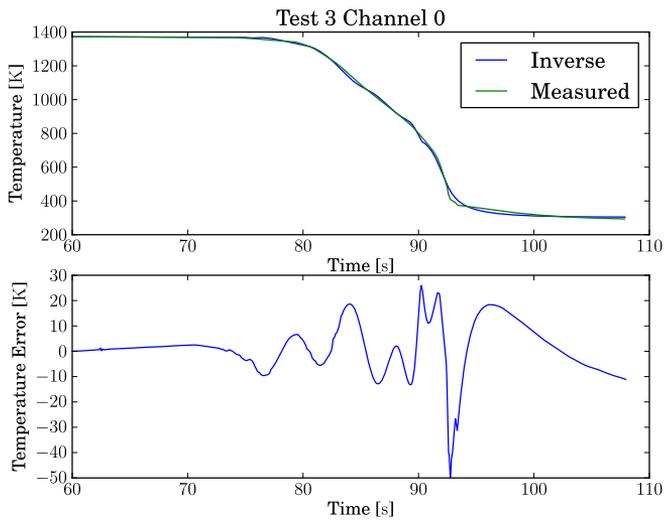


(b) Temperature

Fig 64. Inverse analysis results for Test #2, Channel 1.

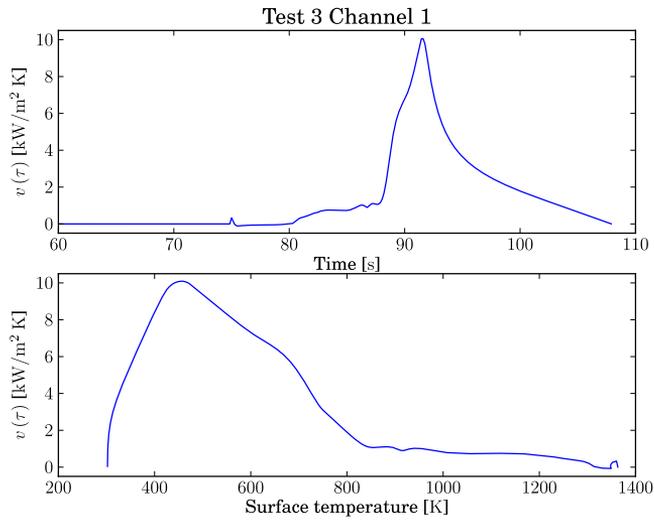


(a) Heat transfer coefficient

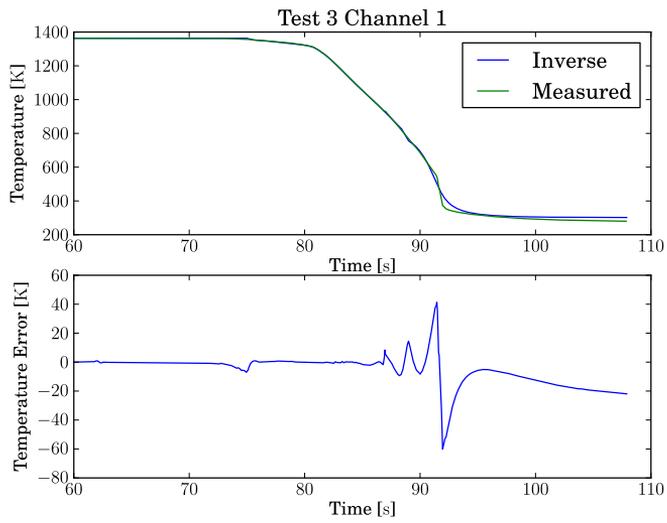


(b) Temperature

Fig 65. Inverse analysis results for Test #3, Channel 0.



(a) Heat transfer coefficient



(b) Temperature

Fig 66. Inverse analysis results for Test #3, Channel 1.

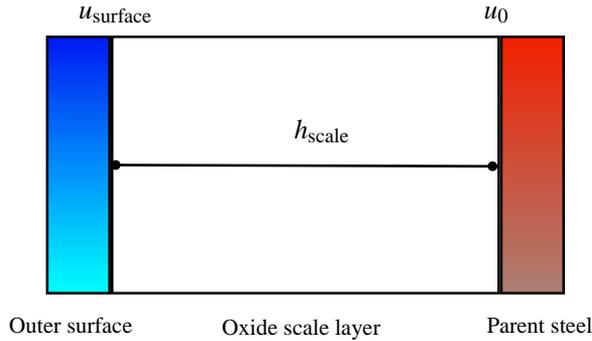


Fig 67. Idealised compact surface oxide scale layer.

As discussed by Wendelstorf *et al* (59), the surface oxide layer on steels at high temperatures can have a pronounced influence on effective heat transfer because of both the thickness of the oxide scale and the low thermal conductivity of the constituent oxides (refer to Appendix 3 for a brief outline of the thermophysical properties of wüstite). In particular, these authors suggest that the presence of oxide scale has the effect of both “broadening” the temperature range over which the peak heat transfer coefficient occurs, as well as reducing the magnitude. A further complicating factor is that the scale itself is, at some unknown point during the process, fractured and removed from the surface of the sample. The same authors’ experimental analysis suggests this phenomena can yield up to $\pm 50\%$ variation in experimentally determined heat transfer coefficients of oxidised carbon steel surfaces purely because of differences in the oxide fracture and removal behaviour during cooling below the Leidenfrost point.

This oxide scale related effect can be illustrated by considering a heat balance across a compact, adherent oxide scale layer covering the surface of the steel. Neglecting capacitance effects and assuming the interface between the oxide layer and parent steel has no thermal resistance. Using the situation depicted in Figure 67 as a model, it follows that

$$\hat{v} [u_{\text{surface}} - u_{\infty}] - \frac{1}{r_{\text{scale}}} [u_0 - u_{\text{surface}}] = 0, \quad (168)$$

where \hat{v} denotes the true surface heat transfer coefficient (as opposed to the effective heat transfer coefficient v which implicitly includes the effects of oxide scale thermal resistance), and r_{scale} is the thermal resistance of the oxide scale layer. Assuming a scale consisting of stoichiometric wüstite, the layer thermal resistance can be calculated

$$r_{\text{scale}} = \frac{h_{\text{wüstite}}}{\lambda_{\text{wüstite}}} \quad (169)$$

with the thermal conductivity of wüstite, $\lambda_{\text{wüstite}}$ calculated using the model proposed by Akiyama *et al* (34) (refer to Appendix 3.3 for full details), and $h_{\text{wüstite}}$ being the scale thickness. When the resistance across the interface between the scale and parent steel is negligible, as should be the case with a compact, adherent scale layer, effective heat transfer coefficient can be calculated as

$$\frac{1}{\hat{v}} = \left[\frac{1}{v} - r_{\text{scale}} \right]. \quad (170)$$

Equations (168) and (170) can be combined to compute the scale surface heat transfer coefficient and temperature from the inverse estimates under the assumption of a given compact scale layer thickness. Because the thermal conductivity of wüstite is also dependent on temperature, we introduce a mean temperature value and thermal conductivity of the oxide scale layer:

$$\begin{aligned} \bar{u}_{\text{wüstite}} &= \frac{1}{2} (u_{\text{surface}} + u_0) \\ &= u_0 + \Delta u \\ \bar{\lambda}_{\text{wüstite}} &= \lambda_{\text{wüstite}} (u_0 + \Delta u) \end{aligned}$$

Using these simplifications, Equation (168) then reformulated as

$$\left[\frac{1}{v} - \frac{h_{\text{wüstite}}}{\bar{\lambda}_{\text{wüstite}}} \right]^{-1} [u_0 - u_{\infty} + 2\Delta u] - 2 \left[\frac{h_{\text{wüstite}}}{\bar{\lambda}_{\text{wüstite}}} \right] \Delta u = 0 \quad (171)$$

and solved for Δu using an iterative method. This then provides the expected oxide surface temperature for a given oxide-steel interface temperature and scale thickness.

Using the solutions for Test #3 channel #1 measurements as an example, Figure 68 shows the estimated effect of the scale layer thickness on the regularized solution for several assumed scale thickness values compared to the original data, calculated using Equation (171). As predicted by Wendelstorf *et al* (59), as the thickness of the scale increases, there is a considerable difference between the solution recovered by the iterative regularisation process without any *a priori* information, and the true heat transfer coefficient. As the scale becomes thicker, the difference between the scale surface temperature and the scale-parent steel interface temperature increases. This has the effect of lowering the effective superheat at the surface compared to that indicated

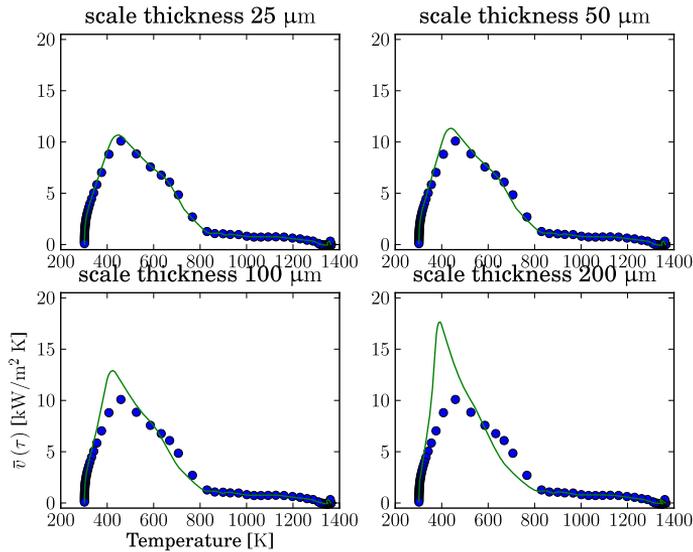


Fig 68. Relationship between true surface heat transfer conditions and those calculated using iterative regularization with no *a priori* information about surface oxide layer thickness. Inverse solution shown with points, estimated true surface heat transfer coefficient shown with lines.

by the inverse solution, which leads to the onset of nucleation boiling and peak heat transfer at a higher inverse indicated temperature than heat transfer theory would predict. Not only does the scale layer have the effect of lowering the effective heat transfer coefficient because of the low thermal conductivity of the scale, it also has the effect of “shifting” the temperature of peak heat transfer to higher values. In the case of descaling a heavily oxidized surface, where the scale is removed at some point during the cooling process, this effect implies that the heat effective heat transfer coefficient may change in unpredictable ways during the transition from a fully adherent scale to a clean, unoxidized surface.

We hypothesise that this effect may go some way to explaining the anomalous result from Test #1. This sample partially jammed when being hand inserted into the holder inside the descaler, and took some seconds to properly seat in the rig before the descaling water was turned on. During this time, the sample was being cooled by the bypass water which is fed through the nozzles to keep them cool and to prevent any air locks or hammer from damaging the high pressure pipework and valves when the descaling water is first introduced into the system. It is possible that the cooling effect

of the bypass water was sufficient to crack or even partially remove the oxide scale before descaling commenced. This would lead to the surface having a lower net thermal resistance and allow far higher rates of heat exchange with the descaling jet. Under such a hypothesis, Test #1 could be viewed as considerably more ill-posed than the other two tests, and this explains the relatively poor fit of the regularised solution obtained in this test compared with the subsequent two test samples. Implicit in this conclusion is that the degree of ill-posedness of the SHE equation for this case is not constant through the cooling process, because the characteristic magnitude and frequency of the true solution is dynamically changing due to the descaling action itself.

Constructing or calibrating a heat transfer boundary condition for this type of situation using the iterative regularized data is, therefore, not a straightforward proposition. Depending on assumptions about the scale removal process, rather different answers can be deduced from the same data. An earlier attempt at this is presented in reference (57), where a considerable amount of existing heat transfer theory was tuned using what was judged to be the better portions of the iterative regularization solution using this measured data and an assumed constant scale thickness. This approach will not be further discussed in the current analysis.

Taking a slightly more pragmatic view point, it would seem acceptable to “replay” the iterative regularized solution as a temperature dependent boundary conditions in numerical simulations in situations for which the experimental conditions are a good approximation. Of course, it is debatable how valid any extrapolation of such results would be to other experimental conditions or industrial scale processes.

3.7 Conclusions

In this chapter, we have considered the problem of reconstructing temperature fields and boundary conditions from temperature data measured by contact thermocouples. The iterative regularisation method described has been demonstrated to be an effective analysis approach for these types of ill-posed inverse problems, exhibiting robustness even with noisy signal, while being relatively straightforward to implement numerically, and computationally inexpensive for the one-dimensional cases studied.

The example computations for measured data taken from hydraulic descaling experiments illustrated many of the practical aspects of inverse analysis of experimental data in thermomechanical treatment processes. This example also highlighted some of the inherent shortcomings that are features of both the physical process, and mathematical

problem itself, even with high quality measured signals from what should be regarded as well controlled experiments.

The heat transfer processes operating in this descaling case were largely transient phenomena whose magnitude and frequency proved to be beyond the domain over which the regularisation process could resolve. As a result, it was not possible to determine the peak heat transfer coefficient associated with fully developed nucleation boiling with any degree of confidence. This is purely related to the ill-posed nature of the problem with respect to high frequency, impulse like boundary phenomena.

The effects of surface oxide scale is another feature which is particular to carbon steel surfaces at hot working temperatures in air, and one which had a direct bearing on the results of the analysis. Even with a robust inverse method and a fairly complete description of the heat conduction and thermophysical behaviour of the steel being cooled, the results obtained were subject to implicit assumptions about the thickness and adherence of the surface oxide scale layer. In a process such as hydraulic descaling, where the cooling and spray impingement pressure are intended to remove surface oxide scale, it becomes necessary to have a sound mechanistic description of the mechanical behaviour of the oxide scale itself.

While the analysis focussed primarily on the interpretation of measured data from contact thermocouples, the iterative regularisation method for the SHE considered has potential application in other areas of thermomechanical treatment simulation. It seems reasonable that the same gradient based optimisation procedure could also be used for cooling schedule design and control, where some target values for the temperature-time history of the steel during austenite decomposition could be substituted for measured data. The solution space would then be a finite dimensional set of parameters for the cooling apparatus which would determine the effective boundary heat flux. In such applications, the use of the adjoint and sensitivity problems should be considerably more efficient than using perturbation of the conduction equation to compute a directional derivative or gradient estimate for using in an separate optimisation procedure.

4 Discussion and final conclusions

This monograph details an analysis of two heat conduction problems which arise in connection with thermomechanical treatment processes of steels. This pair of related problems have a direct bearing on the capacity to accurately predict and control the temperature of hypoeutectoid steels during cooling accompanied by solid state phase transformation, which is a core element of the development of modern, high strength steels.

We have attempted to describe the interaction between heat conduction, thermophysical properties, microstructure evolution and surface phenomena in a consistent fashion, and have demonstrated how the resulting coupled mathematical models can be implemented in a computer code in a relatively straightforward fashion. Their use has been demonstrated through computation of temperature and microstructure evolution during different types of cooling process relevant to both industrial and laboratory scale practice, and through inverse analysis of measured data taken from pilot plant experiments.

The key area of emphasis in this work has been in proposing a workable coupling between microstructure and thermophysical properties. Most of the complexity and non-linearity in heat condition problems involving transforming steel microstructures is connected with thermophysical properties, and it is an area which has probably been oversimplified in many treatments of the subject. The analysis presented in this monograph, relied heavily on the insight provided by Browne (12), particularly relating to the role magnetic latent heat plays in determining the temperature of steel during cooling from an initial austenitic state. We observe that thermodynamic and thermophysical functions for ferrous martensite are more poorly represented in the literature than ferrite and pearlite, and suggest that this might be an area which requires further research as martensitic microstructures become more common in ultra high strength hypoeutectoid steels.

The evolution of microstructure in steels during cooling is complex, and we have relied on empirical rate equations taken from literature sources to formulate the ordinary differential equations for irreversible phase transformation reactions. During the course of this work, we have formed no opinion about the accuracy of these equations, other than to observe that they have been widely used by other authors and a considerable

amount of favourable validation work exists in the literature for each. Coupling of these models with heat conduction to form a robust numerical scheme, however, proved very challenging and we take this opportunity to reiterate our experiences in implementing this type of coupling.

The diffusive transformation theory of Kirkaldy and Venugopalan (32) relies on a sigmoidal power law of the general form

$$\frac{\partial \chi}{\partial \tau} \propto \chi^{\frac{2}{3}(1-\chi)} (1-\chi)^{\frac{2}{3}} \chi, \chi \in [0, 1]$$

to compute the reaction kinetics of the proeutectoid ferrite, pearlite and bainite phase transformations. This equation has very large derivative value magnitudes at small and large volume fraction reacted, and poses considerable difficulties in iterative implicit integration schemes. In the computation examples presented in Chapter 2, we observed that it was this equation which determined integrator step size limits and had considerable negative effects on the overall computation cost of numerical solutions.

The empirical relationship for predicting the volume fraction of retained austenite in quenched steels proposed by Koistinen and Marburger (41)

$$\chi_a = \exp[-\eta (u_{ms} - u)]$$

can be transformed into a rate equation which takes the general form

$$\frac{\partial \chi_m}{\partial \tau} = -\eta \exp[-\eta (u_{ms} - u)] \frac{\partial u}{\partial \tau}.$$

This form also presents difficulties for non-linear implicit schemes because, both because of the requirement for irreversibility and because the transformation rate depends both on cooling rate and temperature. The question of irreversible forms of this model has been addressed in the literature, for example in (33, 16), but the question of interaction between the reaction rate and net heat conduction rate via the source term in the heat conduction equation appears to have received less attention. The solution presented here to this problem involved introducing an extra level of iteration into the method of lines formulation to ensure agreement between the martensite equation rate and the cooling rate during implicit time integration. This solution adds computation expense to the solution and requires additional software complexity.

In the small one- and two-dimensional problems solved in this monograph, the computation time penalties associated with implementation of these phase transformation models were tolerable and presented no barrier to obtaining results in a reasonable time

frame. Simulation of much larger, industrial scale problems, where time scales may be longer and computational grids three dimensional with much higher cell counts, would appear much more problematic. It may be the case that alternative computation methods are going to be necessary to manage the computation cost for larger scale, three dimensional problems.

The measurement of temperature of laboratory and pilot plant scale steel specimens with contact thermocouples plays an important role in the physical simulation of thermomechanical processes as it is practised at the University of Oulu. Interpretation and analysis of the temperature values measured in this way is not necessarily a straightforward task. In this monograph we formulated this problem as an ill-posed sideways heat conduction equation, and proposed an iterative regularisation process based on the family of methods proposed by Alifanov (47, 56) for reconstructing temperature fields in samples and determining boundary conditions from experimental data.

To illustrate the practical use of the iterative regularization with experimental data, we presented a study of temperature data measured during hydraulic descaling test made at the Swerea MEFOS pilot plant in Luleå, Sweden. The hydraulic descaling process involves extremely high rate surface cooling to remove surface oxide scale, and in the stationary tests performed in these experiments, samples were cooled to close to room temperature. Because of this, we included the same irreversible martensite transformation coupling scheme discussed in this monograph, and treat the descaling process as an ultra-fast cooling system.

The resulting analysis showed both the effectiveness of the inverse method at dealing with noisy experimental data, and the shortcomings of the method at resolving such large transient heat fluxes using internal thermocouples. The peak heat flux at around 200 K surface superheat proved to be too transitory for the regularisation scheme to resolve at the 2 mm thermal diffusion distance from the cooled surface to the internal thermocouple junction. This analysis also helped to highlight the uncertainty in cooling behaviour related to the presence of oxide scale layers on the surface of carbon steels at hot working temperatures. The common oxides of iron have a much lower thermal conductivity than the parent steel upon which they form, and this difference in thermal conductivity should be accounted for when performing temperature calculations for oxidised samples. The uncertainty associated with scale behaviour is a further complication in inverse problems, where recovered surface temperatures and boundary conditions may implicitly contain the effects of surface oxide scales. In particular, problems where the surface oxide layer

changes in thickness, either through removal (as was the case in the descaling example), or through the process of further oxidation require some forethought in experimental design and analysis if the results are to be correctly interpreted.

Although not explored in the context of the present work, the iterative regularisation technique would seem to be a good candidate for application in thermomechanical treatment schedule design. A desirable temperature-time history could be substituted for the measured data used in the current work, and a finite dimensional solution containing cooling control parameters obtained which minimised deviation from the target temperature-time history in the least squares sense. We suggest this could be a useful and computationally efficient alternative approach to the usual non-linear optimisation analysis approaches which require numerical computation of directional derivatives or gradients by perturbation.

References

1. Householder AS (1964) The theory of matrices in numerical analysis. Introductions to Higher Mathematics. Blaisdell.
2. Williams JG (2007) Advances in steels for high strength ERW linepipe application in Australia. *Materials Forum - Annual Review Journal of the Institute of Materials Engineering Australia* 31: 1–10.
3. Ouchi C (2001) Development of steel plates by intensive use of TMCP and direct quenching processes. *ISIJ International* 41: 542–553.
4. Deshimaru S, Takahashi K, Endo S, Hasunuma J, Sakata K & Nagahama Y (2004) Steels for production, transportation, and storage of energy. *JFE technical report* 2: 55–67.
5. Asahi H, Hara T, Tsuru E, Morimoto H, Ohkita S, Sugiyama M, Maruyama N, Shinada K, Koyama K, Terada Y, Akasaki H, Ayukawa N, Murata M, Doi N, Miyazaki H & Yoshida T (2004) Development of ultra high strength linepipe, X120. *Nippon Steel Technical Report* 90: 82–90.
6. Hollander F (1970) A model to calculate the complete temperature distribution in steel during hot rolling. *Journal of the Iron and Steel Institute* 208: 46–74.
7. Tomida T, Imai N, Yoshida M & Fukushima S (2007) Effect of ultra-fast cooling after rolling in stable austenite region on grain refinement of C-Mn steel. *Materials Science Forum* 539-543: 4708–4713.
8. Rodrigues PCM, Pereloma EV & Santos DB (2000) Mechanical properties of an HSLA bainitic steel subjected to controlled rolling with accelerated cooling. *Materials Science and Engineering A* 283: 136–143.
9. Suehiro M, Senuma T, Yada H & Sato K (1992) Application of mathematical model for predicting microstructural evolution to high carbon steels. *ISIJ International* 32(3): 433–439.
10. Suehiro M, Senuma T, Oda T & Konishi S (1995) Development of mathematical model for predicting transformation of high carbon steel during cooling on runout table and its application on on-line temperature control of hot strip mill. *Nippon Steel Technical Report* 67: 49–56.
11. Somani S, Porter DA, Pyykönen J & Karjalainen P (2008) Influence of chemical composition and austenite condition on the hardness of direct quenched steels. In: 3rd International conference on thermomechanical processing of steels, pp. 1–13.
12. Browne KM (1998) Modelling the thermophysical properties of iron and steels. In: MFerry (ed.) *Proceedings of Materials '98, The Biennial Conference of the Institute of Materials*, pp. 433–438.
13. Agarwal PK & Brimacombe JK (1981) Mathematical model of heat flow and austenite-pearlite transformation in eutectoid carbon steel rods for wire. *Metallurgical Transactions B* 12: 121–133.
14. Fernandes FM, Denis S & Simon A (1985) Mathematical model coupling phase transformation and temperature evolution during quenching of steels. *Materials Science and Technology* 1: 838–844.
15. Watt DF, Coon L, Bibby M, Goldak J & Henwood C (1988) An algorithm for modelling microstructural development in weld heat affected zones (Part A) reaction kinetics. *Acta Metallurgica* 36(11): 3029–3035.

