The copyright of this thesis rests with the University of Cape Town. No quotation from it or information derived from it is to be published without full acknowledgement of the source. The thesis is to be used for private study or non-commercial research purposes only.
MATHEMATICAL AND COMPUTATIONAL MODELLING OF THE DYNAMIC BEHAVIOUR OF DIRECT CURRENT PLASMA ARCS

Quinn G Reynolds

Thesis presented for the degree of Doctor of Philosophy
In the Department of Mathematics and Applied Mathematics
University of Cape Town
July 2009
Abstract

The problem of direct-current plasma arc behaviour, interaction, and dynamics is considered in the context of metallurgical DC arc furnace applications. Particular attention is paid to the transient flow behaviour of arc systems.

A mathematical formulation of the physics used to describe the arc system is presented, and includes the spatial and temporal evolution of fluid flow, heat transfer, and electromagnetism. A number of strong coupling effects are seen to exist between the various fields that interact to give rise to the plasma arc.

The model is then discretised using a finite difference approach on a regular cartesian grid in both two and three dimensions, with a special focus on robust stability, high resolution modelling, and high performance. Memory usage and optimisation, data storage and analysis, and parallelism regarding the numerical algorithm are also discussed.

A collection of results produced using the numerical model is then presented. These address a wide range of commonly encountered process and design variables and their effect on the numerical model's results. Where possible, quantitative and qualitative behaviour of the model is compared to the limited empirical data available. A number of novel effects and phenomena are seen in the dynamic behaviour of the DC plasma arc model for both single and multiple arc systems, many of which may lead to improved understanding, control, and manipulation of such systems where they occur in industrial applications.
Acknowledgements

This work would not have been possible without the assistance and enthusiasm of several people and institutions. The author wishes to gratefully acknowledge the following:

Mintek, in particular Mr Tom Curr, Mr Rodney Jones, and Mr Kabwika Bisaka, for providing funding, equipment, and the supportive environment necessary for this research.

The Department of Mathematics and Applied Mathematics at the University of Cape Town, in particular Professor Daya Reddy, for administering and supervising this work.

The Centre for High Performance Computing (CHPC) in Cape Town, in particular Dr Happy Sithole, Dr Jeff Chen, and the IT support staff at the Centre, for providing access to their facilities in order to troubleshoot and run high resolution 3D models.

Professor Hans Johnston, for communications and code samples helpful for understanding the Johnston-Liu primitive variable method.

Professor Stewart Zweben and Dr Max Karasik, for correspondence and permission to republish some of their arc photography results.

Dr Anton Lopis, for assistance with data transfers to and from the CHPC, and input on various technical aspects of the facilities there.
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Pg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>ii</td>
</tr>
<tr>
<td>Acknowledgements</td>
<td>iv</td>
</tr>
<tr>
<td>Contents</td>
<td>vi</td>
</tr>
<tr>
<td>Nomenclature</td>
<td>x</td>
</tr>
<tr>
<td>1. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Problem background and description</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Previous work</td>
<td>3</td>
</tr>
<tr>
<td>1.3 Desired advancements in the field achievable with this work</td>
<td>11</td>
</tr>
<tr>
<td>1.4 Overview of aims, and structure of the rest of this thesis</td>
<td>15</td>
</tr>
<tr>
<td>2. Governing equations</td>
<td>19</td>
</tr>
<tr>
<td>2.1 Flow models</td>
<td>19</td>
</tr>
<tr>
<td>2.2 Energy model</td>
<td>27</td>
</tr>
<tr>
<td>2.3 Electromagnetic field model</td>
<td>31</td>
</tr>
<tr>
<td>2.4 Coupling effects</td>
<td>36</td>
</tr>
<tr>
<td>2.5 Summary of governing equations</td>
<td>41</td>
</tr>
<tr>
<td>3. Numerical considerations</td>
<td>45</td>
</tr>
<tr>
<td>3.1 Spatial geometry and discretisation methods</td>
<td>45</td>
</tr>
<tr>
<td>3.2 Fluid flow models</td>
<td>50</td>
</tr>
<tr>
<td>3.3 Energy model</td>
<td>61</td>
</tr>
<tr>
<td>3.4 Electromagnetic field model</td>
<td>65</td>
</tr>
<tr>
<td>3.5 Fast Poisson solver</td>
<td>72</td>
</tr>
</tbody>
</table>
3.6 Geometric multigrid solver 79
3.7 Model algorithm discussion and parallelisation 87
3.8 Data management 99
3.9 Data visualisation 103

4 - 6. Results and discussion of modelling work

4. Test Cases 111

4.1 Introduction to test cases 111
4.2 Test case 1 - evolution of a steady laminar jet 111
4.3 Test case 2 - evolution of an unsteady jet 118
4.4 Test case 3 - electromagnetic solver tests 120
4.5 Test case 4 - parallel performance 127

5. Model Cases (2D) 131

5.1 Description of two-dimensional models and cases studied 131
5.2 Base case model 135
5.3 Region geometry and arc length effects 150
5.4 Arc current effects 155
5.5 Cathode spot current density effects 161
5.6 Sensitivity analysis for constant physical properties 167
5.7 Argon gas 174
5.8 Thermal boundary conditions 179

6. Model Cases (3D) 189

6.1 Description of three-dimensional models and cases studied 189
6.2 Base case model 194
6.3 Electrode separation effects 207
6.4 Arc current asymmetry effects 218
6.5 High resolution cases - triple arc 226
6.6 High resolution cases - twin arc 233
7. Discussion of results, conclusions and recommendations 241

Appendix 1 - Induced current calculation 253
Appendix 2 - Table of data for $\sigma$ and $Q$, air and argon 254
Appendix 3 - Numerical testing of 2D Base Case model 256

References 259
# Nomenclature

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description and units</th>
</tr>
</thead>
<tbody>
<tr>
<td>$j, j_o, j_z$</td>
<td>Electric current density vector and components, A/m²</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>Electrical conductivity of plasma, 1/Ωm</td>
</tr>
<tr>
<td>$\phi$</td>
<td>Electric potential, V</td>
</tr>
<tr>
<td>$B, B_o, B_z$</td>
<td>Magnetic field vector and components, T</td>
</tr>
<tr>
<td>$\mu_0$</td>
<td>Magnetic permeability of free space, N/A²</td>
</tr>
<tr>
<td>$A, A_o, A_z$</td>
<td>Magnetic vector potential and components, Tm</td>
</tr>
<tr>
<td>$\tau, n$</td>
<td>Unit tangent and normal vectors at domain boundaries</td>
</tr>
<tr>
<td>$\Gamma$</td>
<td>Boundary of calculation domain</td>
</tr>
<tr>
<td>$r_a$</td>
<td>Fully-developed arc radius in Bowman shape model, m</td>
</tr>
<tr>
<td>$r_i$</td>
<td>Initial arc radius (at electrode surface) in Bowman shape model, m</td>
</tr>
<tr>
<td>$L_d$</td>
<td>Equivalent arc length (distance from electrode to anode), m</td>
</tr>
<tr>
<td>$I_a$</td>
<td>Arc current, A</td>
</tr>
<tr>
<td>$j_s$</td>
<td>Cathode spot current density, A/m²</td>
</tr>
<tr>
<td>$L$</td>
<td>Characteristic length scale, m</td>
</tr>
<tr>
<td>$\delta_l$</td>
<td>Numerical grid step size, m</td>
</tr>
<tr>
<td>$Re$</td>
<td>Reynolds number</td>
</tr>
<tr>
<td>$v_{max}, v_{rms}$</td>
<td>Maximum or root mean square velocity used in $Re$, m/s</td>
</tr>
<tr>
<td>$\nu$</td>
<td>Dynamic viscosity, m²/s</td>
</tr>
<tr>
<td>$\mu$</td>
<td>Kinematic viscosity, Pa.s</td>
</tr>
<tr>
<td>$\rho$</td>
<td>Density, kg/m³</td>
</tr>
<tr>
<td>$v, v_x, v_y, v_z$</td>
<td>Velocity vector and components, m/s</td>
</tr>
<tr>
<td>$t$</td>
<td>Time, s (independent variable)</td>
</tr>
<tr>
<td>$x, y, z, \hat{e}_x, \hat{e}_y, \hat{e}_z$</td>
<td>Space coordinates, m, and unit vectors (independent variables)</td>
</tr>
<tr>
<td>$P$</td>
<td>Pressure, Pa</td>
</tr>
<tr>
<td>$p$</td>
<td>Reduced pressure, Pa.m³/kg</td>
</tr>
<tr>
<td>$F, F_o, F_x, F_z$</td>
<td>Force vector acting on fluid and components, N/m³</td>
</tr>
<tr>
<td>$\psi, \Psi$</td>
<td>Scalar and vector potential for fluid flow, m²/s</td>
</tr>
<tr>
<td>$\omega, \phi$</td>
<td>Scalar and vector vorticity, 1/s</td>
</tr>
<tr>
<td>$a, a_x, a_y, a_z$</td>
<td>Auxiliary vector and components, m/s</td>
</tr>
<tr>
<td>$\theta$</td>
<td>Gauge variable, m²/s</td>
</tr>
<tr>
<td>$h$</td>
<td>Enthalpy, J/kg</td>
</tr>
<tr>
<td>Symbol, cont'd</td>
<td>Description and units, cont'd</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td>$T$</td>
<td>Temperature, K</td>
</tr>
<tr>
<td>$C_p$</td>
<td>Heat capacity, J/kg.K</td>
</tr>
<tr>
<td>$\kappa$</td>
<td>Thermal conductivity, W/mK</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Thermal diffusivity, m$^2$/s</td>
</tr>
<tr>
<td>$Q$</td>
<td>Energy source/sink, W/m$^3$</td>
</tr>
<tr>
<td>$Q_r$</td>
<td>Energy loss due to thermal radiation, W/m$^3$</td>
</tr>
<tr>
<td>$A_{RD}$</td>
<td>Constant in Richardson-Dushmann equation, A/m$^2$K$^2$</td>
</tr>
<tr>
<td>$e_c$</td>
<td>Electron charge, C</td>
</tr>
<tr>
<td>$W_f$</td>
<td>Work function, eV</td>
</tr>
<tr>
<td>$k_B$</td>
<td>Boltzmann constant, J/K</td>
</tr>
<tr>
<td>$X, Y, Z$</td>
<td>Maximum extent of calculation domain in x, y, and z directions, m</td>
</tr>
<tr>
<td>$i, j, k$</td>
<td>Indices for grid variables in each coordinate direction</td>
</tr>
<tr>
<td>$I, J, K$</td>
<td>Maximum index of grid variables in each coordinate direction</td>
</tr>
<tr>
<td>$\delta x$, $\delta y$, $\delta z$</td>
<td>Discrete grid step sizes in x, y, and z directions, m</td>
</tr>
<tr>
<td>$\delta t$</td>
<td>Discrete time-step size, s</td>
</tr>
<tr>
<td>$u$</td>
<td>Arbitrary grid/field variable used for various examples</td>
</tr>
<tr>
<td>$v_a$</td>
<td>Velocity in one-dimensional convection-diffusion equation, m/s</td>
</tr>
<tr>
<td>$C_1, C_2$</td>
<td>Numerical constants used in $\delta t$ calculation</td>
</tr>
<tr>
<td>$T'$</td>
<td>Intermediate temperature used in fractional step algorithm, K</td>
</tr>
<tr>
<td>$a_{E}, a_{W}, a_{N}, a_{S}, a_{R}, a_b$</td>
<td>Variable coefficients for electric potential equation, 1/Ωm$^3$</td>
</tr>
<tr>
<td>$a, b, c, d$</td>
<td>General coefficients, used in Poisson equation solver description</td>
</tr>
<tr>
<td>$S$</td>
<td>General source term, used in Poisson equation solver description</td>
</tr>
<tr>
<td>$U_y$</td>
<td>Gradient at boundary, used in Poisson equation solver description</td>
</tr>
<tr>
<td>$m$</td>
<td>Grid level counter, used in multigrid solver description</td>
</tr>
<tr>
<td>$R$</td>
<td>Residual, used in multigrid solver description</td>
</tr>
<tr>
<td>$w$</td>
<td>Solution correction, used in multigrid solver description</td>
</tr>
<tr>
<td>$u_r$</td>
<td>Arbitrary grid/field variable, spatially sampled</td>
</tr>
<tr>
<td>$u_f$</td>
<td>Arbitrary grid/field variable, temporally sampled</td>
</tr>
<tr>
<td>$\Delta t_r$</td>
<td>Constant time step size between frames of an animated visualisation, s</td>
</tr>
<tr>
<td>$A_f$</td>
<td>Constant in source term for jet flow test case, N/m$^3$</td>
</tr>
<tr>
<td>$x_s, y_f$</td>
<td>Constants in source term for jet flow test case, m</td>
</tr>
<tr>
<td>$t_E, t_s, t_P$</td>
<td>Various code execution times: total, serial, and parallel, s</td>
</tr>
<tr>
<td>$n_p$</td>
<td>Number of CPUs or equivalent processors used</td>
</tr>
<tr>
<td>$A_{io}$</td>
<td>Proportionality constant for multithreading overhead</td>
</tr>
<tr>
<td>Symbol, cont'd</td>
<td>Description and units, cont'd</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td>$T_{\text{electrode}}$, $T_{\text{anode}}$, $T_{\text{wall}}$</td>
<td>Boundary values of temperature on electrode, anode, and wall, K</td>
</tr>
<tr>
<td>$q$</td>
<td>Energy flux at boundary, W/m$^2$</td>
</tr>
<tr>
<td>$\Gamma_L$, $\Gamma_M$, $\Gamma_N$</td>
<td>Surface boundary regions in 3D model</td>
</tr>
<tr>
<td>$x_e$, $y_e$</td>
<td>Position of electrodes and cathode spots in 3D model, m</td>
</tr>
<tr>
<td>$R_a$</td>
<td>Curvature of arcs in empirical arc deflection model, m</td>
</tr>
<tr>
<td>$A_a$</td>
<td>Proportionality constant in empirical arc deflection model</td>
</tr>
<tr>
<td>$s_e$</td>
<td>Separation between electrodes in empirical arc deflection model, m</td>
</tr>
<tr>
<td>$a_s$</td>
<td>Separation between arc columns in empirical arc deflection model, m</td>
</tr>
</tbody>
</table>
Chapter 1 - Introduction

1.1 Problem background and description

With increasing economic and environmental pressures being brought to bear on the mining and metallurgical industries of the world, the direct current arc furnace (DCAF) is becoming a more attractive option for many pyrometallurgical processes. It can offer a number of advantages over the traditional unit operations of high temperature metallurgy such as fossil-fuel fired furnaces, AC electrical furnaces, and gas-blown converters.

Research and development into the use of the DCAF as a cleaner and more effective replacement for existing processes is a major area of study, as is the development of original processes for the treatment of new materials. Large scale pilot-plant work has been successfully performed on Ferrochrome, Ferronickel, Magnesium, Titanium Dioxide, Platinum Group Metals, Zinc, and many others\textsuperscript{1–9}.

![Figure 1: DCAF schematic](image)

The DCAF (see Figure 1) typically consists of a cylindrical containment vessel, lined with cooling elements and refractory materials. A single electrode made of graphite enters the vessel from above.
Raw feed material is fed into the vessel, usually via ports in the roof, and is heated to the point of melting by electrical power input to the furnace. Chemical reductants, introduced concurrently with the feed, are then able to react very rapidly with the molten material to produce the desired product.

The vessel serves to contain both the molten process material, and to some degree the thermal energy of the furnace. The furnace’s direct-current power supply is connected to the base of the vessel, in contact with the molten bath, and to the electrode.

The principal energy source of the furnace is a direct-current plasma arc, which is struck between the end of the graphite electrode and the surface of the electrically-conductive molten bath. Working along the same principles as arc welding equipment, the plasma arc functions by raising the temperature of the gas between the electrode and the molten bath via ohmic, or resistive, electrical heating. Once the temperature of the gas reaches a certain critical point (around 5000K) its constituent molecules and atoms begin to ionise into positively charged ions and negatively charged electrons, giving a neutral but very strongly-conductive plasma gas. This conducting material permits electrical current to pass from the furnace electrode through to the furnace bath and complete the circuit. Energy escapes from the arc via a multitude of mechanisms, most importantly thermal radiation from the hot plasma gas, and convection to the molten bath surface. Since much of this energy is delivered to a localised area directly beneath the arc, it is a very efficient way of heating the process material.

Flow of the plasma gas in the arc column is driven very strongly by electromagnetic forces, and tends to be directed downward in a jet from the electrode toward the molten bath due to the electrical geometry of the furnace. The velocities within the plasma arc are dependent mainly on the current delivered to the furnace.

An understanding of the fundamental behaviour of DC plasma arcs is thus extremely valuable, as they are central to the operation of the DCAF. Even at pilot plant scale it is an extremely high-intensity environment, operating at elevated temperatures (upwards of 15,000°C) and velocities (hundreds of m/s). Of particular interest is the dynamic behaviour of the arc, and its interaction with other arcs in the case of multiple-electrode furnaces.

Due to the extreme environment in the immediate vicinity of the arc, performing measurements and experimentation on the system is very challenging, and generally yields only partial or
unsatisfactory results. With this in mind, development of a mathematical and numerical model of the physics of the system is the best way to approach the problem of studying the key behaviours associated with the DC plasma arc.

1.2 Previous work

The theoretical study of electric arcs has been actively pursued since the 1950's. Early work focused primarily on empirical descriptions of the arc and analogies to gas jets, and has led to a number of useful engineering approximations still in wide use today.

With the advent of sufficiently powerful and affordable digital computers in the 1970's and 1980's a paradigm shift towards a more fundamental approach to the modelling of the plasma arc occurred, using approximated forms of the governing partial differential equations of fluid flow and energy transfer and solving them numerically to produce the temperature and velocity fields of the arc jet.

Much refinement of this latter method has taken place since, but it is still ultimately the same approach used today for the in-depth study of plasma arcs.

Analytical and semi-empirical studies of the DC plasma arc

The earliest models developed to understand the behaviour of DC plasma arcs were derived from experimental (usually photographic) observations. These led to a view of the arc as a column of plasma, gas that has been heated by the passage of electric current through it to the point at which it begins to ionise and become electrically conductive.

Maecker\textsuperscript{10} was one of the first investigators to consider the plasma arc from a theoretical standpoint, based on extensive experimental studies. A key observation was that the fundamental operating principle of the arc is related to the increased current density near the arc attachment zone on the graphite electrode - the “cathode spot”. This effect is shown in Figure 2.

As the electric current passes through the cathode spot and into the body of the arc, the constriction that occurs results in a self-induced magnetic compression effect that vigorously draws material into the region around the spot, and hence by conservation of mass produces a downward jet leading away from the surface of the electrode.
This cathode jet phenomenon is critical to both the formation of plasma arcs and their dynamic behaviour. The structure of the arc column in its entirety is to a large degree governed by the fluid flow and electromagnetic behaviour near the cathode spot, and is very sensitive to small inconsistencies - these can lead to instability and rapid motion of the arc, or even extinguishment.

Further study of the arc's properties are made possible if the shape of the arc column is known. This approach built upon and succeeded Maecerker's work, with workers such as Bowman, Stenkvist, and others developing and refining empirical shape models based on available experimental evidence. Bowman's work is the most enduring, and describes the shape of the arc column as a function of its length and current with a simple equation fitted to data in the range from 1 to 10kA. That is,

\[ \frac{r_a}{r_k} = 3.2 - 2.2 \exp \left( -\frac{L_A}{5 r_k} \right) \]  

(1)

where \( r_a \) is the radius of the arc column, \( L_A \) is its length, and \( r_k \) is the cathode spot radius, defined by

\[ r_k = \sqrt{\frac{I_a}{\pi j_k}}. \]  

(2)
$I_a$ is the current carried by the arc, and the cathode spot current density $j_s$ is assumed to be a constant. This is due to the fact that the electrode material is usually graphite, which has a sublimation temperature of approximately 4100K. As the plasma arc is generally much, much hotter than this, it is reasonable to assume that the cathode spot itself will be at a fixed temperature, and hence be capable of sustaining a fixed current density related to that temperature by the principles of thermionic emission. A schematic of the shape of the arc is shown in Figure 3.

*Figure 3: The Bowman arc's empirical shape*

Once the empirical shape of the arc is available, it is possible to produce related expressions for a number of variables of engineering interest, such as the voltage of the arc as a function of length and current, the velocity and temperature profiles through the arc, the thrust generated by the arc jet, and so forth - Bowman\textsuperscript{11} summarises the details of many of these calculations. Figure 4 shows an example of this, a calculation of the relationship between the arc's voltage and its length and current.

Many such empirical relationships are still in current use. Reynolds and Jones\textsuperscript{12} used semi-empirical models of the DCAF electrical system based on Bowman's work to perform data reconciliation and
scale up calculations. Simple channel models of the plasma arc are frequently used in AC studies, for example Soevarsdottir et al\textsuperscript{13}, and the more complex models of Sawicki and Krouchinin\textsuperscript{14}. Meng and Irons\textsuperscript{15} considered a number of empirical AC and DC arc models, and performed extensive comparison to industrial-scale experimental data.

Figure 4: Empirical voltage-arc length relationship for various currents

Experimental and theoretical studies of arc deflection, interaction, and instability have also been performed by a variety of authors - instability of the plasma arc is a phenomenon of considerable importance to engineering applications.

Bowman\textsuperscript{11} considers this using analogies to nozzle jets, by using the Maecker expression for arc thrust to derive an arc Reynolds number that is equivalent to that used in jet studies. By this analogy, two main modes of dynamic instability are seen in arcs: helical, in which the entire arc column twists into a helical shape due to oscillatory instability near the cathode spot, and axisymmetric, in which vortex rings form around the arc column and travel downwards along it. Bowman concludes that helical instabilities of the arc are frequently present in the range 0.4 to 2kA, with some evidence to suggest similar behaviour on industrial-scale arcs at up to 120kA.

Evers\textsuperscript{16} performed extensive analysis of magnetic fields induced by the power circuitry around a typical DCAF and its effect on the deflection and positional bias of the arc. Zweben and Karasik\textsuperscript{17} investigated laboratory scale DC plasma arcs and measured the deflection and instabilities induced by both constant and time-varying magnetic fields. They provide high speed photographic evidence that unstable arcs, at least in the relatively low current regime, form helical shapes in space.
Reynolds and Jones\textsuperscript{18} examined experimental and photographic evidence from a test furnace using two electrodes, and offered a semi-empirical model of DC plasma arc interaction in the case where more than one arc is present.

Arc motion and instability has also been studied in the related field of DC plasma torches\textsuperscript{56,57,58,59}. In such devices, the anode and cathode electrical connections of an arc discharge are both contained within a cylindrical housing through which the plasma gas is passed at high velocity. As the gas stream flows through the arc, it is heated and produces a tail flame which can then be used for a variety of high-temperature manufacturing and materials processing applications.

Several modes of short time-scale fluctuation have been identified in plasma torches\textsuperscript{57}. These include:

- Takeover mode, characterised by regular oscillations in the arc column.
- Restrike mode, in which the root of the plasma arc moves across the electrode surface and spontaneously jumps to another location.
- Helmholtz oscillations, caused by pressure vibrations of the volume of gas in the inlet chamber of the torch.

Takeover mode results primarily from the flow of plasma gas in and around the arc jet, causing the arc column to oscillate (a Kelvin-Helmholtz instability). This oscillation then causes the anode root of the arc to periodically detach from one side of the channel in the plasma torch and reattach on the other.

Restrike mode instabilities are caused by the sustained superimposed high-velocity gas flow present in plasma torches. This imparts a lateral drag force on the arc jet near to its attachment point on the anode, and moves the attachment point steadily along the anode channel in the direction of the superimposed flow. As a result the arc is progressively drawn out, and upon reaching a critical length, a new arc root is spontaneously formed upstream of the existing root (the restrike).

Helmholtz instabilities occur predominantly due to the design of DC plasma torch devices. The plasma gas is passed into the torch via an inlet system (the “cold chamber”) before it is heated by the arc. Recent work by Coudert et al\textsuperscript{59} has demonstrated both experimentally and theoretically that the cold chamber can act as a Helmholtz resonator, producing pressure oscillations as a result of the
spring-like compression and expansion behaviour of the gas volume.

Restrike mode and Helmholtz instabilities are effects related specifically to the characteristics of DC plasma torches as opposed to DC plasma arc furnaces, and are unlikely to be relevant to the problem under consideration. Takeover mode instabilities are more closely related to the fluid flow properties of the arc column, and similar behaviour may be expected to occur in any atmospheric pressure DC plasma arc system.

**Early numerical studies**

In the early 1980's, Szekely and his co-workers brought numerical methods to bear on the problem of DC plasma arcs for the first time\textsuperscript{21,22,23}. Their approach relies on the formulation of a description of the plasma arc in terms of the laws of conservation of mass, momentum, and energy, as well as constitutive relations describing the properties of the plasma fluid. These laws manifest in the form of partial differential equations for the fields of interest. Solution is achieved using a variety of numerical approximations to the equations (finite difference and finite volume methods are the most common), and is performed using computers.

This approach signalled a major step forward in understanding from the empirical and analytical models that came before it. The temperature and velocity fields in and around the arc were no longer empirical constructs, but calculated directly from the fundamental physics governing the system.

Szekely's original work was aimed at the modelling of small, low-current DC plasma arcs of interest to welding and related applications. The approach was soon extended to arc furnace scale, in a series of seminal publications from 1981 to 1983. With Ushio et al\textsuperscript{21}, the authors introduced the concept of a partially decoupled system for modelling a large, high-current arc which contained a number of pertinent features:

- An axisymmetric coordinate system, reducing the dimensionality to 2D
- Finite difference discretisations using regular, structured grids
- Steady state (no evolution in time)
- A known, pre-specified current density distribution
- The use of an approximate model of fluid turbulence (K-\(\varepsilon\))
- The use of constant physical properties for the arc plasma
- Treating the arc plasma as optically thin, permitting radiation loss to be expressed approximately as a sink term per unit volume

Although rudimentary in nature - computers of the time could only solve the the model on very coarse grids consisting of small numbers of grid points, and some empirical knowledge regarding the current distribution was still required - this model allowed the authors to predict a number of useful properties of the arc, such as heat transfer and jet velocity behaviour that matched experimental measurements. Critically, it also provided a persuasive qualitative description of the way the temperature and velocity fields varied within the arc column itself.

Building on this approach, McKelliget and Szekely\textsuperscript{22} recognised that earlier models were very sensitive to the boundary conditions used, and applied similar methods to study the cathode spot region of AC plasma arcs in higher detail for currents in the range $10^{-5} \ - \ 50kA$. The numerical method used to solve the fluid flow problem was also altered, to a finite-volume method using the 2/E/FIX algorithm of Pun and Spalding\textsuperscript{45}. Some elementary sensitivity analysis was carried out to investigate the effect of varying the physical and transport properties of the plasma used, and again agreement between the model predictions and experimental data was shown to be good.

Further extending the method, Szekely et al\textsuperscript{23} attempted to partially couple the stand alone models of AC and DC plasma arcs with similar fluid flow and heat transfer models of the molten bath that forms the anode.

The refined model of the plasma arc used in this work used complex boundary condition specifications computed from near-electrode models, as per the authors' earlier paper. A slightly more advanced model for computing radiation energy emitted by the arc column was also developed. In this work, the authors compared experimentally measured two-dimensional profiles of velocity magnitude and temperature to the results from the model, and the data were seen to agree reasonably well.

Coupling between the arc and the molten bath was achieved by using the shear stress at the interface generated by the arc model with the assumption that the anode was a non-slip surface. This shear stress was then passed into the flow model for the molten bath as a surface source term for momentum, creating circulation.
Recent numerical work and refinements

The distributed-parameter numerical approach for the study of DC plasma arc furnaces originated by Szekely and co-workers has subsequently been greatly enhanced and extended by many research groups.