16. Hömberg D (1996) A numerical simulation of the jominy end-quench test. *Acta Materialia* 44(11): 4375–4385.
17. Carslaw HS & Jaeger JC (1959) *Conduction of heat in solids*. Clarendon Press, Oxford, UK, 2nd edition.
18. Rincon MA, Límaco J & Liu IS (2006) A nonlinear heat equation with temperature-dependent parameters. *Mathematical Physics Electronic Journal* 12: 5.
19. Schiesser WE (1991) *The numerical method of lines: Integration of partial differential equations*. Academic Press (San Diego).
20. Strang WG (1986) *Introduction to applied mathematics*. Wellesley-Cambridge Press, Wellesley, Mass.
21. Alexander RK (1977) Diagonally implicit Runge–Kutta methods for stiff O.D.E.'s. *SIAM Journal on Numerical Analysis* 14(6): 1006–1021.
22. Hairer E, Nørsett S & Wanner G (1993) *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*. Springer Series in Computational Mathematics. Springer Verlag.
23. Butcher JC (2003) *Numerical methods for ordinary differential equations*. John Wiley and Sons, second edition.
24. Alexander RK (1994) Stability of Runge-Kutta methods for stiff ordinary differential equations. *SIAM Journal on Numerical Analysis* 31(4): 1147–1168.
25. Cash JR (1979) Diagonally Implicit Runge-Kutta Formulae with Error Estimates. *IMA Journal of Applied Mathematics* 24(3): 293–301.
26. Browne KM (1995) The thermal conductivity of carbon steels. In: *Recent advances in heat transfer and micro-structure modelling for metal processing*, pp. 105 – 115. ASME.
27. Miettinen J (1997) Calculation of solidification-related thermophysical properties for steels. *Metallurgical and Materials Transactions B* 28B: 281–297.
28. Chase MW (1998) *NIST-JANAF Thermochemical Tables*. Journal of Physical and Chemical Reference Data, Monograph 9. American Institute of Physics, fourth edition.
29. Chuang YY, Schmidt R & Chang YA (1985) Magnetic contributions to the thermodynamic functions of pure Ni, Co, and Fe. *Metallurgical Transactions A* 16A: 153–165.
30. Zener C (1946) Kinetics of the decomposition of Austenite. *Transactions of the Metal Society, AIME* 167: 550–595.
31. Chipman J (1972) Thermodynamics and phase diagram of the Fe-C system. *Metallurgical Transactions A* 3: 55–64.
32. Kirkaldy JS & Venugopalan D (1984) Prediction of microstructure and hardenability in low alloy steels. In: *Marder AR & Goldstein JI (eds.) Phase Transformations in ferrous alloys*, pp. 125–148. TMS-AIME.
33. Brokate M & Sprekels J (1996) *Hysteresis and phase transitions*. Springer Verlag.
34. Akiyama T, Ohta H, Takahashi R, Waseda Y & Yagi J (1992) Measurement and modeling of thermal conductivity for dense iron oxide and porous iron ore agglomerates in stepwise reduction. *ISIJ International* 7: 829–837.
35. Jones E, Oliphant T & Peterson P (2001). *SciPy: Open source scientific tools for Python*. URI: <http://www.scipy.org/>.
36. Lehoucq RB, Sorensen DC & Yang C (1998) *Arpack User's Guide: Solution of Large-Scale Eigenvalue Problems With Implicitly Restarted Arnoldi Methods (Software, Environments, Tools)*. SIAM. URI: <http://www.worldcat.org/isbn/0898714079>.
37. Hodgson PD, Browne KM, Collinson DC, Pham TT & Gibbs RK (1991) A mathematical

- model to simulate the thermomechanical processing of steel. In: Quenching and Carburising, pp. 139–159. International federation for heat treatment and surface engineering, Melbourne, Australia.
38. Kaviany M (2001) Principles of Convective Heat Transfer. Springer Verlag, second edition.
 39. Mitsutsuka M & Fukuda K (1979) Boiling phenomena and effect of water temperature on heat transfer in process of immersoin cooling of a heat steel plate. Transactions ISIJ 19: 162–169.
 40. Dierckx P (1993) Curve and surface fitting with splines. Oxford Univeristy Press.
 41. Koistinen DP & Marburger RE (1959) A general equation prescribing the extent of the austenite-martensite transformation in pure iron-carbon alloys and plain carbon steels. Acta Metallurgica 7: 59–60.
 42. Bhadeshia H (1981) Driving force for martensitic transformation in steels. Metal Science 14(4): 175–177.
 43. Denis S, Sjöström S & Simon A (1987) Coupled temperature, stress, phase transformation calculation. Metallurgical and Materials Transactions A 18: 1203–1212. URI: <http://dx.doi.org/10.1007/BF02647190>. 10.1007/BF02647190.
 44. Eldén L (1995) Numerical solution of the sideways heat equation by difference approximation in time. Inverse Problems 11: 913 – 923.
 45. Berntsson F (1999) A spectral method for solving the sideways heat equation. Inverse problems 15: 891–906.
 46. Tikhonov AN, Goncharky A, Stepanov VV & Yagola AG (1995) Numerical Methods for the Solution of Ill-Posed Problems. Springer Verlag.
 47. Alifanov OM (1994) Inverse Heat Transfer Problems. Springer-Verlag, Berlin.
 48. Shidfar A & Pourgholi R (2005) Application of finite difference method to analysis an ill-posed problem. Applied Mathematics and Computation 168: 1400–1408.
 49. Tikhanov AN & Arsenin VY (1977) Solutions of ill-posed problems. Winston press, Washington DC.
 50. Berntsson F (2001) Numerical methods for solving a non-characteristic cauchy problem for a parabolic equation. Technical Report LiTh-MAT-R-2001-17, Linköping Technical University.
 51. Jarny Y, Özişik MN & Bardon JP (1991) A general optimization method using adjoint equation for solving multidimensional inverse heat conduction. International Journal of Heat and Mass Transfer 34(11): 2911–2919.
 52. Schewchuk JA (1994) An introduction to the conjugate gradient method without agonizing pain. Technical report, Carnegie-Mellon University.
 53. Press WH, Flannery BP, Teukolsky SA & Vetterling WT (1986) Numerical Recipes, volume 1. Cambridge University Press, Cambridge, United Kingdom, 1st edition.
 54. Alekseev AK & Navon IM (2001) The analysis of ill-posed problem using multiscale resolution and second order adjoint techniques. Computer Methods in Applied Mechanics and Engineering 190: 1937–1953.
 55. Alekseev AK & Navon IM (2002) On estimation of temperature uncertainty using the second order adjoint problem. International Journal of Computational Fluid Dynamics 16(2): 113–117.
 56. Alifanov OM, Artyukhin EA & Romyantsev SV (1995) Extreme methods for solving ill-posed problems with applications to inverse heat transfer problems. Begell House, New York.
 57. Stewart B (2010) Control of scale during steel processing. European Commission Directorate-General for Research. Final report of contract RFSR-CT-2005-00016.

58. Wendelstorf J, Spitzer KH & Wendelstorf R (2008) Spray water cooling heat transfer at high temperatures and liquid mass fluxes. *International Journal of Heat and Mass Transfer* 51: 4902–4910.
59. Wendelstorf R, Spitzer KH & Wendelstorf J (2008) Effect of oxide layers on spray water cooling heat transfer at high surface temperature. *International Journal of Heat and Mass Transfer* 51: 4892–4901.
60. Monagan MB, Geddes KO, Heal KM, Labahn G, Vorkoetter SM, McCarron J & DeMarco P (2005) *Maple 10 Programming Guide*. Maplesoft, Waterloo ON, Canada.
61. Hosea M & Shampine L (1996) Analysis and implementation of TR-BDF2. *Applied Numerical Mathematics* 20(20): 21–37.
62. Alexander RK (2003) Design and implementation of DIRK integrators for stiff systems. *Applied Numerical Mathematics* 46: 1–17.
63. Swift G (1960) A comment on matrix inversion by partition. *SIAM Review* 2(2): 132–133.
64. Brankin RW & Gladwell I (1989) Shape-Preserving Local Interpolation for Plotting Solutions of ODEs. *IMA Journal of Numerical Analysis* 9(4): 555–566.
65. Shampine LF (1986) Some practical Runge-Kutta formulas. *Mathematics of Computation* 46(173): 135–150.
66. Whaley RC, Petitet A & Dongarra JJ (2001) Automated empirical optimization of software and the ATLAS project. *Parallel Computing* 27(1-2): 3–35.
67. Davis T (2004) Algorithm 832: UMFPACK V4.3—an unsymmetric-pattern multifrontal method. *ACM Transactions on Mathematical Software* 30(2): 196–199.
68. Boggs PT, Byrd RH & Schnabel RB (1987) A stable and efficient algorithm for nonlinear orthogonal distance regression. *SIAM Journal on Scientific and Statistical Computing* 8: 1052–1078.
69. Lee YK & Lusk MT (1999) Thermodynamic prediction of the eutectoid transformation temperatures of low-alloy steels. *Metallurgical and Materials Transactions A* 30(9): 2325–2330.
70. Lee SJ & Lee YK (2007) Thermodynamic formula for the Acm temperature of low alloy steels. *ISIJ International* 47(5): 769–771.
71. Dormand JR & Prince PJ (1980) A family of embedded Runge-Kutta formulae. *Journal of Computational and Applied Mathematics* 6(1): 19 – 26.

Appendix 1 Finite difference methods

Preface

This section summarises the derivation of a set of second and fourth order finite difference formulas based on the generalized Newton divided difference polynomial

$$p(x)_n = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) + \dots + c_n(x - x_0)(x - x_1)\dots(x - x_{n-1}). \quad (172)$$

One-, two- and three-dimensional schemes for a non-linear heat equation are described, including the formulation of generalized Robin boundary condition. These schemes are presented using a straightforward a matrix-vector notation which emphasises ease of assembly and computation for large grids, and simplicity for computer implementation.

Many of the expressions in this section were calculated using the computer algebra facilities available in the MapleTM software package (60). The L^AT_EX used to typeset them was also directly generated by Maple. While this ensures correctness against the original computer algebra calculations, it also has resulted in a degree of untidiness in the ordering of terms and cancellation of common factors in some fractions which are beyond the control of the author.

1.1 Finite difference operators for central schemes

1.1.1 Second order approximations

The second order Newton divided difference polynomial interpolating a uniformly spaced three point stencil, Figure 69, is given by:

$$p(x) = 1/2 \frac{(u_{i+1} - 2u_i + u_{i-1})x^2}{\delta^2} + 1/2 \frac{(u_{i+1} - u_{i-1})x}{\delta} + u_i \quad (173)$$

with first and second derivatives given by

$$p'(x) = \frac{(u_{i+1} - 2u_i + u_{i-1})x}{\delta^2} + 1/2 \frac{u_{i+1} - u_{i-1}}{\delta} \quad (174)$$

$$p''(x) = \frac{u_{i+1} - 2u_i + u_{i-1}}{\delta^2}. \quad (175)$$

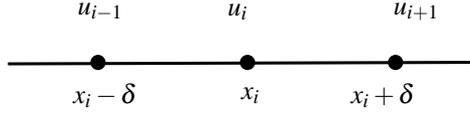


Fig 69. Three point computation stencil on a uniformly spaced grid.

Equation (174) can be evaluated at $x = -\delta$, $x = 0$, and $x = \delta$ to yield three second order formulas for the first derivative:

$$f'(x_{i-1}) = \frac{-u_{i+1} + 4u_i - 3u_{i-1}}{2\delta} \quad (176)$$

$$f'(x_i) = \frac{u_{i+1} - u_{i-1}}{2\delta} \quad (177)$$

$$f'(x_{i+1}) = \frac{3u_{i+1} - 4u_i + u_{i-1}}{2\delta} \quad (178)$$

and Equation (175) a single formula for the second derivative:

$$p''(x) = \frac{u_{i+1} - 2u_i + u_{i-1}}{\delta^2}. \quad (179)$$

For the one dimensional central scheme, the gradient operator ∇u can be written in vector form using Equation (177)

$$\nabla u = f'(x_i) = \frac{1}{2\delta} \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_{i-1} \\ u_i \\ u_{i+1} \end{bmatrix} \quad (180)$$

and the Laplacian $\nabla^2 u$ using Equation (179) as

$$\nabla^2 u = f''(x_i) = \frac{1}{\delta^2} \begin{bmatrix} 1 & -2 & 1 \end{bmatrix} \begin{bmatrix} u_{i-1} \\ u_i \\ u_{i+1} \end{bmatrix}. \quad (181)$$

The one-dimensional stencil can be expanded into a two dimensional molecule, Figure 70, and analogous central gradient and Laplacian operators formed

$$\nabla u = f_x(x_i, y_j) + f_y(x_i, y_j) = \frac{1}{2\delta} \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \left(\begin{bmatrix} u_{i-1, j} \\ u_{i, j} \\ u_{i+1, j} \end{bmatrix} + \begin{bmatrix} u_{i, j-1} \\ u_{i, j} \\ u_{i, j+1} \end{bmatrix} \right) \quad (182)$$

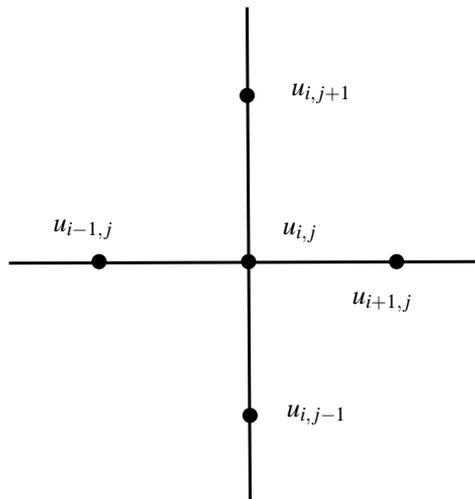


Fig 70. Five point, two-dimensional compact computational molecule on a uniformly spaced grid.

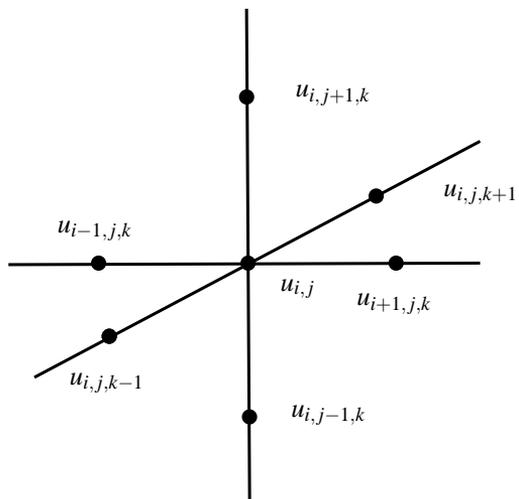


Fig 71. Seven point, three-dimensional compact computational molecule on a uniformly spaced grid.

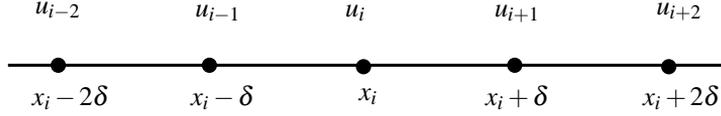


Fig 72. Five point computation stencil on a uniformly spaced grid.

$$\nabla^2 u = f_{xx}(x_i, y_j) + f_{yy}(x_i, y_j) = \frac{1}{\delta^2} \begin{bmatrix} 1 & -2 & 1 \end{bmatrix} \left(\begin{bmatrix} u_{i-1,j} \\ u_{i,j} \\ u_{i+1,j} \end{bmatrix} + \begin{bmatrix} u_{i,j-1} \\ u_{i,j} \\ u_{i,j+1} \end{bmatrix} \right). \quad (183)$$

The same procedure applies to the three dimensional central case, whereby the computation molecule expands to that shown in Figure 71, and the gradient and Laplacian become:

$$\nabla u = f_x(x_i, y_j, z_k) + f_y(x_i, y_j, z_k) + f_z(x_i, y_j, z_k) = \frac{1}{2\delta} \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \left(\begin{bmatrix} u_{i-1,j,k} \\ u_{i,j,k} \\ u_{i+1,j,k} \end{bmatrix} + \begin{bmatrix} u_{i,j-1,k} \\ u_{i,j,k} \\ u_{i,j+1,k} \end{bmatrix} + \begin{bmatrix} u_{i,j,k-1} \\ u_{i,j,k} \\ u_{i,j,k+1} \end{bmatrix} \right) \quad (184)$$

$$\nabla^2 u = f_{xx}(x_i, y_j, z_k) + f_{yy}(x_i, y_j, z_k) + f_{zz}(x_i, y_j, z_k) = \frac{1}{\delta^2} \begin{bmatrix} 1 & -2 & 1 \end{bmatrix} \left(\begin{bmatrix} u_{i-1,j,k} \\ u_{i,j,k} \\ u_{i+1,j,k} \end{bmatrix} + \begin{bmatrix} u_{i,j-1,k} \\ u_{i,j,k} \\ u_{i,j+1,k} \end{bmatrix} + \begin{bmatrix} u_{i,j,k-1} \\ u_{i,j,k} \\ u_{i,j,k+1} \end{bmatrix} \right). \quad (185)$$

1.1.2 Fourth order approximation

The fourth order Newton divided difference polynomial interpolating the uniformly spaced five point stencil shown in Figure 72 is given by:

$$\begin{aligned}
p(x) = & 1/24 \frac{(u_{i+2} - 4u_{i+1} + 6u_i + u_{i-2} - 4u_{i-1})x^4}{\delta^4} + \\
& 1/24 \frac{(4u_{i-1} - 4u_{i+1} - 2u_{i-2} + 2u_{i+2})x^3}{\delta^3} + \\
& 1/24 \frac{(-30u_i - u_{i-2} + 16u_{i-1} + 16u_{i+1} - u_{i+2})x^2}{\delta^2} + \\
& 1/24 \frac{(16u_{i+1} + 2u_{i-2} - 2u_{i+2} - 16u_{i-1})x}{\delta} + u_i \quad (186)
\end{aligned}$$

with first derivate given by

$$\begin{aligned}
p'(x) = & 1/6 \frac{(u_{i+2} - 4u_{i+1} + 6u_i + u_{i-2} - 4u_{i-1})x^3}{\delta^4} + \\
& 1/8 \frac{(4u_{i-1} - 4u_{i+1} - 2u_{i-2} + 2u_{i+2})x^2}{\delta^3} + \\
& 1/12 \frac{(-30u_i - u_{i-2} + 16u_{i-1} + 16u_{i+1} - u_{i+2})x}{\delta^2} + \\
& 1/24 \frac{16u_{i+1} + 2u_{i-2} - 2u_{i+2} - 16u_{i-1}}{\delta} \quad (187)
\end{aligned}$$

and second derivate given by

$$\begin{aligned}
p''(x) = & 1/2 \frac{(u_{i+2} - 4u_{i+1} + 6u_i + u_{i-2} - 4u_{i-1})x^2}{\delta^4} + \\
& 1/4 \frac{(4u_{i-1} - 4u_{i+1} - 2u_{i-2} + 2u_{i+2})x}{\delta^3} + \\
& 1/12 \frac{-30u_i - u_{i-2} + 16u_{i-1} + 16u_{i+1} - u_{i+2}}{\delta^2}. \quad (188)
\end{aligned}$$

Equation (187) can be evaluated at $x = -2\delta$, $x = -\delta$, $x = 0$, $x = \delta$, and $x = 2\delta$ to yield five fourth order formulas for the first derivative:

$$f'(x_{i-2}) = \frac{-3u_{i+2} + 16u_{i+1} - 36u_i - 25u_{i-2} + 48u_{i-1}}{12\delta} \quad (189)$$

$$f'(x_{i-1}) = \frac{u_{i+2} - 6u_{i+1} + 18u_i - 3u_{i-2} - 10u_{i-1}}{12\delta} \quad (190)$$

$$f'(x_i) = \frac{8u_{i+1} + u_{i-2} - u_{i+2} - 8u_{i-1}}{12\delta} \quad (191)$$

$$f'(x_{i+1}) = \frac{3u_{i+2} + 10u_{i+1} - 18u_i - u_{i-2} + 6u_{i-1}}{12\delta} \quad (192)$$

$$f'(x_{i+2}) = \frac{25u_{i+2} - 48u_{i+1} + 36u_i + 3u_{i-2} - 16u_{i-1}}{12\delta} \quad (193)$$

and Equation (188) can be evaluated at $x = -2\delta$, $x = -\delta$, $x = 0$, $x = \delta$, and $x = 2\delta$ to yield five fourth order formulas for the first derivative:

$$f''(x_{i-2}) = \frac{11u_{i+2} - 56u_{i+1} + 114u_i + 35u_{i-2} - 104u_{i-1}}{12\delta^2} \quad (194)$$

$$f''(x_{i-1}) = \frac{-u_{i+2} + 4u_{i+1} + 6u_i + 11u_{i-2} - 20u_{i-1}}{12\delta^2} \quad (195)$$

$$f''(x_i) = \frac{-30u_i - u_{i-2} + 16u_{i-1} + 16u_{i+1} - u_{i+2}}{12\delta^2} \quad (196)$$

$$f''(x_{i+1}) = \frac{11u_{i+2} - 20u_{i+1} + 6u_i - u_{i-2} + 4u_{i-1}}{12\delta^2} \quad (197)$$

$$f''(x_{i+2}) = \frac{35u_{i+2} - 104u_{i+1} + 114u_i + 11u_{i-2} - 56u_{i-1}}{12\delta^2}. \quad (198)$$

For the one dimensional central scheme, the gradient operator ∇u can be written in vector form using Equation (191)

$$\nabla u = f'(x_i) = \frac{1}{12\delta} \begin{bmatrix} 1 & -8 & 0 & 8 & -1 \end{bmatrix} \begin{bmatrix} u_{i-2} \\ u_{i-1} \\ u_i \\ u_{i+1} \\ u_{i+2} \end{bmatrix} \quad (199)$$

and the Laplacian $\nabla^2 u$ using Equation (196) as

$$\nabla^2 u = f''(x_i) = \frac{1}{12\delta^2} \begin{bmatrix} -1 & 16 & -30 & 16 & -1 \end{bmatrix} \begin{bmatrix} u_{i-2} \\ u_{i-1} \\ u_i \\ u_{i+1} \\ u_{i+2} \end{bmatrix}. \quad (200)$$

Following the same procedure used for the second order accurate case, the five point stencil can be extended to a two-dimensional molecule, Figure 73, and the central gradient and Laplacian operators calculated as

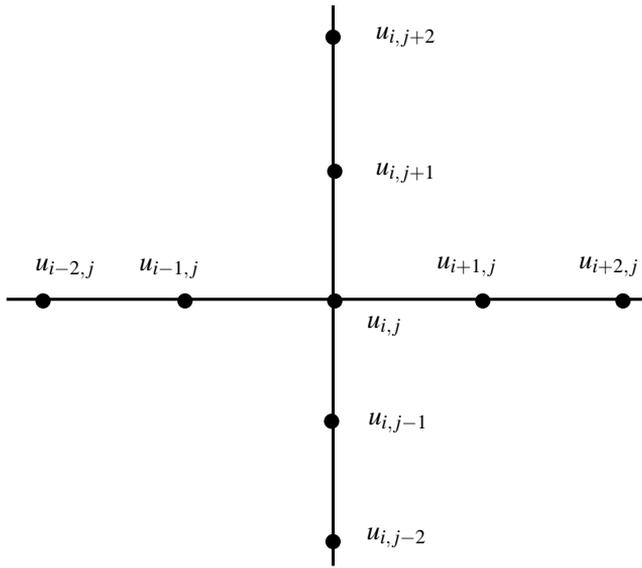


Fig 73. Nine point, two-dimensional computational molecule on a uniformly spaced grid.

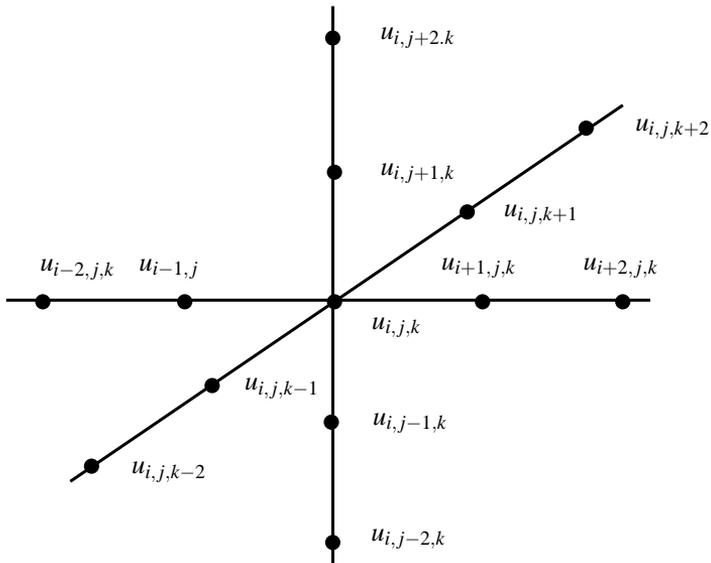


Fig 74. Thirteen point, three-dimensional computational molecule on a uniformly spaced grid.

$$\nabla u = f_x(x_i, y_j) + f_y(x_i, y_j) =$$

$$\frac{1}{12\delta} \begin{bmatrix} 1 & -8 & 0 & 8 & -1 \end{bmatrix} \left(\begin{bmatrix} u_{i-2,j} \\ u_{i-1,j} \\ u_{i,j} \\ u_{i+1,j} \\ u_{i+2,j} \end{bmatrix} + \begin{bmatrix} u_{i,j-2} \\ u_{i,j-1} \\ u_{i,j} \\ u_{i,j+1} \\ u_{i,j+2} \end{bmatrix} \right) \quad (201)$$

$$\nabla^2 u = f_{xx}(x_i, y_j) + f_{yy}(x_i, y_j) =$$

$$\frac{1}{12\delta^2} \begin{bmatrix} -1 & 16 & -30 & 16 & -1 \end{bmatrix} \left(\begin{bmatrix} u_{i-2,j} \\ u_{i-1,j} \\ u_{i,j} \\ u_{i+1,j} \\ u_{i+2,j} \end{bmatrix} + \begin{bmatrix} u_{i,j-2} \\ u_{i,j-1} \\ u_{i,j} \\ u_{i,j+1} \\ u_{i,j+2} \end{bmatrix} \right). \quad (202)$$

The same procedure applies to the three dimensional case, whereby the computation molecule expands to that shown in Figure 74, and the gradient and Laplacian become:

$$\nabla u = f_x(x_i, y_j, z_k) + f_y(x_i, y_j, z_k) + f_z(x_i, y_j, z_k) =$$

$$\frac{1}{12\delta} \begin{bmatrix} 1 & -8 & 0 & 8 & -1 \end{bmatrix} \left(\begin{bmatrix} u_{i-2,j,k} \\ u_{i-1,j,k} \\ u_{i,j,k} \\ u_{i+1,j,k} \\ u_{i+2,j,k} \end{bmatrix} + \begin{bmatrix} u_{i,j-2,k} \\ u_{i,j-1,k} \\ u_{i,j,k} \\ u_{i,j+1,k} \\ u_{i,j+2,k} \end{bmatrix} + \begin{bmatrix} u_{i,j,k-2} \\ u_{i,j,k-1} \\ u_{i,j,k} \\ u_{i,j,k+1} \\ u_{i,j,k+2} \end{bmatrix} \right) \quad (203)$$

$$\nabla^2 u = f_{xx}(x_i, y_j, z_k) + f_{yy}(x_i, y_j, z_k) + f_{zz}(x_i, y_j, z_k) =$$

$$\frac{1}{12\delta^2} \begin{bmatrix} -1 & 16 & -30 & 16 & -1 \end{bmatrix} \left(\begin{bmatrix} u_{i-2,j,k} \\ u_{i-1,j,k} \\ u_{i,j,k} \\ u_{i+1,j,k} \\ u_{i+2,j,k} \end{bmatrix} + \begin{bmatrix} u_{i,j-2,k} \\ u_{i,j-1,k} \\ u_{i,j,k} \\ u_{i,j+1,k} \\ u_{i,j+2,k} \end{bmatrix} + \begin{bmatrix} u_{i,j,k-2} \\ u_{i,j,k-1} \\ u_{i,j,k} \\ u_{i,j,k+1} \\ u_{i,j,k+2} \end{bmatrix} \right). \quad (204)$$

1.2 Spatial discretisation of the non-linear heat equation

The discrete analogue of reference non-linear heat equation used in the present work takes the form

$$\frac{\partial u_i}{\partial t} = \phi_i = \left(\frac{\dot{\lambda}_i}{\gamma_i}\right) (\nabla u_i)^2 + \left(\frac{\lambda_i}{\gamma_i}\right) \nabla^2 u_i + \left(\frac{\psi_i}{\gamma_i}\right) \quad (205)$$

for some discrete spatial point z_i . The gradient and Laplacian operators in the RHS of Equation (205) can be replaced by the finite difference approximations defined in the previous section, so that for the one-dimensional case

$$\phi_i = \left(\frac{\dot{\lambda}_i}{\gamma_i}\right) (\mathbf{q}_i \hat{\mathbf{u}}_i)^T (\mathbf{q}_i \hat{\mathbf{u}}_i) + \left(\frac{\lambda_i}{\gamma_i}\right) \mathbf{k}_i \hat{\mathbf{u}}_i + \sigma_i \quad (206)$$

where \mathbf{q} and \mathbf{k} are row vectors holding coefficients and $\hat{\mathbf{u}}_i$ is the column vector holding the computational stencil or molecule for a given accuracy order about u_i . A convenient simplification of the gradient expression can be obtained by substituting $\mathbf{Q}_i = \mathbf{q}_i^T \mathbf{q}_i$, leading to a standard quadratic form:

$$\phi_i = \left(\frac{\dot{\lambda}_i}{\gamma_i}\right) \hat{\mathbf{u}}_i^T \mathbf{Q}_i \hat{\mathbf{u}}_i + \left(\frac{\lambda_i}{\gamma_i}\right) \mathbf{k}_0 \hat{\mathbf{u}}_i + \sigma_i. \quad (207)$$

For the second order case, \mathbf{k} has only one set of valid coefficients, while \mathbf{Q} has three possible values – a central approximation \mathbf{Q}_0 , a left biased case \mathbf{Q}_{-1} , and a right biased case \mathbf{Q}_1 :

$$\delta^2 \mathbf{k}_0 = \begin{bmatrix} 1 & -2 & 1 \end{bmatrix} \quad (208)$$

$$4\delta^2 \mathbf{Q}_0 = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix} \quad (209)$$

$$4\delta^2 \mathbf{Q}_{-1} = \begin{bmatrix} 9 & -12 & 3 \\ -12 & 16 & -4 \\ 3 & -4 & 1 \end{bmatrix} \quad (210)$$

$$4\delta^2 \mathbf{Q}_1 = \begin{bmatrix} 1 & -4 & 3 \\ -4 & 16 & -12 \\ 3 & -12 & 9 \end{bmatrix}. \quad (211)$$

For the fourth order case, the stencil length expands from three to five entries, and a larger number of coefficient matrices are required to cover all possible stencil combinations. For the Laplacian, we consider three possible stencils: a fully centred stencil, and “left” and “right” biased central stencils, given by

$$12\delta^2\mathbf{k}_0 = \begin{bmatrix} -1 & 16 & -30 & 16 & -1 \end{bmatrix} \quad (212)$$

$$12\delta^2\mathbf{k}_{-1} = \begin{bmatrix} 11 & -20 & 6 & 4 & -1 \end{bmatrix} \quad (213)$$

$$12\delta^2\mathbf{k}_1 = \begin{bmatrix} -1 & 4 & 6 & -20 & 11 \end{bmatrix}. \quad (214)$$

We note that there are additional left and right biased Laplacian approximations for the fourth order scheme, but these are not used in the present work. The gradient coefficient matrix \mathbf{Q} will takes the value of \mathbf{Q}_0 , Equation (215), for the central scheme, or one of two left biased version, Equations (216) and (217), or two right biased versions, Equations (218) and (219).

$$144\delta^2\mathbf{Q}_0 = \begin{bmatrix} 1 & -8 & 0 & 8 & -1 \\ -8 & 64 & 0 & -64 & 8 \\ 0 & 0 & 0 & 0 & 0 \\ 8 & -64 & 0 & 64 & -8 \\ -1 & 8 & 0 & -8 & 1 \end{bmatrix} \quad (215)$$

$$144\delta^2\mathbf{Q}_{-2} = \begin{bmatrix} 9 & -48 & 108 & -144 & 75 \\ -48 & 256 & -576 & 768 & -400 \\ 108 & -576 & 1296 & -1728 & 900 \\ -144 & 768 & -1728 & 2304 & -1200 \\ 75 & -400 & 900 & -1200 & 625 \end{bmatrix} \quad (216)$$

$$144\delta^2\mathbf{Q}_{-1} = \begin{bmatrix} 1 & -6 & 18 & -10 & -3 \\ -6 & 36 & -108 & 60 & 18 \\ 18 & -108 & 324 & -180 & -54 \\ -10 & 60 & -180 & 100 & 30 \\ -3 & 18 & -54 & 30 & 9 \end{bmatrix} \quad (217)$$

$$144\delta^2\mathbf{Q}_1 = \begin{bmatrix} 9 & 30 & -54 & 18 & -3 \\ 30 & 100 & -180 & 60 & -10 \\ -54 & -180 & 324 & -108 & 18 \\ 18 & 60 & -108 & 36 & -6 \\ -3 & -10 & 18 & -6 & 1 \end{bmatrix} \quad (218)$$

$$144\delta^2\mathbf{Q}_2 = \begin{bmatrix} 625 & -1200 & 900 & -400 & 75 \\ -1200 & 2304 & -1728 & 768 & -144 \\ 900 & -1728 & 1296 & -576 & 108 \\ -400 & 768 & -576 & 256 & -48 \\ 75 & -144 & 108 & -48 & 9 \end{bmatrix}. \quad (219)$$

In the two dimensional case, a similar procedure is used, yielding

$$\begin{aligned} \phi_{ij} = & \left(\frac{\dot{\lambda}_i}{\gamma_i}\right) \left(\hat{\mathbf{u}}_i^T \mathbf{Q}_{ii} + \hat{\mathbf{u}}_j^T \mathbf{Q}_{ji}\right) \hat{\mathbf{u}}_i + \left(\frac{\dot{\lambda}_i}{\gamma_i}\right) \left(\hat{\mathbf{u}}_i^T \mathbf{Q}_{ij} + \hat{\mathbf{u}}_j^T \mathbf{Q}_{jj}\right) \hat{\mathbf{u}}_j \\ & + \left(\frac{\lambda_i}{\gamma_i}\right) \mathbf{k}_i \hat{\mathbf{u}}_i + \left(\frac{\lambda_i}{\gamma_i}\right) \mathbf{k}_j \hat{\mathbf{u}}_j + \sigma_{ij}. \end{aligned} \quad (220)$$

In this case, the gradient coefficient matrix \mathbf{Q}_{ij} is defined as the product of the individual direction stencils \mathbf{q}_i and \mathbf{q}_j . For the second order case, this leads to nine possible 3×3 coefficient matrices, and twenty five possible 5×5 matrices for the fourth order case. For the fully centred scheme, this greatly simplifies to

$$\phi_{ij} = \left(\frac{\dot{\lambda}_i}{\gamma_i}\right) (\hat{\mathbf{u}}_i + \hat{\mathbf{u}}_j)^T \mathbf{Q}_0 (\hat{\mathbf{u}}_i + \hat{\mathbf{u}}_j) + \left(\frac{\lambda_i}{\gamma_i}\right) \mathbf{k}_0 (\hat{\mathbf{u}}_i + \hat{\mathbf{u}}_j) + \sigma_{ij} \quad (221)$$

with \mathbf{k}_0 and \mathbf{Q}_{ii} taking the identical values as in the one dimensional case. The three-dimensional case is a natural extension of the one- and two-dimensional versions:

$$\begin{aligned} \phi_{ijk} = & \left(\frac{\dot{\lambda}_{ijk}}{\gamma_{ijk}} \right) (\hat{\mathbf{u}}_i^T \mathbf{Q}_{ii} + \hat{\mathbf{u}}_j^T \mathbf{Q}_{ji} + \hat{\mathbf{u}}_k^T \mathbf{Q}_{ki}) \hat{\mathbf{u}}_i + \\ & \left(\frac{\dot{\lambda}_{ijk}}{\gamma_{ijk}} \right) (\hat{\mathbf{u}}_i^T \mathbf{Q}_{ij} + \hat{\mathbf{u}}_j^T \mathbf{Q}_{jj} + \hat{\mathbf{u}}_k^T \mathbf{Q}_{kj}) \hat{\mathbf{u}}_i + \\ & \left(\frac{\dot{\lambda}_{ijk}}{\gamma_{ijk}} \right) (\hat{\mathbf{u}}_i^T \mathbf{Q}_{ik} + \hat{\mathbf{u}}_j^T \mathbf{Q}_{jk} + \hat{\mathbf{u}}_k^T \mathbf{Q}_{kk}) \hat{\mathbf{u}}_i + \\ & \left(\frac{\lambda_{ijk}}{\gamma_{ijk}} \right) \mathbf{k}_i \hat{\mathbf{u}}_i + \left(\frac{\lambda_{ijk}}{\gamma_{ijk}} \right) \mathbf{k}_j \hat{\mathbf{u}}_j + \left(\frac{\lambda_{ijk}}{\gamma_{ijk}} \right) \mathbf{u}_k \hat{\mathbf{u}}_k + \sigma_{ijk}. \end{aligned} \quad (222)$$

Again, for the central scheme, this simplifies to

$$\begin{aligned} \phi_{ijk} = & \left(\frac{\dot{\lambda}_{ijk}}{\gamma_{ijk}} \right) (\hat{\mathbf{u}}_i + \hat{\mathbf{u}}_j + \hat{\mathbf{u}}_k)^T \mathbf{Q}_0 (\hat{\mathbf{u}}_i + \hat{\mathbf{u}}_j + \hat{\mathbf{u}}_j) \\ & + \left(\frac{\lambda_{ijk}}{\gamma_{ijk}} \right) \mathbf{k}_0 (\hat{\mathbf{u}}_i + \hat{\mathbf{u}}_j + \hat{\mathbf{u}}_j) + \sigma_{ijk}. \end{aligned} \quad (223)$$

1.3 Numerical approximation of boundary conditions

The general discrete Robin boundary condition

$$\alpha_i u_i + (1 - \alpha_i) \frac{\partial u_i}{\partial \mathbf{n}} = \beta_i \quad (224)$$

is presumed to operate at all boundaries on the domain. In the cell centred formulation considered here, the interpolant used to construct the scheme operates at the centroid of the cell, while the boundary condition describes physical or numerical conditions which need to be imposed at the cell boundary. Therefore, in order to satisfy the boundary conditions, the strategy is to introduce fictitious ‘‘ghost-cells’’ at the boundaries of the physical domain, whose values are calculated to enforce the boundary condition at the edge of the true boundary cell. The boundary condition is then imposed by forming a conventional interpolating polynomial over the stencil including the ghost cell, and then the same finite difference discretisation applied to interior cells.

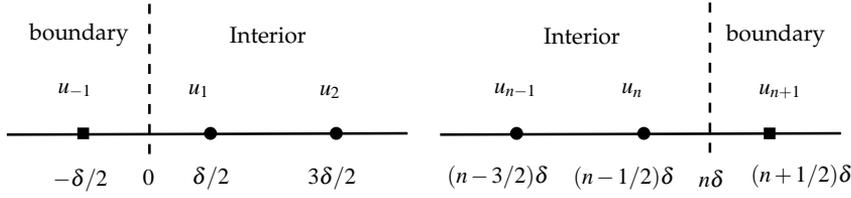


Fig 75. Three point, one dimensional “left” and “right” boundary stencils on a cell centred grid.