Zhu et al\textsuperscript{24}, Shamsi\textsuperscript{25}, Qian et al\textsuperscript{26}, Alexis\textsuperscript{27}, and others have introduced fully-coupled electromagnetic field calculations to eliminate the uncertainty associated with an empirically specified current density. In these solvers, the electrical potential and current density are described by additional partial differential equations arising from the principles of electromagnetism, and are solved numerically in conjunction with the flow and temperature fields. Variations on this approach were considered by Alexis et al\textsuperscript{28} and Ramirez et al\textsuperscript{29}, and focused on including induced current terms in the magnetic transport equation. These terms may become significant in the case of very high current industrial-scale arcs.

The numerical methods used for discretisation of the governing equations by these authors varies considerably. Variations of finite difference and finite volume techniques are the dominant methods however, with the SIMPLE and SIMPLE-R algorithms developed by Patankar\textsuperscript{46} being popular.

A great deal of the research effort on the modelling of DC plasma arcs over the past 25 years has been expended on so-called “unified” models, which seek to combine as many regions of the DCAF as possible in a single numerical framework.

Zhu et al\textsuperscript{24} included a portion of the electrode in their calculations, using a coupling model between the electrode surface and the arc plasma based on plasma sheath physics. Although only applied to the relatively small arcs relevant to welding processes, their results agreed very well with experimental temperature profiles and predicted current densities that matched those calculated by the more simplistic thermionic emission approach.

More sophisticated finite volume and finite element methods were applied to the DC plasma arc problem in the 1990's, as commercial CFD software began to mature. Gu et al\textsuperscript{10} used an early version of FLUENT to compute thermal and flow behaviour of a furnace heated by a plasma torch (a type of DC arc heating system in which the arc is completely contained inside a lance, which
surrounds the electrode and acts as the anode). Their unified model included the torch and arc, the molten bath, and the furnace sidewalls, and also studied the effect of contamination of the plasma with Silicon.

Bakken and Holt\textsuperscript{31} used a similar approach to the unified model of Gu et al, but applied it to a furnace with three inclined DC plasma arcs. They also made extensive study of the radiation heat transfer by including a detailed model of radiative heat exchange between surfaces and gas volumes in the domain of interest.

Much of the fundamental physics of plasmas relevant to DC arcs was summarised and studied in publications by Pfender and coworkers\textsuperscript{32,33,45}. Approximation concepts such as local thermodynamic equilibrium for thermal plasmas enabled the physical and transport properties of plasma fluids to be described as a function of temperature only. The authors also provided substantial tables of property data, calculated from first principles of statistical mechanics, for a range of plasma compositions. Of particular interest, electrical conductivity and radiation energy loss of arc plasmas as a function of temperature was studied, providing a greatly improved set of input data for numerical calculations in which the temperature dependence of these variables is crucial.

1.3 Desired advancements in the field achievable with this work

**Fully coupled, dynamic model**

One of the key shortcomings of the vast majority of the modelling work performed on DC plasma arc systems to date is the steady state approximation.

Much of the academic work related to experimental studies of plasma arc systems suggests that the arc is frequently in an unstable, transient mode of operation\textsuperscript{17,19,20}. While in this mode, the arc column is seen to move rapidly and form complex shapes in space, rendering any approximate steady state modelling effort moot. Such unstable behaviour can have a significant impact on the operation of DC arc furnace equipment, both via direct effects such as difficulties for control and measurement systems and are extinguishment, and indirect effects including noise pollution and undesirable electrical harmonics feeding back into the furnace power supply and grid.

While several theories have been put forward to explain this dynamic behaviour, no work has been
done on directly studying the dynamic behaviour in the context of a distributed parameter numerical model. By establishing the focus on dynamic behaviour early on, the numerical methods chosen for the study can be selected and designed appropriately.

With this in mind, the desirable principal outcomes of the work are related specifically to phenomena of the DC plasma arc that are dependent on time, including oscillatory and helical arc column development, arc stability, and interactions with other arcs and the environment.

Study of the coupling between various components in the model is also a priority. The DC plasma arc system is a multi-physics problem, and the interaction between the flow, energy transfer, and electromagnetic fields is of great importance and interest. It is the intention with this work to focus on the coupling effects that are key to the formation and maintenance of a plasma arc, and simplify or approximate the models only where the effects of doing so are justified and secondary.

**High resolution 2D and 3D models**

The computational demands of fully coupled DC plasma arc models are high, which has traditionally greatly limited the size and scope of numerical investigations. It is however desirable, particularly for dynamic modelling, to have as much spatial resolution available to the model as possible.

Bearing this in mind, relatively simple and robust models able to scale to very high resolution are desirable. Uniform, structured cartesian grids work well for this application, as they are trivial to set up and maintain during the calculation, and the isotropy and structure of the grid greatly simplifies many numerical operations.

It is important to note that a large proportion of the preceding DC plasma arc modelling work performed by other authors has made use of axisymmetric grids in 2D. Unfortunately, the enforced rotational symmetry of such grids means that while they may be more physically realistic than a 2D cartesian grid for the problem of modelling the arc jet, they are unable to capture significant aspects of the arc dynamics that break its rotational symmetry, such as oscillations and twisted/helical column formation. They are also unable to accurately study cases of multiple arcs interacting. In short, the axisymmetric grid is very successful for engineering models of the DC plasma arc, but less useful for more detailed dynamic studies.
An upper limit on necessary resolution for fluid flow problems is set by the viscous length scale. Simulations performed at this resolution or greater are said to be fully resolved, and capture all transient effects of the flow from length scales similar to the dimensions of the domain, down to viscous dissipation levels. They are often termed “direct numerical simulations”, as they are able to study turbulence effects by modelling them directly using the Navier-Stokes equations, as opposed to making use of any approximate or sub-grid scale turbulence models.

In order to produce a fully resolved flow, certain relationships between the density of numerical grid points and the simulation Reynolds number must hold. The origin of these relationships is the requirement that all length scales down to the Kolmogorov length (the smallest dissipative scale) be resolved in the simulation. The Kolmogorov length is defined by the viscosity of the fluid and its rate of kinetic energy dissipation, which is in turn defined in terms of the characteristic length of the domain and the turbulent velocity. Combining these relations gives, for two-dimensional problems,

\[ \frac{L}{\delta l} = O\left(Re^{1/2}\right) \]

while in three dimensions

\[ \frac{L}{\delta l} = O\left(Re^{3/4}\right) \] (4)

The Reynolds number is defined by the root mean square velocity and the characteristic length scale of the simulation:

\[ Re = \frac{v_{rms} L}{\nu} \]

(5)

Achieving a fully resolved calculation is seen to be considerably less demanding in 2D than in 3D. Even using powerful computers, a fully resolved simulation in 3D at moderate Reynolds number will make extremely large demands on memory, data storage, and execution speed.
Multiple arcs

As DC plasma arc furnaces scale to increasingly large power levels, current limitations are reached. This is principally due to the fact that these furnaces use pre-baked graphite electrodes, which are only economically available up to a maximum size (typically 750mm diameter). Graphite electrodes have a maximum specification for the current density they are able to safely carry, which limits furnaces with a single electrode and arc to of the order of 50-70kA, representing < 80MW total power.

The most viable option for an industrial producer who wishes to build a larger furnace is to use two (or more) electrodes in a single vessel and split the current load among them equally. This results in two (or more) arcs operating in close proximity, and can cause considerable interaction to take place depending on the electrode separation and current flow. Understanding the key features of such interaction is important both for fundamental academic reasons, as well as from the point of view of the furnace designer who wishes to know how the DC plasma arcs will behave in order to adequately design the feed ports and cooling systems.

In addition to this, under certain conditions multiple arcs may appear on a single electrode\textsuperscript{20}, even on much smaller scale furnaces. The exact mechanism for multiple arc formation is not clear, however, a model that is capable of handling more than one arc in order to study such cases once they have evolved is clearly desirable.

Parallelisation techniques

With the established emphasis on transient simulations and high-resolution modelling, the demands on the computer systems used to run the models are expected to be large. Modern computing architectures, both high performance and mainstream, are evolving to include an ever increasing degree of parallelism, and some degree of code parallelisation for the DC plasma arc model is thus an important feature.

Two distinct flavours of parallel computing systems are available to the researcher today. Distributed memory multiprocessing (DMP) is loosely defined as a number of nodes, each with their own processor(s) and memory space, connected via a communication system such as Ethernet or more advanced interconnects. Computer programs communicate between nodes by passing
information across the communication layer. Shared memory multiprocessing (SMP) is the traditional parallel model, in which a number of processors share a single large memory resource. Once restricted to high performance servers and workstations, this model has become considerably more mainstream in the past few years with multi-core processors becoming ubiquitous in desktop PCs.

SMP and DMP both have advantages and disadvantages, which make each more or less suited to particular applications (see Table 1). Parallelisation of the DC plasma arc model will be considered carefully with this in mind.

**Table 1: Some characteristics of SMP and DMP systems**

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>SMP</th>
<th>DMP</th>
</tr>
</thead>
<tbody>
<tr>
<td>System performance</td>
<td>up to 1 TeraFLOP</td>
<td>up to 1 PetaFLOP</td>
</tr>
<tr>
<td>System memory</td>
<td>up to 1 Terabyte</td>
<td>up to 100 Terabytes</td>
</tr>
<tr>
<td>Best applications</td>
<td>Any, flexible</td>
<td>Minimal intercommunication, embarrassingly parallel</td>
</tr>
<tr>
<td>Code parallelism</td>
<td>Simple, thread based</td>
<td>Complex, message-passing protocols</td>
</tr>
<tr>
<td>Availability</td>
<td>Common, desktop PCs and upward</td>
<td>Less common, more specialised systems</td>
</tr>
</tbody>
</table>

**1.4 Overview of aims, and structure of the rest of this thesis**

The key aims of the current work are intimately connected with the dynamics of the DC plasma arc in the context of the DC arc furnace environment. Models of the system are to be constructed in order to study the asymmetric behaviour and evolution through time of both single and multiple arc systems. The numerical methods used for the models will focus on high-performance, high-resolution temporal calculations in both two and three dimensions. These models are then to be applied to the study of the behaviour of the DC plasma arc system as certain model parameters and variables of interest are altered. The central focus of this exercise will be a (primarily qualitative) cataloguing of the spatial and temporal phenomena that the model produces, in order to highlight behaviour that occurs on very short time scales and which therefore is frequently missed or ignored in traditional DC arc furnace experiments and operations.
The novelty embodied in the work is expected to be largely twofold. Firstly, the particular numerical description used to model the DC plasma arc in two and three dimensions is unique for this problem, focusing on regular grids, simple finite difference expansions, and stable explicit methods for advancing the solution in time. The particular treatment of the discretisation of the model in 2D is also new for the study of plasma arcs, using cartesian rather than axisymmetric geometry, and employing novel approximations to the governing equations to reduce the dimensionality of the problem. Secondly, the results generated by the modelling study focusing on the dynamics of single and multiple arcs in two and three dimensions are novel in the field. To date, the strong focus of DC plasma arc modelling work has been on steady state engineering models, whereas this work explicitly focuses on the complex dynamics and evolutionary phenomena of arc systems that occur on very short time scales.

In chapter 2, an overview of the model and the governing equations for the DC plasma arc system will be presented. Sections on the individual sub-models relevant to fluid flow, heat transfer, and electromagnetism will examine mathematical details relevant to each component, including the fundamental equations, any approximations made, and the boundary conditions used. The chapter will conclude with a summary of the coupling effects that exist between the various components.

Chapter 3 will put forth a description of the numerical methods brought to bear on the problem. The geometry of the calculation domain and grid structure will be discussed first. Separate sections will then deal with the numerical treatment of individual sub-models, examining the spatial discretisation of the governing equations, the time stepping methods used, and accuracy and stability considerations. The chapter will conclude with a discussion of parallelisation techniques and implementation, data storage and memory issues, and data visualisation methods.

Chapters 4, 5, and 6 will discuss the results obtained from running the model. In Chapter 4, several test cases will be examined to study the performance of the fluid flow and electromagnetic sub-models. In Chapter 5, a number of model cases will be examined using the full 2D DC plasma arc model, examining a range of variables of interest with a focus on single arc systems. Both qualitative and quantitative observations from the model results will be presented. In Chapter 6, the full 3D model will be used to examine a set of model cases changing certain variables of interest, with a focus on interacting multiple arc systems. Again, qualitative and quantitative results will be presented. A final section will discuss various aspects of the results obtained.
In chapter 7, conclusions drawn from the modelling effort will be discussed. Success or failure of the model in terms of the original aims will be assessed.

References and appendices will follow, providing ancillary information to what has been discussed in the earlier chapters.
Chapter 2 - Governing Equations

2.1 Flow models

A model of the fluid dynamics occurring in DC plasma arcs is essential to the success of a complete model of the system. The arc is a high velocity gas jet driven by interactions with other physical phenomena, and is fundamentally a fluid flow problem.

The starting point for all models of fluid flow is the Navier-Stokes equations. These are a set of non-linear partial differential equations describing momentum transfer and conservation in terms of the velocity vector components and fluid pressure, with mass conservation enforced via an additional partial differential equation, the continuity equation.

The Navier-Stokes equations

In the most general case of a Newtonian fluid, the equations for conservation of momentum and mass are given by

\[ \rho \frac{\partial \mathbf{v}}{\partial t} + \rho (\mathbf{v} \cdot \nabla) \mathbf{v} + \nabla P = \nabla \cdot (\mu \nabla \mathbf{v}) + \mathbf{F} \quad , \]

\[ \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \quad . \]

For the DC plasma arc problem, we assume an incompressible flow with constant physical and transport properties. This permits some simplification, so that (6) and (7) become

\[ \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} + \nabla P = \nu \nabla^2 \mathbf{v} + \frac{\mathbf{F}}{\rho} \quad , \]

\[ \nabla \cdot \mathbf{v} = 0 \quad . \]

Equation (9) is derived from (7), which can be rewritten as
where \( \frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{v} = 0 \),

\begin{equation}
\frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{v} = 0 ,
\end{equation}

Hence (9) follows.

We consider the boundaries of the region of interest to be non-slip walls - that is, we assume that the arc and its immediate surroundings form a closed system, completely contained within the calculation domain. This permits the specification of a simple boundary condition for the velocity; that is,

\begin{equation}
v_r = 0 .
\end{equation}

Unfortunately, the equations as posed are not necessarily in the best form for numerical analysis - several variants may be considered.

**Primitive variables**

In this formulation, much of the original form of the Navier-Stokes equations is retained. The most successful class of solvers for fluid flow using this method is undoubtedly the MAC scheme of Harlow and Welch\(^{15}\) and its descendants (including projection methods), which were the first to recognise that much of the difficulty in designing numerical methods for fluid flow lies with the computational enforcement of the continuity equation.

MAC methods generally seek to replace the continuity equation with a Poisson equation for pressure. This is achieved by applying the divergence operator to (8).

\begin{equation}
\nabla \cdot \left( \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} + \nabla \rho \right) = \nabla \cdot \left( \mathbf{v} \nabla^2 \mathbf{v} + \frac{\mathbf{F}}{\rho} \right)
\end{equation}

Rearranging terms and using (9) to eliminate as much as possible gives:

\begin{equation}
\nabla^2 \rho = -\nabla \cdot (\mathbf{v} \cdot \nabla) \mathbf{v} + \frac{\nabla \cdot \mathbf{F}}{\rho}
\end{equation}

\( 20 \)
With (8) and (13) one obtains a system mathematically equivalent to (8) and (9). Numerical solution of the system is somewhat simplified however, due to the fact that (13) is now a standard Poisson equation. Unfortunately, boundary conditions for pressure are now required in order to complete the solution, and these are far from obvious in most cases.

E and Liu$^{39}$ point out that the most obvious candidate for the pressure boundary condition in the case of non-slip walls is given by extending (8) to the boundary, where the viscous terms dominate:

$$\left[ \frac{\partial p}{\partial n} \right] = \nu \cdot \nabla^2 \nu + \frac{n \cdot F}{\rho}$$  \hspace{1cm} (14)

However, it would appear that this condition is difficult to enforce accurately or consistently in a numerical setting, which leads to poor conservation of mass flow continuity.

Johnston and Liu$^{36,37}$ offer a number of interpretations of and alternatives to this boundary condition, and conclude that the best stability and accuracy properties are obtained by leaving the viscous term in (13) in rotational form:

$$\left[ \frac{\partial p}{\partial n} \right] = -\nu \cdot \nabla \nabla \nabla \nu + \frac{n \cdot F}{\rho}$$  \hspace{1cm} (15)

Unfortunately, translating this boundary condition to a numerical approximation can be troublesome for many methods, and direct equivalence with the original Navier-Stokes equations is no longer maintained for the case of steady-state flows.

Vorticity formulation

An early reformulation of the Navier-Stokes equations to make them more amenable to numerical analysis was first used by Fromm$^{40}$ and others, applying methods originally proposed by Thom$^{38}$. They recognised that a number of the difficulties in the equations as presented in primitive variables originate from the presence of the pressure term. The pressure may be completely eliminated by the simple expedient of taking the curl of (8).
$$\nabla \times \left( \frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} + \nabla p \right) = \nabla \times \left( \nu \nabla^2 \mathbf{v} + \frac{\nabla \times \mathbf{F}}{\rho} \right)$$

(16)

After rearranging terms, applying vector identities, and using equation (9), (16) can be expressed as:

$$\frac{\partial |\nabla \times \mathbf{v}|}{\partial t} + \nabla \times |\nabla \times \mathbf{v}| = \nu \nabla^2 |\nabla \times \mathbf{v}| + \frac{\nabla \times \mathbf{F}}{\rho}$$

(17)

Introducing a vector variable, vorticity, allows (17) to be simplified.

$$\mathbf{\omega} = \nabla \times \mathbf{v}$$

(18)

$$\frac{\partial \mathbf{\omega}}{\partial t} + \nabla \times (\mathbf{\omega} \times \mathbf{v}) = \nu \nabla^2 \mathbf{\omega} + \frac{\nabla \times \mathbf{F}}{\rho}$$

(19)

Together with (9) and (11), these provide an equivalent formulation of the Navier-Stokes equations. Since the velocity field is solenoidal by definition of (9), a second intermediate variable, the vector potential, can be defined.

$$\mathbf{v} = \nabla \times \mathbf{\psi}$$

(20)

The vector potential has an associated gauge freedom, which is generally fixed by requiring that it too be divergence free.

$$\nabla \cdot \mathbf{\psi} = 0$$

(21)

This gauge results in a set of Poisson equations relating $\mathbf{\omega}$ and $\mathbf{\psi}$.

$$\mathbf{\omega} = \nabla \times |\nabla \times \mathbf{\psi}| = \nabla |\nabla \cdot \mathbf{\psi}| - \nabla^2 \mathbf{\psi} = -\nabla^2 \mathbf{\psi}$$

(22)

The full formulation of the Navier-Stokes equations in vorticity-vector potential form may then be written as
\[
\frac{\partial \omega}{\partial t} + \nabla \times [\omega \times (\nabla \times \psi)] = \nu \nabla^2 \omega + \frac{\nabla \times F}{\rho},
\]

(23)

\[
\nabla^2 \psi = -\omega.
\]

(24)

Boundary conditions are also necessary for the new intermediate variables in order to complete the specification. However, it is instructive to first consider the 2D case of this formulation. In two dimensions, both the vorticity and the vector potential reduce to scalar variables (the scalar vorticity and the stream function, respectively). The non-linear convection term in (23) can be rearranged to more closely resemble (8):

\[
\frac{\partial \omega}{\partial t} + [\nu \nabla] \omega = \nu \nabla^2 \omega + \frac{1}{\rho} \left( \frac{\partial F_y}{\partial x} - \frac{\partial F_x}{\partial y} \right)
\]

(25)

\[
\nabla^2 \psi = \omega
\]

(26)

\[
v_x = -\frac{\partial \psi}{\partial y}, v_y = \frac{\partial \psi}{\partial x}
\]

(27)

The boundary condition (11) translates into two unique boundary conditions for \( \psi \):

\[
\psi_r = 0, \quad \left[ \frac{\partial \psi}{\partial n} \right]_r = 0
\]

(28)

Unfortunately, while this does completely specify the problem, it does make translation into a numerical method somewhat more difficult. A single boundary condition for each of \( \psi \) and \( \omega \) would be more desirable, and a large amount of research has gone into designing numerical methods of translating one of (28) into a boundary condition for vorticity - as with the pressure boundary conditions in the primitive variable formulation, there is no obvious specification, and many interpretations are possible. However, E and Liu\(^{35}\) clarified the field by summarising a great variety of methods, and performed analysis that indicated virtually all of the more complicated implicit methods are in fact numerically equivalent to one of the simplest and easiest-to-implement boundary conditions, a local method originally proposed by Thom\(^{38}\).
One of the principal difficulties with the vorticity formulation is that the extension from 2D to 3D is not trivial. In 2D, the formulation is very attractive for numerical methods, since it reduces the number of field variables (from three to two), and removes many of the difficulties associated with the pressure variable - relatively simple boundary conditions can be associated with both \( \psi \) and \( \omega \). Additionally, continuity of the flow is directly and implicitly enforced due to the unique specification of \( \nu \) in terms of \( \psi \) (20).

As one moves to 3D, the number of field variables goes up as compared to primitive variables (from four to six), and additional difficulties and ambiguities arise in the numerical interpretation of the convection term in (23). Due to the gauge freedom of the vector potential, continuity of the flow must now also be explicitly enforced by the boundary conditions. Complicating this task, as in the 2D case, no explicit boundary conditions are available for the components of the vorticity vector, while boundary conditions for the vector potential are overspecified - conversion between the two, as per analogies of Thom's method, must be performed. E and Liu\(^4\) address many of these issues in designing numerical methods using this formulation, but extending vorticity-based methods to 3D remains fraught with difficulty.

The gauge method

Recognising many of the problems associated with both the primitive variable approach and vorticity methods, E and Liu\(^39\) introduced the gauge method for incompressible flows. This formulation has its own limitations, as it works exclusively for the incompressible Navier-Stokes equations, but within this limitation it is extremely well suited for numerical implementation.

The formulation begins by replacing the velocity with an auxiliary field \( \mathbf{a} \), supplemented by a gauge variable, \( \theta \).

\[
\mathbf{a} = \mathbf{v} + \nabla \theta
\]  

(29)

Combining this with (8) and rearranging gives
\[
\frac{\partial a}{\partial t} + \mathbf{v} \cdot \nabla |\mathbf{v}| + \left[ \nabla p - \nabla \left( \frac{\partial \theta}{\partial t} - \nu \nabla^2 \theta \right) \right] = \nu \nabla^2 a + \frac{F}{\rho}.
\] (30)

This leads to a natural definition for the relationship between \( p \) and \( \theta \) so as to eliminate the pressure-related terms:

\[
p = \frac{\partial \theta}{\partial t} - \nu \nabla^2 \theta
\] (31)

Finally, taking the divergence of (29) gives a simple Poisson equation relating \( a \) and \( \theta \), suitable for numerical analysis.

\[
\nabla \cdot \mathbf{a} = \nabla \cdot \mathbf{v} + \nabla^2 \theta = \nabla^2 \theta
\] (32)

A key strength of the gauge method is that the boundary conditions for both \( a \) and \( \theta \) may be simply and unambiguously specified by using the gauge freedom. Boundary conditions equivalent to (11) can be defined in either of the following two ways:

\[
\frac{\partial \theta}{\partial n} = 0, \quad a \cdot n = 0, \quad a \cdot \tau = \frac{\partial \theta}{\partial \tau}
\] (33)

\[
\theta = 0, \quad a \cdot n = \frac{\partial \theta}{\partial n}, \quad a \cdot \tau = 0
\] (34)

The full gauge formulation is then expressed as follows.

\[
\frac{\partial a}{\partial t} + \mathbf{v} \cdot \nabla |\mathbf{v}| = \nu \nabla^2 a + \frac{F}{\rho}
\] (35)

\[
\nabla^2 \theta = \nabla \cdot \mathbf{a}
\] (36)

\[
\mathbf{v} = a - \nabla \theta
\] (37)

These are supplemented by either of (33) or (34), whichever best suits the numerical method used to obtain the solution - a choice is possible since both are equivalent to the same physical boundary
condition, that defined by (11).

Wang and Liu\(^42\) show that while the gauge method shares a number of similarities with projection methods, it is free from many of the problems associated with such methods (formation of numerical boundary layers, prescription of pressure boundary conditions).

The gauge method extends easily and logically to 3D, and retains the same number of fields as the primitive variable formulation.

**Source terms**

For the DC plasma arc problem, the forces applied to the fluid have their origin in the electromagnetic fields in and around the arc. In a region with a known current density vector field \(j\) and magnetic field \(B\), the source term for momentum is given by the Lorentz law, a fundamental principle of electromagnetism which states that any charge moving through a magnetic field experiences an applied force.

\[
F = j \times B
\]  

(38)

In the DC plasma arc model, the calculation of \(j\) and \(B\) is performed by a separate sub-model. This will be presented in a later section.

**Approximations**

The single largest approximation in the flow model, and indeed in the DC plasma arc model as a whole, is the decision to use constant values for the physical and transport properties of the fluid.

This was taken in light of the fact that the central aim of the model is to reproduce and study the dynamic behaviour of the DC plasma arc, rather than produce highly accurate engineering predictions. In reality, the physical properties vary widely as functions of temperature, however, the principal effects of interest to the current investigation are primarily related to the very strong coupling between the electromagnetic fields and the flow that generates and sustains the arc - using constant values for the properties is expected to be a secondary effect by comparison.
Good data is available in Boulos et al' and other sources for calculating representative average values of the properties concerned. The variation of properties and the effect it might have on the model results will be considered in this study by a simple sensitivity analysis, as was performed by McKelliget and Szekely'.

After the use of constant physical properties, the use of the incompressible flow equations is the remaining significant approximation used in the derivation of the flow model. For the small to moderate sized arcs considered here, the peak velocities in the plasma are well below supersonic. Values for the speed of sound in an air plasma at temperatures between 10000 and 15000K are of the order of 2700 - 3700 m/s, thus any model predictions indicating velocities of the order of 1000 m/s or less (< 0.3 Ma) may safely be considered as incompressible and will retain their accuracy. As the speed of sound will in general increase with increasing temperature it is important to note that this is likely to be a conservative estimate, as the regions of peak velocity in DC plasma arcs generally coincide with the regions of peak temperature.

The 0.3 factor is chosen to ensure that the degree of compression of the fluid by pressure effects is of the order of 5% by volume or less and therefore has minimal effect on the velocity field. While this is a reasonable approximation for the relatively small scale arc systems considered in this work, at the higher currents more typical of industrial scale furnace operations the arc jet is expected to enter a transonic regime in which parts of the flow (particularly close to the arc attachment zone at the electrode surface) are supersonic. Shock waves and transition effects are therefore likely to develop and contribute significantly to any unstable transient behaviour.

2.2 Energy model

As the DC plasma arc problem is commonly associated with high temperature materials processing systems and operates at significantly elevated temperatures, heat transfer phenomena are of great importance to any model of the problem.

The region of space in and around the arc consists of a mixture of gas, and material in the fourth state of matter, plasma. Plasmas are formed by heating gases to the point at which the component atoms and molecules begin to break up and ionise - this process forms a neutral “soup” of free electrons and heavy charged ions which is able to conduct current electricity very effectively. This behaviour is a key element of the self-sustaining DC plasma arc.
Local Thermodynamic Equilibrium (LTE) approximation

Boulos et al\textsuperscript{45} offer a more detailed description of the physics and mathematics of LTE, but the conditions required may be summarised as follows. A plasma is said to be in LTE if:

- The plasma is optically thin
- Collision processes, not radiative processes, dominate plasma reactions
- There is equilibrium between collision processes and their reverse processes
- Local gradients of plasma properties are small

In cases in which the LTE assumptions are valid, many simplifying assumptions can be made.