1.3.1 Robin condition for the second order scheme

Starting with the simplest one-dimensional case, this concept is illustrated in Figure 75. A ghost cell is added as a neighbour to each boundary cell, with the true boundary lying at the midpoint between the boundary node and its neighbouring ghost node. In this scenario, the one dimensional Robin boundary condition shown in Equation (224) resolves to two possible cases, the “left” boundary with $\partial u / \partial \mathbf{n} = -\partial u / \partial x$:

$$\alpha_0 u(0) - (1 - \alpha_0) u_x(0) = \beta_0 \quad (225)$$

and the “right” boundary with $\partial u / \partial \mathbf{n} = \partial u / \partial x$:

$$\alpha_n u(n\delta) + (1 - \alpha_n) u_x(n\delta) = \beta_n. \quad (226)$$

Using Equation (225) as an example, the second order Newton polynomial through $\{u_{-1}, u_1, u_2\}$ is given by

$$p(x) = 1/8 \frac{(4x^2 - \delta^2) u_2}{\delta^2} + 1/8 \frac{(-8x^2 + 8\delta x + 6\delta^2) u_1}{\delta^2} + 1/8 \frac{(-8\delta x + 4x^2 + 3\delta^2) u_{-1}}{\delta^2} \quad (227)$$

with first derivative given by

$$p'(x) = \frac{u_2 x}{\delta^2} - \frac{(2x - h) u_1}{\delta^2} - \frac{(\delta - x) u_{-1}}{\delta^2}. \quad (228)$$

We remark that u_{-1} is a ghost value whose value must be determined such that the Laplacian calculated on the same stencil will satisfy the boundary condition. Equations (227) and (228) can be substituted into (225), yielding an equation containing the unknown ghost cell value

$$\alpha_0 (3/4 u_1 - 1/8 u_2 + 3/8 u_{-1}) - \frac{(1 - \alpha_0)(u_1 - u_{-1})}{\delta} = \beta_0. \quad (229)$$

This can be solved for the unknown ghost cell value:

$$u_{-1} = -\frac{(-8 + 8\alpha_0 + 6\alpha_0\delta)u_1}{3\alpha_0\delta + 8 - 8\alpha_0} + \frac{\alpha_0\delta u_2}{3\alpha_0\delta + 8 - 8\alpha_0} + 8\frac{\beta_0\delta}{3\alpha_0\delta + 8 - 8\alpha_0}. \quad (230)$$

In explicit schemes, the computational domain is usually expanded to include ghost cell values, so that the same differencing stencils and coefficients can be used for all cells with negligible increase in total computational cost. In the case of implicit schemes, increasing the size of the computational domain typically incurs a computational cost penalty because most implicit schemes employ some form of linear algebra solver. As an alternative, the expressions for the ghost cell values can be substituted into the difference equations, eliminating the ghost cells from the assembled system of equations. For the second-order central scheme, the Laplacian at the boundary node is given by

$$\nabla^2 u = u_{xx}(x_1) = \frac{u_{-1} - 2u_1 + u_2}{\delta^2} \quad (231)$$

and the ghost value can be substituted to eliminate u_{-1} , yielding

$$\nabla^2 u = u_{xx}(x_1) = -4\frac{(2 - 2\alpha_0 + 3\alpha_0\delta)u_1}{(3\alpha_0\delta + 8 - 8\alpha_0)\delta^2} - 4\frac{(-\alpha_0\delta - 2 + 2\alpha_0)u_2}{(3\alpha_0\delta + 8 - 8\alpha_0)\delta^2} + 8\frac{\beta_0}{\delta(3\alpha_0\delta + 8 - 8\alpha_0)}. \quad (232)$$

In the matrix-vector notation introduced in earlier sections, this result becomes

$$\delta^2 \nabla^2 u = \begin{bmatrix} \frac{-9\alpha_0\delta}{\eta_0} - 1 & \frac{\alpha_0\delta}{\eta_0} + 1 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} + \frac{8\beta_0\delta}{\eta_0} \quad (233)$$

with

$$\eta_0 = 3\alpha_0\delta - 8\alpha_0 + 8. \quad (234)$$

For the common Dirichlet condition $\partial u/\partial x = 0$, $\alpha_0 = 0$ and $\beta_0 = 0$, which simplifies the boundary condition to

$$\delta^2 \nabla^2 u = \begin{bmatrix} -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}. \quad (235)$$

This is equivalent to substituting $u_{-1} = u_1$ into the Laplacian, a condition which can also be deduced from the second order gradient approximation at the boundary:

$$f'(0) = \frac{1}{\delta} (u_1 - u_{-1}). \quad (236)$$

The same procedure for the “right” condition defined by Equation (226) yield the following expression for the ghost value

$$u_{n+1} = \frac{\alpha_n \delta u_{n-1}}{3 \alpha_n \delta + 8 - 8 \alpha_n} - \frac{(6 \alpha_n \delta - 8 + 8 \alpha_n) u_n}{3 \alpha_n \delta + 8 - 8 \alpha_n} + 8 \frac{\beta_n \delta}{3 \alpha_n \delta + 8 - 8 \alpha_n} \quad (237)$$

and substitution into the Laplacian gives

$$\nabla^2 u = u_{xx}(x_n) = -4 \frac{(-\alpha_n \delta - 2 + 2 \alpha_n) u_{n-1}}{\delta^2 (3 \alpha_n \delta + 8 - 8 \alpha_n)} - 4 \frac{(3 \alpha_n \delta + 2 - 2 \alpha_n) u_n}{\delta^2 (3 \alpha_n \delta + 8 - 8 \alpha_n)} + 8 \frac{\beta_n}{\delta (3 \alpha_n \delta + 8 - 8 \alpha_n)} \quad (238)$$

or in matrix-vector form

$$\delta^2 \nabla^2 u = \begin{bmatrix} 0 & \frac{\alpha_n \delta}{\eta_n} + 1 & -\frac{9 \alpha_n \delta}{\eta_n} - 1 \end{bmatrix} \begin{bmatrix} u_{n-2} \\ u_{n-1} \\ u_n \end{bmatrix} + \frac{8 \beta_n \delta}{\eta_n} \quad (239)$$

with

$$\eta_n = 3 \alpha_n \delta - 8 \alpha_n + 8. \quad (240)$$

For the Dirichlet condition, the “right” approximation simplifies to

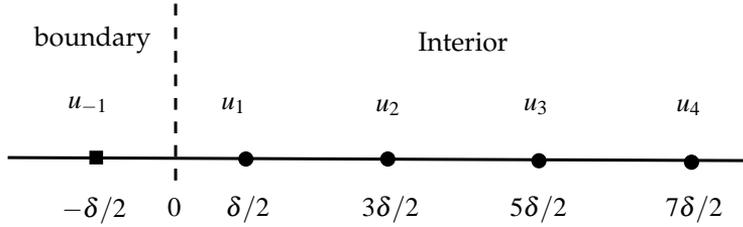


Fig 76. Five point, one dimensional “left” boundary stencil on a cell centred grid.

$$\delta^2 \nabla^2 u = \begin{bmatrix} 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} u_{n-2} \\ u_{n-1} \\ u_n \end{bmatrix}. \quad (241)$$

For non-linear Robin conditions where either of the coefficients α and β may be temperature dependent, it is necessary to compute these coefficients using the temperature at the cell boundary. For the second order scheme, the following interpolants are used:

$$u_0 = \frac{15u_1 - 10u_2 + 3u_3}{8} \quad (242)$$

$$u_{n\delta} = \frac{3u_{n-2} - 10u_{n-1} + 15u_n}{8}. \quad (243)$$

1.3.2 Robin condition for the fourth order scheme

For the fourth order case, the procedure is similar to that just illustrated for the second order case. We define a third order accurate interpolating polynomial through the stencil $\{u_{-1}, u_1, u_2, u_3\}$, shown in Figure 76, and given by

$$\begin{aligned} p(x) = & 1/48 \frac{(-24u_2 + 24u_1 - 8u_{-1} + 8u_3)x^3}{\delta^3} \\ & + 1/48 \frac{(36u_{-1}\delta + 60u_2\delta - 84u_1\delta - 12u_3\delta)x^2}{\delta^3} \\ & + 1/48 \frac{(-46u_{-1}\delta^2 + 6u_2\delta^2 + 42u_1\delta^2 - 2u_3\delta^2)x}{\delta^3} \\ & + 1/48 \frac{15u_{-1}\delta^3 + 3u_3\delta^3 - 15u_2\delta^3 + 45u_1\delta^3}{\delta^3} \end{aligned} \quad (244)$$

with a first derivative:

$$\begin{aligned}
 p'(x) = & 1/16 \frac{(-24u_2 + 24u_1 - 8u_{-1} + 8u_3)x^2}{\delta^3} \\
 & + 1/24 \frac{(36u_{-1}\delta + 60u_2\delta - 84u_1\delta - 12u_3\delta)x}{\delta^3} \\
 & + 1/48 \frac{-46u_{-1}\delta^2 + 6u_2\delta^2 + 42u_1\delta^2 - 2u_3\delta^2}{\delta^3}. \quad (245)
 \end{aligned}$$

For the “left” boundary case, the interpolant and first derivative can be evaluated at the cell boundary, $x = 0$, yielding

$$u(0) = \frac{5}{16}u_{-1} + 1/16u_3 - \frac{5}{16}u_2 + \frac{15}{16}u_1 \quad (246)$$

$$u'(0) = 1/24 \frac{-23u_{-1} + 3u_2 + 21u_1 - u_3}{\delta} \quad (247)$$

and these can be substituted into the boundary condition, Equation (225), to form an equation containing the unknown ghost cell value u_{-1} :

$$\begin{aligned}
 \alpha_0 \left(\frac{5}{16}u_{-1} + 1/16u_3 - \frac{5}{16}u_2 + \frac{15}{16}u_1 \right) \\
 - 1/24 \frac{(1 - \alpha_0)(-23u_{-1} + 3u_2 + 21u_1 - u_3)}{\delta} = \beta_0. \quad (248)
 \end{aligned}$$

which leads to a self contained expression for the ghost value in terms of interior temperatures and Robin condition coefficients:

$$\begin{aligned}
 u_{-1} = & - \frac{(45\alpha_0\delta + 42\alpha_0 - 42)u_1}{15\alpha_0\delta + 46 - 46\alpha_0} - \frac{(-6 - 15\alpha_0\delta + 6\alpha_0)u_2}{15\alpha_0\delta + 46 - 46\alpha_0} \\
 & - \frac{(3\alpha_0\delta + 2 - 2\alpha_0)u_3}{15\alpha_0\delta + 46 - 46\alpha_0} + 48 \frac{\beta_0\delta}{15\alpha_0\delta + 46 - 46\alpha_0} \quad (249)
 \end{aligned}$$

For the fourth order scheme, we use the “centre-left” biased approximation of the Laplacian defined on the stencil $\{x_{-1}, x_1, x_2, x_3, x_4\}$, shown in Figure 76 and given by

$$u_{xx}(x_1) = -1/12 \frac{u_4 - 4u_3 - 6u_2 - 11u_{-1} + 20u_1}{\delta^2}. \quad (250)$$

The ghost value in the Laplacian is eliminated by substituting Equation (249), leading to the final fourth order Laplacian which satisfies the Robin boundary condition:

$$\begin{aligned}
12\delta^2\nabla^2u = & \left(-11\frac{45\alpha_0\delta+42\alpha_0-42}{15\alpha_0\delta+46-46\alpha_0}-20\right)u_1 \\
& + \left(6-11\frac{-6-15\alpha_0\delta+6\alpha_0}{15\alpha_0\delta+46-46\alpha_0}\right)u_2 + \left(4-11\frac{3\alpha\delta+2-2\alpha_0}{15\alpha_0\delta+46-46\alpha_0}\right)u_3 \\
& - u_4 + 528\frac{\beta\delta_0}{15\alpha_0\delta+46-46\alpha_0} \quad (251)
\end{aligned}$$

or in matrix-vector form:

$$\begin{aligned}
12\delta^2\nabla^2u = & \begin{bmatrix} -20-11\frac{45\alpha_0\delta+42\alpha_0-42}{15\alpha_0\delta+46-46\alpha_0} \\ 6-11\frac{-6-15\alpha_0\delta+6\alpha_0}{15\alpha_0\delta+46-46\alpha_0} \\ 4-11\frac{3\alpha_0\delta+2-2\alpha_0}{15\alpha_0\delta+46-46\alpha_0} \\ -1 \\ 0 \end{bmatrix}^T \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix} \\
& + \frac{528\beta_0\delta}{15\alpha_0\delta+46-46\alpha_0}. \quad (252)
\end{aligned}$$

The equivalent Dirichlet condition for this case corresponds to $\alpha_0 = 0$ and $\beta_0 = 0$, and is given by

$$12\delta^2\nabla^2u = \begin{bmatrix} -229/23 & 171/23 & 81/23 & -1 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix}. \quad (253)$$

We remark that this condition is only third order accurate, with larger truncation order accuracy than a comparable approximate Dirichlet condition made using a fourth order gradient estimate.

For the ‘‘right’’ boundary case, a similar procedure yields a ghost cell value of

$$\begin{aligned}
u_{n+1} = & -\frac{(-2\alpha_n+2+3\alpha_n\delta)u_{n-2}}{15\alpha_n\delta+46-46\alpha_n} - \frac{(6\alpha_n-6-15\alpha_n\delta)u_{n-1}}{15\alpha_n\delta+46-46\alpha_n} \\
& - \frac{(42\alpha_n-42+45\alpha_n\delta)u_n}{15\alpha_n\delta+46-46\alpha_n} + 48\frac{\beta_n\delta}{15\alpha_n\delta+46-46\alpha_n}. \quad (254)
\end{aligned}$$

The “centre-right” biased approximation of the Laplacian defined on the stencil $\{x_{n-3}, x_{n-2}, x_{n-1}, x_n, x_{n+1}\}$, is calculated as

$$u_{xx}(x_n) = -1/12 \frac{u_{n-3} - 4u_{n-2} - 6u_{n-1} + 20u_n - 11u_{n+1}}{\delta^2} \quad (255)$$

and the resulting Laplacian include the boundary condition is given by

$$\begin{aligned} 12\delta^2\nabla^2 u = & -u_{n-3} + \left(-11 \frac{-2\alpha_n + 2 + 3\alpha_n\delta}{15\alpha_n\delta + 46 - 46\alpha_n} + 4\right) u_{n-2} \\ & + \left(6 - 11 \frac{6\alpha_n - 6 - 15\alpha_n\delta}{15\alpha_n\delta + 46 - 46\alpha_n}\right) u_{n-1} + \left(-20 - 11 \frac{42\alpha_n - 42 + 45\alpha_n\delta}{15\alpha_n\delta + 46 - 46\alpha_n}\right) u_n \\ & + 528 \frac{\beta_n\delta}{15\alpha_n\delta + 46 - 46\alpha_n} \end{aligned} \quad (256)$$

$$12\delta^2\nabla^2 u = \begin{bmatrix} & & & & 0 \\ & & & & -1 \\ & 4 - 11 \frac{3\alpha_0\delta + 2 - 2\alpha_0}{15\alpha_0\delta + 46 - 46\alpha_0} & & & \\ & 6 - 11 \frac{-6 - 15\alpha_0\delta + 6\alpha_0}{15\alpha_0\delta + 46 - 46\alpha_0} & & & \\ -20 - 11 \frac{45\alpha_0\delta + 42\alpha_0 - 42}{15\alpha_0\delta + 46 - 46\alpha_0} & & & & \end{bmatrix}^T \begin{bmatrix} u_{n-4} \\ u_{n-3} \\ u_{n-2} \\ u_{n-1} \\ u_n \end{bmatrix} + \frac{528\beta_0\delta}{15\alpha_0\delta + 46 - 46\alpha_0}. \quad (257)$$

The equivalent Dirichlet condition is taken as

$$12\delta^2\nabla^2 u = \begin{bmatrix} 0 & -1 & 171/23 & 81/23 & -229/23 \end{bmatrix} \begin{bmatrix} u_{n-4} \\ u_{n-3} \\ u_{n-2} \\ u_{n-1} \\ u_n \end{bmatrix}. \quad (258)$$

For non-linear Robin conditions where either of the coefficients α and β may be temperature dependent, it is necessary to compute these coefficients using the temperature at the cell boundary. For the fourth order scheme, a third order interpolant using the same stencil as the boundary condition formulation is used:

$$u_0 = \frac{35u_1 - 35u_2 + 21u_3 - 5u_4}{16} \quad (259)$$

$$u_{n\delta} = \frac{-5u_{n-3} + 21u_{n-2} - 35u_{n-1} + 35u_n}{16}. \quad (260)$$

1.4 Assembly of systems of equations for large grids

The generalised ODE for the one-dimensional non-linear heat equation is assumed to take the form

$$\frac{\partial \mathbf{u}}{\partial t} = f(\mathbf{u}) = f \left(\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{l-1} \\ u_l \end{bmatrix} \right) = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_{l-1} \\ \phi_l \end{bmatrix} \quad (261)$$

with each ϕ_i value computed using a difference formulation of the form

$$\begin{aligned} \phi_i &= \left[\left(\frac{\dot{\lambda}_i}{\gamma_i} \right) \hat{\mathbf{u}}_i^T \mathbf{Q}_i + \left(\frac{\lambda_i}{\gamma_i} \right) \mathbf{k}_i \right] \hat{\mathbf{u}}_i + \frac{\sigma_i}{\gamma_i} \\ &= \mathbf{a}_i \hat{\mathbf{u}}_i + \psi_i. \end{aligned}$$

Each of these discrete approximations can be assembled into a system of linear equations

$$f(\mathbf{u}) = \begin{bmatrix} \mathbf{a}_1 & & \dots & & 0 \\ & \ddots & & & \vdots \\ & & \mathbf{a}_i & & \\ \vdots & & & \ddots & \\ 0 & \dots & & & \mathbf{a}_l \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_i \\ \vdots \\ u_l \end{bmatrix} + \begin{bmatrix} \psi_1 \\ \vdots \\ \psi_i \\ \vdots \\ \psi_l \end{bmatrix}, \quad (262)$$

where suitable zero padding is implied in each row of the system matrix. The bandwidth of the resulting sparse matrix is either three in the case of the second order scheme, or five in the case of the fourth order scheme. For higher dimensional approximations, the system remains sparse and banded, with additional bands in the lower and upper triangular halves of the matrix. Typical matrix structures for the one-, two- and three-dimensional cases are illustrated in Figures 77 and 78.

At first inspection, the matrix notation used to describe spatial discretisation of the non-linear heat conduction equation and boundary conditions may seem somewhat

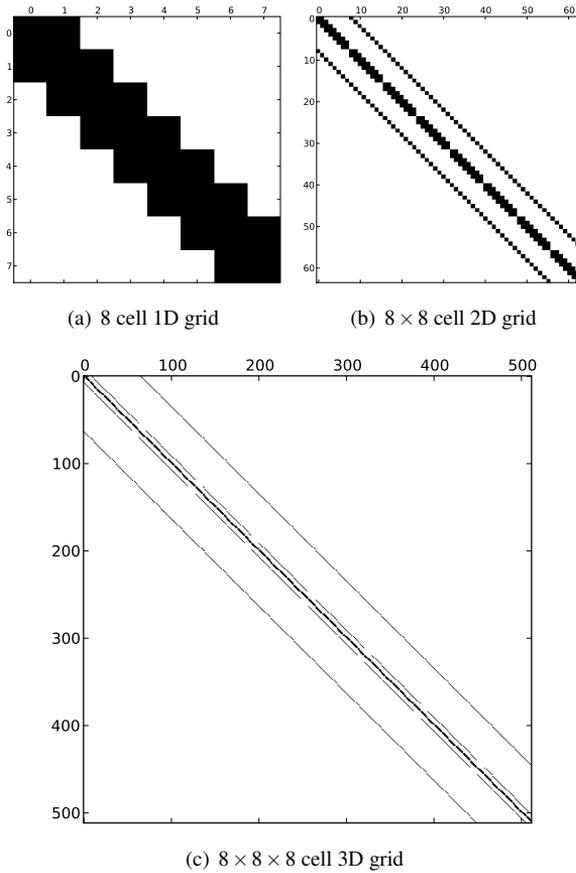
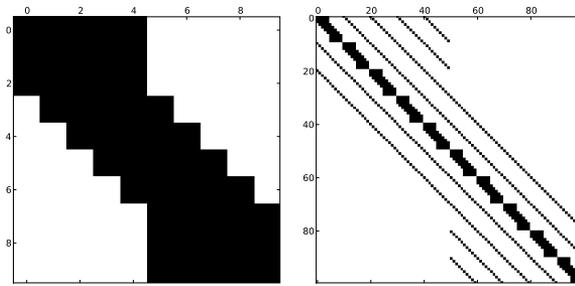


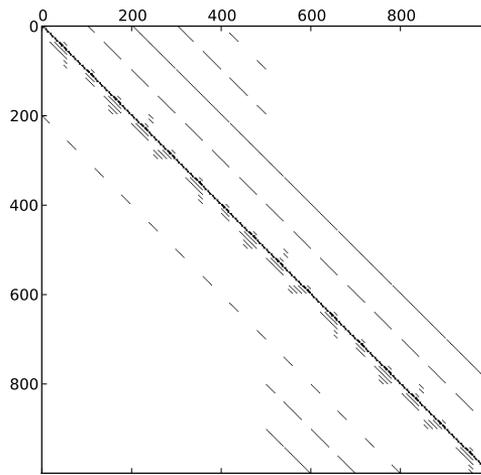
Fig 77. Sparsity patterns for the cell-centred, second order finite difference discretisation of the non-linear heat equation grid with Dirichlet boundary condition.

elaborate for the purpose. We justify its use for two major reasons. Firstly, the matrix-vector form is particularly well suited to simple and efficient computer implementation in programming languages which provide intrinsic matrix and vector operators, such as Python and Fortran 90. Secondly, as will be demonstrated a following section, the matrix-vector form lends itself to an intuitive and straightforward method of constructing the Jacobian matrix of $f(\mathbf{u})$, which is a key requirement of many non-linear solution methods.



(a) 10 cell 1D grid

(b) 10×10 cell 2D grid



(c) $10 \times 10 \times 10$ cell 3D grid

Fig 78. Sparsity patterns for the cell-centred, fourth order finite difference discretisation of the non-linear heat equation grid with Dirichlet boundary condition.

1.5 Calculation of the Jacobian matrix

Non-linear ODE solvers often require an approximate Jacobian matrix to use as part of a Newton-Raphson style iterative scheme. The implicit Runge-Kutta schemes used to solve the method of lines discretisations of the heat equation in the present work have this requirement. While numerical approximation of the Jacobian can be used, the simplest forward difference approximation requires N^2 function evaluations for a system of N cells. This can be reduced somewhat if the band structure of the Jacobian can be predicted *a priori*, so that the forward differences of zero entries are not computed, but the resulting Jacobian is still only a first order accurate approximation. As an alternative to a numerical Jacobian, we apply matrix calculus to the matrix-vector expressions presented in Section 1.2 to compute individual directional derivatives, and assemble the Jacobian from these.

Starting with the one-dimensional second order central case, Equation (207), the directional derivative of a given scalar function value ϕ_i on the one-dimensional stencil $\hat{\mathbf{u}}_i = \{u_{i-1}, u_i, u_{i+1}\}$:

$$\begin{aligned} \frac{\partial \phi_i}{\partial \hat{\mathbf{u}}_i} = & -\frac{\phi_i}{\gamma_i^2} \begin{bmatrix} 0 \\ \dot{\gamma}_i \\ 0 \end{bmatrix}^T + \frac{1}{\gamma_i} \left[\hat{\mathbf{u}}_i^T \mathbf{Q} \hat{\mathbf{u}}_i \begin{bmatrix} 0 \\ \dot{\lambda}_i \\ 0 \end{bmatrix}^T + \dot{\lambda}_i \hat{\mathbf{u}}_i^T [\mathbf{Q}^T + \mathbf{Q}] \right] \\ & + \frac{1}{\gamma_i} \left[\mathbf{k} \hat{\mathbf{u}}_i \begin{bmatrix} 0 \\ \dot{\lambda}_i \\ 0 \end{bmatrix}^T + \dot{\lambda}_i \mathbf{k} \right] + \frac{1}{\gamma_i} \begin{bmatrix} 0 \\ \dot{\sigma}_i \\ 0 \end{bmatrix}^T, \quad (263) \end{aligned}$$

where

$$\begin{aligned} \dot{\lambda}_i &= \frac{\partial \lambda_i}{\partial u}, \quad \ddot{\lambda}_i = \frac{\partial}{\partial u} \left(\frac{\partial \lambda_i}{\partial u} \right) \\ \dot{\gamma}_i &= \frac{\partial \gamma_i}{\partial u}, \quad \dot{\sigma}_i = \frac{\partial \sigma_i}{\partial u}. \end{aligned}$$

As was outlined earlier in the chapter, boundary cells where Robin conditions act have stencils for the Laplacian which potentially have temperature dependent coefficients. An additional derivative term for the stencil \mathbf{k} is required in such cases:

$$\begin{aligned} \frac{\partial \phi_i}{\partial \hat{\mathbf{u}}_i} = & -\frac{\phi_i}{\gamma_i^2} \begin{bmatrix} 0 \\ \dot{\gamma}_i \\ 0 \end{bmatrix}^T + \frac{1}{\gamma_i} \left[\hat{\mathbf{u}}_i^T \mathbf{Q} \hat{\mathbf{u}}_i \begin{bmatrix} 0 \\ \dot{\lambda}_i \\ 0 \end{bmatrix}^T + \lambda_i \hat{\mathbf{u}}_i^T [\mathbf{Q}^T + \mathbf{Q}] \right] \\ & + \frac{1}{\gamma_i} \left[\mathbf{k} \hat{\mathbf{u}}_i \begin{bmatrix} 0 \\ \dot{\lambda}_i \\ 0 \end{bmatrix}^T + \lambda_i \mathbf{k} + \lambda_i \left[\frac{\partial \mathbf{k}}{\partial \hat{\mathbf{u}}_i} \right] \hat{\mathbf{u}}_i \right] + \frac{1}{\gamma_i} \begin{bmatrix} 0 \\ \dot{\sigma}_i \\ 0 \end{bmatrix}^T \end{aligned} \quad (264)$$

with $[\partial \mathbf{k} / \partial \hat{\mathbf{u}}_i]$ denoting the jacobian matrix of the Laplacian operator with respect to the stencil temperatures.

The Jacobian matrix can then be assembled row-wise from each of these row vectors of directional derivatives:

$$J = \begin{bmatrix} \frac{\partial \phi_1}{\partial \hat{\mathbf{u}}_1} & & \dots & 0 \\ & \ddots & & \vdots \\ & & \frac{\partial \phi_i}{\partial \hat{\mathbf{u}}_i} & \\ \vdots & & & \ddots \\ 0 & \dots & & \frac{\partial \phi_N}{\partial \hat{\mathbf{u}}_N} \end{bmatrix} \quad (265)$$

with suitable zero padding implied at each row. The resulting Jacobian is a sparse banded matrix with a band structure which mirrors the equivalent matrix of the linear approximation described in the previous section. This is an important result, because it implies that the Jacobian matrix of the function $f(\mathbf{u})$ can be directly calculated to the same truncation order accuracy as the spatial discretising operator at approximately the same computational expense as the function itself.

The same approach can be extended to the two and three dimensional cases with either second or fourth order approximations in a straightforward manner, with the primary difference being the band structure of the resulting Jacobian matrix.

Appendix 2 Implicit integrators for stiff ordinary differential equations

Preface

Throughout the monograph, implicit integration schemes for the general ordinary differential equation

$$\frac{\partial \chi}{\partial \tau} = \Phi(\tau, \chi), \tau > 0 \tag{266}$$

have been used with limited discussion of their mathematical foundations or details of their computer implementation. In this section, we briefly review the implicit methods used in the temporal integration of the various ordinary differential equations employed in the earlier chapters, and present complete algorithms for solving a single implicit Runge-Kutta step, and a simple integrator with step size control.

2.1 Implicit Runge-Kutta methods

In the present work, all of the solution methods used to solve ODEs of the form given by Equation (266) are implicit methods, all based on one or more repetitions of the implicit Euler method:

$$\chi_1 = \chi_0 + h\Phi(\tau_0 + h, \chi_1) \tag{267}$$

or more generally for any method with s stages:

$$\phi_i = \Phi\left(t_0 + c_i h, \chi_0 + h \sum_{j=1}^s a_{ij} \phi_j\right) \tag{268}$$

$$\chi_1 = \chi_0 + h \sum_{i=1}^s b_i \phi_i, \tag{269}$$

where c_i defines the set of s quadrature points for the scheme, a_{ij} are the set of s^2 stage coefficients, and b_i are the set of s weights for the scheme. Throughout the rest of this section, we employ a device known as a Butcher Tableau (23, §202) to conveniently represent the Runge-Kutta scheme as:

$$\begin{array}{c|ccc}
c_1 & a_{11} & \dots & a_{1s} \\
\vdots & \vdots & \ddots & \vdots \\
c_s & a_{s1} & \dots & a_{ss} \\
\hline
& b_1 & \dots & b_s
\end{array} = \frac{\mathbf{c}}{\mathbf{b}} \Big| \frac{\mathbf{A}}{\mathbf{b}},$$

which encapsulates all of the features of Equations (268) and (269) in a single, pictorial form. We note that schemes where the coefficient matrix \mathbf{A} is strictly lower triangular are the more familiar explicit Runge-Kutta schemes, while those schemes where \mathbf{A} contains non-zero coefficients on the diagonal or in the upper triangular half are implicit. In the present work we focus on *diagonally implicit* Runge-Kutta or DIRK schemes, where the coefficient matrix is lower triangular, implying that the stage is effectively a backward Euler step as defined by Equation (267).

2.2 Second order accurate diagonally implicit Runge-Kutta schemes

2.2.1 The DIRK(2, 2) scheme

The first scheme described is the widely used implicit trapezoidal method, most commonly written as

$$\chi_1 = \chi_0 + \frac{1}{2}h\Phi(\tau_0, \chi_0) + \frac{1}{2}h\Phi(\tau_0 + h, \chi_1) \quad (270)$$

is known to possess an $\mathcal{O}(h^4)$ truncation order, although is it normally classified as a second order scheme. Equation (270) can be re-expressed in a different form as

$$\phi_1 = \Phi(\tau_0, \chi_0) \quad (271)$$

$$\phi_2 = \Phi\left(\tau_0 + h, \chi_0 + \frac{1}{2}h\phi_1 + \frac{1}{2}h\phi_2\right) \quad (272)$$

$$\chi_1 = \chi_0 + \frac{1}{2}h\phi_1 + \frac{1}{2}h\phi_2, \quad (273)$$

which recognisable as a Runge-Kutta scheme of the form given by Equations (268) and (269), with an explicit first stage and an implicit second stage. The Butcher tableau for this is given by

$$\begin{array}{c|cc} 0 & & 0 \\ 1 & 1/2 & 1/2 \\ \hline & 1/2 & 1/2 \end{array}.$$

The structure of the Butcher tableau emphasises the diagonally implicit characteristics of the scheme: the \mathbf{A} coefficient matrix is lower diagonal, with an explicit first stage. This explicit stage imparts the so called “FSAL”, or first-stage-same-as-last, property to the scheme and makes the first stage value effectively free of additional computational expense.

The stage values given in Equations (271) and (272) can also be combined to form a first order approximation using a forward Euler step:

$$\hat{\chi}_1 = \chi_0 + h\phi_2, \tag{274}$$

which can be appended to the basic Butcher tableau, which becomes

$$\begin{array}{c|cc} 0 & & 0 \\ 1 & 1/2 & 1/2 \\ \hline & 1/2 & 1/2 \\ & 0 & 1 \end{array}.$$

We will refer to this scheme as the DIRK(2, 2) scheme through the rest of the document.

As discussed by Butcher (23, §232), the absolute difference between the solution estimates of different orders, Equations (273) and (274), can be used to form a local error estimate for the stage. In this case the local error estimate is taken as

$$\varepsilon = \left| \chi_1 - \hat{\chi}_1 \right| = \left| \frac{1}{2}h\phi_1 - \frac{1}{2}h\phi_2 \right|. \tag{275}$$

As will be discussed in more detail in the description of the computer implementation, this local error estimate can be used to form the basis of a variable step size algorithm, which is a key algorithmic feature when the ordinary differential equation system is strongly non-linear or stiff.

2.2.2 The DIRK(3, 2) scheme

The other second order accurate scheme considered consists of an implicit trapezoidal step (identical to the method just discussed), followed by a second order backward differentiation step. so that a step computed sequentially would have the form

$$\chi_\theta = \chi_0 + \frac{\theta}{2}h\Phi\left(\tau_0, \chi_0\right) + \frac{\theta}{2}h\Phi\left(\tau_0 + \theta h, \chi_\theta\right) \quad (276)$$

$$\chi_1 = -\frac{(1-\theta)^2}{\theta(2-\theta)}\chi_0 + \frac{1}{\theta(2-\theta)}\chi_\theta + \frac{1-\theta}{2-\theta}h\Phi\left(\tau_0 + h, \chi_1\right), \quad (277)$$

where $\theta \in (0, 1)$ controls the relative magnitude of the trapezoidal and BDF stages, with the optimal value for L-stability taken as $\theta = 2 - \sqrt{2}$. This approach can written as a three stage, second order accurate diagonally implicit Runge-Kutta scheme, with a Butcher Tableau given by

0			
$2 - \sqrt{2}$	$(2 - \sqrt{2})/2$	$(2 - \sqrt{2})/2$	
1	$\sqrt{2}/4$	$\sqrt{2}/4$	$(2 - \sqrt{2})/2$
	$\sqrt{2}/4$	$\sqrt{2}/4$	$(2 - \sqrt{2})/2$
	$(4 - \sqrt{2})/12$	$(4 + 3\sqrt{2})/12$	$(2 - \sqrt{2})/6$

which we will refer to as the DIRK(3, 2) scheme.

This scheme has the property that the first stage is explicit (and can be taken from either initial conditions or the results of a preceding stage), and all subsequent stages can be calculated sequentially as with a conventional explicit Runge-Kutta scheme, but with each stage resembling an implicit Euler step. We note this scheme is the same as the TR-BDF2 method described by Hosea and Shampine (61). As with the DIRK(2, 2) scheme, we can compute the local truncation error from the absolute difference between a solution made with the weight vector \mathbf{b} and one made with $\hat{\mathbf{b}}$. In this scheme, the weight vector $\hat{\mathbf{b}}$ corresponds to a third order truncation accuracy solution. The resulting local error estimate is then taken as

$$\varepsilon = \left| \chi_1 - \hat{\chi}_1 \right| = \left| \frac{1-\omega}{3}h\phi_1 + \frac{3\omega+1}{3}h\phi_2 + \frac{2-\omega}{6}h\phi_3 \right| \quad (278)$$

with

$$\omega = \frac{\sqrt{2}}{4}. \quad (279)$$

We remark that although the truncation order accuracy of both the two stage and three stage methods are the same, in most cases, at a given step size, the magnitude of the local error estimate of the three stage method, Equation (278), will be lower than the magnitude of the local error estimate of the two stage method, Equation (275). We therefore expect that the DIRK(3, 2) scheme should be able to admit larger step sizes for a given local error tolerance than the DIRK(2, 2) scheme, which should be advantageous in situations where minimising the total step count yields a reduction in the total solution time.

2.3 Third order accurate diagonally implicit Runge-Kutta schemes

2.3.1 The DIRK(4, 3) scheme

For third order accurate cases, the four stage, L-stable scheme proposed by Alexander (62) has been adopted, given by

$$\begin{array}{c|cccc} 0 & 0 & & & \\ 2\alpha & \alpha & \alpha & & \\ \beta & a_{31} & a_{32} & \alpha & \\ 1 & b_1 & b_2 & b_3 & \alpha \\ \hline & \hat{b}_1 & \hat{b}_2 & \hat{b}_3 & \hat{b}_4 \end{array}$$

with stage coefficients

$$\begin{aligned} \alpha &= 1 - \frac{\sqrt{2}}{2} \cos\left(\frac{1}{3} \arctan \frac{\sqrt{2}}{4}\right) + \frac{\sqrt{6}}{2} \sin\left(\frac{1}{3} \arctan \frac{\sqrt{2}}{4}\right) \approx 0.43586652 \\ \beta &= \frac{18}{13} \alpha^2 - 2\alpha + \frac{14}{13} \approx 0.46823875 \\ a_{31} &= -\frac{4\alpha^2 - 6\alpha\beta + \beta^2}{4\alpha} \approx 0.14073777 \\ a_{32} &= -\frac{\beta(2\alpha - \beta)}{4\alpha} \approx -0.10836556 \end{aligned}$$

and weights

$$b_1 = -\frac{12\alpha^2(\beta-1) - 18\alpha\beta + 12\alpha + 3\beta - 2}{12\alpha\beta} \approx 0.1023994$$

$$b_2 = \frac{6\alpha(\beta-1) - 3\beta + 2}{12\alpha(2\alpha-\beta)} \approx -0.37687845$$

$$b_3 = -\frac{6\alpha^2 - 6\alpha + 1}{3\beta(2\alpha-\beta)} \approx 0.83861253.$$

This local error estimator in this scheme is fourth order accurate, with weights calculated as

$$\hat{b}_1 = \frac{4\alpha(3\beta-1) - 2\beta + 1}{24\alpha\beta} \approx 0.1570249$$

$$\hat{b}_2 = \frac{2\beta-1}{24\alpha(2\alpha-\beta)(\beta-1)} \approx 0.11733044$$

$$\hat{b}_3 = -\frac{4\alpha-1}{\beta(2\alpha-\beta)(\beta-1)} \approx 0.61667803$$

$$\hat{b}_4 = \frac{4\alpha(3\beta-2) - 4\beta}{12(2\alpha-1)(\beta-1)} \approx 0.10896663.$$

The calculation procedure for this method largely follow that of the DIRK(3, 2) scheme described above, but with three rather than two sequential implicit Euler stages following an explicit or FSAL first stage value calculation.

2.4 Vectorised DIRK method calculations

In most cases in the present work, the ODE systems being considered are the result of method of lines discretizations of partial differential equations, or rate equations which have been mapped over a vector space. As a result, any DIRK scheme should be able to be efficiently formed and solved for the vector valued analogue of Equation (266)

$$\frac{\partial \mathbf{x}}{\partial t} = \Phi(t, \mathbf{x}), t > 0, \Phi : \mathfrak{R}^n \rightarrow \mathfrak{R}^n \quad (280)$$

for which the reference implicit Euler step, Equation (267), can be written as

$$\mathbf{x}_1 = \mathbf{x}_0 + h\Phi(\tau_0 + h, \mathbf{x}_1). \quad (281)$$

For the sake of computational efficiency, it is possible to simultaneously solve all stages of the equivalent vector valued version of any DIRK integrator for Equation

(280). Using the example of a three stage scheme, we can compute a stacked vector of intermediate stage solution values as

$$\mathbf{v} = (\mathbf{z} \otimes \mathbf{x}_0) + (\mathbf{A} \otimes \mathbf{I}) h \mathbf{f} \quad (282)$$

with

$$\mathbf{x}_0 = \begin{bmatrix} \chi_0 \\ \vdots \\ \chi_n \end{bmatrix}, \mathbf{v} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \end{bmatrix}, \mathbf{z} = [1, 1, 1], \mathbf{f} = \begin{bmatrix} \Phi(\mathbf{x}_1) \\ \Phi(\mathbf{x}_2) \\ \Phi(\mathbf{x}_3) \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \end{bmatrix}.$$

The stacked vector of stage function values, \mathbf{f} , can then be used to calculate the final step solution as

$$\mathbf{x}_1 = \mathbf{x}_0 + (\mathbf{I} \otimes \mathbf{b}) \mathbf{f} \quad (283)$$

and the local error estimate similarly calculating using

$$\mathbf{e} = \left| (\mathbf{I} \otimes (\mathbf{b} - \hat{\mathbf{b}})) \mathbf{f} \right|. \quad (284)$$

The system of implicit equations defined by Equation (282) can be solved using a simple Newton-Raphson iteration of the form

$$\mathbf{r}^{(i)} = \mathbf{v}^{(i)} - (\mathbf{z} \otimes \mathbf{x}_0) - (\mathbf{A} \otimes \mathbf{I}) h \mathbf{f}^{(i)} \quad (285)$$

$$\mathbf{d}^{(i)} : \mathbf{J} \mathbf{d}^{(i)} = -\mathbf{r}^{(i)} \quad (286)$$

$$\mathbf{v}^{(i+1)} = \mathbf{v}^{(i)} + \mathbf{d}^{(i)}, \quad (287)$$

where \mathbf{J} is the Jacobian matrix of the residual function, Equation (285), itself given by

$$\mathbf{J} = \begin{bmatrix} \frac{\partial \mathbf{r}}{\partial \mathbf{v}} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} - (\mathbf{A} \otimes \mathbf{I}) h \begin{bmatrix} \frac{\partial \mathbf{f}}{\partial \mathbf{v}} \end{bmatrix}. \quad (288)$$

Typically, Equation (286) is solved using an LU factorization of the Jacobian matrix. For computational efficiency reasons, an optimal scheme would aim to compute and factorise the Jacobian only once per step at some suitable point within the step. In the simplest (although certainly not the most efficient) implementation, forward differences

can be used to approximate each of the matrices of directional derivatives in the block matrix

$$\mathbf{F}' = \left[\frac{\partial \mathbf{f}}{\partial \mathbf{v}} \right] = \begin{bmatrix} \mathbf{F}'_{11} & \mathbf{F}'_{12} & \mathbf{F}'_{13} \\ \mathbf{F}'_{21} & \mathbf{F}'_{22} & \mathbf{F}'_{23} \\ \mathbf{F}'_{31} & \mathbf{F}'_{32} & \mathbf{F}'_{33} \end{bmatrix} \quad (289)$$

with

$$\mathbf{F}'_{ij} = \left[\frac{\partial \mathbf{f}_i}{\partial \mathbf{v}_j} \right], \quad (290)$$

which requires $9N$ evaluations of $\Phi(\chi)$ for this (or any) three stage method. However, by examining each of the three stage function values:

$$\mathbf{f}_1 = \Phi(\mathbf{x}_0) \quad (291)$$

$$\mathbf{f}_2 = \Phi(\mathbf{x}_0 + a_{21}h\mathbf{f}_1 + a_{22}h\mathbf{f}_2) \quad (292)$$

$$\mathbf{f}_3 = \Phi(\mathbf{x}_0 + a_{31}h\mathbf{f}_1 + a_{32}h\mathbf{f}_2 + a_{33}h\mathbf{f}_3) \quad (293)$$

it can be immediately deduced that

$$\mathbf{F}'_{11} = \mathbf{F}'_{12} = \mathbf{F}'_{13} = \mathbf{0} \quad (294)$$

$$\mathbf{F}'_{23} = \mathbf{0}, \quad (295)$$

which reduces \mathbf{F}' to lower triangular form:

$$\mathbf{F}' = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{F}'_{21} & \mathbf{F}'_{22} & \mathbf{0} \\ \mathbf{F}'_{31} & \mathbf{F}'_{32} & \mathbf{F}'_{33} \end{bmatrix} \quad (296)$$

and decreases the upper bound of the computational cost from $9N$ to $5N$. In some cases this cost can be further reduced by either exploiting structure of individual non-zero \mathbf{F}_{ij} matrices, or by direct algebraic computation, if derivations of such directional derivatives are available. The resulting Jacobian matrix for this three stage scheme takes the final form

$$\mathbf{J} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ -a_{22}h\mathbf{F}'_{21} & \mathbf{I} - a_{22}h\mathbf{F}'_{22} & \mathbf{0} \\ -a_{32}h\mathbf{F}'_{21} - a_{33}h\mathbf{F}'_{31} & -a_{32}h\mathbf{F}'_{22} - a_{33}h\mathbf{F}'_{32} & \mathbf{I} - a_{33}h\mathbf{F}'_{33} \end{bmatrix}, \quad (297)$$

which is amenable to factorization by any one of a number of fast, triangular block factorization algorithms. A further optimization can be applied to any three stage scheme when the component matrices are easily invertible, as is the case for diagonal, non-singular matrices. This can be achieved by applying a variant of the procedure described by Swift (63), and solving the block matrix equation

$$\begin{bmatrix} \mathbf{G}_{11} & \mathbf{G}_{12} & \mathbf{G}_{13} \\ \mathbf{G}_{21} & \mathbf{G}_{22} & \mathbf{G}_{23} \\ \mathbf{G}_{31} & \mathbf{G}_{32} & \mathbf{G}_{33} \end{bmatrix} \begin{bmatrix} \mathbf{J}_{11} & \mathbf{0} & \mathbf{0} \\ \mathbf{J}_{21} & \mathbf{J}_{22} & \mathbf{0} \\ \mathbf{J}_{31} & \mathbf{J}_{32} & \mathbf{J}_{33} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \quad (298)$$

to directly yield the inverse Jacobian matrix and allow a modified version of the Newton-Raphson scheme to be used without the need for matrix factorization. Equation (298) solves to yield

$$\mathbf{J}^{-1} = \begin{bmatrix} \mathbf{J}_{11}^{-1} & 0 & 0 \\ -\mathbf{J}_{22}^{-1}\mathbf{J}_{21}\mathbf{J}_{11}^{-1} & \mathbf{J}_{22}^{-1} & 0 \\ -\mathbf{J}_{33}^{-1}(\mathbf{J}_{32}\mathbf{J}_{22}^{-1}\mathbf{J}_{21} + \mathbf{J}_{31}) & \mathbf{J}_{33}^{-1}\mathbf{J}_{32}\mathbf{J}_{22}^{-1} & \mathbf{J}_{33}^{-1} \end{bmatrix}, \quad (299)$$

which requires only the computation of the inverse of the diagonal component matrices. The cost of this can be as low as computing the reciprocal of each matrix diagonal in the best case. The Newton-Raphson iterations given by Equations (285), (286), and (287) can then be simplified by substituting the inverse Jacobian shown for the factorized version used in the original, yielding

$$\mathbf{r}^{(i)} = \mathbf{v}^{(i)} - (\mathbf{z} \otimes \mathbf{x}_0) - (\mathbf{A} \otimes \mathbf{I})h\mathbf{f}^{(i)} \quad (300)$$

$$\mathbf{d}^{(i)} = -\mathbf{J}^{-1}\mathbf{r}^{(i)} \quad (301)$$

$$\mathbf{v}^{(i+1)} = \mathbf{v}^{(i)} + \mathbf{d}^{(i)}. \quad (302)$$

Similar simplifying calculations can be made for the two stage and four stage schemes as those shown here for the three stage case.