The first condition relates to how the plasma interacts with radiation passing through it. Plasmas may be optically opaque, absorbing all incident radiation, or optically thin, permitting all incident radiation to pass through unaffected. For thermal plasmas typical of DC plasma arcs at atmospheric pressure and small scale (< 1kA current), optically thin behaviour dominates across most wavelengths, however it is important to note that this is not in general the case. Bowman\textsuperscript{11} and Boulos et al\textsuperscript{45} have demonstrated that the reabsorption of thermal radiation by atmospheric plasmas may become very significant in the physically larger, higher-current arcs typical of industrial furnace operations. Using the present model for such large arc systems may result in considerable errors - one would expect the local temperatures in the vicinity of the arc to be lower than in reality, causing the arc voltage to be overpredicted by the model.

The second and third conditions are central to LTE, and relate to the state variables required to describe the plasma. In the most general case a plasma may be understood as a complex reacting flow, consisting of chemical interactions between a number of different neutral and ionised species. Local compositions of each species, including free electrons, as well as their energy levels would be needed for a complete description of the state of the material, and due to the differences in mass of the various charge carriers (radical in the case of electrons versus heavy ions), separate temperatures for each species would be required to describe the system. However, with the conditions given, the plasma may be assumed to be in a state of local equilibrium with regard to the chemical reactions and transitions occurring, and moreover, the differences in the temperature state variable for each species may be safely ignored. This permits the plasma properties to be
completely characterised at any localised point by a single state variable, the plasma temperature.

The fourth condition supports the equilibrium requirements. It indicates that any particle moving from one location to another should find adequate time for equilibration, and that the time for equilibration should be of similar magnitude or smaller than the time required for the movement.

For the case of DC plasma arcs at atmospheric pressure, Boulos et al.⁵ and others report that the conditions of LTE are for the most part adequately satisfied. Deviations may be expected to occur in regions of low free-electron concentrations, near walls and surfaces, and in the outer regions of the arc near to the plasma transition temperature. However, given the mathematical and computational advantages that the assumption of LTE confers, these deviations may be ignored for the purposes of the present modelling effort.

*Equations for conservation of energy*

The fundamental partial differential equation governing the conservation of energy in a moving fluid is

\[ \rho \frac{\partial h}{\partial t} + \rho v \cdot \nabla h = \nabla \left( \frac{k}{C_p} \nabla h \right) + Q. \]  

(39)

For the DC plasma arc model, as in the case of the flow equations (and for the same reasons), we make the assumption that the fluid has approximately constant physical and transport properties. We additionally include the assumption of LTE, by which a single temperature can be used to characterise the plasma. The enthalpy may then be expressed simply in terms of a plasma temperature:

\[ h = C_p T \]  

(40)

The thermal diffusivity is defined by:

\[ \alpha = \frac{k}{\rho C_p} \]  

(41)
With these, (39) simplifies to:

\[
\frac{\partial T}{\partial t} + \mathbf{v} \cdot \nabla T = \alpha \nabla^2 T + \frac{Q}{\rho C_p}
\]  \hspace{1cm} (42)

To complete the specification of (42), boundary conditions for temperature are required. Due to the boundary conditions used for the velocity, it is logical to apply wall-like boundaries to the temperature field as well. However, since the boundary conditions used can have a significant impact on the temperature field and hence the arc behaviour (particularly in the vicinity of the electrode and cathode spot) it is prudent to investigate the effect of changing these boundary conditions. With this in mind, two sets of boundary conditions for (42) are proposed.

TBC 1 is defined by the use of constant temperatures on all surfaces. Different temperatures are specified according to the different wall materials possible on the boundaries of the model. Temperatures on the anode and electrode are governed by the vaporisation temperature of the materials they are composed of - as the arc is expected to be at greatly elevated temperatures compared to these boundaries, the temperature at these surfaces will be primarily governed by the phase change of the surface material into gaseous form.

\[
T_r = T_\infty
\]  \hspace{1cm} (43)

At the other extreme, TBC 2 is defined by the use of thermally insulating conditions on all boundaries. This is less physically realistic than (43), since very strong temperature gradients are usual in regions where high temperature plasmas come into contact with walls. It is however useful from a systems study point of view, in order to examine the role that thermal gradients near the boundary play in generating the dynamics of the DC plasma arc. Thus TBC 2 is

\[
\left[ \frac{\partial T}{\partial n} \right]_r = 0
\]  \hspace{1cm} (44)

on all boundaries.
Source term for temperature equation

The total energy source term $Q$ is related to the Ohmic heating of the DC plasma arc by the electrical current passing through it, and to the net volumetric radiation loss from the plasma.

$$Q = \frac{j \cdot j}{\sigma} - Q_r$$  \hspace{1cm} (45)

The current density is calculated using a separate sub-model for the electromagnetic fields, as described in a later section.

The thermal radiation term is calculated using look-up tables based on the data for optically thin radiation from the various plasma gases considered. Although it is not considered in this work, partially or completely optically thick plasmas could be dealt with in the model in a number of ways. For example, a simple method would be to assume that a fraction of the total emitted radiation is immediately re-absorbed by the plasma, and define an effective $Q_{\text{eff}}$ to take this into account. $Q_{\text{eff}}$ would in general be a strong function of temperature, just as $Q_r$ is. A somewhat better method - although computationally less trivial - would be to assume that the absorbed radiation fraction is redistributed as a positive source term in the energy balance equation within a certain small radius (the “optical thickness”) of the emission point.

2.3 Electromagnetic field model

A fundamental aspect of the DC plasma arc is the creation of and interactions with electromagnetic fields. At its most basic, electromagnetism is governed by Maxwell's laws\textsuperscript{48}, which describe how electric and magnetic fields interact and evolve through space and time.

For the problem of plasma arcs, we are typically interested in time scales much longer than those relevant to the transient parts of Maxwell's laws, in which field propagation occurs at the speed of light. We may thus safely use the simplified versions of the laws, in which transient terms and behaviour is disregarded, and the electromagnetic system is treated as being at steady-state at every point in time. These are referred to as the laws of electrostatics and magnetostatics, and combine with the equations of fluid flow to produce the magneto-hydrodynamic approximation.
Electric Potential

The electric potential in the gas space between the electrode and the anode formed by the molten bath surface is governed by an elliptic equation. This equation may be derived from the charge continuity relationship and the definition of current density in terms of the electric potential:

\[ \nabla \cdot j = 0 \quad (46) \]

\[ j = -\sigma \left| \nabla \phi + \nu \times B \right| \quad (47) \]

The second term in the current expression is the induced current, which is generated by the motion of the plasma across field lines. For the small to moderate sized arcs considered in this study, an order of magnitude analysis (see appendix 1) shows this term to be negligible in comparison with the electric potential, and it can be safely ignored. With this approximation, (47) becomes

\[ j = -\sigma \nabla \phi \quad . \quad (48) \]

Hence (46) becomes:

\[ \nabla \cdot (\sigma \nabla \phi) = 0 \quad (49) \]

Further simplification to a Laplace equation is possible in cases of constant electrical conductivity, however, this is not the case in plasma flows - conductivity is a strong function of temperature and provides much of the fundamental coupling between the various transport phenomena.

Boundary conditions are required for the complete solution of (49). These are generally supplied either in the form of a fixed potential at a surface (Dirichlet boundary condition) or a fixed current density at a surface (Neumann boundary condition, after (48)). For the DC plasma arc problem, the anode is generally set to a fixed potential of zero:

\[ \phi_{\text{f, anode}} = 0 \quad (50) \]
The area inside the cathode spot(s) is set to the fixed current density $j_k$:

$$\left[-\sigma \frac{\partial \phi}{\partial n}\right]_{r,cs} = j_k$$  \hspace{1cm} (51)$$

All other surfaces are treated as insulating:

$$\left[\frac{\partial \phi}{\partial n}\right] = 0$$  \hspace{1cm} (52)$$

**Current Density**

The current density field is determined by direct calculation once the electric potential field is known, via (48) or its discrete analogue.

**Magnetic Vector Potential**

Once the current density distribution in space is known, the steady-state magnetic field generated by it may be found from Ampere's and Gauss's laws of electromagnetism:

$$\nabla \times \mathbf{B} = \mu_0 \mathbf{j}$$  \hspace{1cm} (53)$$

$$\nabla \cdot \mathbf{B} = 0$$  \hspace{1cm} (54)$$

However, the laws in these forms are not well suited to numerical treatment. A more tractable form is obtained if we introduce an auxiliary gauge variable $\mathbf{A}$, the magnetic vector potential, such that:

$$\mathbf{B} = \nabla \times \mathbf{A}$$  \hspace{1cm} (55)$$

We then have:

$$\nabla \times \nabla \times \mathbf{A} = \mu_0 \mathbf{j}$$

$$\nabla \nabla \cdot \mathbf{A} - \nabla^2 \mathbf{A} = \mu_0 \mathbf{j}$$  \hspace{1cm} (56)$$
Using the gauge freedom to specify that $A$ is solenoidal (the Coulomb gauge), we have:

$$\nabla \cdot A = 0$$  \hspace{1cm} (57)

$$\nabla^2 A = -\mu_0 j$$  \hspace{1cm} (58)

Additionally, Gauss's law follows conveniently from the definition of $A$:

$$\nabla \cdot B = \nabla \cdot |\nabla \times A| \equiv 0$$  \hspace{1cm} (59)

With this definition, the solution for the magnetic field in two (or three) dimensions reduces to solving two (or three) decoupled Poisson equations for the components of $A$, followed by a curl calculation to retrieve $B$ from the auxiliary field.

In an enclosed region, boundary conditions must be applied to the Poisson equation for each component of $A$. For the problem at hand, we consider the surrounding boundaries to be magnetically insulating, that is, lines of magnetic flux are only permitted to pass into or out of the calculation region on parts of the boundary through which current is flowing.

There is some physical basis for this assumption, namely that the outer sidewall and roof of a DC arc furnace is typically constructed out of steel or other ferromagnetic materials. This will indeed act as a very effective shield, preventing the magnetic fields inside the vessel from emanating out into the space around it. It is nonetheless possible that the magnetic shielding of the furnace shell may not be perfect in reality, in which case more complex boundary conditions would be required for $A$.

Using the following set of boundary conditions guarantees that this condition will be met at all times:

$$|A \cdot \mathbf{n}| = 0$$  \hspace{1cm} (60)

This fixes one (in 2D) or two (in 3D) of the components of the magnetic vector potential at zero on
any given domain boundary. From the solenoidal definition of \( \mathbf{A} \), (57), it then follows that:

\[
\left[ \frac{\partial (\mathbf{A} \cdot \mathbf{n})}{\partial n} \right]_{r} = 0
\]  

(61)

This fixes the remaining component and completes the specification for \( \mathbf{A} \).

**Setting the current density at the cathode spot**

In order for charge-carrying electrons to pass through the DC plasma arc as they move around the electric circuit in the furnace, they first need to be emitted from the electrode surface and enter the plasma (they also need to be absorbed on the anode surface, in order to continue around the circuit, but energy barriers must only be overcome during the emission process). The rate at which electrons can be emitted from hot surfaces is governed by a process called thermionic emission, described by the Richardson-Dushman equation:

\[
j_{e} = A_{RD} T^{2} \exp \left( - \frac{e_{e} W_{f}}{k_{B} T} \right)
\]  

(62)

\( A_{RD} \) is an assembly of fundamental physical constants and may additionally depend on the surface material properties. \( T \) is the temperature at the emitting surface. \( e_{e} \) and \( k_{B} \) are the electron charge and the Boltzmann constant respectively. \( W_{f} \) is the work function of the emitting surface, a constant that depends only on the material the surface is made of.

The Richardson-Dushman equation gives the current density on the surface that is emitting electrons. As a result, for a fixed current \( I \) and temperature, it also determines the physical size of the emitting surface; assuming the surface is round, its radius is given by (2).

In the DC plasma arc model, the emitting surface is the graphite electrode. Assuming a very high temperature arc, it is then reasonable to assume that the surface temperature on the electrode is fixed at or near the sublimation temperature of graphite, at approximately 4100K. The work function of graphite is given as 4.75eV. Using these values and allowing for some uncertainties produces a \( j_{e} \) value of the order of \( 10^{7} - 10^{8} \) A/m\(^2\). Bowman\(^\text{10}\) correlated a large amount of data for DC arc furnaces to produce a value of \( 3.5 \times 10^{7} \) A/m\(^2\). This high value implies that the radius of the
cathode spot on the electrode surface is quite small, typically much smaller than the size of the electrode. This fact is borne out in visual and photographic observations.

**Approximations**

It is important to note that the power supply of the furnace and arc is assumed to be a perfect constant-current device, that is, it has an impedance of zero. There is some evidence to suggest that practical design of the power supply can influence the dynamics of the plasma arc\textsuperscript{17}, however, these effects are not studied in the present model.

The cathode spot is assumed to be stationary on the surface of the electrode in the model. While the cathode spot is in general quite mobile, and this has the potential to be a source of unstable motion of the arc in itself\textsuperscript{7,20}, observations of real arcs in the 1 – 5kA range indicate that it often remains stationary for several seconds at a time before moving rapidly to another location on the electrode, and unstable behaviour of the rest of the arc column persists even while the spot is stationary. As will be seen in later chapters the time scales used in the model are of the order of \(1/100\)th of a second or shorter, and the stationary-spot approximation is therefore deemed reasonable.

**2.4 Coupling effects**

The complete DC plasma arc model as embodied by the fluid flow, energy, and electromagnetic sub-models exhibits a number of important coupling effects between the various fields. It is these coupling effects which give rise to the formation of an arc and sustain it over time, and it is thus instructive to provide some additional commentary on the interactions present in the model.

The model is not fully coupled in the sense of each field variable being explicitly involved in every governing equation. However, as discussed in the preceding chapter, the critically important coupling effects that give rise to the formation of a sustainable plasma arc are accounted for, and are shown schematically in Figure 5.

To highlight an example - while the energy transfer does not explicitly affect the fluid flow equations, it very strongly affects the electric and magnetic fields, which then feed back into the fluid flow equations via the source term.

36
The result is that no part of the model is completely independent of the others for the duration of a numerical calculation based on these governing equations, although components may be partially or completely decoupled at the individual time-step level to enhance performance as much as possible. While the generality could certainly be extended in many ways, the model is believed to be comprehensive enough for the kinds of short-term, transient effects it is designed to study.

**Electromagnetics to fluid flow coupling**

This effect is embodied by the Lorentz force (38), a source term for the velocity equations that originates from the current density and magnetic fields. The Lorentz force exhibits a number of interesting features, most notably invariance with regard to the direction of current flow, and forming an irrotational vector field in 2D.

Consider reversing the direction of current flow through the arc, while leaving all other parameters and dimensions of the system unchanged - essentially this implies that the components of the $j$ field simply change sign while retaining the same magnitude. As a result, because $j$ and $B$ are related by a linear operator in (53), making $B$ a function of $j$, the $B$ field also becomes -$B$. With both $j$ and $B$ identical except for a change of sign, the Lorentz force $F$ (which is a vector product of the two) remains exactly the same.

This is an interesting result, although of more direct significance to the study of AC furnaces in
which the polarity of the circuit changes frequently, as opposed to DC, where the polarity is fixed. It
does demonstrate however that the Lorentz force is primarily a function of arc and electrode
geometry, and not the direction of current flow.

It is interesting to note the structure of the Lorentz force term in the 2D non-axisymmetric case. In
two dimensions, the magnetic field reduces to a scalar quantity \( B_z \). The components of the force
vector acting on the fluid become:

\[
F_x = j_y B_z \\
F_y = -j_x B_z
\]  

Similarly, the components of (53) become:

\[
\frac{\partial B_z}{\partial y} = \mu_0 j_x \\
\frac{\partial B_z}{\partial x} = \mu_0 j_y
\]  

If we now consider the curl of (63) to form a scalar quantity, we obtain:

\[
\nabla \times F = \left[ -\frac{\partial (j_y B_z)}{\partial x} - \frac{\partial (j_x B_z)}{\partial y} \right] \hat{\varepsilon}_z = \left[ -B_z \left( \frac{\partial j_z}{\partial x} + \frac{\partial j_x}{\partial y} \right) - j_x \frac{\partial B_z}{\partial x} - j_y \frac{\partial B_z}{\partial y} \right] \hat{\varepsilon}_z
\]  

The first term in brackets in (65) is equivalent to (46), and is zero by definition of current
continuity. Substituting \( j_x, j_y \) from (64) into (65) we then have:

\[
\nabla \times F = \left[ -\frac{1}{\mu_0} \left( \frac{\partial B_z}{\partial x} \frac{\partial B_z}{\partial y} - \frac{\partial B_z}{\partial y} \frac{\partial B_z}{\partial x} \right) \right] \hat{\varepsilon}_z = 0
\]  

In 2D cartesian coordinates, the curl of the Lorenz force is therefore irrotational.

This is particularly problematic for 2D fluid flow solvers that rely on the vorticity formulation (and
those that are numerically related or equivalent, such as the MAC scheme), since the source term
for vorticity is defined in this way. Certain modifications must be made to the electromagnetic
solver routines in order to perform such solutions in two dimensions - these will be discussed in a later section.

**Electromagnetics to energy coupling**

This coupling effect occurs as a result of the Ohmic heating of the plasma in the arc by the current passing through it. Also termed “resistance heating”, this effect is straightforward, as a larger current density and/or a lower electrical conductivity produces a larger heating effect. As for the Lorentz force term, and unsurprisingly as it acts on a scalar field, Ohmic heating is independent of the direction of current flow, depending only on the magnitude of the current density vector.

**Energy to energy coupling**

This coupling effect occurs via the energy sink term due to thermal radiation emitted from the hot plasma gases in the model.

By the assumption of local thermodynamic equilibrium, $Q_e$ becomes a function of the plasma temperature only. However, due to the complexity of the dependence of thermal radiation emission and absorption on the composition of the plasma material, this function is highly non-linear. It also tends to rise extremely rapidly as temperature increases. An example of such a function from the data of Naghizadeh-Kashani et al.\textsuperscript{43} for the total emission coefficient of air is shown in Figure 6.

*Figure 6: $Q_e$ for air as a function of temperature*
Convective coupling

As the DC plasma arc system is primarily a fluid flow problem, the convective transport of certain fields provides an important coupling effect.

In the case of the Navier-Stokes equations, the velocity fields feed back into the equations in a non-linear fashion by convecting themselves, producing the familiar complexity and potential for chaotic behaviour of fluid flow. In the case of the energy equation, the fluid flow acts to transport the temperature field and hence feed forward into other parts of the model.

Effects of convective coupling manifest themselves in the form of turbulence - the equations of fluid flow frequently undergo a transition between laminar flow, in which a steady state exists, and turbulent, in which the flow is increasingly unsteady and possibly chaotic, as the velocity and other parameters change. For models in which the physical properties of the fluid are treated as constant, the transition is governed in part by the Reynolds number of the flow:

\[
Re = \frac{Lv_{max}}{v}
\]  

(67)

For the small to moderate sized DC plasma arcs considered in this work the fluid flow is expected to be in the transition region between laminar and turbulent, and occasionally into the turbulent regime, where dynamic modelling becomes necessary to capture transient effects in the flow.

Energy to electromagnetics coupling

The coupling between the energy and electromagnetics fields is an important part of the DC plasma arc formation process. It occurs via the plasma electrical conductivity.

As with thermal radiation, the LTE assumption implies that the electrical conductivity of the plasma is a function of the plasma temperature only. Various molecules and atoms lose electrons and contribute to the electron density of the material as it heats up, thus the electrical conductivity rises with temperature. The behaviour with temperature is however quite non-linear, with the conductivity rising abruptly from close to zero (a minimum value is set in the model) as the ionisation temperature of the gas is passed. An example of this dependence from the data of Boulos...
et al\textsuperscript{45} for air plasmas is shown in Figure 7.

![Figure 7: $\sigma$ of air as a function of temperature](image)

It is important to note that the shape of the arc column in distributed parameter arc models such as this is not specified in advance - it is a calculated result. The location and shape of the boundary of the arc is primarily due to the non-linear behaviour of electrical conductivity with temperature, as shown in Figure 7. The radius of the arc at the cathode surface is well-defined - it is one of the boundary conditions and is determined by the specified value of $j_i$. In the body of the arc the radius becomes more poorly defined, but may be identified by either the limit at which the gas/plasma stops conducting electricity appreciably (the “electrical boundary”), or the limit at which it stops emitting visible light (the “visible boundary”). Both of these would be determined by contour surfaces of constant temperature.

The location of the plasma ionisation temperature isotherm in the temperature field at any given time serves to separates the electrically-conducting region ("arc") from the non-conducting region ("not-arc"). This isotherm is calculated automatically as part of the model, and no special treatment of the governing equations is required at the arc boundary.

### 2.5 Summary of governing equations

Fluid flow in primitive variable formulation:

$$
\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} + \nabla p = \nu \nabla^2 \mathbf{v} + \frac{\mathbf{F}}{\rho}
$$

(8)
\[ \nabla^2 p = -\nabla \cdot (|v| \nabla |v|) + \frac{\nabla \cdot F}{\rho} \]  

(13)

Fluid flow in vorticity-stream function formulation (2D):

\[ \frac{\partial \omega}{\partial t} + |v| \nabla \omega = \nu \nabla^2 \omega + \frac{1}{\rho} \left( \frac{\partial F}{\partial x} - \frac{\partial F}{\partial y} \right) \]  

(25)

\[ \nabla^2 \psi = \omega \]  

(26)

\[ v_x = -\frac{\partial \psi}{\partial y}, \quad v_y = \frac{\partial \psi}{\partial x} \]  

(27)

Fluid flow in gauge method formulation:

\[ \frac{\partial a}{\partial t} + |v| \nabla a = \nu \nabla^2 a + \frac{F}{\rho} \]  

(35)

\[ \nabla^2 \theta = \nabla \cdot a \]  

(36)

\[ v = a - \nabla \theta \]  

(37)

\[ F = j \times B \]  

(38)

Energy conservation:

\[ \frac{\partial T}{\partial t} + v \cdot \nabla T = \alpha \nabla^2 T + \frac{Q}{\rho C_p} \]  

(42)

Electromagnetic fields:

\[ j = -\sigma \nabla \phi \]  

(48)

\[ \nabla \cdot (\sigma \nabla \phi) = 0 \]  

(49)
\[ B = \nabla \times A \]  \hspace{1cm} (55)

\[ \nabla^2 A = -\mu_0 j \]  \hspace{1cm} (58)
Chapter 3 - Numerical Considerations

3.1 Spatial geometry and discretisation methods

The governing equations for the DC plasma arc model are solved approximately by the use of finite differences on a regular 2D or 3D structured cartesian mesh. A finite region immediately surrounding the arc in the central area of the furnace is modelled, consisting of the gas space between the bottom surface of the graphite electrode, and the top surface of the molten bath that forms the anode. For three-dimensional models, the geometry is shown in Figure 8.

Figure 8: Geometry of the solution region for 3D models

The domain is rectangular in shape, and its size is given by the range of each dimension. The $x$-dimension runs from $x = 0$ to $x = X$, similarly the $y$-dimension runs from $y = 0$ to $y = Y$, and the $z$-dimension from $z = 0$ to $z = Z$.

The electrode surface and cathode spot are located on Boundary 1, and extend over a predefined area - the remainder of Boundary 1 is treated as though it was a wall surface. The molten bath forming the anode of the furnace is represented by Boundary 2, and Boundaries 3-6 are treated as the walls of the furnace vessel for the purposes of the calculation.

In two dimensions, the solution region is confined to the $x$-$y$ plane, and is assumed to extend indefinitely in the $z$-direction. Cartesian geometry is retained for the 2D solvers, in order to capture asymmetric effects that may arise from the arc dynamics - these asymmetries are impossible to
capture with the axisymmetric geometry more traditionally used in two-dimensional arc modelling. The geometry is shown schematically in Figure 9.

*Figure 9: Geometry of the solution region for 2D models*

Grid nomenclature and indexing

The basic spatial discretisation of the solution region is accomplished by dividing each dimension into a number of steps of equal length. The position of each successive step locates a grid point along that dimension, and a set of two (2D) or three (3D) \(i, j, k\) indices locates a grid point in space.

Harlow and Welch\(^{14}\) introduced an alternative spatial discretisation method as part of the original MAC solver for fluid flow. This involved offsetting the grids used for the calculation of the velocity components by half a grid step in the direction of the component - that is, \(v_x\) was offset along the \(x\) dimension, \(v_y\) offset along the \(y\) dimension, and so on. This confers a number of advantages to any scheme using primitive (or related) variables. The divergence and curl operators have more natural numerical interpretations in such a scheme, of particular importance to incompressible flows where the enforcement of various divergence-free conditions is a requirement.

The MAC offset grid geometry is discussed in some detail in Harlow and Welch\(^{14}\), E and Liu\(^{35}\), Johnston and Liu\(^{36}\) and many others, and still finds use in contemporary numerical fluid flow solvers.

It is important to note that parts of the grids for the velocity components and pressure or gauge variable may lie outside the physical boundary of the domain (indicated by \(i = 0\) and \(j = 0\) in Figure
10). This adds some complexity to the specification of boundary conditions using these grids.

The classical MAC grid uses variables defined at both cell centres and grid vertices. A slightly simpler staggered grid based on the MAC concept can be introduced by requiring that all non-offset variables (including the pressure or gauge variables) use the same grid. Such a grid retains all the useful features of the classical MAC grid but reduces the geometrical complexity to some degree, as well as making numerical implementation of the gauge method for fluid flow considerably easier.

This grid is shown below, in Figure 10. The standard finite difference grid is defined at the points indicated with circles, the x-offset variables are defined at the points indicated by triangles, and the y-offset variables at the points indicated by diamonds.

Figure 10: Finite difference grids used, staggered and non-staggered

On the non-staggered grid, all the field variables of interest are defined at the same set of discrete grid points \((i,j)\), for \(i = 0, 1, 2, \ldots I\), and \(j = 0, 1, 2, \ldots J\). Logical extension to 3D is performed using an additional index \(k\) \((k = 0, 1, 2, \ldots K)\) and grid spacing \(\delta z\) for the \(z\) direction, with the \(v_z\) component of velocity being offset in an equivalent fashion.
Discrete operators

It is useful for the discussions that follow to define a number of spatial finite difference operators that act on the grids shown. These operators will replace partial derivatives and other terms in the DC plasma arc model equations with their finite difference analogues, producing a numerical approximation.

Consider the non-staggered grid first. For an arbitrary variable \( u \) defined on such a grid, define the second order centred difference operator in the \( x \)-direction by:

\[
\frac{\partial u}{\partial x} \approx D_x^u u_{i,j,k} = \frac{u_{i+1,j,k} - u_{i-1,j,k}}{2 \delta x}
\]  
(68)

The operators for the partial derivatives in the \( y \)- and \( z \)-direction follow logically.

The second order approximation for the second derivative in the \( x \)-direction is given by:

\[
\frac{\partial^2 u}{\partial x^2} \approx D_x^u D_x^u u_{i,j,k} = \frac{u_{i+1,j,k} - 2u_{i,j,k} + u_{i-1,j,k}}{\delta x^2}
\]  
(69)

Again, the second derivative expressions for the \( y \)- and \( z \)-directions are similar, using the \( j \) index and \( \delta y \), or the \( k \) index and \( \delta z \) respectively.

It is also useful to define the second order finite difference approximation to the Laplace operator in 3D:

\[
\nabla^2 u \approx \Delta_h u_{i,j,k} = D_x^2 u_{i,j,k} + D_y^2 u_{i,j,k} + D_z^2 u_{i,j,k}
\]  
(70)

and in 2D:

\[
\nabla^2 u \approx \Delta_h u_{i,j} = D_x^2 u_{i,j} + D_y^2 u_{i,j}
\]  
(71)

These definitions can be applied to the case of staggered grids too. For example, the centred difference difference operator acting on a variable offset in the \( x \)-direction would be written as
\[ D_x u_{i+1/2,j,k} = \frac{u_{i+3/2,j,k} - u_{i-1/2,j,k}}{2 \delta x} . \] (72)

It is also useful to define half-step centred difference operators for use on staggered grids. In the \( x \)-direction this is given by:

\[ \frac{\partial u}{\partial x} \approx d_x u_{i,j,k} = \frac{u_{i+1/2,j,k} - u_{i-1/2,j,k}}{\delta x} . \] (73)

Again, the expressions for the \( y \)- and \( z \)-related derivatives follow naturally.