2.5 Initialization and interpolation for DIRK scheme

A important property of the DIRK methods presented in this work is their ability to yield better than second order accurate function values and derivatives at both the beginning and end of a step, and at some intermediate stage values in cases where the scheme has more than two stages. The availability of such data naturally suggests that interpolation could be used to construct C_1 continuous estimates of the solution within a stage which are of comparable accuracy to the solution itself. The availability of a suitably accurate interpolant would allow for the solution at particular time values of interest to be estimated without the need to modify otherwise optimal step sizes.

In the present work we adopt some of the methods described by Brankin and Gladwell (64). A cubic Hermite spline of the general form

$$\tilde{\chi}(\theta) = \left[2(\chi_0 - \chi_1) + h(\phi_0 + \phi_1) \right] \theta^3 - \left[3(\chi_0 - \chi_1) + h(2\phi_0 + \phi_1) \right] \theta^2 + h\phi_0\theta + \chi_0 \quad (303)$$

with

$$\theta = \frac{\tau - \tau_0}{h} \quad (304)$$

is used to interpolate the solution value at some time $\tau \in [\tau_0, \tau_0 + h]$ using the stage starting and ending solutions and function values. This operation requires no additional calculations beyond simple scalar and vector addition. We also note that the derivative of Equation (303)

$$h\tilde{\phi}(\theta) = 3 \left[2(\chi_0 - \chi_1) + h(\phi_0 + \phi_1) \right] \theta^2 - 2 \left[3(\chi_0 - \chi_1) + h(2\phi_0 + \phi_1) \right] \theta + h\phi_0 \quad (305)$$

can be used to interpolate a function value in the same way as the solution value itself.

One of the distinct advantages of implicit Runge-Kutta schemes is that they require only function and solution values from the beginning of the step in order to complete a new integration step. This is in contrast to backward differentiation formulas, for example, which require values from one or more previous steps for the computation of a

new step. This lack of requirements for previous data greatly simplifies the calculation of the first step of any integration, and also allows simple implementation of variable step schemes with disrupting the truncation order accuracy of the scheme, unlike backward differentiation formulas.

One open question for the iterative solution scheme described in Section 2.4 is how to compute initial stage solution values to start the Newton-Raphson iteration. The success and rate of convergence of Newton methods is known to be strongly dependent on the proximity of the initial solution estimate to the true solution, therefore applying suitable starting solution values for the Newton-Raphson search is important for minimising the number of iterations required to reach convergence. These starting values are calculated by interpolating or extrapolating function values to form the vector \mathbf{f} . We chose to interpolate the function value rather than solution value because, in general, forming an initial solution approximation from estimated stage function values is far less expensive than computing stage function values from initial solution estimates.

For each of the three schemes, we apply slightly different versions of the same basic procedure:

- For the DIRK(2,2) scheme, the FSAL property provides $\phi_1 = \phi_0$, with ϕ_2 calculated by extrapolation from the values of the preceding step using (305), or using an explicit Euler step for the first step. If CFL limits apply to the explicit Euler step size, the first step size is controlled accordingly.
- For the DIRK(3,2) scheme, we use two distinct phases:
 1. the FSAL property is utilised for the first stage value, so that $\phi_1 = \phi_0$
 2. the second stage value is calculated by extrapolation from the values of the preceding step using (305), or using an explicit Euler step for the first step.
 3. the third stage value is approximated using Equation (305) by extrapolating from $t + \theta h$ to $t + h$ using the values of ϕ_1 and ϕ_2
- For the DIRK(4,3) scheme, the procedure is similar to that used for DIRK(3,2)
 1. the FSAL property is utilised for the first stage value, so that $\phi_1 = \phi_0$
 2. the second stage value is extrapolated from values of the preceding step using (305), or using an explicit Euler step for the first step.
 3. the values of ϕ_1 and ϕ_2 are used to interpolate the value of ϕ_3 using Equation (305) (noting that $\beta < 2\alpha$).
 4. the values of ϕ_3 and ϕ_2 are then used to extrapolate to $\tau_0 + h$ using Equation (305).

2.6 Integrator algorithm design

The previous two sections describe most of the mathematical apparatus necessary to complete a single integration step using any of the second or third order accurate methods used in the present work. The step computations presented are, however, not self contained methods for solving ODEs, and they require a supporting computational framework for managing integrator step size, tolerances, and convergence.

To expand these DIRK step computations into complete integrators, we largely follow the local error evaluation and step size adjustment procedure outlined by Shampine (65). We note that while this procedure was originally developed for high order explicit Runge-Kutta schemes, most of the concepts are equally applicable to implicit Runge-Kutta schemes with only minor modifications.

The step computation and integrator algorithms are defined here only in pseudo-code. The reference computer implementation used through the calculations in the present work is written as a class in the Python language, making extensive use of the `numpy` module, which provide native Python matrix and vector types and operators which are directly interface to high performance, start of the art compiled libraries, include the ATLAS linear algebra subroutines (66) and the UMFPACK sparse factorization suite (67).

2.6.1 *Newton-Raphson iterations and convergence criteria*

For a given step, we begin by computing stage function value estimates by extrapolation and interpolation from the first stage value using the methods discussed in Section 2.5. This initial function estimate is then used to calculate an initial approximation of the scheme stage values. From these, the Jacobian matrix can be computed and factorized. These initial values can also be used as the starting point for a line search to define a bracketing interval for a solution – the importance of this is discussed in more detail below. With this set of initial data available, we define an error limit for each solution value from a predefined absolute and relative tolerance:

$$\mathbf{m} = \text{atol} + \frac{1}{2} \text{rtol} (\mathbf{x}_0 + \mathbf{x}_1) \quad (306)$$

using a solution value for the relative component of the tolerance which is the average of the average of the step starting and ending values. We do this to allow for cases where components of the solution may vary rapidly within a given step.

The Newton-Raphson iteration defined by Equations (285), (286), and (287) is

repeated until the values of the residual vector satisfy

$$\left| [r_i/m_i] \right|_{\infty} < 1, \quad (307)$$

which should ensure that every member of \mathbf{r} has converged to a value less than the absolute of its tolerance. We prefer this form of convergence test to a standard L_2 norm or similar, because often the system of linear equations may contain more than one coupled ODE, and the solution values of each may be of dissimilar magnitudes. In such cases, an L_2 norm is not necessarily a representative measure of convergence compared to the tolerance.

In addition to the standard Newton-Raphson iteration, we include an idea borrowed from the `rtsafe` code described by Press et al. (53, §9) and maintain a pair of solution estimates at each Newton iteration, \mathbf{v}_l and \mathbf{v}_h which form a bracketing interval about a true solution vector such that residual vectors for each satisfies

$$\max(\mathbf{r}_l * \mathbf{r}_u) \leq 0.$$

At each iteration, we require that the Newton-Raphson solution update \mathbf{d} should lie on the bracketing interval so that each solution value satisfies

$$|d_i| < |v_{hi} - v_{li}|. \quad (308)$$

In cases where this condition is not met, that is cases where the Newton-Raphson update would move a solution value outside the interval known to contain a solution, the Newton-Raphson step is rejected in favour of a bisection step instead:

$$v_i^{(k+1)} = \frac{1}{2} (v_{hi} + v_{li}), \quad (309)$$

The bisection step also helps ensure that each iteration will advance the solution towards convergence in cases where the Newton-Raphson method may otherwise fail. Both the bracketing interval and tolerance vectors are updated at the completion of each Newton-Raphson iteration.

At convergence of the stage values, the final solution is computed using Equation (283) and the local error estimate using Equation (284). Lastly, the local error estimate is used to calculate a local error estimate ratio

$$\mathbf{p} = \left[(m_i/e_i)^{\frac{1}{s}} \right], \quad (310)$$

where m_i and e_i are the solution tolerance and local error, and s is the order of the local error estimate. For a solution to have acceptable local error, this ratio should be greater than unity.

The complete single step algorithm is given in pseudo code in Figure 1.

2.6.2 Time integration and step size adaptation

The step solver shown in Figure 1 computes the desired DIRK step to a specified tolerance, returning a trial solution estimate \mathbf{x}_1 , function value \mathbf{g}_1 , and local error estimate ratio \mathbf{p} . This local error estimate is an error-per-step value, rather than an error-per-unit-step value, and the basic logic for determining the acceptability of a give step is based on the error-per-step procedure discussed by Shampine (65).

For any given integration step to be accepted, we require that

$$\min(\mathbf{p}) \geq 1, \quad (311)$$

which implies that the local error of every solution value is less than the prescribed tolerance.

In cases where this condition is not satisfied, the trial solution is rejected and the step repeated with a smaller step size, calculated as

$$h_{new} = \beta h_{old} \quad (312)$$

with

$$\beta = \max(0.1, \min(\mathbf{p}, 0.9)), \quad (313)$$

which restricts the change in step size to be between 0.1 and 0.9 times the size of the rejected step. This prevents overly large reductions in step size per trial step, but at the same time ensures that a sufficient reduction is made in cases where $\min(\mathbf{p})$ is close to unity.

In cases where the step has acceptable local error, the integrator will permit the step size to be extended, providing the previous step was not rejected. This is done to prevent instability in the step size algorithm. When it is determined that the step size can be increased, the new step size is calculated as

$$h_{new} = \alpha h_{old} \quad (314)$$

with

$$\alpha = \min(\mathbf{p}, 2), \quad (315)$$

which restricts step size increases to a maximum of twice the current step size.

The integrator allows the solution to be interpolated within a step, so that the solution can be recorded at events or times of interest. This is done using the interpolation scheme described in Section 2.5. The complete integrator algorithm is given in Figure 2.

Algorithm 1 Newton-Raphson/bisection method for computing single DIRK integrator steps.

```

1: Input: solution at  $\tau_0$ , step size
2: Output: Step solution,function value,local error estimate
3: Data:  $\mathbf{x}_0, h$ 

4: Interpolate initial function approximation  $\mathbf{f}$ 
5:  $\mathbf{v} \leftarrow (\mathbf{z} \otimes \mathbf{x}_0) + (\mathbf{A} \otimes \mathbf{I})h\mathbf{f}$ 
6: Search for a bracketing interval for the step solution  $\mathbf{v}_l, \mathbf{v}_h$ 
7:  $\mathbf{x}_1 \leftarrow \mathbf{x}_0 + h(\mathbf{I} \otimes \mathbf{b})\mathbf{f}$ 
8:  $\bar{\mathbf{v}} \leftarrow \frac{1}{2} \left[ (\mathbf{z} \otimes \mathbf{x}_0) + \mathbf{v} \right]$ 
9:  $\mathbf{k} \leftarrow \left| \mathbf{v}_u - \mathbf{v}_l \right|$ 
10: Compute and factorize the Jacobian matrix for the step  $\mathbf{J}$ 
11: repeat
12:    $\mathbf{m} \leftarrow \text{atol} + \text{rtol} * \bar{\mathbf{v}}$ 
13:    $\mathbf{f} \leftarrow \Phi(\mathbf{v})$ 
14:    $\mathbf{r} \leftarrow \mathbf{v} - (\mathbf{z} \otimes \mathbf{x}_0) - (\mathbf{A} \otimes \mathbf{I})h\mathbf{f}$ 
15:   solve  $\mathbf{J}\mathbf{d} = -\mathbf{r}$  for  $\mathbf{d}$ 
16:   for all  $d_i$  do ▷ Check result of Newton-Raphson update
17:     if  $d_i < k_i$  then
18:        $v_i \leftarrow v_i + d_i$  ▷ Accept the Newton-Raphson step
19:     else
20:        $v_i \leftarrow \frac{1}{2}(v_l + v_h)$  ▷ Reject in favour of Bisection step
21:     end if
22:   end for

```

Algorithm 1 Newton-Raphson/bisection method for computing single DIRK integrator steps *continued*

```

23:   for all  $v_i$  do ▷ Update the bracketing interval
24:     if  $r_i \leq 0$  then
25:        $v_{li} \leftarrow v_i$ 
26:     else
27:        $v_{hi} \leftarrow v_i$ 
28:     end if
29:   end for
30:    $\mathbf{k} \leftarrow \left| \mathbf{v}_u - \mathbf{v}_l \right|$ 
31:    $\bar{\mathbf{v}} \leftarrow \frac{1}{2} \left[ (\mathbf{z} \otimes \mathbf{x}_0) + \mathbf{v} \right]$ 
32:   until  $\left\| [r_i/m_i] \right\|_{\infty} \leq 1$  or iterations limits exceeded
33:    $\mathbf{x}_1 \leftarrow \mathbf{x}_0 + h(\mathbf{I} \otimes \mathbf{b}) \mathbf{f}$ 
34:    $\mathbf{e} \leftarrow \left| (\mathbf{I} \otimes (\mathbf{b} - \hat{\mathbf{b}})) \mathbf{f} \right|$ ;
35:    $\mathbf{m} \leftarrow \text{atol} + \text{rtol} * \frac{1}{2} (\mathbf{x}_1 + \mathbf{x}_0)$ 
36:    $\mathbf{p} \leftarrow \left[ (m_i/e_i)^{\frac{1}{5}} \right]$ 
37:   Return  $\mathbf{x}_1, \Phi(\mathbf{x}_1), \mathbf{p}$ 

```

Algorithm 2 Variable step integrator for DIRK schemes

```
1: Input: initial conditions, starting and ending times, minimum and maximum
   timesteps, output frequency
2: Output: solution values at each requested output time and at the ending time
3: Data;  $\mathbf{x}_0$ ,  $\tau_0$ ,  $\tau_1$ ,  $\Delta\tau_{min}$ ,  $\Delta\tau_{max}$ ,  $\omega_{out}$ 

4:  $\Delta\tau \leftarrow \tau_1 - \tau_0$ 
5:  $\mathbf{x}_1 \leftarrow \mathbf{x}_0$ 
6:  $\tau \leftarrow \tau_0$ 
7:  $\tau_{out} \leftarrow 0$ 
8: while  $\Delta\tau > 0$  do
9:   Solve trial step of  $h$  returning  $\hat{\mathbf{x}}_1$ ,  $\Phi(\hat{\mathbf{x}}_1)$ ,  $\mathbf{p}$ 
10:  if  $\min(\mathbf{p}) \geq 1$  then ▷ Accept local error
11:     $\mathbf{x}_1 \leftarrow \hat{\mathbf{x}}_1$ 
12:     $\Delta\tau \leftarrow \Delta\tau - h$ 
13:     $\Delta\tau_{out} \leftarrow \Delta\tau_{out} + h$ 
14:     $\delta \leftarrow \Delta\tau_{out} - (\omega_{out})^{-1}$ 
15:    if  $\delta \geq 0$  then ▷ Check output frequency
16:      Compute and store an interpolated solution at  $\tau + h - \delta$ 
17:       $\Delta\tau_{out} \leftarrow 0$ 
18:    end if
19:     $\tau \leftarrow \tau + h$ 
20:    if previous step was not rejected then
21:       $\alpha \leftarrow \min(\mathbf{p}, 2)$ 
22:       $h \leftarrow \min(\Delta\tau_{max}, \beta * h)$ 
23:    end if
24:  else ▷ Reject trial step and compute smaller step size
25:     $\beta \leftarrow \max(1/10, \min(\mathbf{p}, 9/10))$ 
26:     $h \leftarrow \min(\Delta\tau_{min}, \beta * h)$ 
27:  end if
28: end while
```

Appendix 3 Thermophysical properties and microstructure evolution calculations for carbon steels

Preface

This appendix summarises the methods of calculation of thermophysical coefficients and microstructure evolution during continuous cooling used throughout the monograph.

3.1 Heat capacity of Carbon steels

In (12), a computationally inexpensive, semi-empirical model for the specific heat capacity of low alloy steels is described, itself based on the earlier work of Chuang, Schmid and Chang (29). Unfortunately, the model as published contains several typographical errors and omissions which make it impossible to reproduce the authors' results, nor match the datasets used to fit the model. What follows is a version of this model which has been corrected for these errors and uses recalculated coefficients to yield the final specific heat capacity in SI units – J/kg K in each case.

In the Browne heat capacity model, the total specific heat capacity κ is assumed to be the sum of four components:

$$\kappa = C + (\kappa - C) + \kappa_e + \kappa_m, \quad (316)$$

where C is the heat capacity at constant volume, $(\kappa - C)$ is the lattice expansion contribution, κ_e is the electronic specific heat, and κ_m is the magnetic specific heat. For the specific heat at constant volume, an approximation of the Debye integral is used:

$$C = \frac{3R}{55.847 \times 10^{-3}} \left[1 - 0.0488 \left(\frac{u_d}{u} \right)^{1.87} + 5 \times 10^{-4} \right] \quad (317)$$

with the Debye temperature u_d is taken as 425 K for α iron and 400 K for γ iron. The lattice expansion contribution is calculated using an approximation of the Gruneisen equation as:

$$(\kappa - C) = aC^2u \quad (318)$$

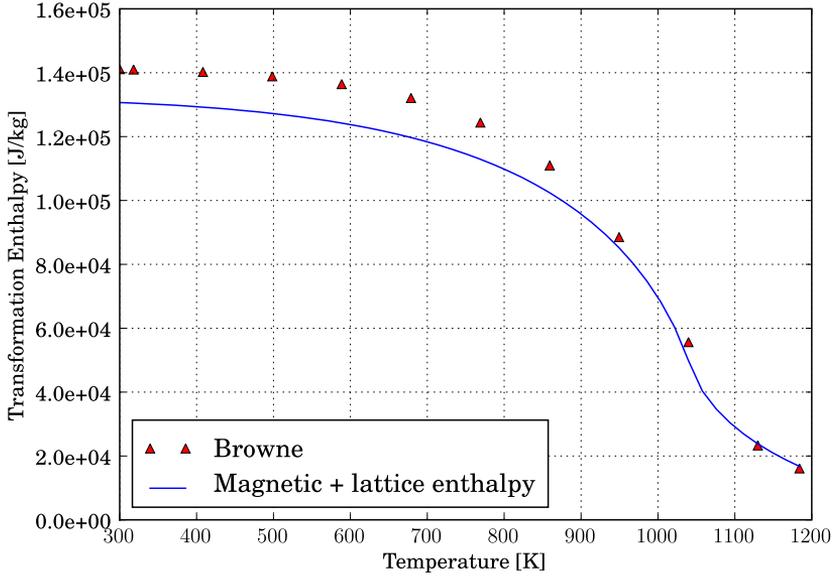


Fig 79. Combined lattice transformation and magnetic latent over the temperature range 400 K–1198 K. Model predictions after Browne (12).

with $a = 3.4044 \times 10^{-7}$ for α iron and $a = 3.9874 \times 10^{-7}$ for γ iron. The electronic specific heat is assumed to be linearly dependent on temperature:

$$\kappa_e = bu \quad (319)$$

with $b = 8.9889 \times 10^{-2}$ for α iron and $b = 5.9985 \times 10^{-2}$ for γ iron. Finally, for α iron, the magnetic specific heat is taken as:

$$\kappa_m = \begin{cases} 9.1321 \times 10^2 \left(\frac{u}{u_c}\right)^{1.235} \exp\left[-3.5\left(1 - \frac{u}{u_c}\right)^{0.45}\right] & u \leq u_c \\ 9.1321 \times 10^2 \left(\frac{u}{u_c}\right)^{1.235} \exp\left[-51.5\left(\frac{u}{u_c} - 1\right)^{0.41}\right] & u > u_c \end{cases}, \quad (320)$$

where u_c is the Curie temperature of iron, taken as 1042 K.

3.2 Magnetic transformation enthalpy of alpha iron

Browne (12) also presents a simplified calculation for the transformation enthalpy of pure alpha iron which combines both the lattice transformation enthalpy and the magnetic specific heat enthalpy, given in J/kg as

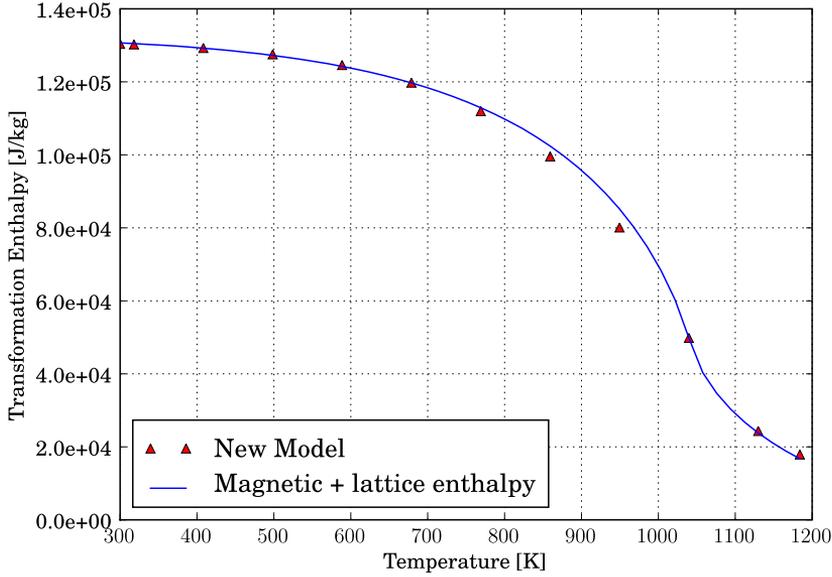


Fig 80. Combined lattice transformation and magnetic latent heat over the temperature range 400 K–1198 K. Model predictions made using Equation (322), with an assumed lattice transformation heat of 16750 J/kg.

$$\mu_{\gamma \rightarrow \alpha} + \mu_{\text{magnetic}} = -141995 \left[1 - \frac{0.8865}{\cosh(0.0064[u - u_c])} \right]. \quad (321)$$

Compared with either the predictions of Equation (320) or the difference between the JANAF tabulations and Equations (317), (318), and (319), however, this model seems to over predict the magnetic specific heat enthalpy by approximately 10% at temperatures below the Curie temperature of pure iron (1042 K). This is shown in Figure 79 The form of Equation (320) has no closed integral form, so numerical quadrature based on the Romberg integration method was used to evaluate the integral:

$$\mu_{\text{magnetic}} = - \int_u^{u_{\text{Ae3}}} C_p du$$

over the temperature range 300 K to 1184 K. A model based on modifications to the magnetic enthalpy equations proposed by Chuang, Schmidt and Chang (29) was then fitted to this data by an orthogonal distance regression method using the odrpack package (68). For paramagnetic ferrite, the integral from the Curie temperature to the Ae3 temperature of pure iron is approximated as

$$-\mu_{\text{paramagnetic}} = \psi_p \left[u_c (1 + \varepsilon_p) \varepsilon_p - (u - u_c \varepsilon_p) \exp \left(\frac{1}{\varepsilon_p} \left(1 - \frac{u}{u_c} \right) \right) \right]$$

with fitted values

$$\begin{aligned} \varepsilon_p &= \frac{1}{8\kappa_p} \\ \kappa_p &= 1.7043 \\ \psi_p &= 25.360 \end{aligned}$$

and for the ferromagnetic case, the integral from 300 K to the Curie Temperature is approximated as:

$$-\mu_{\text{Ferromagnetic}} = \psi_f \left[u_c (1 - \varepsilon_f) \varepsilon_f + (u - u_c \varepsilon_f) \exp \left(-\frac{1}{\varepsilon_f} \left(1 - \frac{u}{u_c} \right) \right) \right]$$

with fitted values

$$\begin{aligned} \varepsilon_f &= \frac{1}{4} \\ \psi_f &= 23.335. \end{aligned}$$

These can be combined to give a final expression for the magnetic transformation enthalpy over the temperature range 300 K to 1184 K in J/kg as

$$-\mu_{\text{magnetic}} = \begin{cases} 6360.1 - 5.8425 [u - 260.50] \exp \left(-4 \left[1 - \frac{u}{u_c} \right] \right) & u \leq u_c \\ -340.10 + 1.9027 [u + 79.747] \exp \left(13.066 \left[1 - \frac{u}{u_c} \right] \right) & u_c < u \leq u_{\text{Ae}3} \\ 0 & \text{otherwise} \end{cases} \quad (322)$$

A comparison of this final function and numerically integrated values is shown in Figure 80.

3.3 Thermophysical properties of wüstite

Although oxidation behaviour of steel surfaces has been ignored in most of the models described in the present work, we do consider the presence of a surface wüstite layer

in calculating the surface thermal resistance as part of the Robin boundary condition formulation. For this we require thermophysical coefficients for wüstite. For thermal conductivity, we use the relationship proposed by Akiyama *et al* (34):

$$\lambda_{\text{wüstite}} = \begin{cases} [1.136 \times 10^{-1} + 2.335 \times 10^{-4}u]^{-1} & 298 < u < 825 \\ [3.926 \times 10^{-1} - 1.047 \times 10^{-4}u]^{-1} & 825 \leq u < 1500 \end{cases} \quad (323)$$

for other thermophysical properties, the equations for stoichiometric iron monoxide from the JANAF tables (28) as used:

$$\kappa_{\text{wüstite}} = 45.75120 + 18.78555u - 5.952201u^2 + 0.852779u^3 - \frac{0.081265}{u^2}. \quad (324)$$

The density of wüstite is taken as 5580 kg/m³ at all temperatures, and a standard heat of formation as $\Delta H = -3786$ kJ/kg.

3.4 Simplified microstructure evolution calculations for C-Mn steels

In order to demonstrate the influence of microstructure evolution on thermal-physical and thermal behaviour, simplified calculations based on the transformation theory of Kirkaldy and Venugopalan (32) have been used for diffusive transformations, and the relationship proposed by Koistinen and Marburger (41) for the martensite transformation.

In this treatment of microstructure evolution, the microstructure of any proeutectoid steel obeys a volume conservation statement:

$$\chi_a + \chi_f + \chi_p + \chi_b + \chi_m = 1, \quad (325)$$

where χ_j is the volume fraction of a given phase, and the subscripts a , f , p , b and m denote austenite, proeutectoid ferrite, pearlite, bainite and martensite respectively. During continuous cooling from an initial austenitic state, steel microstructure is assumed to undergo one or more irreversible austenite decomposition processes. The volume fraction of transformation product from the j^{th} austenite decomposition reaction, χ_j , is presumed to be calculable as the temporal integral of an evolution equation, taking the general form

$$\chi_j = \int_0^t \frac{\partial \chi_j}{\partial \tau} d\tau \quad (326)$$

with Equation 325 requiring $\chi_j \in [0, 1] \forall \tau$. The kinetics of diffusive transformations are assumed to obey the general rate equation

$$\frac{\partial \chi_j}{\partial \tau} = \beta_j(\xi_j, \iota, u) \xi_j^{\frac{2}{3}(1-\xi_j)} (1 - \xi_j)^{\frac{2}{3}\xi_j} \quad (327)$$

$$\xi_j = \frac{\chi_j}{\chi_j^{\max}}, \quad (328)$$

where $\xi_j \in [0, 1]$ is the fraction of the reaction completed, χ_j^{\max} is the maximum volume fraction of the phase which will form, ι is the ASTM grain size number of the transforming austenite, and β_j is a reaction specific function. In the present analysis, we reduce the full computations described by Kirkaldy and Venugopalan to consider only the effects of carbon, manganese and silicon on transformation behaviour.

For the pro-euctectoid ferrite transformation, β is calculated as:

$$\beta_f = \frac{2^{(\iota-1)/2}}{59.6w\{\text{Mn}\}} [u_{Ae3} - u]^3 \exp\left[-\frac{96278}{Ru}\right] \quad (329)$$

for the austenite-pearlite transformation, β is calculated as:

$$\beta_p = \frac{2^{(\iota-1)/2}}{1.79} [u_{Ae1} - u]^3 \exp\left[-\frac{115115}{Ru}\right] \quad (330)$$

and for the austenite-bainite reaction, β is calculated using

$$\beta_b = \frac{10^4 \times 2^{(\iota-1)/2} [u_{bs} - u]^2 \exp\left[-\frac{112968}{Ru}\right]}{[2.34 + 10.1w\{\text{C}\}] \exp[-\xi^2 (1.9w\{\text{C}\} + 2.5w\{\text{Mn}\})]}. \quad (331)$$

The formation of martensite is presumed to obey the Koistinen-Marburger relationship (41), so that the volume fraction of retained austenite at a given degree of undercooling below the martensite start temperature is calculated as

$$\chi_a = \chi_a^{\max} \exp(-\eta (u_{ms} - u)) \quad (332)$$

with χ_a^{\max} denoting the volume fraction of austenite at the martensite start temperature, and the reaction constant taken as $\eta=0.011$ /K. The mass balance given by equation (325) can be used to transform Equation (332) into an expression for the volume fraction of martensite:

$$\chi_m = \chi_m^{\max} (1 - \exp(-\eta (u_{ms} - u))). \quad (333)$$

This can be differentiated with respect to time to yield a martensite transformation rate equation of the form

$$\frac{\partial \chi_m}{\partial \tau} = \chi_m^{\max} \eta \exp(-\eta (u_{ms} - u)) \frac{\partial u}{\partial \tau}. \quad (334)$$

We further require that the martensite reaction should be irreversible. This implies enforcing two conditions – firstly, the rate equation must be zero above the martensite start temperature, and secondly that any temperature increase below the martensite start temperature will not result in a change in sign of the reaction rate. These constraints are functionally similar to the modified Koistinen and Marburger formulation proposed by Hömberg (16). In the present work, we enforce these via a modified cooling rate calculation

$$\widehat{\frac{\partial u}{\partial \tau}} = \mathcal{H} \left(-(u_{ms} - u) \frac{\partial u}{\partial \tau} \right) \frac{\partial u}{\partial \tau}, \quad (335)$$

where \mathcal{H} denotes the Heaviside function. This leads to the final rate equation

$$\frac{\partial \chi_m}{\partial \tau} = -\eta \chi_m^{\max} \exp[-\eta (u_{ms} - u)] \widehat{\frac{\partial u}{\partial \tau}}. \quad (336)$$

The maximum volume fractions of each transformation product are determined as follows:

1. The maximum volume fraction of proeutectoid ferrite formed is taken from a lever rule calculation between the Ae2 and Ae3 lines above the Ae1 temperature, or the Ae2 and Acm line below the Ae1 temperature, This is illustrated in Figure 81, which implies

$$\chi_f^{\max} = \begin{cases} \frac{\ell_0}{\ell_0 + \ell_1} & u_{\text{Ae3}} \geq u \geq u_{\text{Ae2}} \\ \frac{\ell_2}{\ell_2 + \ell_3} & u < u_{\text{Ae2}} \end{cases}$$

The austenite-ferrite transformation is assumed to complete when either $\chi_f = 1$, or the Acm line is reached.

2. The maximum volume fraction of pearlite is similarly the results of the proeutectoid ferrite level rule calculation below the Ae1 line such that

$$\chi_p^{\max} = 1 - \chi_f^{\max}$$

3. The maximum volume fraction of bainite is taken to be the volume fraction of untransformed austenite at bainite start temperature, so that

$$\chi_b^{\max} = 1 - \chi_f - \chi_p.$$

Additionally, in the Kirkaldy-Venugopalan theory, the pearlite and bainite transformations are presumed to be additive, so that the bainite transformation proceeds from the same pearlite-austenite interface, if pearlite has been formed above the bainite start temperature. This implies that

$$\frac{\chi_p + \chi_b}{1 - \chi_f^{\max}} \leq 1.$$

4. The maximum volume fraction of martensite is taken to be the volume fraction of untransformed austenite at the martensite start temperature, so that

$$\chi_m^{\max} = 1 - \chi_f - \chi_p - \chi_b$$

In the current implementation, we restrict the calculations to simple mild steels, allowing simplifications to include only the influence terms for C, Mn and Si. The transformation temperatures for a given hypo-eutectoid C-Mn steel are calculated from the formulas proposed by Watt *et al* (15):

$$u_{\text{Ae}3} = 1185 - 203\sqrt{w\{\text{C}\}} + 44.7w\{\text{Si}\} - 30w\{\text{Mn}\}$$

$$u_{\text{Ae}2} = 1185 - 8486w\{\text{C}\}$$

$$u_{\text{Ae}1} = 996 - 10.7w\{\text{Mn}\} + 29w\{\text{Si}\}$$

$$u_{\text{bs}} = 929 - 58w\{\text{C}\} - 35w\{\text{Mn}\} - 75w\{\text{Si}\}$$

$$u_{\text{ms}} = 834 - 474w\{\text{C}\} - 33w\{\text{Mn}\},$$

where $w\{\circ\}$ denotes the weight percent of any given alloying element. For the Acm line, we use the expression proposed by Lee and Lee (70) :

$$u_{\text{Acm}} = 497.4 + 992.4w\{\text{C}\} - 465.1w\{\text{C}\}^2 + 7.6w\{\text{Mn}\} + 16.7w\{\text{Si}\}.$$

For the lever rule calculations necessary to determine the maximum equilibrium amounts of proeutectoid ferrite and pearlite, the Ae1, Ae3 and Acm expressions can be solved for the equilibrium weight percentage carbon at a given temperature. The

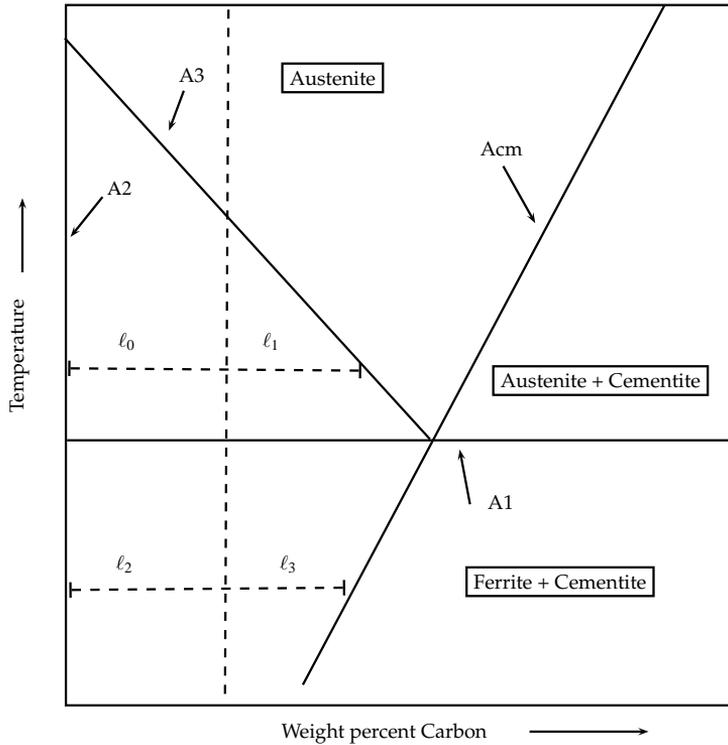


Fig 81. Simplified Iron-Carbon diagram showing lever arm calculations for proeutectoid steels above and below the A1 temperature.

effective carbon content in residual austenite as a result of carbon partitioning during the proeutectoid ferrite transformation is calculated using the method described in Section 3.5.

3.5 Carbon partitioning calculations during the pro-eutectoid ferrite transformation

For an Iron-Carbon binary alloy undergoing an austenite-ferrite phase transformation, molar conservation requires

$$n\{\text{Fe}\}_f + n\{\text{Fe}\}_a = n\{\text{Fe}\}_0 \quad (337a)$$

$$n\{\text{C}\}_\alpha + n\{\text{C}\}_a = n\{\text{C}\}_0, \quad (337b)$$

where $n\{x\}$ is the number of moles of a give species in a reference mass, and the 0 subscript denotes the homogenous austenised state before the commencement of transformation.

We define the solid state volume fraction of the two phases present as χ_f and χ_a and require that $\chi_f + \chi_a = 1$. The molar concentrations of each species in each phase of the solid solution are defined as

$$\begin{aligned} [\text{Fe}]_f &= \frac{n\{\text{Fe}\}_f}{V_f} & [\text{Fe}]_a &= \frac{n\{\text{Fe}\}_a}{V_a} \\ [\text{C}]_f &= \frac{n\{\text{C}\}_f}{V_f} & [\text{C}]_a &= \frac{n\{\text{C}\}_a}{V_a}, \end{aligned}$$

where V_j is the volume fraction of a given phase in the control volume. Using these definitions and assuming constant volume, the molar balance can be transformed into a concentration balance as

$$\chi_f [\text{Fe}]_f + (1 - \chi_f) [\text{Fe}]_a = [\text{Fe}]_0 \quad (338a)$$

$$\chi_f [\text{C}]_f + (1 - \chi_f) [\text{C}]_a = [\text{C}]_0. \quad (338b)$$

As the phase transformation proceeds, the concentration of Iron and Carbon in the residual untransformed austenite can then calculated by

$$[\text{Fe}]_a = \frac{[\text{Fe}]_0 - \chi_f [\text{Fe}]_f}{(1 - \chi_f)} \quad (339a)$$

$$[\text{C}]_a = \frac{[\text{C}]_0 - \chi_f [\text{C}]_f}{(1 - \chi_f)}. \quad (339b)$$

The molar fraction of Carbon in the residual untransformed austenite can then be calculated as

$$\eta\{C\}_a = \frac{[\text{C}]_a}{[\text{Fe}]_a + [\text{C}]_a} \quad (340)$$

and the equivalent weight percent Carbon in the residual untransformed austenite is then given by

$$w\{C\}_a = 100 [\text{C}]_a \frac{m\{C\}}{\rho_a}, \quad (341)$$

where $m\{C\}$ is the atomic mass of carbon.

3.6 Computer Implementation of the Kirkaldy-Venugopalan transformation model

The austenite decomposition model described by Kirkaldy and Venugopalan in reference (32) is based on a sigmoidal rate law of the general form

$$\phi(u, \chi) = \beta(u, \chi) \left[\chi^{\alpha(1-\chi)} (1-\chi)^{\alpha\chi} \right], \chi \in (0, 1) \quad (342)$$

with α usually taken as 2/3 and $\beta(u, \chi)$ given by one of three reaction dependent formula already described in Appendix 3.4. In the present work, this is recast as an ordinary partial differential equation, Equation (344).

$$\frac{\partial \chi}{\partial t} = \phi(\chi), t > 0, \phi : \mathfrak{R}^N \rightarrow \mathfrak{R}^N \quad (343)$$

$$\chi(0) = 0, t = 0. \quad (344)$$

A typical first approach to integrating this expression would be to use an explicit Runge-Kutta scheme, such as the step adaptive embedded methods discussed by Dormand and Prince (71) and Shampine (65). As was discussed in Chapter 2, the objective is to solve both microstructure evolution and heat conduction simultaneously using the same integration scheme. The stiff properties of the conduction problem demand the use of an implicit integrator, therefore we seek an implicit solution method for (344) which is compatible with both the heat and microstructure evolution problems.

To gain some insight into the stability of this ODE, it is instructive to examine the properties of ODE function (Equation 344), and its first derivative, given by

$$\frac{\partial \phi}{\partial \chi} = \phi(u, \chi) \frac{\partial \beta}{\partial \chi} + \beta(u, \chi) \left[a \chi^{\alpha(1-\chi)-1} (1-\chi)^{\alpha\chi-1} \right] \left[(\chi^2 - \chi) (\ln \chi - \ln(1-\chi) - 2\chi + 1) \right]. \quad (345)$$

For the proeutectoid ferrite and pearlite reactions, β depends only on temperature and composition, and $\partial \beta / \partial \chi = 0$. For the bainite reaction, β takes the form

$$\beta = \frac{a}{c \exp(-b\chi^2)} \quad (346)$$

with a derivative

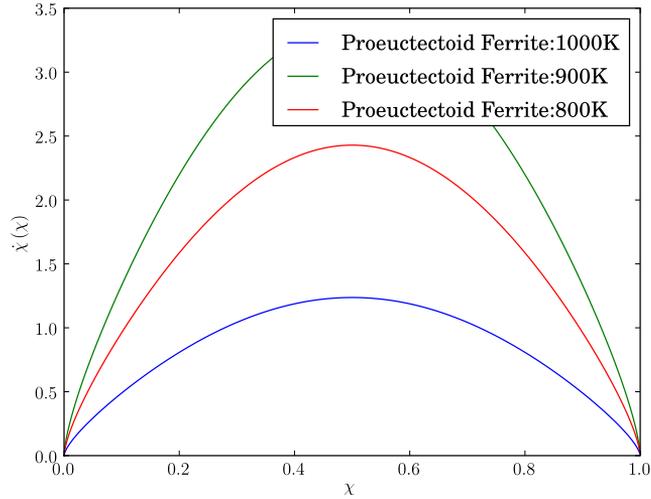


Fig 82. Austenite decomposition model proposed by Kirkaldy and Venugopalan, Equation (342), computed for the proeutectoid-ferrite case on $\chi \in (0, 1)$.

$$\frac{\partial \beta}{\partial \chi} = \frac{2ab\chi}{c \exp(-b\chi^2)}, \quad (347)$$

where a , b , and c are composition and temperature dependent parameters.

Figure 82 shows the ODE function plotted for three different temperatures using β values for the pro-eutectoid ferrite reaction, Equation 329, with a composition corresponding to an AISI-1016 mild steel. The ODE function for the bainite case of the same steel is shown in Figure 83.

Although at first inspection the rate equation looks relatively benign, in fact, it exhibits severe non-linear local behaviour at both the lower and upper ends of the feasible solution domain $\chi \in (0, 1)$. This property is illustrated in Figures 84 and 85, which shows Equation (345) plotted under the same two sets of conditions. The very rapid changes in derivative at each end of the function greatly hinders the convergence properties of derivative based solvers, such as the Newton-Raphson method, because the local slopes about small and large values of χ can become near orthogonal. In such cases, the solver will tend to oscillate about the true solution without converging at any useful rate, if at all. This, in turn, requires any solver to use excessively small step sizes for integration, and for the same reasons outlined in the discussion of the non-linear heat equation, we classify this ODE as mildly stiff in its own right, if only at the extremities of the feasible solution domain.

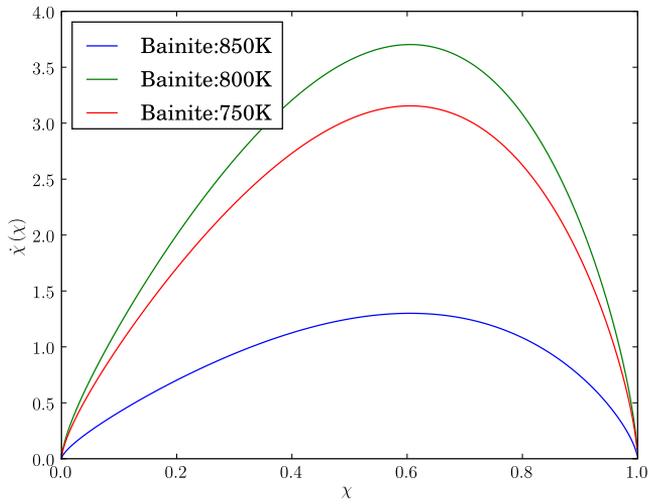


Fig 83. Austenite decomposition model proposed by Kirkaldy and Venugopalan, Equation (342), computed for the bainite case on $\chi \in (0, 1)$.

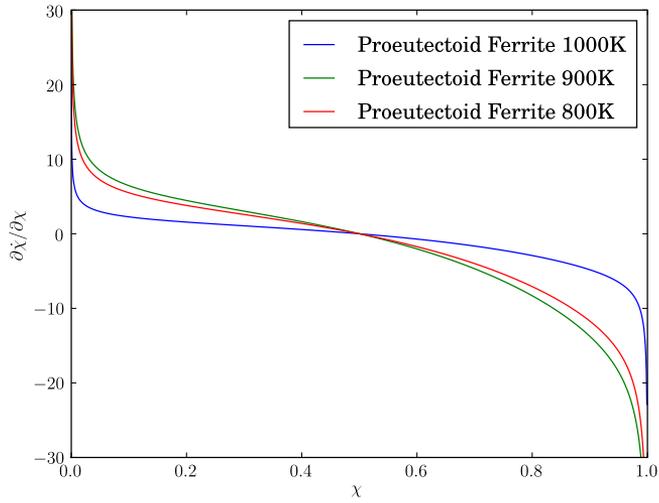


Fig 84. First derivative of the austenite decomposition model proposed by Kirkaldy and Venugopalan, Equation (345), computed for the proeutectoid ferrite case on $\chi \in (0, 1)$.