It is important to note that for the DC plasma arc model, we will generally use \( \delta x = \delta y = \delta z = \delta l \) to simplify the numerical mathematics and improve computational performance, and these approximations hence give overall spatial accuracy of order \( O(\delta F) \). All derivations will however be performed considering independent values of \( \delta x \), \( \delta y \) and \( \delta z \).

**Time discretisation and stability**

E and Liu35 made a detailed study of finite difference approximations to the convection-diffusion equation (74), which can be regarded as a very simple model of the Navier-Stokes equations and other transport phenomena.

\[ \frac{\partial u}{\partial t} + v_u \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2} . \] (74)

In particular, for the case of \( \nu = 0 \), the use of second-order centred differences for the spatial derivatives and explicit forward Euler numerical discretisation in time is known to be unconditionally unstable. This has negative implications for solvers that aim to study fluid flow at moderate to high Reynolds numbers, as in this situation \( v_u \) is generally much larger than \( \nu \) and the time step has to satisfy the onerous constraint

\[ \delta t < \frac{\delta x^2}{2 \nu} , \delta t < \frac{2 \nu}{v_u^2} . \] (75)
to ensure stability.

These constraints also lead to the well-known cell Reynolds number limitations for explicit centred-difference finite difference methods.

The stability constraints have traditionally been overcome by the use of semi- or fully-implicit methods in time which eliminate the difficulties associated with (75), but at great computational cost, as the solution of the set of discretised equations produced at each time step becomes considerably more expensive. Weighted discretisation schemes for the convection term such as upwinding may also be used to enhance stability, however, this can reduce the spatial accuracy of the numerical scheme as well as introducing artificial numerical diffusion.

E and Liu demonstrate that explicit solvers can still be used in conjunction with centred-difference finite differences, provided the solvers possess certain properties relating to their stability when solving (74). These properties are not present in the forward Euler method, however, they are found in higher order Runge-Kutta methods. In particular, a solver using standard fourth-order Runge Kutta (RK4) time stepping in conjunction with second-order centered differences is stable subject to the following conditions:

\[ \delta t < \frac{C_1 \delta x^2}{4 \nu}, \quad \delta t < \frac{C_2 \delta x}{\nu_a} \]  

(76)

\( C_1 \) and \( C_2 \) are arbitrary constants, generally given values between 1 and 2. Compared with (75), (76) offers a much less restrictive constraint on the size of the time step to be used, and permits transient calculations of high velocity flows to be performed in a reasonable amount of time.

### 3.2 Fluid flow models

The numerical discretisation of the various models of fluid flow is performed using second-order centered finite differences in space and an explicit forward time-step solver for the temporal variable. The spatial discretisations are dependent on the grids used.

The key decision factors in the choice of methods were algorithm simplicity, performance, and easy
transition to three dimensions. While there is little doubt that finite element and finite volume techniques are much more flexible and robust tools for general purpose computational fluid dynamics, in this particular application it was felt that their flexibility is not required and would have simply added computational overhead.

Derivations of the primitive variable and vorticity formulations are restricted to the two-dimensional case as these formulations are of interest for comparison and testing purposes only. The discretisation of the gauge method is presented in full three dimensions.

**Primitive variable formulation**

For the solution in primitive variables, methods developed by Johnston and Liu\(^{36,37}\) were used. These authors examined numerical discretisations of the Navier-Stokes equations in primitive variables using non-staggered grids, which are desirable as they permit simpler extension to more general finite volume and element methods, as well as arbitrary unstructured meshes.

For the two-dimensional case, (13) may be written as:

\[
\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} = 2 \left[ \frac{\partial v_x}{\partial x} \frac{\partial v_y}{\partial y} - \frac{\partial v_x}{\partial y} \frac{\partial v_y}{\partial x} \right] + \frac{1}{\rho} \left( \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} \right) \tag{77}\]

We then apply standard second-order centered differencing to the spatial derivative operators in (8) and (77). With these definitions, we obtain the finite difference approximation for this formulation:

\[
\frac{d}{dt} v_{x,i,j} + v_{x,i,j} \left( D_x v_{x,i,j} \right) + v_{y,i,j} \left( D_y v_{x,i,j} \right) + D_x p_{i,j} = \nu \Delta_h v_{x,i,j} + \frac{F_{x,i,j}}{\rho} \tag{78}\]

\[
\frac{d}{dt} v_{y,i,j} + v_{x,i,j} \left( D_x v_{y,i,j} \right) + v_{y,i,j} \left( D_y v_{y,i,j} \right) + D_y p_{i,j} = \nu \Delta_h v_{y,i,j} + \frac{F_{y,i,j}}{\rho} \tag{79}\]

\[
\Delta_h p_{i,j} = 2 \left[ \left| D_x v_{x,i,j} \right| \left| D_y v_{x,i,j} \right| - \left| D_y v_{x,i,j} \right| \left| D_x v_{y,i,j} \right| \right] + \frac{1}{\rho} \left| D_x F_{x,i,j} + D_y F_{y,i,j} \right| \tag{80}\]

Boundary conditions supplemented to (78) and (79) are given by (11), and on a rectangular domain using a non-staggered grid become
\[ \begin{align*}
    v_{x,i,0} &= 0, v_{x,i,J} = 0, v_{x,0,i} = 0, v_{x,1,i} = 0, \\
    v_{y,i,0} &= 0, v_{y,i,J} = 0, v_{y,0,i} = 0, v_{y,1,i} = 0.
\end{align*} \tag{81} \]

Boundary conditions for (80) are more complicated, and can be specified in a number of ways. The simplest and most intuitive of these is derived by Johnston and Liu\textsuperscript{36} and referred to as a “local pressure boundary condition”.

The principal justification for the use of this type of boundary condition is that it closely resembles a highly successful local vorticity boundary condition derived by Thom in the vorticity-stream function formulation. This boundary condition was shown in E and Liu\textsuperscript{35} to be equivalent to several other considerably more complex non-local boundary conditions for vorticity, and when used in conjunction with a second-order or higher spatial discretisation scheme, to preserve overall second order accuracy of the flow fields. Similarly, the local pressure boundary condition of Johnston and Liu maintains at least second order spatial accuracy while at the same time being very simple to compute due to its local nature.

We consider a grid point on the \((i,0)\) \((y = 0)\) boundary, as per Figure 11.

*Figure 11: Example boundary grid point on a non-staggered grid*
The discrete analogue of (14) on the boundary may be written as:

\[
\left[ \frac{\partial p}{\partial y} \right]_{j,0} \approx \nu \left( D_x v_{y,i,0} + D_y v_{y,i,0} \right) + \frac{F_{y,i,0}}{\rho} + \left( v_{y,i+1,0} - 2 v_{y,i,0} + v_{y,i-1,0} + \frac{v_{y,i,1} - 2 v_{y,i,0} + v_{y,i,-1}}{\delta y^2} \right) \frac{F_{y,i,0}}{\rho} \quad (82)
\]

Here, the -1 index refers to an imaginary grid point outside the calculation domain. Due to (81), we can remove many of the terms in (82) and write it as

\[
\left[ \frac{\partial p}{\partial y} \right]_{j,0} = \nu \left( \frac{v_{y,i+1} + v_{y,i-1}}{\delta y^2} \right) + \frac{F_{y,i,0}}{\rho} \quad (83)
\]

Analysing the discrete version of the continuity equation (9) in a similar way on the same boundary suggests a definition for the imaginary grid point:

\[
\begin{align*}
D_x v_{x,i,0} + D_y v_{y,i,0} &= 0 \\
v_{x,i+1,0} - v_{x,i-1,0} + \frac{v_{y,i+1} - v_{y,i-1}}{2 \delta y} &= 0 \quad (84)
\end{align*}
\]

With this, (83) becomes:

\[
\left[ \frac{\partial p}{\partial y} \right]_{j,0} = 2 \nu \left( \frac{v_{y,i+1} + v_{y,i-1}}{\delta y^2} \right) + \frac{F_{y,i,0}}{\rho} \quad (85)
\]

The relationship for other boundaries follows similarly. This can be solved together with (80) to produce a complete pressure field.

In order to impart stability to the scheme subject to reasonable constraints on the time step size, the temporal discretisation of the convective and diffusive components of the Navier-Stokes equations is accomplished using RK4. The explicit RK4 algorithm - starting with a known velocity and pressure field \((v_{x0}, v_{y0}, p_0)\) at the beginning of the step - operates by calculating two estimates for the \(v_x\) and \(v_y\) fields at the halfway point of the time step (at \(\frac{1}{2}\delta t\)), and two at the end (at \(\delta t\)). These four estimates are then combined linearly to obtain the final values of \(v_x\) and \(v_y\) for the end of the time

53
step.

The first stage calculates the $v_{il}$ and $v_{jil}$ intermediate fields. Dropping the $(i,j)$ subscripts for clarity and using the $x$-component as an example, we have

$$\frac{v_{ixl} - v_{ix0}}{\frac{1}{2} \delta t} + v_{ix0} \left[ D_x v_{ix0} \right] + v_{jx0} \left[ D_x v_{jx0} \right] + D_x p_0 = v \Delta_h v_{ix0} + F_{x0} \quad . \tag{86}$$

$v_{jil}$ follows similarly. The intermediate pressure $p_j$ is then found by solving (80) using $v_{xj}$ and $v_{jij}$ for the velocities on the right-hand side, and the appropriate boundary conditions. The second stage of the calculation produces intermediate fields $v_{ix2}$, $v_{jx2}$, and $p_2$ in a similar fashion:

$$\frac{v_{ix2} - v_{ix0}}{\frac{1}{2} \delta t} + v_{ix0} \left[ D_x v_{ix0} \right] + v_{jx0} \left[ D_x v_{jx0} \right] + D_x p_1 = v \Delta_h v_{ix1} + \frac{F_{x0}}{\rho} \quad . \tag{87}$$

$v_{jx2}$ follows similarly, and $p_2$ is calculated from (80) using $v_{x2}$ and $v_{jx2}$. The third stage commences with the calculation of the $v_{ix3}$ and $v_{jx3}$ intermediate fields by

$$\frac{v_{ix3} - v_{ix0}}{\delta t} + v_{ix0} \left[ D_x v_{ix0} \right] + v_{jx0} \left[ D_x v_{jx0} \right] + D_x p_2 = v \Delta_h v_{ix2} + \frac{F_{x0}}{\rho} \quad , \tag{88}$$

with $v_{jx3}$ following similarly. $p_3$ is calculated by solving (80) using the new intermediate velocities, $v_{ix3}$ and $v_{jx3}$. The fourth stage computes the last intermediate velocity field, and the final value of the field for the time step by linear combination of the various intermediate fields:

$$\frac{v_{ix4} - v_{ix0}}{\delta t} + v_{ix0} \left[ D_x v_{ix0} \right] + v_{jx0} \left[ D_x v_{jx0} \right] + D_x p_3 = v \Delta_h v_{ix3} + \frac{F_{x0}}{\rho}$$
$$v_x = \frac{1}{h} \left[ -3 v_{x0} + 2 v_{ix1} + 4 v_{ix2} + 2 v_{ix3} + v_{ix4} \right] \tag{89}$$

$v_{jx4}$ and $v_j$ are calculated similarly. The final pressure $p$ for the time step is then calculated by solving (80) using the velocities $v_i$ and $v_j$.

After each set of stages is complete, the source terms are updated for the following step, the time step is adjusted according to the constraints (76), and the calculation for the next step may begin.
Vorticity formulation

In the vorticity formulation, the velocity and pressure are replaced by intermediate variables, the vorticity vector and vector potential. These are defined on standard non-staggered grids.

In the two-dimensional case these both reduce to scalar variables, the vorticity and the stream function. These are governed by equations (25) and (26). Replacing \( v_h \) and \( v_l \) in (25) with the definitions in (27) and applying second-order finite difference operators produces the discrete version of this formulation:

\[
\frac{d}{dt} \omega_{i,j} - D_y \psi_{i,j} D_x \omega_{i,j} + D_x \psi_{i,j} D_y \omega_{i,j} = v \Delta_h \omega_{i,j} + \frac{1}{\rho} \left[ D_x F_{y,i,j} - D_y F_{x,i,j} \right] \quad (90)
\]

\[
\Delta_h \psi_{i,j} = \omega_{i,j} \quad (91)
\]

To produce a boundary condition for \( \omega \), consider (91) at a point \((i,0)\) on the boundary of the domain, as per Figure 11. This may be written as:

\[
\frac{\psi_{i+1,0} - 2\psi_{i,0} + \psi_{i-1,0}}{\delta x^2} + \frac{\psi_{i,1} - 2\psi_{i,0} + \psi_{i,-1}}{\delta y^2} = \omega_{i,0} \quad (92)
\]

The boundary conditions for \( \psi \) as given by (28) may be expressed as follows in a finite difference setting:

\[
\psi_{i,-1,0} = \psi_{i,0} = \psi_{i+1,0} = 0, \quad \frac{\psi_{i,1} - \psi_{i,-1}}{2 \delta y} = 0 \quad (93)
\]

With this, (92) may be simplified to:

\[
\omega_{i,0} = \frac{2}{\delta y^2} \psi_{i,1} \quad (94)
\]

The relationship for other boundaries follows similarly. This is the familiar boundary condition.
derived by Thom\textsuperscript{38}. The procedure is quite similar to that used to produce the local pressure boundary condition in the primitive variable formulation.

As for the primitive variable formulation, the discretisation of (90) in time is accomplished by using RK4 time stepping, comprising four separate calculation stages per time step. The explicit RK4 algorithm starts with known fields $\psi_0$ and $\omega_0$ at the beginning of the step.

The first stage calculates the $\omega_1$ intermediate field. Dropping the $(i,j)$ subscripts for clarity, we have

$$\frac{\omega_1 - \omega_0}{\frac{1}{2} \delta t} - D_j \psi_0 D_x \omega_0 + D_x \psi_0 D_j \omega_0 = \nu \Delta_h \omega_0 + \frac{1}{\rho} |D_j F_{x0} - D_x F_{y0}| + \frac{1}{\rho} |D_j F_{x0} - D_x F_{y0}| \ .$$  \hspace{1cm} (95)

The intermediate field $\psi_1$ is then calculated from $\omega_1$ by solving (91) using the appropriate boundary conditions. The second stage calculates the $\omega_2$ intermediate field:

$$\frac{\omega_2 - \omega_0}{\frac{1}{2} \delta t} - D_j \psi_1 D_x \omega_1 + D_x \psi_1 D_j \omega_1 = \nu \Delta_h \omega_1 + \frac{1}{\rho} |D_j F_{x0} - D_x F_{y0}| \ .$$ \hspace{1cm} (96)

$\psi_2$ is again calculated from $\omega_2$ via solution of (91). In the third stage, the calculation of $\omega_3$ is performed using

$$\frac{\omega_3 - \omega_0}{\delta t} - D_j \psi_2 D_x \omega_2 + D_x \psi_2 D_j \omega_2 = \nu \Delta_h \omega_2 + \frac{1}{\rho} |D_j F_{x0} - D_x F_{y0}| \ ,$$ \hspace{1cm} (97)

with $\psi_3$ calculated using $\omega_3$ in the solution of (91) as before. The final stage computes the last intermediate value $\omega_4$, and the final value of the $\omega$ field for the time step by linear combination:

$$\frac{\omega_4 - \omega_0}{\delta t} - D_j \psi_3 D_x \omega_3 + D_x \psi_3 D_j \omega_3 = \nu \Delta_h \omega_3 + \frac{1}{\rho} |D_j F_{x0} - D_x F_{y0}| \ .$$

$$\omega = \frac{1}{8} [-3 \omega_0 + 2 \omega_1 + 4 \omega_2 + 2 \omega_3 + \omega_4] \ .$$ \hspace{1cm} (98)

The final value of $\psi$ is calculated by solution of (91), using the updated $\omega$ field.
Each Runge-Kutta stage consists of a single explicit calculation, and a single Poisson solve. Once the procedure is completed, $dt$ and $F$ are updated (if necessary) using the new values of $\psi$ and $\omega$, and the calculation for the following step begins.

**Gauge formulation**

In the gauge method, the velocity and pressure are replaced by an auxiliary vector field and a gauge variable in order to simplify the Navier-Stokes equations. Both the two- and three-dimensional cases are of interest, as this is the formulation of choice for the full DC plasma arc model. We therefore present the derivation of the finite difference approximation in 3D - reduction to the 2D case is generally trivial and involves omitting the $k$ index and any terms involving the $z$-component of fields or derivatives.

The numerical discretisation of the gauge method equations, (35) - (37), is accomplished using the alternative staggered MAC grid, as shown in Figure 10. The components of the $a$ and $v$ variables are defined on grids offset from the main grid by a half-step. With this in mind, the finite difference approximation of (35) using second-order centered differences is given by:

$$
\begin{align*}
\frac{d}{dt} a_{x,i+1/2,j,k} + & v_{x,i+1/2,j,k} D_x v_{x,i+1/2,j,k} + v_{y,avg} D_y v_{x,i+1/2,j,k} + v_{z,avg} D_z v_{x,i+1/2,j,k} \\
& = \nu \Delta t a_{x,i+1/2,j,k} + \frac{1}{\rho} F_{x,i+1/2,j,k}
\end{align*}
$$

(99)

$$
\begin{align*}
\frac{d}{dt} a_{y,i,j+1/2,k} + & v_{y,avg} D_y v_{y,i,j+1/2,k} + v_{x,avg} D_x v_{y,i,j+1/2,k} + v_{z,avg} D_z v_{y,i,j+1/2,k} \\
& = \nu \Delta t a_{y,i,j+1/2,k} + \frac{1}{\rho} F_{y,i,j+1/2,k}
\end{align*}
$$

(100)

$$
\begin{align*}
\frac{d}{dt} a_{z,i,j+1/2} + & v_{x,avg} D_x v_{z,i,j+1/2} + v_{y,avg} D_y v_{z,i,j+1/2} + v_{z,avg} D_z v_{z,i,j+1/2} \\
& = \nu \Delta t a_{z,i,j+1/2} + \frac{1}{\rho} F_{z,i,j+1/2}
\end{align*}
$$

(101)

Due to the component grids being offset from each other, some interpolation is required when using $y$-components of $v$ in the $x$-equation, and so forth - this is performed using bilinear interpolation.
The gauge variable equation (36) is discretised on the main grid, offset from all three of the vector component grids. The divergence of the $a$ field then follows naturally using half-step derivatives.

$$\Delta_h \theta_{i,j,k} = d_x a_{x,i,j,k} + d_y a_{y,i,j,k} + d_z a_{z,i,j,k}$$

(102)

Calculation of $v$ on the offset grids is accomplished via the discrete version of (37):

$$v_{x,i+1/2,j,k} = a_{x,i+1/2,j,k} - d_x \theta_{i+1/2,j,k}$$

(103)

and similarly for the $v_{y,i,j,k+1/2}$ and $v_{z,i,j,k+1/2}$ components.

Boundary conditions for the $v$, $a$, and $\theta$ variables are given by (11) together with (33) or (34). As (33) and (34) are equivalent, we can choose either - for the offset grid in use, (33), with homogeneous Neumann boundary conditions supplemented to (36), is easier to implement.

Due to the offset positions of the vector component grids, the boundary conditions are not entirely straightforward. Using a point on the lower $x$ boundary at $(i,0,k)$ as an example, as per Figure 13, it is easy to see that the $v_x$ and $a_x$ components do not coincide with the physical boundary of the problem - they are defined only at $(i,-1/2,k)$ and $(i,1/2,k)$. In order to overcome this, we use the reflection boundary condition as originally studied in the context of the MAC scheme by Harlow and Welch\textsuperscript{34} - to achieve $v_{y,i,0,k} = 0$, take the average of the grid points either side of $(i,0,k)$:

$$v_{y,i,0,k} = \frac{1}{2} \left( v_{y,i,-1/2,k} + v_{y,i,1/2,k} \right)$$

$$v_{y,i,1/2,k} = -v_{y,i,-1/2,k}$$

(104)

The grids for $v_y$ and $v_z$ are offset only in the $x$- and $z$-directions respectively, and setting $v_{x,i,0,k} = v_{z,i,0,k} = 0$ is thus trivial.

The normal component of the $a$ field at the boundary, $a_y$, is also required to be zero, so the same procedure applies:

$$a_{y,i,0,k} = \frac{1}{2} \left( a_{y,i,-1/2,k} + a_{y,i,1/2,k} \right)$$

$$a_{y,i,1/2,k} = -a_{y,i,-1/2,k}$$

(105)

The tangential components of the $a$ field, $a_x$ and $a_z$, do coincide with the physical boundary at $(i,0,k)$, and are given by the tangential derivatives of the gauge variable in the $x$- and $z$-directions.
respectively. Recall that these components are also defined on offset grids, but they are offset in the in the x- and z-directions only, as for \( v_x \) and \( v_z \). Hence (33) is approximated using simple half-step centered differences:

\[
a_{x,i+1/2,0,k} = d_x \theta_{i+1/2,0,k}, \quad a_{z,i,0,k+1/2} = d_z \theta_{i,0,k+1/2}
\] (106)

The extension of (104) - (106) to the other boundary surfaces in the domain is performed in an analogous fashion.

The time stepping procedure for the gauge method follows the same scheme developed for the primitive variable and vorticity based solvers - RK4. Starting with initial values of the fields \((v_{i0}, v_{j0}, a_{i0}, a_{j0}, a_{k0}, \theta_{0})\), four separate calculation stages must be performed in order to advance the simulation by a single time step.

The first stage begins with calculation of the \( a_{i1}, a_{j1}, \) and \( a_{k1} \) intermediate fields. We drop the \((i,j,k)\) subscripts for clarity, although it should be remembered that the calculations involving \( a_i, a_j, a_k, v_i, v_j, \) and \( v_k \) are performed on offset grids as per (99) - (103). Using the x-component as an example, we have

\[
\frac{a_{ix} - a_{ix0}}{\frac{1}{2} \delta t} + v_{ix0} \left( D_x v_{ix} \right) + v_{ix0} \left( D_y v_{ix} \right) + v_{ix0} \left( D_z v_{ix} \right) = \nu \Delta_h a_{ix0} + \frac{F_{x0}}{\rho} .
\] (107)

The calculations for the \( a_{ix} \) and \( a_{jx} \) components are performed similarly. The intermediate gauge variable for the stage is found by using \( a_{ix1}, a_{jx1}, \) and \( a_{kx1} \) in the solution of (102) to give \( \theta_i \).

Intermediate velocities \( v_{ix1}, v_{jx1} \) and \( v_{kx1} \) are then calculated from \( a_{ix1}, a_{jx1}, a_{kx1} \) and \( \theta_i \) using (103). The second stage calculates the \( a_{ix2}, a_{jx2}, \) and \( a_{kx2} \) intermediate fields using

\[
\frac{a_{ix2} - a_{ix0}}{\frac{1}{2} \delta t} + v_{ix0} \left( D_x v_{ix} \right) + v_{ix0} \left( D_y v_{ix} \right) + v_{ix0} \left( D_z v_{ix} \right) = \nu \Delta_h a_{ix1} + \frac{F_{x0}}{\rho} ,
\] (108)

with the \( a_{jx2} \) and \( a_{kx2} \) calculations following similarly. These intermediate fields are then used in (102), which is solved to give \( \theta_i \). Intermediate velocities \( v_{ix2}, v_{jx2} \) and \( v_{kx2} \) are again calculated from \( a_{ix2}, a_{jx2}, a_{kx2} \) and \( \theta_i \) using (103). The third stage calculations begin with the \( a_{ix3}, a_{jx3}, \) and \( a_{kx3} \)
intermediate fields:

\[
\frac{a_{i,x}-a_{i,0}}{\delta t} + v_{i,x}\left[D_{x}v_{x,i}\right] + v_{i,y}\left[D_{y}v_{x,i}\right] + v_{i,z}\left[D_{z}v_{x,i}\right] = \nu \Delta_{h} a_{i,x} + \frac{F_{x,0}}{\rho}
\] (109)

The \(a_{i,x}\) and \(a_{i,z}\) calculations follow similarly. The intermediate fields are used as before in (102), which is solved to give \(\theta_{i}\). \(a_{i,x}, a_{i,y}, a_{i,z}\) and \(\theta_{i}\) are used to calculate intermediate velocities \(v_{i,x}, v_{i,y}\) and \(v_{i,z}\) from (103). The fourth and final stage begins with the calculation of the last intermediate variable and the final value for the time step:

\[
\frac{a_{i,x}-a_{i,0}}{\delta t} + v_{i,x}\left[D_{x}v_{x,i}\right] + v_{i,y}\left[D_{y}v_{x,i}\right] + v_{i,z}\left[D_{z}v_{x,i}\right] = \nu \Delta_{h} a_{i,x} + \frac{F_{x,0}}{\rho}
\]

\[
a_{i,z} = \frac{1}{6}\left[-3a_{i,0} + 2a_{i,x} + 4a_{i,y} + 2a_{i,z} + a_{i,0}\right]
\] (110)

The calculation for \(a_{i}\) and \(a_{z}\) follows similarly. The final value of \(\theta\) for the time step is then found by solving (102) using \(a_{i}, a_{x},\) and \(a_{z}\). Finally, \(v_{i}, v_{x},\) and \(v_{z}\) for the time step are calculated using (103).

Each Runge-Kutta stage consists of three explicit calculations and a single Poisson solve. Once the procedure is completed, \(\delta t\) and \(F\) are updated (if necessary) using the new values of \(v\) and other variables, and the calculation for the following step begins.

The gauge method in three dimensions is seen to be very efficient - it incurs minimal extra calculation overhead as compared to the 2D version. The major computational load is a single Poisson equation solve at every RK4 step, regardless of the dimensionality of the problem.

**Adaptive time stepping**

For all the fluid flow solvers considered here, the use of the RK4 time stepping algorithm produces a constraint on the maximum size of the time step allowable to ensure numerical stability, as expressed by the inequalities in (76). For the DC plasma arc model, the velocity used in (76) is not a constant, but varies from step to step as the arc dynamics evolve over time.

After each set of RK4 stages is computed, the velocity component fields are searched for the
maximum. This maximum value is used in place of \( v_a \) in (76). Two limiting time step sizes are then determined from the two inequalities and compared, with the smaller of the two being used for the following time step calculation.

**Source terms**

For the DC plasma arc system, the source terms for fluid flow are given by the Lorentz force, described by (38). In three dimensions, the components of (38) are:

\[
F_x = j_y B_z - j_z B_y
\]  
(111)

\[
F_y = j_z B_x - j_x B_z
\]  
(112)

\[
F_z = j_x B_y - j_y B_x
\]  
(113)

As \( j \) and \( B \) are defined on the non-staggered grid, the finite difference definition of (111) - (113) is trivial for solvers whose variables are also defined on such grids. For the staggered grids used by the gauge method, linear interpolation is applied according to

\[
F_{x,i+1/2,j,k} = \frac{1}{2} \left[ F_{x,i,j,k} + F_{x,i+1,j,k} \right]
\]  
(114)

with the \( F_{y,i+1/2,j,k} \) and \( F_{z,i,j,k+1/2} \) components following similarly.

In two dimensions, only the components \( j_x \) and \( j_y \) of \( j \), and the component \( B_z \) of \( B \), are non-zero - the finite difference approximation of (38) in 2D follows accordingly.

**3.3 Energy model**

The energy model for the DC plasma arc problem is discretised on a standard non-staggered grid. Certain numerical difficulties are posed by the energy conservation equation, which may be overcome using operator splitting and a partially-implicit treatment of the source term.

We present the numerical analysis for three-dimensional problems below. Obtaining the finite difference discretisation in 2D from this is trivial, and is accomplished by dropping the \( k \) index and
any terms involving the $z$-component of fields or derivatives.