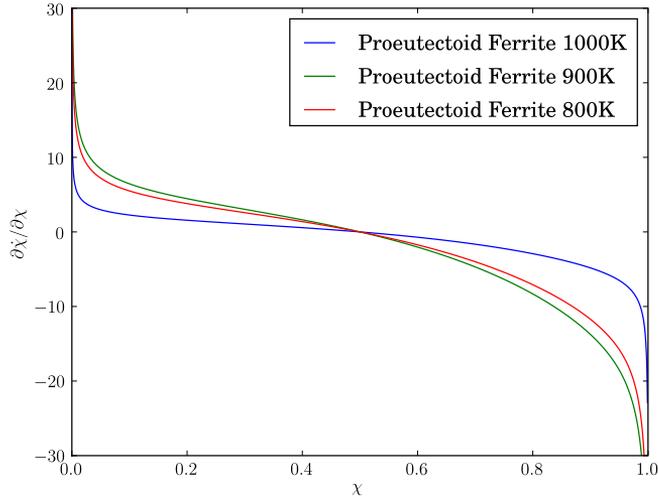


Fig 85. First derivative of the austenite decomposition model proposed by Kirkaldy and Venugopalan, Equation (345), computed for the bainite case on $\chi \in (0, 1)$.

As was discussed in some detail in Appendix 2, this difficulty can be circumvented by using an implicit Runge-Kutta scheme solved using a solution procedure based on a modified version of the `rtsafe` algorithm described by Press et. al. (53, §9). Here, the basic approach is to maintain a working interval which brackets the function root. The Newton-Raphson solver is then restricted to operating only within that interval. If a given Newton-Raphson update step would move the solution estimate outside the bracketing interval, it is rejected in favour of a simple bisection step. While the convergence properties of the bisection step are considerably inferior to the Newton-Raphson method, the bisection step at least guarantees that the bracketing interval over which the root search takes place narrows at every iteration, and the solution will not diverge.

Introducing a bisection stage to the solver brings with it the need to compute an initial bracketing interval to start the solution procedure. In the present implementation, this is done by exploiting several physical assumptions about the austenite decomposition process itself, namely that irreversibility requires

$$\chi_{i+1} \geq \chi_i, \forall \chi \quad (348)$$

and the physical limits of the reaction itself imply

$$(\chi_{i+1} - \chi_i) \leq (1 - \chi_i), \forall \chi. \quad (349)$$

This allows a bracketing interval to be established at all times, such that

$$(\chi_{i+1} - \chi_i) \in [0, 1 - \chi_i], \forall \chi. \quad (350)$$

While the interval defined by Equation (350) is guaranteed to bracket the solution, it is the effective maximum admissible interval on which the solution exists. This is safe, but not necessarily conducive to rapid convergence, particularly in cases where the bisection search must be relied to find the root. To somewhat improve this situation, we adopt a bracketing strategy which seeks to find a usefully small starting interval without excessive additional computation. We do this by starting with an interval $[\chi_i, \chi_u]$, which is a small fraction of that defined by Equation (350), and increase it logarithmically until it satisfies

$$r(\chi_i) * r(\chi_u) \leq 0, \quad (351)$$

where $*$ denotes the Hadamar product. Any set of conditions which cannot be bracketed in this fashion are considered to be insoluble and regarded as an algorithm failure. While we have no mathematical proof that this possibility can be excluded, we considered it unlikely enough to neglect, at least in scope of the present work. The pseudocode for this is shown in Figure 3

An unanswered question in the implementation of an implicit scheme for the Kirkaldy-Venugopalan model was which accuracy order and number of stages amongst the three DIRK schemes would be the best suited to solving this ODE. In order to investigate this, test solutions were made using the same conditions described earlier in this section with each of the DIRK(2, 2), DIRK(3, 2) and DIRK(4, 3) integrators and a reference fourth-fifth order, step adaptive Runge-Kutta-Fehlberg explicit intergrator described by Dormand and Prince (71). All computations were made simultaneously as a vector valued problem, with relative and absolute error tolerances set at 2.5×10^{-5} . The results this experiment is summarised in Tables 6 and 7, which shows the relative computation time, number of integrator steps and RMS deviation from the reference explicit solution for each integrator for the proeutectoid ferrite and bainite cases.

Interesting, the worst performing of the three conditions for the ferrite reaction (1000 K, 900 K, and 800 K) is the highest temperature case. At first inspection this is a somewhat surprising result, because as shown in Figure 82, this case has the slowest kinetics and the magnitude of the resulting ODE is smallest across the entire computational domain. The reason this case proves more challenging to the implicit integrators lies in the behaviour of the ODE derivative at small and large values of χ .

Algorithm 3 Solution bracketing algorithm for the Kirkaldy-Venugopalan transformation model

```

1:  $a \leftarrow -2$ 
2:  $b \leftarrow 0$ 
3:  $r_l \leftarrow f(\chi_i, \chi_i)$ 
4:  $\eta \leftarrow 10^a$ 
5:  $\Delta\chi_u \leftarrow (1 - \chi_0)$ 
6: for  $j \in \{1 \dots n\}$  do
7:    $p \leftarrow a + (j/n)(b - a)$ 
8:    $\eta \leftarrow 10^p$ 
9:    $\chi_u \leftarrow \chi_i + \eta\Delta\chi_u$ 
10:   $r_u \leftarrow f(\chi_i, \chi_u)$ 
11:  if  $r_l * r_u \leq 0$  then
12:    break
13:  end if
14: end for

```

Table 6. Relative performance of the implicit integrators for the sample ferrite transformation reaction cases shown in Figure 82.

(a) Computational efficiency

Scheme	Relative run time	Steps	Step Failures
DIRK(2,2)	1	252	53
DIRK(3,2)	0.3777	57	14
DIRK(4,3)	0.4809	59	22

(b) RMS deviation from reference RK4/5 solution

Scheme	1000 K	900 K	800 K
DIRK(2,2)	0.01181	0.00863	0.00956
DIRK(3,2)	0.00791	0.00784	0.00786
DIRK(4,3)	0.00856	0.00855	0.00859

Table 7. Relative performance of the implicit integrators for the sample bainite transformation reaction cases shown in Figure 83 .

(a) Computational efficiency			
Scheme	Relative run time	Steps	Step Failures
DIRK(2,2)	1	271	72
DIRK(3,2)	0.4435	61	17
DIRK(4,3)	0.5366	59	23

(b) RMS deviation from reference RK4/5 solution			
Scheme	1000 K	900 K	800 K
DIRK(2,2)	0.01854	0.01309	0.01384
DIRK(3,2)	0.01267	0.01221	0.01239
DIRK(4,3)	0.01392	0.01330	0.01348

This is clearly illustrated in Figure 84, which shows how rapidly the ODE derivative at 1000 K changes as χ approaches either 0 or 1. So although this case looks to be the most straightforward to integrate because of the slow reaction rate, it turns out to be the most difficult because of localized gradient changes and the difficulties they present to the iterative Newton solver.

Figures 91(a) and 91(b) show the average timestep, calculated as the time between output data points divided by the number of timesteps calculated during that interval for the proeutectoid ferrite and bainite examples. We see that the DIRK(3,2) scheme is the most efficient in both cases, accommodating the largest step sizes through most of the integration. In contrast, the two stage DIRK(2,2) scheme requires almost an order of magnitude smaller step size, despite being of the same truncation order accuracy. Overall, the bainite reaction is somewhat stiffer than the ferrite reaction, requiring smaller average timestep sizes to solve to the same tolerance.

In both of these examples, the DIRK(4,3) integrator is marginally less accurate than the DIRK(3,2) scheme, as well as computationally more expensive. As discussed by Alexander and Hosea and Shampine (62, 61), this scheme has superior stability properties for stiff ODEs compared to the two alternatives used here, but those are not greatly tested in this example and, therefore, there is little reason to prefer it over the DIRK(3,2) integrator.

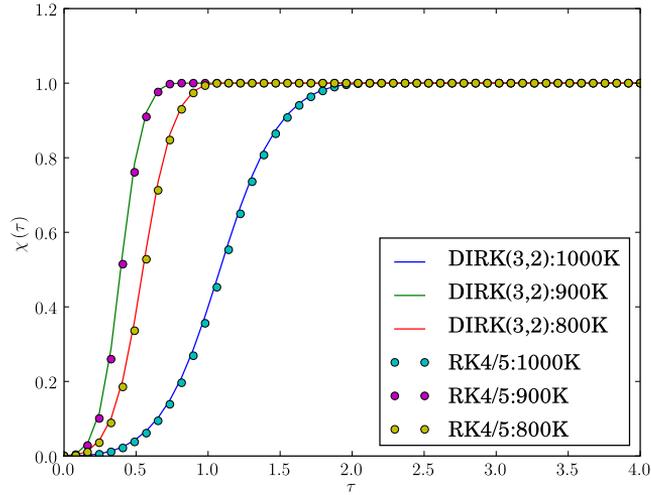


Fig 86. Austenite-proeutectoid ferrite decomposition model proposed by Kirkaldy and Venugopalan, Figure 82, integrated with the DIRK(3,2) implicit scheme and a reference explicit RK4 scheme described by Dormand and Prince (71) integrated on $\tau \in (0,4)$.

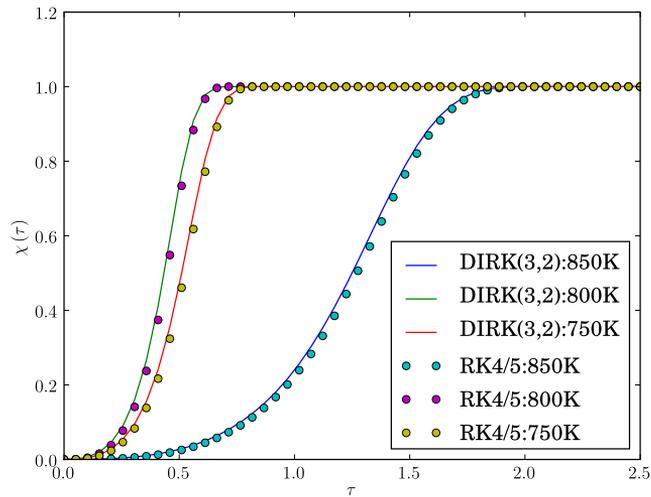


Fig 87. Austenite-bainite ferrite decomposition model proposed by Kirkaldy and Venugopalan, Figure 82, integrated with the DIRK(3,2) implicit scheme and a reference explicit RK4 scheme described by Dormand and Prince (71) integrated on $\tau \in (0,2.5)$.

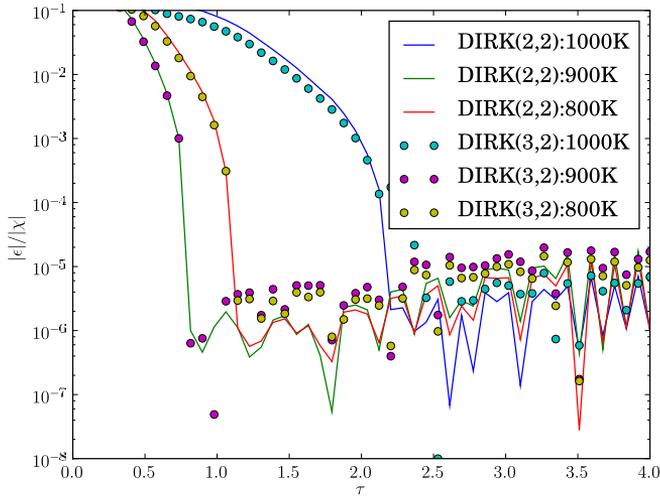


Fig 88. Relative error differences between the DIRK22 and DIRK32 integrators for the ferrite rate equation cases shown in Figure 82. Relative and absolute error tolerances set to 2.5×10^{-5} .

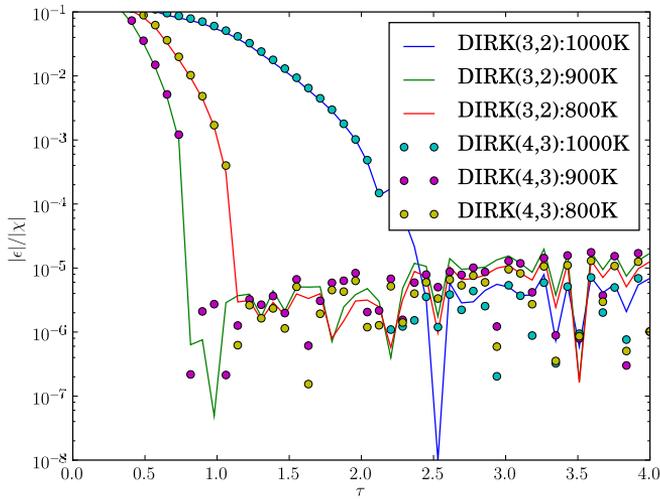


Fig 89. Relative error differences between the DIRK32 and DIRK43 integrators for the ferrite rate equation cases shown in Figure 82. Relative and absolute error tolerances set to 2.5×10^{-5} .

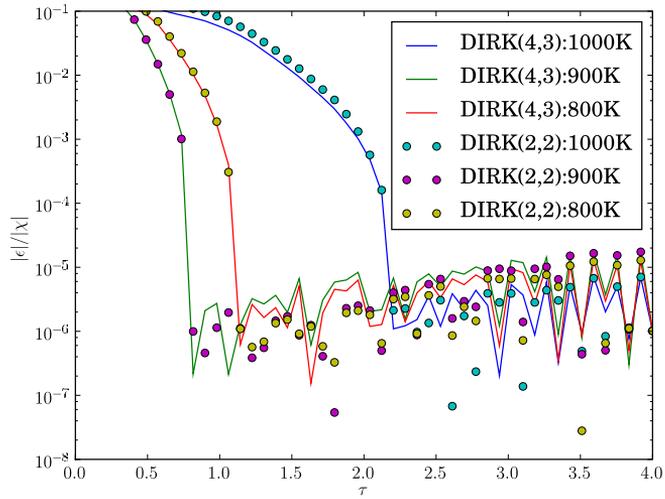
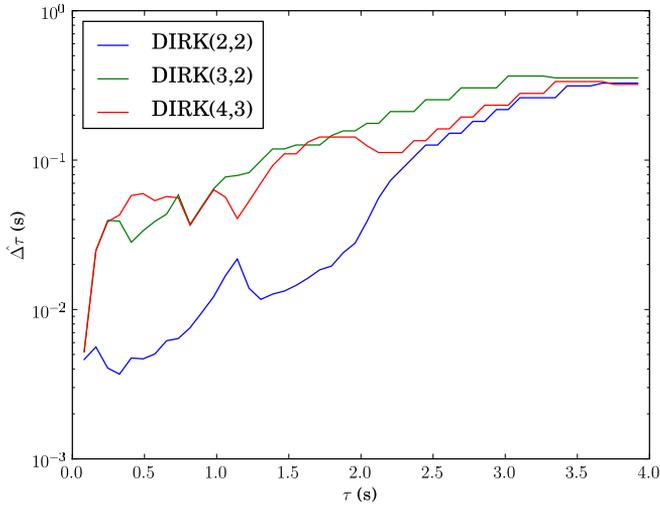
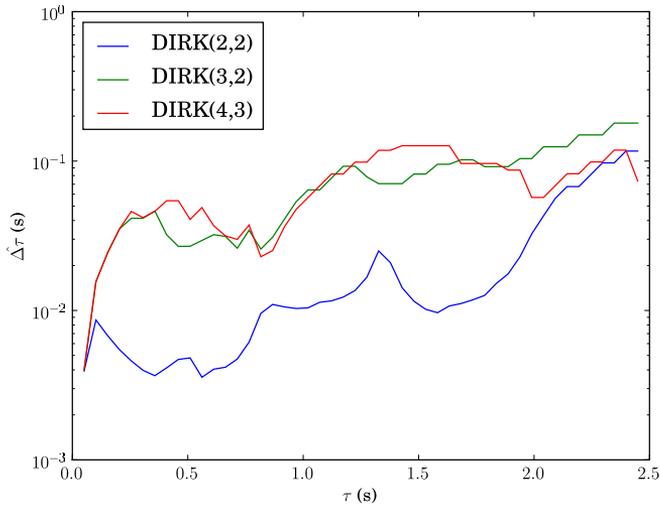


Fig 90. Relative error differences between the DIRK43 and DIRK22 integrators for the ferrite rate equation cases shown in Figure 82. Relative and absolute error tolerances set to 2.5×10^{-5} .



(a) Proeutectoid ferrite rate equation cases shown in Figure 82.



(b) Bainite rate equation cases shown in Figure 83.

Fig 91. Comparison of average time step size required by DIRK22, DIRK32 and DIRK43 integrators. Relative and absolute error tolerances set to 2.5×10^{-5} .

Appendix 4 Sample simulation program listing

```
1  #!/usr/bin/env python
2
3  import gc
4  import numpy as np
5  import math
6
7  # Grid - 32x32 2D grid with h = 0.25mm
8  h = 0.25e-3
9  M=32
10 N=32
11 celln=M*N
12 gridshape=(M,N)
13 grid=(gridshape,h)
14 xvals=(h/2) + h*np.arange(0,M)
15 yvals=(h/2) + h*np.arange(0,N)
16
17 idxx,idxy = np.indices(gridshape)
18 meshx = xvals[idxx.flatten()].reshape(gridshape)
19 meshy = yvals[idxy.flatten()].reshape(gridshape)
20
21 #
22 # initial conditions
23 #
24
25 # Temperature
26 uhat0 = 1250.
27 uhat1 = 1200.
28 r = np.sqrt(meshx**2. + meshy**2.)
29 uhat = uhat0 - (uhat0-uhat1)*r/np.max(r)
30 u0=uhat.flatten('F')
31
```

```

32  # Composition and microstructure
33  C = 0.45
34  Mn = 0.75
35  Si = 0.25
36  chem = (C,Mn,Si)
37
38  X=np.zeros((celln,4),order='F')
39  x0 = np.hstack((u0.reshape(celln,1),X))
40
41  # Problem integration time
42  t0 = 0.
43  t1 = 75.
44  nout = 400
45  tref0 = (t1-t0)/nout
46  tref = np.linspace(tref0,t1,nout,endpoint=True)
47
48  # Solver/Intergrator setup
49  import fdintegrator as fdi
50  import fdintegrator43 as fd43
51  import fd4
52  import fdbc2d
53
54  # Fourth order, 2D stencils and operators
55  [xstenb,ystenb,kstensb,Qstensb]=fd4.makeStens2D()
56  stensb=dict( stenwidth=len(xstenb),\
57              xsten=xstenb,ysten=ystenb, \
58              kstens=kstensb,Qstens=Qstensb)
59  fd2argsb = [grid,stensb]
60
61  # Spray cooling boundary condition -
62  # 2.0 l/m/m/s flow rate,
63  # 299K, 0.75 emissivity
64  bcfunc = fdbc2d.bcSpray(flux=2.0, uinf=299., emiss=0.75)
65
66  # Fourth Order spatial discretisation using DIRK43 scheme

```

```
67 run1b = fdi.solverrun2D(x0, t0, t1, grid, chem, \  
68     tref, fd4, bcfunc, fd43.DIRK43_heatscheme, fd2argsb)  
69  
70 # Perform the simulation run and write out the results to disk  
71 if __name__ == '__main__':  
72     run1b.run()  
73     run1b.save('runs/spray/sprayfd4dirk43.npz')  
74  
75 # vim:ts=4:sw=4:expandtab:number
```

372. Plekh, Maxim (2010) Ferroelectric performance for nanometer scaled devices
373. Lee, Young-Dong (2010) Wireless vital signs monitoring system for ubiquitous healthcare with practical tests and reliability analysis
374. Sillanpää, Ilkka (2010) Supply chain performance measurement in the manufacturing industry : a single case study research to develop a supply chain performance measurement framework
375. Marttila, Hannu (2010) Managing erosion, sediment transport and water quality in drained peatland catchments
376. Honkanen, Seppo (2011) Tekniikan ylioppilaiden valmistumiseen johtavien opintopolkujen mallintaminen — perusteena lukiossa ja opiskelun alkuvaiheessa saavutettu opintomenestys
377. Malinen, Ilkka (2010) Improving the robustness with modified bounded homotopies and problem-tailored solving procedures
378. Yang, Dayou (2011) Optimisation of product change process and demand-supply chain in high tech environment
379. Kalliokoski, Juha (2011) Models of filtration curve as a part of pulp drainage analyzers
380. Myllylä, Markus (2011) Detection algorithms and architectures for wireless spatial multiplexing in MIMO-OFDM systems
381. Muhos, Matti (2011) Early stages of technology intensive companies
382. Laitinen, Ossi (2011) Utilisation of tube flow fractionation in fibre and particle analysis
383. Lasanen, Kimmo (2011) Integrated analogue CMOS circuits and structures for heart rate detectors and other low-voltage, low-power applications
384. Herrala, Maila (2011) Governance of infrastructure networks : development avenues for the Finnish water and sewage sector
385. Kortelainen, Jukka (2011) EEG-based depth of anesthesia measurement : separating the effects of propofol and remifentanyl
386. Turunen, Helka (2011) CO₂-balance in the atmosphere and CO₂-utilisation : an engineering approach
387. Juha, Karjalainen (2011) Broadband single carrier multi-antenna communications with frequency domain turbo equalization

S E R I E S E D I T O R S

A
SCIENTIAE RERUM NATURALIUM

Senior Assistant Jorma Arhippainen

B
HUMANIORA

Lecturer Santeri Palviainen

C
TECHNICA

Professor Hannu Heusala

D
MEDICA

Professor Olli Vuolteenaho

E
SCIENTIAE RERUM SOCIALIUM

Senior Researcher Eila Estola

F
SCRIPTA ACADEMICA

Director Sinikka Eskelinen

G
OECONOMICA

Professor Jari Juga

EDITOR IN CHIEF

Professor Olli Vuolteenaho

PUBLICATIONS EDITOR

Publications Editor Kirsti Nurkkala

ISBN 978-951-42-9518-8 (Paperback)

ISBN 978-951-42-9519-5 (PDF)

ISSN 0355-3213 (Print)

ISSN 1796-2226 (Online)