Discretisation of the energy balance equation

We apply second order centered difference operators to obtain the numerical approximation of (42). In three dimensions, this is written as follows:

$$
\frac{d}{dt} T_{i,j,k} + v_{x,i,j,k} D_x T_{i,j,k} + v_{y,i,j,k} D_y T_{i,j,k} + v_{z,i,j,k} D_z T_{i,j,k} = \alpha \Delta_k T_{i,j,k} + \frac{Q_{i,j,k}}{\rho C_p} \tag{115}
$$

Two points should be noted. Firstly, when (115) is used in conjunction with flow models that use staggered grids, $v_x$, $v_y$, and $v_z$ do not coincide with the grid used for calculation of $T$, and should be found by linear interpolation.

Secondly, the source term $Q$ as defined by (45) is a very strong function of temperature, and can dominate the other terms in (115) at certain points. This has the potential to cause numerical instability if naive temporal discretisations are used.

Temperature dependence of $Q_r$ and $\sigma$

As seen in the examples shown in Figures 6 and 7, both the volumetric radiation emission and the electrical conductivity are functions of temperature. As they are highly non-linear, these functions are brought into the model by piecewise linear approximations - the actual data for the plasma gas of interest over the range of 1000K to 33000K are discretised at a series of temperatures $T_n$ and input into the model algorithm. Values at any temperature $T_n < T < T_{n+1}$ are then determined by:

$$
\sigma = \sigma_n + \left( \frac{T - T_n}{T_{n+1} - T_n} \right) \left( \sigma_{n+1} - \sigma_n \right) \tag{116}
$$

$$
Q_r = Q_{r,n} + \left( \frac{T - T_n}{T_{n+1} - T_n} \right) \left( Q_{r,n+1} - Q_{r,n} \right) \tag{117}
$$

If the temperature of interest falls outside the range over which the functions are discretised, the data are extrapolated linearly from the end point temperatures.
Operator splitting of (115)

To facilitate development of a more numerically stable time stepping procedure for (115), we separate the convection/diffusion terms from the source terms by using first-order operator splitting. Introduce an intermediate variable $T^*$ defined as follows:

$$
\frac{d}{dt} T^*_{i,j,k} = \frac{Q_{i,j,k}}{\rho C_p}
$$

(118)

Different time discretisations may now be used for (118) and the convection/diffusion terms over the interval $\delta t$. The source term step is performed using semi-implicit backward Euler time stepping - although this is only first order in time, this is acceptable as the operator splitting is itself only first order accurate. Assuming a temperature $T_0$ at the beginning of the time step, the discretised version of (118) is:

$$
\frac{T^*_{i,j,k} - T_{0,i,j,k}}{\delta t} = \frac{1}{\rho C_p} \left( \frac{j_j f_{0,i,j,k}}{\sigma_{i,j,k}} + Q_{r,i,j,k} \right) = f_Q(T^*_{i,j,k})
$$

(119)

Due to the definitions (116) and (117), $f_Q$ is only known exactly at a set of discrete temperatures $T_n$. Applying linear interpolation for values of temperature $T_n < T < T_{n+1}$, we obtain a description of $f_Q$ that allows a solution to (119) to be found rapidly:

$$
f_Q(T^*_{i,j,k}) = f_Q(T_n) + \left( \frac{T^*_{i,j,k} - T_n}{T_{n+1} - T_n} \right) \left( f_Q(T_{n+1}) - f_Q(T_n) \right)
$$

(120)

As (120) is locally linear, the solution of (119) is trivial once the location of the solution has been narrowed down to a particular interval $(T_n, T_{n+1})$. This can easily be performed by a direct or binary search evaluating (119) over the sequence of $T_n$ values, provided the length of the sequence is not very large.

Once $T^*$ is known, the solution for $T$ at the end of the time step is then completed by solving for the
remaining terms in (115):

\[
d\frac{T_{i,j,k}+v_{x,i,j,k} D_x T_{i,j,k}+v_{y,i,j,k} D_y T_{i,j,k}+v_{z,i,j,k} D_z T_{i,j,k}}{dt} = \alpha \Delta_h T_{i,j,k}
\] (121)

As (121) embodies the convection and diffusion behaviour, RK4 is used to explicitly step it forward in time. Dropping the \((i,j,k)\) subscripts for clarity, the four stages may be written as:

\[
\frac{T_1 - T^*}{\frac{1}{\delta t}} + v_x D_x T^* + v_y D_y T^* + v_z D_z T^* = \alpha \Delta_h T^*
\] (122)

\[
\frac{T_2 - T^*}{\frac{1}{\delta t}} + v_x D_x T_1 + v_y D_y T_1 + v_z D_z T_1 = \alpha \Delta_h T_1
\] (123)

\[
\frac{T_3 - T^*}{\delta t} + v_x D_x T_2 + v_y D_y T_2 + v_z D_z T_2 = \alpha \Delta_h T_2
\] (124)

\[
\frac{T_4 - T^*}{\delta t} + v_x D_x T_3 + v_y D_y T_3 + v_z D_z T_3 = \alpha \Delta_h T_3
\] (125)

The value of \(T\) at the end of the time step is calculated as:

\[
T = \frac{1}{\delta t} \left(-3 T^* + 2 T_1 + 4 T_2 + 2 T_3 + T_4\right)
\] (126)

Spatial boundary conditions are supplied to (121) according to (43) or (44), depending on which of TBC 1 or 2 is in use. By way of example, consider a point on the \((i,0,k)\) boundary as per Figure 11. The second order centered difference approximation of (44) on the \(j = 0\) boundary gives:

\[
\frac{T_{i,1,k} - T_{i,-1,k}}{2 \delta y} = 0
\]

\[
T_{i,-1,k} = T_{i,1,k}
\] (127)

This expression is substituted into (121) as evaluated on the boundary to give the numerical boundary condition required.
For the specification of constant temperature boundary conditions, a somewhat different approach is taken. As the boundaries are generally much colder than the ionisation temperature of the plasma, severe numerical instability is possible in the electromagnetic field solution if the boundary grid points are simply set to a low constant value in the temperature solver (this would amount to setting $\sigma = 0$ on the boundaries). In order to ameliorate this effect, the boundary conditions are set approximately by assuming that the physical constant-temperature boundary is positioned one grid point outside the boundary of the main calculation grid. In the example above on the boundary $(i,0,k)$, this would amount to specifying:

$$T_{i,-1,k} = T_\infty$$

(128)

and then using the full expression of (121) evaluated on the grid boundary at $(i,0,k)$.

**Adaptive time stepping**

As part of the temperature solver makes use of the RK4 algorithm to solve the convection-diffusion problem for energy transport, it introduces an additional time step size constraint to the pair given by (76) that apply to the fluid flow problem. This additional constraint relates to the thermal diffusivity of the plasma material, and is given by:

$$\delta t < \frac{C_1 \delta x^2}{4 \alpha}$$

(129)

This is calculated and compared to the time steps calculated from (76), and the smallest is chosen for the following time step in the overall algorithm.

**3.4 Electromagnetic field model**

The numerical treatment of the model of electrodynamics used in the DC plasma arc model parallels that of the other sub-models in that it uses second order centered difference finite difference methods on regular rectangular grids. However, the electrical and magnetic fields follow quite different fundamental physical laws compared to the transport quantities such as velocity and temperature, and some subtleties must be considered during their expression in discretised form.
All electromagnetic field variables are solved on the main non-staggered grid, although intermediate variables in the solver algorithm make use of the staggered grid concept to facilitate the numerical analysis.

**Electric potential**

For the purposes of deriving a compact numerical definition of (49), we consider the components of the current density vector $j$ as being defined on a staggered MAC-style grid - that is, $j_i$ is offset by a half grid step in the $x$-direction, and so forth.

In three dimensions, the charge conservation equation (46) is then expressed using half-step finite difference operators as follows:

$$d_x j_{x,i,j,k} + d_y j_{y,i,j,k} + d_z j_{z,i,j,k} = 0$$

(130)

The relationship between current density and electric potential, (48), may be written in a similar fashion for discrete current densities defined on the staggered grid:

$$j_{x,i+1/2,j,k} = -\sigma_{i+1/2,j,k} d_x \phi_{i+1/2,j,k}$$

(131)

Components $j_{x,i+1/2,j,k}$ and $j_{z,i,k+1/2,j}$ follow similarly.

These together provide a finite difference approximation to (49). It is useful for clarity purposes to write out the discrete version of (130) using (131) in longhand form:

$$\sigma_{i+1/2,j,k} \left( \frac{\phi_{i+1,j,k} - \phi_{i,j,k}}{\delta x} \right) - \sigma_{i-1/2,j,k} \left( \frac{\phi_{i,j,k} - \phi_{i-1,j,k}}{\delta x} \right) + \sigma_{i,j+1/2,k} \left( \frac{\phi_{i,j+1,k} - \phi_{i,j,k}}{\delta y} \right) - \sigma_{i,j-1/2,k} \left( \frac{\phi_{i,j,k} - \phi_{i,j-1,k}}{\delta y} \right) + \sigma_{i,j,k+1/2} \left( \frac{\phi_{i,j,k+1} - \phi_{i,j,k}}{\delta z} \right) - \sigma_{i,j,k-1/2} \left( \frac{\phi_{i,j,k} - \phi_{i,j,k-1}}{\delta z} \right) = 0$$

(132)
The values of $\sigma$ at the offset grid locations are computed using linear interpolation. For example, in the x-component direction,

$$\sigma_{i+1/2,j,k} = \frac{1}{2} \left( \sigma_{i,j,k} + \sigma_{i+1,j,k} \right).$$

(133)

There is some evidence from Patankar and others to suggest that the use of a harmonic mean is more accurate for diffusion problems on coarse rectangular grids, however, since one of the aims of the DC plasma arc model is to use very high density grids, the computational cost associated with computing the harmonic mean along with the matrix-dependent transfers and operators needed for the multigrid algorithm is deemed to outweigh any accuracy improvement it is able to provide.

Define coefficient variables as follows:

$$a_E = \frac{\sigma_{i+1/2,j,k}}{\delta x^2}, \quad a_W = \frac{\sigma_{i-1/2,j,k}}{\delta x^2}, \quad a_N = \frac{\sigma_{i,j+1/2,k}}{\delta y^2},$$

$$a_S = \frac{\sigma_{i,j-1/2,k}}{\delta y^2}, \quad a_T = \frac{\sigma_{i,j,k+1/2}}{\delta z^2}, \quad a_B = \frac{\sigma_{i,j,k-1/2}}{\delta z^2}.$$  

(134)

With these definitions, we can write (132) as a difference equation:

$$a_E \phi_{i+1,j,k} + a_W \phi_{i-1,j,k} + a_N \phi_{i,j+1,k} + a_S \phi_{i,j-1,k} + a_T \phi_{i,j,k+1} + a_B \phi_{i,j,k-1}$$

$$- (a_E + a_W + a_N + a_S + a_T + a_B) \phi_{i,j,k} = 0$$

(135)

Boundary conditions for (135) are specified using (50) - (52). Use of the Dirichlet boundary condition on the anode is trivially implemented by setting:

$$\phi_{i,j,0} = 0$$

(136)

The numerical approximation for current flow boundary conditions (of which the insulating boundary is a special case) is accomplished by considering (130) at the boundary. As for the earlier
solvers, we use the boundary at \((i,0,k)\) as per Figure 11 as an example case:

\[
\frac{j_{x,i+1/2,0,k} - j_{x,i-1/2,0,k}}{\delta x} + \frac{j_{y,i,1/2,k} - j_{y,i,-1/2,k}}{\delta y} + \frac{j_{z,i,0,k+1/2} - j_{z,i,0,k-1/2}}{\delta z} = 0
\]  

(137)

We wish to specify the normal component of the current density vector on the boundary, that is, \(j_{y,i,0,k}\) - we do this approximately by taking a linear average between the \((i,1/2,k)\) and \((i,-1/2,k)\) points:

\[
j_{I} = j_{y,i,0,k} = \frac{1}{2} \left( j_{y,i,1/2,k} + j_{y,i,-1/2,k} \right)
\]  

(138)

Substituting (138) into (137) and then using the definition (131) produces the boundary condition relationship used for (50) and (51):

\[
a_E \phi_{i+1,0,k} + a_w \phi_{i-1,0,k} + 2a_{N} \phi_{i,1,k} + a_T \phi_{i,0,k+1} + a_B \phi_{i,0,k-1} - (a_E + a_w + 2a_N + a_T + a_B) \phi_{j,i,k} = \frac{2j_I}{\delta y}
\]  

(139)

This definition extends logically to the other boundaries in the problem. For insulating boundaries, \(j_I = 0\) is used.

Boundary conditions (136) and (139) are supplemented to (135) in order to produce a complete numerical description for the electric potential field. Unfortunately this is a non-constant coefficient problem, and cannot be solved directly using the spectral transform methods applied to standard Poisson and Laplace equations - as the electric potential is discretised on regular grids, an iterative geometric multigrid solver was used to solve the system efficiently. This will be detailed in a later section.

The numerical treatment of the electric potential for two-dimensional problems proceeds similarly, omitting the \(k\) index and any terms involving the \(z\)-component of fields or derivatives.

**Calculation of current density**

Once the electric potential is known, the current density vector is calculated directly via (131) in three dimensions. However, these relationships only give the current density vector components at
points offset from the main finite difference grid - linear interpolation is used to calculate their values on the main grid.

As this gives two-sided averages, the calculation involves values of the electric potential and electrical conductivity on either side of the point \((i,j,k)\). On the boundaries a different approach must be taken, as the grid used for \(\phi\) and \(\sigma\) does not extend beyond the physical boundary of the problem. For constant current density boundary conditions, this is overcome trivially by setting the normal component of the current density vector equal to the specified boundary value, \(j_r\). For constant voltage boundary conditions, a one-sided average using only the current density a half-step inside the boundary is used for the calculation of the normal component.

In 2D, the procedure is very similar, the only change being the dropping of the \(k\) index and the calculation for \(j_z\).

**Magnetic field**

In order to solve for the magnetic field, the magnetic vector potential must first be evaluated - for a known current density field, it is governed by (58).

In discussing the numerical treatment of \(A\) and \(B\), it is most convenient to begin with the three-dimensional case and then reduce the dimensionality of the problem to acquire the 2D description. This is due to the fact that the 3D case develops logically from the governing equations, whereas the 2D case for this particular part of the model requires some special treatment.

In 3D, second order centered difference operators are used to discretise (58) on a non-staggered finite difference grid as follows:

\[
\Delta_h A_{x,i,j,k} = -\mu_0 j_{x,i,j,k}
\]  

(140)

The expressions for the \(A_y\) and \(A_z\) components follow similarly.

The boundary condition description for each component, given by (60) and (61), is
\[
\left[ \frac{\partial A_x}{\partial x} \right]_{j,k} = 0, \quad A_{x,i,0,k} = A_{x,i,J,k} = 0, \quad A_{x,i,j,0} = A_{x,i,j,K} = 0, \quad (141)
\]

\[
A_{y,0,j,k} = A_{y,J,j,k} = 0, \quad \left[ \frac{\partial A_y}{\partial y} \right]_{j,k} = 0, \quad A_{y,i,0,j} = A_{y,i,J,j} = 0, \quad (142)
\]

\[
A_{z,0,j,k} = A_{z,J,j,k} = 0, \quad A_{z,i,0,j} = A_{z,i,J,j} = 0, \quad \left[ \frac{\partial A_z}{\partial z} \right]_{j,k} = 0. \quad (143)
\]

These together with (140) are standard discrete Poisson equations with mixed homogeneous boundary conditions, and may be solved using the rapid solvers described in a later section.

Once the \( \mathbf{A} \) field is known, the magnetic field may be calculated from it using the discrete version of (55). At the interior points, using second order centered difference operators, this is given by:

\[
B_{x,i,j,k} = D_y A_{z,i,j,k} - D_z A_{y,i,j,k} \quad (144)
\]

\[
B_{y,i,j,k} = D_z A_{x,i,j,k} - D_x A_{z,i,j,k} \quad (145)
\]

\[
B_{z,i,j,k} = D_x A_{y,i,j,k} - D_y A_{x,i,j,k} \quad (146)
\]

On boundaries, the terms for derivatives normal to the boundary involve grid points outside the calculation domain. This problem is overcome using quadratic extrapolation, for example:

\[
D_x A_{z,0,j,k} = -\frac{3 A_{z,0,j,k} + 4 A_{z,1,j,k} - A_{z,2,j,k}}{2\delta x} \quad (147)
\]

Similar expressions apply at the remaining boundaries.

**Magnetic field - reduction to 2D**

In two dimensions, the obvious procedure to reduce the dimensionality of (58) would be to assume the 2D current density and magnetic fields to extend infinitely in the \( z \)-direction and simply drop all \( z \)-related terms. However, as was shown earlier in section 2.4, this produces an irrotational field for
the velocity equations' source term, which is physically unrealistic and numerically troublesome. An alternative means of reducing (58) to two dimensions is thus sought.

If we consider a pseudo-2D scenario in which the size of the domain in the $z$-direction is small and the current is confined to the $x$-$y$ plane, we have $A_z = 0$, and (140) describes the two remaining components of the $A$ field.

Now consider approximating $A$ with only three grid points in the $z$ direction, $k = \{0,1,2\}$. The boundary condition (141) together with (140) then gives:

$$
\frac{A_{x,i+1,j,1} + A_{x,i-1,j,1} - 2A_{x,i,j,1}}{\delta x^2} + \frac{A_{x,i,j+1,1} + A_{x,i,j-1,1} - 2A_{x,i,j,1}}{\delta y^2} - \mu_0 j_{x,i,j,1} \frac{2A_{x,i,j,1}}{\delta z^2} = 0
$$

The expression for the $A_z$ component is found in a similar manner. If we drop the $k = 1$ subscripts, we may obtain the reduced finite difference relationship for the magnetic vector potential in two dimensions:

$$
\Delta_h A_{x,i,j} - \frac{2}{\delta z^2} A_{x,i,j} = -\mu_0 j_{x,i,j}
$$

The expression for $A_y$ follows similarly. These are supplemented with boundary conditions as per (141) and (142):

$$
\left[ \frac{\partial A_x}{\partial x} \right]_{j,0} = \left[ \frac{\partial A_x}{\partial x} \right]_{j,J} = 0, \quad A_{x,i,0} = A_{x,i,J} = 0
$$

$$
A_{y,0,j} = A_{y,J,j} = 0, \quad \left[ \frac{\partial A_x}{\partial y} \right]_{j,0} = \left[ \frac{\partial A_y}{\partial y} \right]_{j,J} = 0
$$

Assuming a constant value is used for $\delta z$, this formulation retains symmetric constant coefficients in the finite difference equation (149) and can therefore be solved using the rapid spectral solvers developed for Poisson equations. These will be discussed in a later section.
Once $A_x$ and $A_y$ are known, the calculation of the magnetic field (a scalar in 2D) is performed by:

$$B_z = D_x A_{y,i,j} - D_y A_{x,i,j}$$  \hspace{1cm} (152)

Approximations to the normal derivatives on the boundaries are handled by extrapolation, via identical procedure to that used for (147).

The expressions (149) - (151) can be seen as a somewhat empirical generalisation of the naive translation of (58) to 2D finite differences, which would correspond to the case of allowing $\delta z$ to tend to infinity. This formulation does introduce a new problem, namely the choice of an appropriate (finite) $\delta z$ for a given 2D problem. Loosely speaking, this amounts to a decision on how far in the $z$-direction we would expect the electrically-conducting body of the arc to extend. In the absence of better information, we may consider an arc column as described by the empirical model of Bowman, with shape determined by (1) and (2). Based on this, a reasonable choice would be given by $r_x < \delta z < 3.2r_x$. The effect of this number on the results of the magnetic field calculation will be examined in a later section.

3.5 Fast Poisson solver

Consider a general finite-difference equation governing the behaviour of a variable in two or more dimensions. If the equation can be assumed to have symmetric constant coefficients, it may be written as follows for 2D:

$$b[u_{i+1,j} + u_{i-1,j}] + c[u_{i,j+1} + u_{i,j-1}] + au_{i,j} = S_{i,j}$$  \hspace{1cm} (153)

and for 3D:

$$b[u_{i+1,j,k} + u_{i-1,j,k}] + c[u_{i,j+1,k} + u_{i,j-1,k}] + d[u_{i,j,k+1} + u_{i,j,k-1}] + au_{i,j,k} = S_{i,j,k}$$  \hspace{1cm} (154)

Equations of the form of (153) and (154) arise a number of times during the finite difference discretisation of the DC plasma arc model. In particular, the solution for the pressure $p$ (80), the stream function $\psi$ (91), the fluid flow gauge variable $\Theta$ (102), and the components of the magnetic vector potential $A$ (140) and (149) are all governed by discrete Poisson or Poisson-like equations.
which reduce to the forms above.

We consider only the three-dimensional case - the 2D formulations are derived in an identical manner, omitting the terms depending on the $k$ grid index. Consider homogeneous Dirichlet boundary conditions applied to (154):

$$u_{0,j,k} = u_{i,j,k} = 0$$  \hspace{1cm} (155)

$$u_{i,0,k} = u_{i,j,k} = 0$$  \hspace{1cm} (156)

$$u_{i,j,0} = u_{i,j,k} = 0$$  \hspace{1cm} (157)

The use of the discrete sine transform and its inverse guarantee these boundary conditions:

$$\hat{u}_{l,m,n} = \sum_{i=0}^{I} \sum_{j=0}^{J} \sum_{k=0}^{K} u_{i,j,k} \sin \left( \frac{\pi i l}{I} \right) \sin \left( \frac{\pi j m}{J} \right) \sin \left( \frac{\pi k n}{K} \right)$$  \hspace{1cm} (158)

$$u_{i,j,k} = \frac{1}{IJK} \sum_{l=0}^{I} \sum_{m=0}^{J} \sum_{n=0}^{K} \hat{u}_{l,m,n} \sin \left( \frac{\pi i l}{I} \right) \sin \left( \frac{\pi j m}{J} \right) \sin \left( \frac{\pi k n}{K} \right)$$  \hspace{1cm} (159)

It is useful to note that the sine transform (and its inverse) of a 3D grid $(i,j,k)$ of variables can be broken down into successive one-dimensional transforms, which can be performed in any order. For example:

$$u_{i,j,n}^{*1} = \sum_{k=0}^{K} u_{i,j,k} \sin \left( \frac{\pi k n}{K} \right)$$

$$u_{i,m,n}^{*2} = \sum_{j=0}^{J} u_{i,j,n}^{*1} \sin \left( \frac{\pi j m}{J} \right)$$  \hspace{1cm} (160)

$$\hat{u}_{l,m,n} = \sum_{i=0}^{I} u_{i,m,n}^{*2} \sin \left( \frac{\pi i l}{I} \right)$$

We take the definition of $u_{i,j,k}$ from the inverse sine transform, (159), and substitute it into (154). Using the identity
\[
\sin \left( \frac{\pi (i+1) l}{I} \right) + \sin \left( \frac{\pi (i-1) l}{I} \right) = 2 \sin \left( \frac{\pi i l}{I} \right) \cos \left( \frac{\pi l}{I} \right)
\]  
(161)

to simplify then gives:

\[
\sum_{l=0}^{I-1} \sum_{m=0}^{J-1} \sum_{n=0}^{K-1} \left[ a \hat{u}_{i,m,n} + 2 b \hat{u}_{i,m,n} \cos \left( \frac{\pi l}{I} \right) \right] + 2 c \hat{u}_{i,m,n} \cos \left( \frac{\pi m}{J} \right) + 2 d \hat{u}_{i,m,n} \cos \left( \frac{\pi n}{K} \right) = 0
\]

(162)

Equating coefficients, we have:

\[
\hat{u}_{i,m,n} = \frac{\hat{S}_{i,m,n}}{2 b \cos \left( \frac{\pi l}{I} \right) + 2 c \cos \left( \frac{\pi m}{J} \right) + 2 d \cos \left( \frac{\pi n}{K} \right) + a}
\]  
(163)

This gives the algorithm for solution of (154) with (155) – (157):

- Take the sine transform (158) of the Poisson source term \( S_{i,j,k} \)
- Compute the solution in transformed space by (163)
- Use the inverse transform (159) to acquire the solution for \( u_{i,j,k} \)

Rapid methods based on the Fast Fourier Transform (FFT) algorithm are available to perform forward and backward discrete sine transforms in \( O(N \log_2 N) \) time for a series of length \( N \).

Now we consider homogeneous Neumann boundary conditions supplied to (154). This implies:

\[
\left[ \frac{\partial u}{\partial x} \right]_{0,j,k} = 0
\]

(164)

\[
\left[ \frac{\partial u}{\partial y} \right]_{j,0,k} = 0
\]

(165)

\[
\left[ \frac{\partial u}{\partial z} \right]_{j,0,k} = 0
\]

(166)
Note that non-homogeneous Neumann boundary conditions can be converted into (164) - (166) in a discrete setting by considering (154) at the boundary. Using the point \((i,0,k)\) as an example as per Figure 11, we have:

\[
b |u_{i+1,0,k} + u_{i-1,0,k}| + c |u_{i,1,k} + u_{i,-1,1,k}| + d |u_{i,0,k+1} + u_{i,0,k-1}| + a u_{i,0,k} = S_{i,0,k}
\]  

(167)

Approximate (165) using standard second order centered differences at the same boundary:

\[
\frac{u_{i,1,k} - u_{i,-1,k}}{2 \delta y} = 0
\]

\[
u_{i,-1,k} = u_{i,1,k}
\]  

(168)

Substituting into (167) gives:

\[
b |u_{i+1,0,k} + u_{i-1,0,k}| + 2 c u_{i,1} + d |u_{i,0,k+1} + u_{i,0,k-1}| + a u_{i,0,k} = S_{i,0,k}
\]  

(169)

A non-homogeneous Neumann condition with a gradient \(U_{y,i,k}\) specified on the \((i,0,k)\) boundary would be given by:

\[
\left[ \frac{\partial u}{\partial y} \right]_{y,0,k} = U_{y,i,k}
\]  

(170)

Discretising and substituting as per (168) and (169) gives:

\[
b |u_{i+1,0,k} + u_{i-1,0,k}| + 2 c u_{i,1} + d |u_{i,0,k+1} + u_{i,0,k-1}| + a u_{i,0,k} = S_{i,0,k} + 2 c \delta y U_{y,i,k}
\]  

(171)

Comparing (171) and (169), it is clear that solving (154) with non-homogeneous boundary condition (170) is equivalent to solving it with homogeneous boundary condition (165), provided the source term \(S_{i,j}\) is modified on the boundary as per the right-hand side of (171).

Having established that the homogeneous problem is no less unique, we return to the solution of (154) with boundary conditions (164) - (166). This is performed via a procedure analogous to that used for Dirichlet boundary conditions, with the primary difference being the transform used - for
Neumann type boundary conditions, (164) - (166) are guaranteed if the cosine transform is used. In three dimensions, the transform and its inverse are given by:

\[ u_{l,m,n} = \sum_{i=0}^{I} \sum_{j=0}^{J} \sum_{k=0}^{K} u_{i,j,k} \cos \left( \frac{\pi i l}{I} \right) \cos \left( \frac{\pi j m}{J} \right) \cos \left( \frac{\pi k n}{K} \right) \]  

(172)

\[ u_{i,j,k} = \frac{1}{IJK} \sum_{i=0}^{I} \sum_{j=0}^{J} \sum_{k=0}^{K} u_{l,m,n} \cos \left( \frac{\pi i l}{I} \right) \cos \left( \frac{\pi j m}{J} \right) \cos \left( \frac{\pi k n}{K} \right) \]  

(173)

Again, the cosine transform in 3D may be seen as successive applications of the one-dimensional transform along each coordinate direction. As the transform is linear, the order in which the coordinates are calculated is unimportant:

\[ u_{i,j,n}^{r1} = \sum_{k=0}^{K} u_{i,j,k} \cos \left( \frac{\pi k n}{K} \right) \]  

\[ u_{i,m,n}^{r2} = \sum_{j=0}^{J} u_{i,j,n}^{r1} \cos \left( \frac{\pi j m}{J} \right) \]  

\[ \tilde{u}_{l,m,n} = \sum_{i=0}^{I} u_{i,m,n}^{r2} \cos \left( \frac{\pi i l}{I} \right) \]  

(174)

Now we consider substituting the inverse cosine transform definition of \( u_{i,j,k} \) into (154) and rearranging terms within the summation signs. Use of the identity

\[ \cos \left( \frac{\pi (i+1) l}{I} \right) + \cos \left( \frac{\pi (i-1) l}{I} \right) = 2 \cos \left( \frac{\pi i l}{I} \right) \cos \left( \frac{\pi l}{I} \right) \]  

(175)

then results in:

\[ \sum_{i=0}^{I} \sum_{j=0}^{J} \sum_{k=0}^{K} \left[ a \tilde{u}_{i,m,n} + 2 b \bar{u}_{i,m,n} \cos \left( \frac{\pi l}{I} \right) + 2 c \tilde{u}_{i,m,n} \cos \left( \frac{\pi m}{J} \right) + 2 d \tilde{u}_{i,m,n} \cos \left( \frac{\pi n}{K} \right) \right] = 0 \]  

(176)
Comparing coefficients gives:

\[
\tilde{u}_{l,m,n} = \frac{\tilde{S}_{l,m,n}}{2b \cos \left( \frac{\pi l}{I} \right) + 2c \cos \left( \frac{\pi m}{J} \right) + 2d \cos \left( \frac{\pi n}{K} \right) + a}
\]  

(177)

The algorithm for solution of (154) with (164) - (166) is then:

- Take the cosine transform (172) of the Poisson source term \( S_{i,j,k} \)
- Compute the solution in transformed space by (177)
- Use the inverse transform (173) to acquire the solution for \( u_{i,j,k} \)

As for the sine transform, rapid methods based on the FFT are available to perform forward and backward discrete cosine transforms.

For solving a problem with mixed boundary conditions, a solution may still be obtained using these methods under certain circumstances. The boundary conditions along a coordinate direction must be of the same kind, for example, Dirichlet conditions for both of the \( i/x \) boundaries, Neumann conditions for both \( j/y \) boundaries, and Neumann conditions for both \( k/z \) boundaries. In such cases, it is possible to define a mixed transform - we use sine transforms in the direction of coordinates with Dirichlet boundary conditions, and cosine transforms in the direction of those with Neumann boundary conditions. Using the example given:

\[
\tilde{u}_{l,m,n} = \sum_{i=0}^{I} \sum_{j=0}^{J} \sum_{k=0}^{K} u_{i,j,k} \sin \left( \frac{\pi il}{I} \right) \cos \left( \frac{\pi jm}{J} \right) \cos \left( \frac{\pi kn}{K} \right)
\]  

(178)

\[
u_{i,j,k} = \frac{1}{IJK} \sum_{i=0}^{I} \sum_{j=0}^{J} \sum_{k=0}^{K} \tilde{u}_{l,m,n} \sin \left( \frac{\pi il}{I} \right) \cos \left( \frac{\pi jm}{J} \right) \cos \left( \frac{\pi kn}{K} \right)
\]  

(179)

This transform is comprised of successive 1D transforms in each coordinate direction - sine transforms in the direction of coordinates given Dirichlet boundary conditions, cosine transforms in the direction of those given Neumann conditions.
Using the example mentioned:

\[ u_{i,j,n}^* = \sum_{k=0}^{K} u_{i,j,k} \cos \left( \frac{\pi k n}{K} \right) \]

\[ u_{i,m,n}^* = \sum_{j=0}^{J} u_{i,j,n}^1 \cos \left( \frac{\pi j m}{J} \right) \]

\[ \tilde{u}_{i,m,n} = \sum_{i=0}^{I} u_{i,m,n}^2 \sin \left( \frac{\pi i l}{I} \right) \]

A number of such transforms can be defined, depending on which combination of Dirichlet and Neumann boundary conditions are required.

Applying the inverse mixed transform to (154), followed by term rearrangement and the use of (161) and (175) gives

\[
\sum_{l=0}^{I} \sum_{m=0}^{J} \sum_{n=0}^{K} \left[ a \tilde{u}_{i,m,n} + 2b \tilde{u}_{i,m,n} \cos \left( \frac{\pi i l}{I} \right) + 2c \tilde{u}_{i,m,n} \cos \left( \frac{\pi j m}{J} \right) + 2d \tilde{u}_{i,m,n} \cos \left( \frac{\pi k n}{K} \right) \right] = 0
\]

(181)

following which, comparison of coefficients results in the usual expression:

\[
\tilde{u}_{i,m,n} = \frac{\tilde{S}_{i,m,n}}{2b \cos \left( \frac{\pi i l}{I} \right) + 2c \cos \left( \frac{\pi j m}{J} \right) + 2d \cos \left( \frac{\pi k n}{K} \right) + a}
\]

(182)

The algorithm for solution of (154) with mixed boundary conditions is therefore:

- Take the mixed transform of the Poisson source term \( S_{i,j,k} \) by successive applications of 1D sine and cosine transforms as appropriate
- Calculate the solution in transformed space by (182)
- Compute the inverse mixed transform to acquire the solution for \( u_{i,j,k} \)
Implementation of FFT calculations

For the computer code used to implement the DC plasma arc model, routines from the third-party FFTW library\textsuperscript{50} were used to perform the one-dimensional discrete sine and cosine transforms. 2D and 3D transforms were computed by sweeping through each coordinate direction, as per (160), (174), and (180).

3.6 Geometric multigrid solver

While most of the computational load in the DC plasma arc model algorithm can be handled by fast spectral solvers, the solution for electric potential in the electromagnetic field sub-model cannot be treated this way. This is due to the fact that the finite difference discretisation of the governing equation for electric potential produces a difference equation with non-constant coefficients, (135).

We consider only grids with $\delta x = \delta y = \delta z = \delta l$. It is possible to write a general version of (135) as follows. For 2D:

$$a_E \phi_{i+1,j} + a_W \phi_{i-1,j} + a_N \phi_{i,j+1} + a_S \phi_{i,j-1} - a_p \phi_{i,j} = S_{i,j}$$

$$a_p = a_E + a_W + a_N + a_S$$  \hspace{1cm} (183)

In 3D:

$$a_E \phi_{i+1,j,k} + a_W \phi_{i-1,j,k} + a_N \phi_{i,j+1,k} + a_S \phi_{i,j,k-1} + a_T \phi_{i,j,k+1} - a_p \phi_{i,j,k} = S_{i,j,k}$$

$$a_p = a_E + a_W + a_N + a_S + a_T + a_B$$  \hspace{1cm} (184)

The $a_i$ are all assumed to be functions of the grid point location $(i,j)$ or $(i,j,k)$ via the electrical conductivity $\sigma$, as well as the grid spacing $\delta l$, as per the definitions in (134).

For such problems, which are linear and elliptic but otherwise intractable for solution via direct methods, Brandt\textsuperscript{44} introduced the multigrid family of algorithms, later reinterpreted and summarised by Press et al\textsuperscript{47}.
Two-dimensional iterative geometric multigrid algorithm

We begin with some definitions for the 2D case.

A key element of multigrid algorithms is the numerical approximation of equations like (183) on a series of successively finer grids until the desired resolution is reached. This is achieved by inserting new grid points half way between coarse grid locations to produce a finer grid, and then repeating the process. This is shown schematically in Figure 12, assuming the finest grid is Grid 1, and there are $M$ grids in total.

*Figure 12: Multiple grids used in multigrid algorithm*

![Multiple grids used in multigrid algorithm](image)

The value of $\delta l$ is different for each grid level, as the grid spacing gets coarser and coarser from the finest grid which is equivalent to the grid resolution used for the rest of the numerical model.

$$
\begin{align*}
\delta l_m &= 2 \delta l_{m-1} \\
\delta l_m &= 2^m \delta l_i
\end{align*}
$$

(185)

Similarly, the size of the grid in $i$ and $j$ also changes:

$$
\begin{align*}
I_m &= \left[ \frac{I_{m-1} - 1}{2} \right] + 1 \\
J_m &= \left[ \frac{J_{m-1} - 1}{2} \right] + 1
\end{align*}
$$

(186)

We are now in a position to describe the iterative multigrid algorithm. Assuming an estimate of a
variable $\phi$, denoted $\phi^*$, on the finest (main) grid, we begin by performing an approximate solution or smoothing operation by calculating a single Gauss-Seidel iteration of (183). This is written as:

$$\phi_{1,i,j}^* = \frac{1}{a_{p,1}} \left( a_{E,1} \phi_{1,i+1,j}^* + a_{W,1} \phi_{1,i-1,j}^* + a_{N,1} \phi_{1,i,j+1}^* + a_{S,1} \phi_{1,i,j-1}^* - S_{1,i,j} \right)$$  \hspace{1cm} (187)

Here, the additional subscript refers to the grid level the values are defined at. The residual $R$ is then calculated as follows:

$$R_{1,i,j} = a_{E,1} \phi_{1,i+1,j}^* + a_{W,1} \phi_{1,i-1,j}^* + a_{N,1} \phi_{1,i,j+1}^* + a_{S,1} \phi_{1,i,j-1}^* - a_{p,1} \phi_{1,i,j}^* - S_{1,i,j}$$  \hspace{1cm} (188)

This result is then interpolated to get an estimate of the residual on the next coarsest grid $(i,j)$ by using a full-weighting restriction operator defined by:

$$R_{m+1,i,j}^* = \frac{1}{2} R_{m,2i,j} + \frac{1}{2} R_{m,2i+1,j} + \frac{1}{2} R_{m,2i,j+1} + \frac{1}{2} R_{m,2i+1,j+1} + \frac{1}{16} R_{m,2i+2,j+1} + \frac{1}{16} R_{m,2i,j+2} + R_{m,2i+1,j+1} + R_{m,2i+2,j+1} + R_{m,2i+2,j+2}$$  \hspace{1cm} (189)

Using $m = 1$ gives the estimated residual $R^*$ on the coarser grid 2. Discretising (183) on the second grid, we have:

$$a_{E,2} \phi_{2,i+1,j} + a_{W,2} \phi_{2,i-1,j} + a_{N,2} \phi_{2,i,j+1} + a_{S,2} \phi_{2,i,j-1} - a_{p,2} \phi_{2,i,j} = S_{2,i,j}$$  \hspace{1cm} (190)

Due to the linearity of (183), if we then define the correction $w = \phi - \phi^*$ and use (188) defined on grid 2, we have the following relationship on the coarser grid:

$$a_{E,2} w_{2,i+1,j} + a_{W,2} w_{2,i-1,j} + a_{N,2} w_{2,i,j+1} + a_{S,2} w_{2,i,j-1} - a_{p,2} w_{2,i,j} = R_{2,i,j} = R_{2,i,j}^*$$  \hspace{1cm} (191)

This is seen to be similar in form to (183), with the source term replaced by the coarse-grid residual calculated from (189). A similar routine to (187) - (189) can then be executed to improve the estimate of $w$ and move to an even coarser grid. Beginning with a Gauss-Seidel smoothing step:

$$w_{2,i,j} = \frac{1}{a_{p,2}} \left( a_{E,2} w_{2,i+1,j} + a_{W,2} w_{2,i-1,j} + a_{N,2} w_{2,i,j+1} + a_{S,2} w_{2,i,j-1} - R_{2,i,j}^* \right)$$  \hspace{1cm} (192)
We then calculate the residual of (191):

$$R_{2,i,j} = a_{E,2}w_{2,i+1,j} + a_{W,2}w_{2,i-1,j} + a_{N,2}w_{2,i,j+1} + a_{S,2}w_{2,i,j-1} - a_{P,2}w_{2,i,j} - R^*_2,i,j$$  \hspace{1cm} (193)$$

and estimate it on the next coarsest grid, grid 3, by using (189) with $m = 2$.

The coarsening-grid calculation procedure (192), (193), and (189) is repeated for grids 3, 4, ... $M$. When grid $M$ is reached, (191) is written as:

$$a_{E,M}w_{M,i+1,j} + a_{W,M}w_{M,i-1,j} + a_{N,M}w_{M,i,j+1} + a_{S,M}w_{M,i,j-1} - a_{P,M}w_{M,i,j} = R^*_M,i,j$$  \hspace{1cm} (194)$$

This must be solved for $w_M$. Generally the grid size $I_M$ and $J_M$ will be small, so an iterative solver is used to find the solution of (194) very rapidly.

For the next phase of the multigrid algorithm, we require an operator to estimate $w$ values on a finer grid based on the values from a coarse grid. We denote these estimates by $w^*$. The bilinear interpolation operator from grid $m$ to grid $m - 1$ is used - it is defined as follows:

$$w^*_{m-1,2i,2j} = w_{m,i,j}$$
$$w^*_{m-1,2i+1,2j} = \frac{1}{4} [w_{m,i,j} + w_{m,i+1,j}]$$
$$w^*_{m-1,2i,2j+1} = \frac{1}{4} [w_{m,i,j} + w_{m,i,j+1}]$$
$$w^*_{m-1,2i+1,2j+1} = \frac{1}{4} [w_{m,i,j} + w_{m,i+1,j} + w_{m,i,j+1} + w_{m,i+1,j+1}]$$  \hspace{1cm} (195)$$

We start with $m = M$ in (195). The estimate produced is added to the most recent value of $w_M$ from the coarsening-grid calculation, to generate an updated value of $w$ on the fine grid:

$$w_{M-1,i,j} = w_{M-1,i,j} + w^*_{M-1,i,j}$$  \hspace{1cm} (196)$$

The result is smoothed using a single Gauss-Seidel iteration:

$$w_{M-1,i,j} = \frac{1}{a_{P,M-1}} \left[ a_{E,M-1}w_{M-1,i+1,j} + a_{W,M-1}w_{M-1,i-1,j} + a_{N,M-1}w_{M-1,i,j+1} + a_{S,M-1}w_{M-1,i,j-1} - R^*_{M-1,i,j} \right]$$  \hspace{1cm} (197)$$
The process (195) - (197) is then repeated for all successively finer grids $M - 2, M - 3, \ldots 2$. The final step is slightly different, in that in place of (196), the correction on the finest grid is applied directly to the variable $\phi^*$:

$$
\phi_{1,i,j}^* = \phi_{1,i,j}^* + w_{1,i,j}^*
$$

(198)

A final Gauss-Seidel smoothing step is then performed by (187).

The complete procedure (187) through (198) is described as a single “V-cycle”, calculating down to the coarsest grid and then back up to the finest (main) grid. Several V-cycles or variants thereof (F- and W-cycles are popular and execute more calculations at the coarse grid levels) may be executed successively, with the advantage of the multigrid algorithm being that the accuracy of the solution at each iteration is generally greatly improved. For problems with constant $\sigma$ (Poisson equations), solution to beyond the discretised equation's truncation error is generally possible in only one or two iterations of the cycle.

It is important to note that for highly variable and anisotropic $\sigma$ the convergence of the method does deteriorate, and it can stall after a small number of cycle iterations - typically $< 10$. However, reasonable accuracy with relative errors of the order of $10^4$ or less may still be obtained efficiently, especially if a good initial estimate of $\phi$ is combined with a more complex cycle such as F or W.

**Boundary conditions and coarse-grid coefficients**

For the particular case of the electric field solver, boundary conditions are supplied to the $\phi_{i,j}$ field in the form of either constant voltage conditions (136), or constant current density conditions (139).

The implementation of the former in the multigrid solver is simple, and amounts to setting $\phi$, $w$, and $R$ to zero on the boundary concerned at all grid levels.

For the latter, any non-homogeneous boundary conditions are first converted to homogeneous ones by observing that conditions of the sort (139) are equivalent to (183) at the boundary with homogeneous Neumann boundary conditions, provided the source term is modified. Considering the $(i,0)$ boundary as an example (see Figure 11), this would involve setting the source term equal to
\[ S_{i,0} + 2 j_f \phi l \]. Once this is done, the boundary is handled in the multigrid algorithm by applying homogeneous Neumann conditions to \( \phi \) and \( w \) on the boundary concerned at all grid levels.

The restriction operator also requires special treatment for Neumann boundary conditions. Again using the \((i,0)\) boundary as an example:

\[
\begin{align*}
R_{m,2i+1,0}^r & = \frac{1}{2} R_{m,2i,0} + \frac{1}{8} \left[ R_{m,2i+1,0} + R_{m,2i-1,0} + R_{m,2i+1,1} + R_{m,2i-1,1} \right] \\
& + \frac{1}{16} \left[ R_{m,2i+1,1} + R_{m,2i-1,1} + R_{m,2i+1,-1} + R_{m,2i-1,-1} \right]
\end{align*}
\]  

(199)

This is seen to involve values at ghost points outside the physical range of the grid. To overcome this, we treat the gradient of \( R \) normal to the boundary as zero:

\[
\frac{R_{m,i,1} - R_{m,i,-1}}{\delta l} = 0
\]  

(200)

\[
R_{m,i,-1} = R_{m,i,1}
\]

\[
R_{m+1,i,0}^r = \frac{1}{2} R_{m,2i,0} + \frac{1}{8} \left[ R_{m,2i+1,0} + R_{m,2i-1,0} + 2 R_{m,2i,1} \right] + \frac{1}{16} \left[ 2 R_{m,2i+1,1} + 2 R_{m,2i-1,1} \right]
\]  

(201)

The various \( a_i \) coefficients in (183) and (191) etc are defined by (134), for which values for the electrical conductivity are required at every grid level. These are calculated by successive applications of the restriction operator, for \( m = 1 \) to \( M - 1 \):

\[
\begin{align*}
\sigma_{m+1,i,j} & = \frac{1}{2} \sigma_{m,2i,j} + \frac{1}{8} \left[ \sigma_{m,2i+1,j} + \sigma_{m,2i-1,j} + \sigma_{m,2i,j+1} + \sigma_{m,2i,j-1} + \sigma_{m,2i,j+1} + \sigma_{m,2i,j-1} \right] \\
& + \frac{1}{16} \left[ \sigma_{m,2i+1,j+1} + \sigma_{m,2i-1,j+1} + \sigma_{m,2i+1,j-1} + \sigma_{m,2i-1,j-1} \right]
\end{align*}
\]  

(202)

Extension of the multigrid algorithm to three dimensions

The 2D multigrid algorithm extends naturally to three dimensions, for the solution of general variable-coefficient equations such as (184). The multiple grids as described in Figure 12 are produced in the same fashion, however they now have an additional dimension in the \( z \)-direction, indexed by \( k \).

The grids' size and shape are governed by (185) and (186), together with a similar relationship for
the K variable:

\[ K_m = \frac{K_{m-1} - 1}{2^m} + 1 \quad (203) \]

Starting as before with an estimate \( \phi^* \) of \( \phi \), we replace (187) and (188) with:

\[ \phi^*_{i,j,k} = \frac{1}{a_{p_1}} \left( a_{E,i} \phi^*_{i+1,j,k} + a_{W,i} \phi^*_{i-1,j,k} + a_{N,i} \phi^*_{i,j+1,k} + a_{S,i} \phi^*_{i,j-1,k} + \right. \]

\[ \left. a_{T,i} \phi^*_{i,j,k+1} + a_{B,i} \phi^*_{i,j,k-1} - S_{i,j,k} \right) \quad (204) \]

\[ R^*_{i,j,k} = a_{E,i} \phi^*_{i+1,j,k} + a_{W,i} \phi^*_{i-1,j,k} + a_{N,i} \phi^*_{i,j+1,k} + a_{S,i} \phi^*_{i,j-1,k} + \]

\[ a_{T,i} \phi^*_{i,j,k+1} + a_{B,i} \phi^*_{i,j,k-1} - a_{p_1} \phi_{i,j,k} - S_{i,j,k} \quad (205) \]

The full weighting restriction operator in three dimensions, with \( m = 1 \), is then used to calculate \( R^* \) on the next coarser grid:

\[ R^*_{m+1,i,j,k} = \frac{1}{2^m} \left( R_{m,i,j-1,k} + R_{m,i,j,k} + R_{m,i,j+1,k} + R_{m,i,j,k+1} + R_{m,i,j,k-1} + \right. \]

\[ \left. R_{m,i,j,k+1} + R_{m,i,j,k-1} + R_{m,i+1,j,k} + R_{m,i-1,j,k} + R_{m,i+1,j,k+1} + R_{m,i-1,j,k+1} + R_{m,i+1,j,k-1} + R_{m,i-1,j,k-1} \right) \quad (206) \]

Equivalent to (191), we then have the relationship for the error \( w \) on grid 2 in 3D:

\[ a_{E,i} w_{i+1,j-1,k} + a_{W,i} w_{i-1,j,k} + a_{N,i} w_{i,j+1,k} + a_{S,i} w_{i,j-1,k} + \]

\[ a_{T,i} w_{i,j,k+1} + a_{B,i} w_{i,j,k-1} - a_{p_2} w_{i,j,k} = R^*_{2,i,j,k} \quad (207) \]

Gauss-Seidel smoothing is applied as per (192):

\[ w_{i,j,k} = \frac{1}{a_{p_2}} \left( a_{E,i} w_{i+1,j-1,k} + a_{W,i} w_{i-1,j,k} + a_{N,i} w_{i,j+1,k} + a_{S,i} w_{i,j-1,k} + \right. \]

\[ \left. a_{T,i} w_{i,j,k+1} + a_{B,i} w_{i,j,k-1} - R^*_{2,i,j,k} \right) \quad (208) \]
The residual on grid 2 is then given by the equivalent of (193):

\[
R_{2,i,j,k} = a_{E,2} w_{2,i+1,j,k} + a_{W,2} w_{2,i-1,j,k} + a_{N,2} w_{2,i,j+1,k} + a_{S,2} w_{2,i,j-1,k} + a_{T,2} w_{2,i,j,k+1} + a_{B,2} w_{2,i,j,k-1} - a_{P,2} w_{2,i,j,k} - R^*_2 \tag{209}
\]

The estimated residual for grid 3 is then obtained from (206), using \( m = 2 \).

The coarsening-grid calculation procedure (207), (208), and (206) is then repeated for grids 3, 4, ..., \( M \). On the coarsest grid \( M \), (194) is replaced with:

\[
a_{E,M} w_{M,i+1,j,k} + a_{W,M} w_{M,i-1,j,k} + a_{N,M} w_{M,i,j+1,k} + a_{S,M} w_{M,i,j-1,k} + a_{T,M} w_{M,i,j,k+1} + a_{B,M} w_{M,i,j,k-1} - a_{P,M} w_{M,i,j,k} \approx R^*_M \tag{210}
\]

and solved for \( w_M \) using an iterative solver.

Trilinear interpolation of the correction values is required for the second part of the V-cycle in 3D. This is defined, interpolating from grid \( m \) to grid \( m - 1 \), as:

\[
\begin{align*}
  w^*_{m-1,2,i,2}, 2k &= w_{m,i,j} \\
  w^*_{m-1,2,i+1,2}, 2k &= \frac{1}{2} \left( w_{m,i,j,k} + w_{m,i,j,k+1} \right) \\
  w^*_{m-1,2,i,2}, 2k+1 &= \frac{1}{2} \left( w_{m,i,j,k} + w_{m,i,j,k+1} \right) \\
  w^*_{m-1,2,i,2}, 2k+1 &= \frac{1}{2} \left( w_{m,i,j,k} + w_{m,i,j,k+1} \right) \\
  w_{m-1,2,i+1,2}, 2k+1 &= \frac{1}{2} \left( w_{m,i,j,k} + w_{m,i,j,k+1} \right) + w^*_{m-1,2,i,2}, 2k+1 \\
  w^*_{m-1,2,i+1,2}, 2k+1 &= \frac{1}{2} \left( w_{m,i,j,k} + w_{m,i,j,k+1} \right) + w^*_{m-1,2,i,2}, 2k+1 \\
  w^*_{m-1,2,i+1,2}, 2k+1 &= \frac{1}{2} \left( w_{m,i,j,k} + w_{m,i,j,k+1} \right) + w^*_{m-1,2,i,2}, 2k+1 \\
  w^*_{m-1,2,i+1,2}, 2k+1 &= \frac{1}{2} \left( w_{m,i,j,k} + w_{m,i,j,k+1} \right) + w^*_{m-1,2,i,2}, 2k+1 \\
  w^*_{m-1,2,i+1,2}, 2k+1 &= \frac{1}{2} \left( w_{m,i,j,k} + w_{m,i,j,k+1} \right) + w^*_{m-1,2,i,2}, 2k+1 \\
  w^*_{m-1,2,i+1,2}, 2k+1 &= \frac{1}{2} \left( w_{m,i,j,k} + w_{m,i,j,k+1} \right) + w^*_{m-1,2,i,2}, 2k+1 \\
\end{align*} \tag{211}
\]

We start with \( m = M \) in (211). The result is added to the most recent value of \( w_{M,i,j,k} \) from the coarsening-grid calculation, as per (196):

\[
w_{M-1,i,j,k} = w_{M-1,i,j,k} + w^*_{M-1,i,j,k} \tag{212}
\]
The result is smoothed using a single Gauss-Seidel iteration:

\[
\begin{align*}
 w_{M-1,i,j,k} & = \frac{1}{a_{P,M-1}} \left[ a_{E,M-1} w_{M-1,i+1,j,k} + a_{W,M-1} w_{M-1,i-1,j,k} + a_{N,M-1} w_{M-1,i,j+1,k} + \\
 & a_{S,M-1} w_{M-1,i,j-1,k} + a_{T,M-1} w_{M-1,i,j,k+1} + a_{B,M-1} w_{M-1,i,j,k-1} - R^*_{M-1,i,j,k} \right]
\end{align*}
\]  (213)

(211) - (213) are then repeated for all successively finer grids \( M - 2, M - 3, \ldots 2 \). As in the 2D algorithm, the final step is slightly different in that the correction on the finest grid is applied to the variable \( \phi^* \) directly:

\[
\phi^*_{i,j,k} = \phi^*_{i,j,k} + w^*_{i,j,k}
\]  (214)

The final Gauss-Seidel smoothing step is then performed by (204).

The procedure (204) through (214) forms a single V-cycle step for a three-dimensional problem. Successive cycles (which can be more complex than V-cycles as in the 2D case) improve \( \phi^* \) until the desired accuracy is reached.

Boundaries in 3D problems are treated identically to the 2D case, using the three-dimensional analogues of (199) - (201). The electrical conductivity is estimated on all grid levels by restriction as per (202).

### 3.7 Model algorithm discussion and parallelisation

Although designed for maximum efficiency, using simple discretisation techniques and explicit procedures wherever possible, the DC plasma arc model is still extremely computationally intensive once it is scaled to very large grid sizes. A clear understanding of the overall algorithm and the load it places on the computers it is intended to run on is thus of value in considering optimisation techniques such as parallelisation.

**Algorithm overview**

The full DC plasma arc algorithm, using the gauge method based fluid flow solver, is illustrated in Figure 13. Detail of the RK4 time stepping algorithm and the electromagnetic field solver are
presented in Figures 14 and 15.

The algorithm begins with several initialisation routines, allocating memory to grid variables and reading in parameters of the simulation and physical property data. Initial values are then set for the variables with temporal dependence, the flow and temperature fields. Typically these are impulsive-start-constant-temperature conditions, with velocity and related variables being set to zero and the temperature being set to a constant initial value. This would be equivalent to flooding the area between the molten bath anode and the graphite electrode with a stagnant, hot plasma to initiate the arc.

Once initialised, the time-stepping routine begins. Each time step starts with the calculation of the electromagnetic fields, as per Figure 15. The electrical conductivity field is evaluated using the most recent temperature field, via linear interpolation of the input $\sigma - T$ data as described in section 3.3. Once the electrical conductivity field is known, the iterative multigrid solver is used to solve the governing equation (49) to give the electric potential field, $\phi$. The current density vector field $j$ is then calculated from both the $\phi$ and $\sigma$ fields, via the discrete version of (48). Once the current density is known, the magnetic vector potential $A$ is calculated by solving (58). This is performed using the fast Poisson solver, with appropriate modifications to the coefficients in the 2D case as discussed in section 3.4. The magnetic field $B$ is then calculated directly from $A$ by taking the discrete curl, as per (55).

Once the electromagnetic field calculation is complete, the value of the source term for the velocity equations is given by (38), and is computed by taking the cross product of the $j$ and $B$ fields.

The solution of the temperature equation (42) over the time step is accomplished using operator splitting, as discussed in section 3.3. The first part of this calculation, the solution of (118) to give the intermediate temperature $T^*$, is performed using the value of $T$ at the beginning of the time step and the current density field. The semi-implicit procedure used is described in section 3.3.

Once $T^*$ is known, the Runge-Kutta calculation for the values of both $T$ and the $a$ and $v$ fields at the end of the time step begins - these fields are governed by equations (35) and (42) respectively. This proceeds in a stepwise fashion for each of the successive RK4 stages, as shown in Figure 14. Starting from $T^*$ in the case of the temperature equation, and from the value of $a$ at the beginning of
the time step for the velocity equations, each stage begins with the calculation of intermediate fields of \( \mathbf{a} \) and \( T \) for that stage, as per the discrete RK4 equations presented in sections 3.2 and 3.3. The intermediate \( \mathbf{a} \) field is then used to calculate the gauge variable \( \theta \). This is governed by a simple Poisson equation, equation (36), and the calculation is performed using the fast Poisson solver. Once \( \theta \) is known, the value of the velocity field \( \nu \) is updated for the next RK4 step by direct calculation using equation (37).

Once all four stages are complete, the final values of \( \mathbf{a} \) and \( T \) at the end of the time step are calculated by linear combination of the intermediate values from each stage as per the standard RK4 algorithm, described in sections 3.2 and 3.3. The final value of \( \nu \) is then found, again by solving (36) for \( \theta \) using the final value of \( \mathbf{a} \), and then direct calculation from (37).

At this point in the algorithm the calculation for the current time step is complete, and the final steps proceed as shown in Figure 13. The values of the field variables of interest are stored to data files where necessary, and the time step size is recalculated according to the various inequalities presented in sections 3.1 and 3.3 that result from the use of the RK4 explicit time-stepping procedure. The values of the time-dependent variables at the start of the following time step are set equal to the current values, and the algorithm loops back to the electromagnetic field solver to begin the calculation for the next time step. The total time (sum of successive \( \delta t \) values) is monitored during the calculation, and the algorithm exits once a certain predefined maximum time has been reached.
Figure 13: Algorithm overview flowsheet

Begin

Initialisation routines
Set up grids and dimensions
Read in property data

Set initial values
\( a_0 = 0, v = 0, T_s = T_{\text{ini}}, \Phi = 0 \)

Electromagnetic Field Solver
\( T_s \rightarrow j, B \)

v source term calculation
\( j, B \rightarrow F \)

\( T \) semi-implicit calculation
\( T_s, i \rightarrow T \)

Runge-Kutta time step solver
\( a_{c_{n}} v, F, T' \rightarrow a_{n}, v, T \)

Write data to output files
\( v, T, j, B, \Phi \)

Calculate \( \delta t \) for next time step
\( v \rightarrow \delta t \)

Set new time step variables
\( a_{n} = a, T_{n} = T \)

Total time \( \geq t_{\text{sim}} \) ?

No

Yes

End
Figure 14: Runge-Kutta time step solver algorithm flowsheet
Estimated-load analysis

The algorithm is seen to be a series of calculation steps enclosed in a time stepping loop. Beginning with the electromagnetic field solver, we consider each step in terms of the approximate number of floating point calculations required. These are summarised in Table 2.

It is easy to see from this simple analysis that for the moderate to large sized grids \((I, J, K > 64)\) of interest, the DC plasma arc model algorithm will spend a large proportion of its time in the multigrid and fast Poisson solver routines. An effective means of parallelising these routines is therefore of some importance.
Table 2: Computational load estimates for full DC plasma arc model

<table>
<thead>
<tr>
<th>Algorithm step</th>
<th>Load, 2D</th>
<th>Load, 3D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electrical conductivity calculation</td>
<td>$IJ$</td>
<td>$IJK$</td>
</tr>
<tr>
<td>Electric field calculation (Multigrid solver)</td>
<td>$50IJ - 150IJ$</td>
<td>$50IJK - 150IJK$</td>
</tr>
<tr>
<td>Current calculation</td>
<td>$2IJ$</td>
<td>$3IJK$</td>
</tr>
<tr>
<td>Magnetic vector potential calculation (Poisson solver)</td>
<td>$2 \times (2\log_2 I + 2\log_2 J + 1)IJ$</td>
<td>$3 \times (2\log_2 I + 2\log_2 J + 2\log_2 K + 1)IJK$</td>
</tr>
<tr>
<td>Magnetic field calculation</td>
<td>$2IJ$</td>
<td>$3IJK$</td>
</tr>
<tr>
<td>Source term calculation, $\nu$</td>
<td>$2IJ$</td>
<td>$3IJK$</td>
</tr>
<tr>
<td>$T^*$ semi-implicit calculation</td>
<td>$4IJ - 6IJ$</td>
<td>$4IJK - 6IJK$</td>
</tr>
<tr>
<td>Intermediate $a$ calculation (RK4)</td>
<td>$4 \times 2IJ$</td>
<td>$4 \times 3IJK$</td>
</tr>
<tr>
<td>Intermediate $T$ calculation (RK4)</td>
<td>$4 \times IJ$</td>
<td>$4 \times IJK$</td>
</tr>
<tr>
<td>Gauge variable calculation (RK4) (Poisson solver)</td>
<td>$4 \times (2\log_2 I + 2\log_2 J + 1)IJ$</td>
<td>$4 \times (2\log_2 I + 2\log_2 J + 2\log_2 K + 1)IJK$</td>
</tr>
<tr>
<td>Recalculation of $\nu$ (RK4)</td>
<td>$4 \times 2IJ$</td>
<td>$4 \times 3IJK$</td>
</tr>
<tr>
<td>Calculate $\delta t$ for next time step</td>
<td>$2IJ$</td>
<td>$3IJK$</td>
</tr>
</tbody>
</table>

**SMP vs DMP for the DC plasma arc model**

As discussed in the introduction, there are two broad approaches to the problem of computer power parallelisation - shared memory multiprocessing, and distributed memory multiprocessing. The difference is illustrated schematically in Figure 16 and 17.

As discussed in the introduction, these two different parallel computing paradigms have advantages and disadvantages depending on the kind of problem that is required to be solved.

A particular weakness of DMP systems is that they do not scale well for problems in which a significant amount of information exchange between processors is required, as this information must be explicitly passed back and forth over the communication layer. For certain kinds of communication-intensive problems, most notably discrete problems using iterative solvers, this can be overcome by scaling to very large problem sizes - for this type of solver, the communication
required between nodes scales more slowly than the computational load required on each node, and parallel performance therefore increases with problem size.

Figure 16: Classical SMP architecture - many processors, one memory resource

Figure 17: Classical DMP architecture - many processors, each with their own memory resource

Unfortunately, a large proportion of the computational load in the DC plasma arc model is borne by the fast direct Poisson solvers. These solvers rely on FFT-based transforms for their performance, and as the FFT is a data-dense algorithm, it is somewhat poorly suited to DMP architectures. For two-dimensional transforms, this means that a large amount of inter-processor communication would be needed in the DMP case - since 2D transforms are computed by successive 1D transforms calculated along each coordinate direction, the first such sweep could be done locally on each node without the need for any communication at all. However, for the following sweep in the perpendicular direction, every value on the entire \( I \times J \) grid would have to be passed between nodes.
before the calculation could take place. This procedure would need to be executed twice (once for the forward transform, and once for the inverse). In 3D the situation is somewhat ameliorated by the fact that two of the three 1D transform steps could be performed locally before any communication is required, but the final step would still incur a communication penalty of $I \times J \times K$.

The important point is that the communication requirements of the 2D and 3D fast Poisson solvers would be of roughly the same order as the computational load requirements. Unless an extremely fast and low-latency communication layer is used the parallel performance of the algorithm will suffer, and scaling to higher grid resolutions would not significantly improve matters.

SMP overcomes the communication problems with the fast Poisson solvers, as the entire $I \times J \times K$ grid is located in the system's shared memory, and is therefore visible to all processors at all times. Additionally, SMP parallelism may be easily applied to the remainder of the the algorithm, which is very dependent on loop/sweep calculations over the numerical grids.

SMP is thus chosen as the parallelisation paradigm for the DC plasma arc model.

**Implementation of parallelism using OpenMP**

In an SMP system, the executing computer program splits into a number of parallel threads at certain points, each of which executes parts of the code in the program simultaneously. The threads may have access to shared global variables stored in system memory, or private local variables declared and used within the scope of the thread.

Using the fast direct Poisson solvers in 2D as an example (the 3D solvers are very similar, and simply involve more for loops), we may write them in pseudocode form to facilitate the parallelisation discussion. Given an input source term $S_{ij}$ and a variable we wish to solve for $u_{ij}$, the serial code would be written as shown in Figure 18.
Figure 18: Pseudocode for 2D fast Poisson solver

for i = 0 to I
  Do 1D sin/cos transform of $S_{i,j}$ in j-direction to get $S'_{i,j}$
for j = 0 to J
  Do 1D sin/cos transform of $S'_{i,j}$ in i-direction to get $S^t_{i,j}$

for i = 0 to I
  for j = 0 to J
    Calculate $u^b_{i,j}$ from $S^t_{i,j}$
for i = 0 to I
  Do inverse 1D sin/cos transform of $u^b_{i,j}$ in j-direction to get $u'^{b}_{i,j}$
for j = 0 to J
  Do inverse 1D sin/cos transform of $u'^{b}_{i,j}$ in i-direction to get $u_{i,j}$

Using SMP parallelisation, the outer loops may be split up into parts, each handled by a separate thread, and thus reducing the total execution time if each thread is executed on a separate processor. This is illustrated for the first for loop above in Figure 19, for the case of two threads.

Figure 19: Thread parallelisation of a for loop
As the vast majority of the code for the 2D and 3D DC plasma arc model is calculation-in-loop based, this approach is simple to implement (in principle) for most of the computationally intensive parts of the algorithm. With the use of the OpenMP SMP model, this can be achieved trivially by the insertion of a few directives into the serial code, and the explicit thread parallelisation shown in Figure 19 is then handled automatically by the code compiler.

Using C directives as an example, Figure 18 would be written as shown in Figure 20.

*Figure 20: Pseudocode for 2D fast Poisson solver; using OpenMP directives*

```c
#pragma omp parallel
{
    ...
    ...
    #pragma omp for
    for i = 0 to I
        Do 1D sin/cos transform of \( S_i, j \) in j-direction to get \( S'_i, j \)
    #pragma omp for
    for j = 0 to J
        Do 1D sin/cos transform of \( S'_i, j \) in i-direction to get \( S'^i, j \)
    #pragma omp for
    for i = 0 to I
        for j = 0 to J
            Calculate \( u^i, j \) from \( S'^i, j \)
    #pragma omp for
    for i = 0 to I
        Do inverse 1D sin/cos transform of \( u^i, j \) in j-direction to get \( u'^i, j \)
    #pragma omp for
    for j = 0 to J
        Do inverse 1D sin/cos transform of \( u'^i, j \) in i-direction to get \( u_{i, j} \)
    ...
    ...
    }
```

The “#pragma omp parallel” directive creates a team of threads from the single master thread running the program and maintains them for a block of code. The “#pragma omp for” directive used within this block then splits the *for* loop it is associated with among the available threads, and executes it in parallel.

The creation and destruction of thread teams is a computationally costly process, and it is thus advantageous to include as much code as possible within “#pragma omp parallel” blocks for
optimal performance. This is achieved in the DC plasma arc model algorithm by including everything from the $v$ source term calculation to the RK4 solver steps within a single parallel block (see Figure 13). The various solvers and loops within the block are then parallelised with for directives where necessary.

Due to the iterative and multi-resolution nature of the multigrid algorithm, the electromagnetic field solver must be treated somewhat differently. The portions of the solver before and after the electric field calculation in Figure 14 are each enclosed in their own separate parallel blocks, isolating the multigrid solver. The serial multigrid solver can be written in pseudocode as shown in Figure 21. Here a V-cycle solver is used as an example, but any of the more complex cycles will analyse similarly.

![Pseudocode for multigrid solver](image)

This is parallelised by enclosing each step of the downward and upward stroke of the V-cycle in a parallel block. This is rather fine-grained parallelism, however, it is necessary to maintain parity between the various grid calculations at each level, and maintains reasonable computational efficiency as there are a number of separate for loops within each of the relaxation, residual calculation, restriction, and interpolation subroutines.

The pseudocode for the multigrid solver using OpenMP C directives would look like Figure 22. As usual, any for loops within the subroutines inside the parallel blocks are parallelised with “#pragma omp for” directives.
This completes the discussion of the parallel implementation of the DC plasma arc model. A performance analysis will be presented in the results section to given an indication of the gains to be had using SMP over serial code on multiprocessor machines.

3.8 Data management

One of the main aims in designing the DC plasma arc model is for it to run at high grid resolutions in both two and three dimensions. With this focus, the size of the data used internally by the models, as well as that written to storage media for data logging during a model exercise, can become excessive and cause performance degradation.

Internal data structures

The live data used to represent the variables of interest within the model while it is running are stored as double- or triple-indexed arrays of floating point variables, representing the 2D or 3D finite difference grids used by the algorithm. If all intermediate and other variables are given their own array storage, the arrays required are shown in Table 3.
Table 3: Number of grid variables required for 2D and 3D algorithms

<table>
<thead>
<tr>
<th>Algorithm section</th>
<th>Number of IJ arrays in 2D, IJK arrays in 3D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multigrid solver</td>
<td>4/3 x 5, 8/7 x 5</td>
</tr>
<tr>
<td>Fast Poisson solver</td>
<td>2, 1</td>
</tr>
<tr>
<td>Runge-Kutta time step solver</td>
<td>3 x 4, 4 x 4</td>
</tr>
<tr>
<td>Electromagnetic field solver</td>
<td>7, 11</td>
</tr>
<tr>
<td>Main variables</td>
<td>10, 13</td>
</tr>
</tbody>
</table>

This is a total of ~37 for 2D, and ~47 for 3D. This factor can be used to obtain an estimate of the amount of system memory the algorithm requires. It is instructive to consider the case of grids of size $I = J = K$; some results of this calculation are shown in Table 4 below, assuming 32-bit precision data (these values would double for 64-bit precision).

Table 4: Total system memory requirement for 2D and 3D algorithms

<table>
<thead>
<tr>
<th>Grid density $I$ (grid size = $I^2$, $I'$)</th>
<th>Memory required - 2D</th>
<th>Memory required - 3D</th>
</tr>
</thead>
<tbody>
<tr>
<td>128</td>
<td>2.3 MB</td>
<td>374 MB</td>
</tr>
<tr>
<td>256</td>
<td>9.2 MB</td>
<td>2.9 GB</td>
</tr>
<tr>
<td>512</td>
<td>37 MB</td>
<td>23 GB</td>
</tr>
<tr>
<td>1024</td>
<td>147 MB</td>
<td>187 GB</td>
</tr>
<tr>
<td>2048</td>
<td>587 MB</td>
<td>1.5 TB</td>
</tr>
</tbody>
</table>

What is immediately obvious from this analysis is that the 3D DC plasma arc model needs enormous system memory resources as it scales to higher resolutions. In the range of resolutions of interest, typically $I = 256$ to 1024, the 3D model requires around three orders of magnitude more system memory than the 2D model.

**Memory usage optimisation**

Some streamlining of the DC plasma arc model algorithm is possible in terms of the number of intermediate variables required.
The procedure required for the Runge-Kutta time stepping solver can be greatly optimised. Each intermediate $a$ and $T$ variable in the RK4 solver makes a linear contribution to the final value at the end of the time step. As a result, if the final values for the $a$ and $T$ variables are updated incrementally at every stage of the solver instead of only at the end, only two instead of four intermediate variables can be used. Additionally, any other grid variables that are idle during the RK4 calculation may be used to store the intermediate variables temporarily. This requires some rearrangement of the order of certain steps in the DC plasma arc model algorithm (most importantly, positioning the file output and data capture at a point in the algorithm where all variables possess the correct values), but it gives a large advantage in terms of the memory saving.

Some optimisation can also be performed on the multigrid algorithm. One intermediate variable can be eliminated, and pointers can be used to refer the highest level of the correction grid directly to the $\phi$ variable, and similarly the highest level of the $\sigma$ grid directly to the $\sigma$ variable. These optimisations can reduce the total memory requirement by more than $\frac{1}{3}$.

**File I/O and data storage**

The DC plasma arc model produces two kinds of data as it is running - spatial data, describing the values of a particular variable over the region of space covered by the grids in the model, and temporal data, describing how variables change at particular locations over time. A representative amount of this data must be collected and stored for analysis after the model has completed its calculations.

Recording all such data generated by the model would imply writing each grid variable out to a data file at every time step. This is extremely expensive in terms of storage space and I/O performance, even in 2D. The total data storage space required for a single run may be anywhere between 17 GB and 270 GB, using moderately conservative estimates.

Performance of the algorithm can also suffer, as the time taken to write output data files onto slow storage media like hard disk drives is anticipated to be significant in comparison with the time spent by the processor(s) in computing the rest of the algorithm for a given time step. This problem is likely to be further exacerbated by SMP machines, which increase the algorithm's calculation speed but retain similar I/O performance to serial computers.
In three dimensions, these difficulties are even worse, particularly as regards the storage space requirement. Some means of reducing the amount of data logged by the model as it runs is thus highly desirable.

The first approach that may be used involves decoupling the need for spatial data from the need for temporal data. Full spatial data is particularly useful for qualitative analysis of the model results, producing data visualisations both static and animated, and using these to study the dynamic behaviour of the DC plasma arc. Temporal data is more useful for quantitative study of the system, examining how the numerical values of the variables change over time. As it is quantitative, the temporal data must in general be reduced in dimensionality from the original spatial grid variables in some way, by simply reading the variable values at a small number of preselected locations, estimating maxima and minima across the grid, and so forth. We can thus envisage the data sampling scheme which is shown schematically in Figure 23.

*Figure 23: Decoupled spatial and temporal data logging for the 2D algorithm*

Storing of the full spatial data for the grid variables of interest is performed only once every certain number of time steps. This reduces the temporal resolution of the spatial data, but greatly reduces the file I/O overhead and data storage requirements while retaining enough temporal detail for visualisation and qualitative analysis. Meanwhile, the values of the variables of interest are logged to a data file for *every* time step only at a certain (small) number of pre-specified locations in the calculation domain - one such location is shown in Figure 23.
With this scheme, the overhead involved in storing data output from the model may be reduced by an order of magnitude or more.

**Further data compression - 3D resolution reduction**

The 3D model poses major challenges for data storage once it scales to high resolutions - reducing the data load by using the decoupled spatial/temporal logging scheme alone is insufficient.

The full spatial grid data for 3D algorithms is again used predominantly for qualitative visualisation work. With this in mind, it is subjected to additional lossy data compression by the simple expedient of downsampling the grid dimensions prior to file storage. Data values on the downsampled grid are found by linear interpolation from the native grid - if power-of-two grids are used, downsampling ratios of 2, 4 or 8 results in a simple read from the native grid without additional computational overhead, and greatly reduces the system storage space required. This process is shown schematically in Figure 24.

*Figure 24: Schematic of grid downsampling process*

3.9 Data visualisation

In order to discuss qualitative aspects of the DC plasma arc model, the data output by the models must be post-processed into a form more digestible by human beings. For the very large spatial grids typically used, presenting the data in a visual form is the most obvious and intuitive way to do this. Due to the dynamic nature of the models, collections of spatial data sets varying in time may
also be visualised and then stacked together to form an animation.

2D data - static image visualisation

In two dimensions, output data from the model are represented by grid variables of dimension \( I \times J \). Vector variables such as velocity or current may be represented in scalar form by magnitude, streamlines, and/or vorticity.

The first step is performed by sampling from the native grid data using simple bilinear interpolation, as shown in Figure 25. The desired output resolution, \( I \times J \), should be of the same aspect ratio as the native grid \( I \times J \), in order to avoid distortion.

![Bilinear interpolation from native grid variables to pixel grid variables](image)

Colour translation is then performed by normalising the values \( u_{i,j} \) on the pixel grid to give values \( u' \). Any values of \( u' \) less than zero or greater than one are truncated to those values. \( u' \) is then translated into red, green, and blue colour component values using colour functions.

Once converted into colour values, the pixel data are written out to a standard bitmap image file format for examination. The complete flowsheet for visualisation of a single frame in 2D is shown in Figure 26.
2D data - animation

Dynamic behaviour is a very important component of the DC plasma arc modelling study. The natural way to qualitatively interpret this behaviour is by compiling a number of static frame images at different model times into an animation.

As the successive data files output by the model are generally not spaced at constant model-time intervals (due to the adaptive time stepping of the RK4 algorithm), temporal sampling must be performed. We create an evenly-spaced sequence by using linear interpolation between the data files. Each interpolated frame is then rendered using the standard procedure for static images described earlier, given a sequentially-numbered file name, and the set of images produced is compiled into an animation using the open source FFmpeg software⁵¹. The algorithm flowsheet for producing an animation visualisation is shown in Figure 27.
3D data - static image visualisation

In three dimensions, visualisation techniques are somewhat complicated by the fact that the dimensionality of the data is no longer equivalent to the dimensionality of the computer screen on which it is to be displayed. A 3D rendering method must be employed to convert the three-dimensional data set into a two-dimensional image - for visualisation of the output data from the DC plasma arc model, the open source VTK library\textsuperscript{52} was used in conjunction with the Python scripting language\textsuperscript{53} to perform this task. Various methods exist within the VTK library for
rendering both scalar and vector fields. Some of these methods are shown in Table 5.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scalar</td>
<td>vtkContourFilter (contour surfaces)</td>
</tr>
<tr>
<td>Vector</td>
<td>vtkStreamLine (streak/stream lines)</td>
</tr>
<tr>
<td>Vector</td>
<td>vtkHedgeHog (vector arrows)</td>
</tr>
</tbody>
</table>

Table 5: Some useful VTK library methods for rendering of 3D data

The general algorithm used involves reading in a data set output from the 3D DC plasma arc model along with some user-specified information about the plot method to be used, parsing the data set into a form compatible with the VTK library, automatic writing of a script file to perform the render operation, and execution of the script to generate an image file.

The algorithm for scalar data is shown schematically in Figure 28.

Figure 28: Algorithm flowsheet for rendering of scalar data using contours
For vector data, additional steps may be required. Using a streamline calculation as an example, the user must supply the number and location of start points for the calculation of streamlines, which is performed by numerical integration in the VTK library. The streamlines are coloured automatically by VTK, using scalar data given by the magnitude of the vector field. The algorithm for rendering a vector field with streamlines is shown in Figure 29.

*Figure 29: Algorithm flowsheet for rendering of vector data using streamlines*

```
Input number and location of start points
Input colour scheme to use
Input viewpoint location

Read in vector grid variable
u_{x,i}, u_{y,j}, u_{z,k}, each I x J x K

Calculate vector magnitude
u_{x,i}^2 + u_{y,j}^2 + u_{z,k}^2 = |u|

Write vector components and magnitude data to VTK binary file

Write Python script to create and render streamlines using VTK file

Execute Python script to produce rendered image
```

**3D data - animation**

The flow diagram and procedure for animating 3D data is identical to Figure 27 for 2D, with the exception that the data to be interpolated in time now lies on grids of three (as opposed to two) dimensions, and may be in vector form, requiring three separate interpolations to be performed for each component. Once the interpolated data is available, either of the 3D visualisation blocks from
Figure 28 or 29 is substituted for the 2D data-to-image operation in Figure 27, and the algorithm proceeds as for the two dimensional case.
Chapter 4 - Test Cases

4.1 Introduction to test cases

In the following chapter, several numerical tests will be conducted to examine various aspects of the performance of the two-dimensional DC plasma arc model. The aims of this chapter are twofold: To demonstrate the capabilities of the individual sub-models for fluid flow and electromagnetism, and to tune, compare, and select parameters for the sub-models where this is required. The tests are confined to the two-dimensional models as these make up the majority of the cases examined in later chapters. Due to the similarity in algorithms, the results should also give a reasonable indication of the behaviour of the three-dimensional models.

The chapter will begin by examining the properties of the various fluid flow solvers as applied to the case of evolution of a steady gas jet in a cavity. Following on from this, the stability and quality of the solutions produced by the solvers for the case of an unsteady jet will be presented. A third section will then examine aspects of the electromagnetic field solver, including selection of an optimal value of δz for the 2D magnetic field model using an analytical problem, and performance tuning of the multigrid algorithm. Finally, the complete DC plasma arc model is evaluated from a parallel performance standpoint using a representative set of parameters.

4.2 Test case 1 - evolution of a steady laminar jet

The first test case for the DC plasma arc model examines the performance of the flow solvers in calculating the evolution of a two-dimensional steady-state gas jet. The aim is to compare the three flow models discussed earlier, primitive variables, vorticity-stream function, and gauge method, each of which are used to solve the same problem.

Geometry and spatial layout

For the two-dimensional jet tests, the geometry used describes a rectangular region of space with a velocity source term region located at its centre. This is shown schematically in Figure 30.
The origin of a cartesian coordinate system is placed at point D, as shown in Figure 30. The height of the domain is given by $y_{AD} = y_{BC}$, and the width is given by $x_{AB} = x_{CD}$. The source region is square, with dimensions $2x_f$ and $2y_f$, and is located at the centre of the domain, $x = x_{AB}/2, y = y_{AD}/2$.

The boundaries ABCD are treated as no-slip walls, with boundary conditions $v_x = v_y = 0$. All jet test models are subject to impulsive start initial conditions, with the velocity field or equivalent variables set to zero at the beginning of the calculation. The jet column produced by the source region evolves and settles to steady state as the simulation proceeds.

Several parameters must be given to define the physical and numerical behaviour. Firstly, the (constant) density and viscosity of the fluid are required. For a particular case, the grid dimensions $I$ and $J$ must also be given. From these, and the dimensions $y_{AD}$ and $x_{AB}$, values for $\delta x$ and $\delta y$ can be calculated.

For all 2D models, test cases and DC plasma arc models included, the region dimensions and grid resolution will always be chosen to give $\delta x = \delta y = \delta l$ in order to simplify the numerical mathematics and improve performance. $I$ and $J$ will additionally be chosen to have large factors that can be expressed as $2^n$ for integral $n$, in order to maximise the performance of the FFT-based Poisson solvers and multigrid solver.

The number of threads to be used, and the maximum model time (the final time $t$ at which the model will complete its calculations), are the last pieces of information needed by the model.
The parameters used in the simulation are described in Table 6.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Region height, ( y_{AD} )</td>
<td>0.1 m</td>
</tr>
<tr>
<td>Region width, ( x_{AB} )</td>
<td>0.4 m</td>
</tr>
<tr>
<td>Fluid density, ( \rho )</td>
<td>0.01647 kg/m(^3)</td>
</tr>
<tr>
<td>Fluid viscosity, ( \mu )</td>
<td>1.579 x 10(^{-4}) Pa.s</td>
</tr>
<tr>
<td>Number of threads</td>
<td>4</td>
</tr>
<tr>
<td>Maximum model time</td>
<td>100 ms</td>
</tr>
</tbody>
</table>

**Source term**

The source terms for the steady-state jet problem are given by

\[
F_x = A_f \left[ \frac{x - \frac{1}{2} x_{AB}}{x_f} + \frac{1}{1000} \left( 1 - \left( \frac{x - \frac{1}{2} x_{AB}}{x_f} \right)^2 \right) \left( 1 - \left( \frac{y - \frac{1}{2} y_{AD}}{y_f} \right)^2 \right) \right],
\]

\[
F_y = -A_f \left[ 1 - \left( \frac{x - \frac{1}{2} x_{AB}}{x_f} \right)^2 \right] \left( 1 - \left( \frac{y - \frac{1}{2} y_{AD}}{y_f} \right)^2 \right),
\]

\[
\frac{1}{2} x_{AB} - x_f < x < \frac{1}{2} x_{AB} + x_f, \quad \frac{1}{2} y_{AD} - y_f < y < \frac{1}{2} y_{AD} + y_f.
\]

The source terms are defined as zero outside the ranges of \( x \) and \( y \) specified. For the steady state jet case, the parameters in (215) - (217) are specified in Table 7. These values give a peak jet velocity of the order of 70 m/s.

**Convergence results for gauge method**

The gauge method is used in the full DC plasma arc model, as applied to the various problems presented in Chapters 5 and 6. As such, it is prudent to examine the convergence of this method.
more closely.

In order to do this, the gauge method solver was run at a variety of grid resolutions from 256 x 64 up to 2048 x 512. After 10ms of model time, an error estimate for the method was performed by calculating the discrete divergence of the velocity fields. The L¹ norm of the divergence field (exact solution should give zero) was then computed as an estimate of the error and plotted against the grid size δl using logarithmic scales. The L¹ norm is defined as the arithmetic mean of the set of discrete values of the divergence field as represented on the numerical grid used for the problem.

Table 8: Divergence errors at 10ms using gauge method

<table>
<thead>
<tr>
<th>Simulation resolution, I x J</th>
<th>δl</th>
<th>L¹ error</th>
</tr>
</thead>
<tbody>
<tr>
<td>256 x 64</td>
<td>1.563 x 10⁻³ m</td>
<td>2.652 x 10⁻²</td>
</tr>
<tr>
<td>384 x 96</td>
<td>1.042 x 10⁻³ m</td>
<td>1.187 x 10⁻²</td>
</tr>
<tr>
<td>512 x 128</td>
<td>7.813 x 10⁻⁴ m</td>
<td>6.693 x 10⁻³</td>
</tr>
<tr>
<td>768 x 192</td>
<td>5.208 x 10⁻⁴ m</td>
<td>2.975 x 10⁻³</td>
</tr>
<tr>
<td>1024 x 256</td>
<td>3.906 x 10⁻⁴ m</td>
<td>1.372 x 10⁻³</td>
</tr>
<tr>
<td>1536 x 384</td>
<td>2.604 x 10⁻⁴ m</td>
<td>4.063 x 10⁻⁴</td>
</tr>
<tr>
<td>2048 x 512</td>
<td>1.953 x 10⁻⁴ m</td>
<td>1.714 x 10⁻⁴</td>
</tr>
</tbody>
</table>

Figure 31: Graph of L¹ divergence error vs δl for gauge method

As can be seen from the gradient of the graph in Figure 31, the rate of convergence is approximately quadratic with respect to spatial discretisation over the range of resolutions tested,
confirming that the gauge method does indeed have predominantly spatially second-order accuracy.

Additional grid-independence tests, time step size tests, and initial condition testing for the full 2D DC plasma arc model is presented in Appendix 3.

**Comparison of models**

To compare the performance of the different fluid flow solvers, each was used for the steady state jet calculation at a resolution of 1024 x 256. The evolution of the jet velocity at a single point was then compared, together with the spatial jet profile at the end of the simulation (at 100ms). The results are shown in Figures 32 - 34.

*Figure 32: Temporal evolution of $v_y$ at point $x = 0.2m, y = 0.025m$*

*Figure 33: Spatial variation of $v_y$ along jet centreline, $x = 0.2m$, at time = 100ms*

*Figure 34: Temporal evolution of $v_x$ at point $x = 0.2m, y = 0.025m$, initial period only*
There is very good agreement between the gauge method and the primitive variable formulation in this test; this may be expected, as their functional forms are quite closely related. There is however some difference between these two methods and the vorticity-stream function method, particularly in regions of elevated velocity where the deviation can be up to 5%.

The difference in results between vorticity-stream function and the other methods arises predominantly because the method is structurally quite different to the other two. It implicitly enforces incompressibility while computing variables that are related to the derivative of the velocities, as opposed to the gauge and primitive variable methods which calculate incompressibility explicitly via pressure or gauge variables and work directly with velocity or velocity-related variables.

Comparison of resolutions

To show the effect of resolution scaling on the flow models, a similar analysis was performed using only the gauge method model, varying model resolution between 256 x 64 and 1024 x 256. Evolution in time and space of the \( v_y \) variable was graphically compared for the various resolutions. The results are shown in Figures 35 – 37.

For this test case problem, the gauge method at 512 x 128 resolution performs nearly identically to the method at 1024 x 256 resolution. At 100ms, spatial agreement between all three resolutions tested is extremely good - the method is able to retain reasonable accuracy for moderate grid sizes.

---

**Figure 35:** Gauge method evolution of \( v_y \)

at point \( x = 0.2m, y = 0.025m \)

**Figure 36:** Gauge method - spatial variation of \( v_y \)

along jet centreline, \( x = 0.2m, \) at time = 100ms
Visualisation plots

A qualitative comparison between the model results at steady state may be obtained by calculating and plotting the vorticity field at the end of the steady state simulation, at 100ms simulation time. These are shown below in Figures 38 - 40 for the three different model types, each at resolution 1024 x 256.

Figures 38 - 40: Plot of vorticity variable for various flow models, scale range -3500:3500 s⁻¹

Fig. 38: Primitive variable model  
Fig. 39: Vorticity-stream function method  
Fig. 40: Gauge method

Qualitatively, there is very little to choose between the different flow models for the steady state jet case. Each produces a dominant central jet flanked by symmetric recirculation cells.

Animations of jet formation and steady state profile evolution for each method may be found in the animations/testing2D/steadyjet directory on the accompanying disk.
4.3 Test case 2 - evolution of an unsteady jet

The aim of this case is to test the various flow models' ability to deal with high-velocity unsteady state flow calculations, particularly as regards stability of their algorithms and boundary conditions. The calculation performed is of an unsteady state gas jet. The source term is described as for the steady state case, by (215) - (217).

The parameters used for the models were identical to those shown in Tables 6 and 7, with the exception of the maximum model time which was shortened to 10ms, and the value of $A_f$ which was increased to $5 \times 10^3$ N/m$^3$ - this produces jet velocities of the order of 600 m/s.

At elevated velocities, the non-linear momentum components of the Navier-Stokes equations begin to dominate the linear viscous terms in and around the source term region. As material is accelerated through the source region, a large shear gradient develops between the fluid in the jet and the fluid immediately outside it. For sufficiently large velocities, this shear gradient leads to the development of Kelvin-Helmholtz instabilities at the jet boundary. These result in the generation of transient vortices that travel down the jet column and into the region, where they dissipate and are recycled by the circulating flow that develops.

The jet column is initially highly symmetric (reflecting the symmetry of the problem) but small irregularities in the initial conditions are rapidly magnified by the non-linear nature of the momentum-dominated fluid mechanics and the jet breaks down into asymmetrical behaviour.

The simulations were performed at a resolution of 512 x 128.

**Primitive variable formulation**

This flow solver method has difficulty with the high velocity flow, and becomes unstable and ultimately crashes as the gas jet reaches the lower boundary and begins interacting with it. Considerable oscillation of the solution is observed near the boundary immediately prior to the model crashing - this is shown in Figure 41.

This is believed to be a fundamental failure of the primitive variable method, and may be related to
use of a non-staggered grid and the resulting spatial discretisation at the boundary. This is a somewhat surprising result, since this method was explicitly derived for use on non-staggered grids\textsuperscript{36}.

\textit{Figure 41: Plot of vorticity variable at time = 0.3ms, scale range -80000:80000 s\textsuperscript{-1}}

An animation of the jet formation phase of this model is available on the accompanying disk, in the directory animations/testing2d/unsteadyjet. Note how the solution deteriorates from smooth behaviour as the head of the jet encounters the boundary.

\textbf{Vorticity-stream function formulation}

The vorticity-stream function method copes somewhat better with the high velocity flow, maintaining stability where the primitive variable solver failed. It is able to calculate the full 10ms of unsteady jet behaviour stably, however, the quality of the solution is fairly poor. As can be seen in Figure 42 below, the solution using this model exhibits oscillations at various points in the field during the calculation, and particularly in regions of high shear.

\textit{Figure 42: Plot of vorticity variable at time = 0.3ms, scale range -80000:80000 s\textsuperscript{-1}}
Animations of the jet formation phase and full 10ms run of this model are available on the accompanying disk, in the directory animations/testing2d/unsteadyjet.

**Gauge method**

By contrast with the other two methods, the gauge method model is able to calculate the high velocity flow reasonably well. The solution remains smooth even during the initial contact between the jet and the lower boundary, and stability is maintained throughout the 10ms simulation time. Figure 43, plotting the vorticity variable at the same model time as the two Figures above, shows the difference clearly.

*Figure 43: Plot of vorticity variable at time = 0.3ms, scale range -80000:80000 s⁻¹*

Animations of the jet formation phase and full 10ms run of this model are available on the accompanying disk, in the directory animations/testing2d/unsteadyjet. Note the symmetry breaking of the jet soon after it develops, and the very chaotic turbulent motion of the fluid that results.

Comparing these with the animations for the earlier methods shows that the gauge method appears to preserve smoothness and a “physically realistic” solution far more readily than the vorticity-stream function and primitive variable methods. The gauge method would seem to be the best choice for high velocity gas jet flows, which the DC plasma arc is understood to be a form of.

**4.4 Test case 3 - electromagnetic solver tests**

The third test case examines various aspects of the electromagnetic field solver developed for the DC plasma arc model. The aim of the first part of the case is to examine the effect of various values of δz on the 2D magnetic field solver. The second aims to demonstrate the effect of various
algorithm-design decisions on the performance of the geometric multigrid solver for the electric potential.

**Magnetic field solver test case**

In order to test the magnetic field solver, we set up a constant conductivity linear conductor model in two dimensions. This is shown schematically in Figure 44.

*Figure 44: Model geometry and boundaries for 2D magnetic field solver test*

The origin of a cartesian coordinate system is placed at point F, as shown in Figure 44. The height of the domain is given by \( y_{AF} = y_{DE} \), and the width is given by \( x_{AD} = x_{EF} \). The width of the conductor region is given by \( x_{BC} \), and the conductor is located along the centreline of the region.

The electrical conductivity is specified to be a constant value within the conductor, and a small value close to zero outside it. The conductor's width is calculated from a given current, by assuming a radially symmetric conductor carrying a constant current density. These and other parameters for the model are given in Table 9.

*Table 9: Properties and dimensions for linear conductor test*

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Region height, ( y_{AF} )</td>
<td>0.1 m</td>
</tr>
<tr>
<td>Region width, ( x_{AD} )</td>
<td>0.4 m</td>
</tr>
<tr>
<td>Electrical conductivity</td>
<td>5000 ( \Omega^{-1} \text{m}^{-1} )</td>
</tr>
<tr>
<td>Current density</td>
<td>3.5 x 10^7 A/m²</td>
</tr>
</tbody>
</table>
In order to accurately represent the behaviour of the magnetic field in the vicinity of a DC plasma arc, the calculations of the two-dimensional linear conductor model should closely match the theoretical variation of magnetic field strength around a current carrying conductor. For an infinitely long conductor of radius \( r_k \) carrying current \( I_a \), the magnetic field at a distance \( r \) from the conductor's centreline is given by the relationship (218).

\[
B_z = \frac{\mu_0 I_a r}{2 \pi r_k^2}, \quad r < r_k, \quad B_z = \frac{\mu_0 I_a r}{2 \pi r^2}, \quad r \geq r_k
\]

(218)

\( B_z \) increases linearly with radial distance until it reaches a peak at the surface of the conductor, from which point it decreases inversely proportional to radial distance.

Some examples of the results of the two-dimensional model at a grid resolution of 1024 x 256 are shown below. The first is current density, shown in Figure 45.

*Figure 45: Plot of \( j_r \) for a linear conductor at current 1 kA, scale range -4 x 10^7 to 4 x 10^7 A/m²*

As can be seen, the low conductivity of the regions surrounding the central conductor concentrate all the current inside it. This current density distribution gives rise to a magnetic field, shown in Figure 46.

*Figure 46: Plot of \( B_z \) for a linear conductor at current 1 kA, scale range -0.07 to 0.07 T*

With the linear symmetry of the conducting region, the resultant magnetic field is symmetric around the conductor's centreline. The exact behaviour of the field is governed by the value of \( \delta z \) used in
the 2D magnetic field model.

In order to compare the 1D theoretical behaviour with the 2D model results, the \( B_z \) data is sampled on a horizontal line in the \( x \)-direction from the centreline of the conductor out to the wall. This data is compared to the theoretical model \( (218) \) at each value of \( x (= r) \), taking \( x = 0 \) at the centreline of the conductor.

As accuracy is most important in the region near to the conductor where the highest values of \( B_z \) occur, two errors are calculated - the peak error, which measures the difference in values of \( B_z \) at the surface of the conductor at \( r = r_s \), and the close range \( L^1 \) error, which is an average of the errors between \( r = 0 \) and \( r = 2r_s \).

This error calculation was performed for a variety of currents and model resolutions in order to locate an optimal value of \( \delta z \). Since the dimensions of the conductor change with current, the errors are plotted against a dimensionless \( r_s \) multiplier, defined as \( \delta z/r_s \). The results are shown in Figures 47 - 50.

For all cases, \( \delta z/r_s = 2 \) appears to be an optimal choice in order to produce the lowest close range errors between the 2D magnetic field solver and the theoretical variation of magnetic field around a cylindrical conductor. A plot of the \( r (= x) \) dependence of the magnetic field for various multipliers including the optimal value of 2 is shown in Figure 51.
It is important to note that while the optimal value of the $r_k$ multiplier gives very good agreement with the theoretical curve within and near to the surface of the conductor, the agreement becomes quite poor as distance from the conductor surface increases. This is due to the fact that the approximate 2D magnetic field model produces an exponential decay of $B_z$ with distance, whereas the theoretical expression changes more slowly as the inverse of distance.

As a result of this limitation, the complete two-dimensional model for DC plasma arcs may give highly inaccurate results in cases where the interactions of multiple arcs a significant distance apart are being studied. The 2D arc model is thus best suited for the study of the dynamics of single arc systems where the close-range interactions are dominant. The full 3D model is reserved for multiple arc systems.
Multigrid solver performance

For this test case, the parameters are the same as shown in Table 9, with the exception that a variable electrical conductivity field is supplied to the electric potential solver. The aim of this test is to evaluate the performance of the iterative geometric multigrid solver using variations on the standard algorithm.

The electrical conductivity field was constructed from the Bowman arc shape model perturbed with sinusoidal variations, to be representative of the strongly anisotropic structure that is expected in the region in and around the DC plasma arc column - it is shown in Figure 52.

*Figure 52: Variable electrical conductivity field used for multigrid tests, scale range 0-10000 $\Omega$ m*^{-1}

Using this and the specified current density at the top surface, the multigrid solver produces a calculated electric potential field. Since the solver is iterative, an initial guess of $\phi = 0$ is supplied. An example at resolution 1024 x 256 is shown in Figure 53.

*Figure 53: Electric potential calculated by multigrid tests, scale range 0-234 V*

Several design choices can be made with regard to the exact algorithm used by the multigrid solver. For each option, the convergence rate of the solver was computed by evaluating the residual of the electric potential field after each cycle iteration and normalising it based on the residual at the beginning of the calculation. The change of this normalised residual as a function of cycle iterations is then examined.
The first option is the choice of cycling scheme to use. Several variations on the standard V-cycle are possible, and we use the observation of Tackley\(^\text{49}\) that for elliptic problems with strongly anisotropic coefficients, multigrid cycles offering more coarse grid level iterations are more effective. F-cycles perform additional coarser-grid calculations and are simple to implement. The difference in performance is shown in Figure 54.

**Figure 54: Effect of V- vs F-cycles on normalised residual, resolution 1024 x 256**

![Graph showing the effect of V- and F-cycles on normalised residual](image)

The change of algorithm has a pronounced effect, reducing the number of iterations needed to converge the solution to an accuracy of \(10^{-6}\) from 10 to 6. As the F-cycle requires only a moderate increase in computational power compared to the V-cycle, it is more efficient.

**Figure 55: Effect of number of relaxation iterations on normalised residual, resolution 1024 x 256**

![Graph showing the effect of number of relaxation iterations](image)

A significant difference can also be made by changing the number of relaxation calculations
performed at each grid level in the multigrid algorithm. As shown in Figure 55, increasing these improves the rate of convergence to some degree, particularly if the algorithm enters a stalled phase.

The performance penalty for additional relaxation steps is quite high, although still less than the cost of an entire multigrid step, so the trade-off of increased convergence rate vs performance is not cut and dried. In practice, two relaxation steps is seen to be a reasonable choice. With the choice of F-cycle iterations with two relaxation calculations at each grid level, it is instructive to examine the behaviour of the multigrid solver as a function of grid resolution. This is shown in Figure 56 for the test case using a variety of grid resolutions.

![Figure 56: Effect of resolution on performance of multigrid solver](image)

The scaling of the multigrid solver with resolution is fairly good. Even though the convergence rate of the solver begins to stall at around five iteration steps, by this point it has already calculated the solution to an error of less than $10^{-6}$ at all the grid resolutions considered. Given that the multigrid solver in the full DC plasma arc model will be starting with a much better initial guess of the electric potential field (the values of the field from the previous time step) than has been used here, good accuracy is assured if at least five cycle iterations are used.

### 4.5 Test case 4 - parallel performance

The aim of the final test case is to examine the performance of the full DC plasma arc model algorithm in SMP parallel environments. Due to the computational load faced by the high resolution
two- and three-dimensional models, parallelism is an important optimisation to ensure reasonable performance of the models on modern computers.

For each test, the complete 2D model including gauge method fluid flow solver, temperature solver, and electromagnetic field solver was used. The geometry and various model parameters were equivalent to those used in the 2D base case model in the following section. 300 time-step iterations were performed, with the last 100 timed to give the average wall-clock execution time for a single step.

The first part of the test examines the overhead incurred by multiple threads. This can become an important factor in the overall performance of the model on highly-parallel architectures where large numbers of threads must be created and maintained. This was compared by running the model on a single processor computer, with a fixed resolution (1024 x 256) and other parameters, while changing the number of threads. The performance of the model was compared to that of the single-thread code to determine the additional computational cost associated with threading - the results are shown in Figure 57.

*Figure 57: Thread overhead vs number of threads used, Intel Pentium M 1.8GHz*

The overhead associated with threading is seen to increase approximately linearly with the number of threads. This is to be expected, as the extra work associated with creating and managing a set of threads will increase in proportion to the number of threads involved. It is important to note that the overhead can become a significant component of the model execution speed as the number of threads increases. For threaded code with overheads, a slightly modified version of Amdahl's law
can be written for the execution time of the code. This is given by

\[
t_E \propto t_S + \frac{t_p}{n_p} + A_{t_o} |n_p - 1|, \quad S = \frac{t_S + t_p}{t_S + \frac{t_p}{n_p} + A_{t_o} |n_p - 1|}.
\] (219)

Here, \( t_s \) and \( t_p \) are the times taken by a single thread for the serial and parallel portions of the code respectively. \( A_{t_o} \) is the thread overhead proportionality constant associated with additional threads, as per Figure 57. The relative speed up of the code, \( S \), indicates how much faster it will perform in comparison to a single thread. In Amdahl's theoretical model of performance, \( S \) is seen to first increase to a maximum and then decrease as the number of threads \( n_p \) is increased. The value of the optimal \( n_p \) is entirely problem- and computer-architecture-dependent, and should thus be carefully studied before any time-consuming use of the code is considered.

With this in mind the second part of the test case for parallelism examines the performance increase of the DC plasma arc model on computers with multiple processors. The model was run at various resolutions between 512 x 128 and 2048 x 512, and the speed up factor \( S \) was calculated as a function of the number of threads used, up to the total number of physical processors available in the machine. The results are shown in Figure 58.

*Figure 58: Speed up vs number of threads and model resolution, Intel Core2 Q6600 2.4GHz*

Several points are immediately obvious. For this particular problem and computer architecture, the maximum number of threads/CPUs can be safely used, as the optimal \( n_p \) is clearly > 4. It is interesting to note that the achievable speed up degrades quite quickly compared to the theoretical
performance curve, particularly at lower resolutions as expected. The model does seem to reach a point at which increased resolution does not improve the performance, however - this is likely due to other peripheral factors such as memory access speed and latency coming into play.
Chapter 5 - Model Cases (2D)

In this chapter, the two-dimensional DC plasma arc model will be applied to a number of problems with parameters related to DC arc furnaces at small to medium pilot plant scale - in two dimensions, the focus is on systems with a single electrode and arc. The modelling work will begin with a detailed study of the results generated using a base set of parameters, identifying interesting phenomena relating to the transient behaviour. Some comparison will also be performed between the model predictions, established empirical relationships, and photographic experiments. Variations of selected parameters of the base case model will then be examined to develop an understanding of the range of possible phenomena the model system may exhibit.

5.1 Description of two-dimensional models and cases studied

Geometry and spatial layout

For all two-dimensional modelling of the DC plasma arc, the geometry used describes a rectangular region of space between the tip of the graphite electrode and the surface of the molten bath in the furnace, which serves as the anode. This is depicted in Figure 59.

*Figure 59: Model geometry and boundaries for 2D cases*

```
A-----------------------------F
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
|                             |
In order to completely specify a particular model case, several pieces of information must be provided. Firstly, the dimensions of all the boundaries in Figure 59 must be given. In most cases the desired value is simply specified directly, however, the dimensions of the cathode spot are determined by assuming a circular spot and a constant current density - together with a given current, these enable the calculation of a representative value for $x_{CD}$. The location of the centreline of the cathode spot and arc root must also be specified, to locate the boundary regions BE and CD along the upper surface - for all the 2D models, unless otherwise stated, they are assumed to be located along the centreline of the model region.

Secondly, values for the constant physical properties of the plasma gas ($\mu, \rho, \kappa, C_p$), must be provided. These are specified directly, either by using average values for the plasma gas in use over a range of relevant temperatures, or regular multiples of these values for the purpose of sensitivity analysis.

Temperature-dependent values of $\sigma$ and $Q_\kappa$ are also required. These are specified by providing the model with a discrete set of values at constant 500K intervals, between 1000K and 33000K. The values are obtained from literature\textsuperscript{43,45} dealing with physical properties and thermal radiation behaviour of plasmas, and are again specific to the type of plasma gas and thermal radiation model (optically thin or thick) considered.

Next, the electrical parameters of the arc must be specified. This is given by the value of $j_e$ on the cathode spot boundary CD, and the total current provided to the furnace. These in turn define the width of the cathode spot as mentioned above.

Finally, some parameters must be given to define the numerical behaviour. The grid dimensions $I$ and $J$ must be given. Using these and the dimensions of the region, the value of $\delta x = \delta y = \delta l$ can be calculated. The number of threads to be used, and the maximum model time to calculate until, then complete the data needed by the DC plasma arc model.

**Initial conditions**

The initial conditions supplied to the model are constant-temperature impulsive start conditions.
The velocity (or equivalent field) is set to zero at the start of the calculation, allowing the full evolution of the arc jet to be observed. The initial temperature field is set to a moderately high constant value, 10000K, to provide a conductive path for the electric current during the early stages of arc establishment.

**Boundary conditions**

The spatial boundary conditions applied to the various components of the model are described in Table 6. The intermediate variables such as the gauge field for fluid flow and the magnetic vector potential for the solution of the electromagnetic fields are supplied with additional or related boundary conditions, as described in Section 2.

The nature of and values for the thermal boundary conditions at various parts of the model must also be given. Depending on whether TBC 1 or TBC 2 is in use, temperatures at the various parts of the model region boundary (anode, walls, electrode surface) may be required, as described earlier in Section 2.2.

**Table 6: Boundary conditions for 2D plasma arc model**

<table>
<thead>
<tr>
<th>Variable</th>
<th>CD</th>
<th>BC &amp; DE</th>
<th>AH &amp; FG</th>
<th>AB &amp; EF</th>
<th>GH</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v_x, v_y$</td>
<td>$v_x = v_y = 0$</td>
<td>$v_x = v_y = 0$</td>
<td>$v_x = v_y = 0$</td>
<td>$v_x = v_y = 0$</td>
<td>$v_x = v_y = 0$</td>
</tr>
<tr>
<td>$T$ (TBC 1)</td>
<td>$T = T_{electrode}$</td>
<td>$T = T_{electrode}$</td>
<td>$T = T_{wall}$</td>
<td>$T = T_{wall}$</td>
<td>$T = T_{anode}$</td>
</tr>
<tr>
<td>$T$ (TBC 2)</td>
<td>$\frac{\partial T}{\partial y} = 0$</td>
<td>$\frac{\partial T}{\partial y} = 0$</td>
<td>$\frac{\partial T}{\partial x} = 0$</td>
<td>$\frac{\partial T}{\partial y} = 0$</td>
<td>$\frac{\partial T}{\partial x} = 0$</td>
</tr>
<tr>
<td>$\phi$</td>
<td>$\frac{\partial \phi}{\partial y} = -\frac{j_k}{\sigma}$</td>
<td>$\frac{\partial \phi}{\partial y} = 0$</td>
<td>$\frac{\partial \phi}{\partial x} = 0$</td>
<td>$\frac{\partial \phi}{\partial y} = 0$</td>
<td>$\phi = 0$</td>
</tr>
</tbody>
</table>

The boundary conditions chosen for the flow model represent no-slip surfaces. This is considered to be reasonable for a DC arc furnace, which contains the electrode and plasma arc within refractory or water-cooled metal walls. This is also the reasoning for the constant temperature boundary conditions in TBC 1. On the anode and cathode spot surfaces, the temperature is assumed to be fixed by phase changes of the anode and cathode materials, whilst on the walls it will be governed by the nature and degree of cooling applied to the furnace vessel. The boundary conditions for $\phi$ result from the fact that electric current can only enter or leave the vessel via the cathode spot.
(which operates at a fixed current density \(j_k\)), or the anode (which is assumed to be a conductive material and fixed at ground potential).

**Base reference case**

The two-dimensional modelling exercise begins with the examination of a base set of parameters, chosen to represent reasonable conditions for a DC plasma arc at moderate sized pilot plant furnace scale. These are shown in Table 7. Variations of the model parameters independent of the others are then studied, and the effects documented.

*Table 7: Specification for base case 2D model*

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Region height, (y_{all})</td>
<td>0.05 m</td>
</tr>
<tr>
<td>Region width, (x_{GH})</td>
<td>0.2 m</td>
</tr>
<tr>
<td>Electrode diameter, (x_{BE})</td>
<td>0.05 m</td>
</tr>
<tr>
<td>Plasma gas</td>
<td>Air</td>
</tr>
<tr>
<td>Thermal radiation model</td>
<td>Optically thin</td>
</tr>
<tr>
<td>Viscosity, (\mu)</td>
<td>(1.307 \times 10^{-4}) Pa.s</td>
</tr>
<tr>
<td>Density, (\rho)</td>
<td>0.02593 kg/m³</td>
</tr>
<tr>
<td>Thermal conductivity, (\kappa)</td>
<td>3.067 W/mK</td>
</tr>
<tr>
<td>Heat capacity, (C_p)</td>
<td>9420 J/kgK</td>
</tr>
<tr>
<td>Cathode spot current density, (j_k)</td>
<td>(3.5 \times 10^7) A/m²</td>
</tr>
<tr>
<td>Current</td>
<td>500 A</td>
</tr>
<tr>
<td>Temperature boundary condition</td>
<td>TBC 1</td>
</tr>
<tr>
<td>Wall temperature, (T_{wall})</td>
<td>2000 K</td>
</tr>
<tr>
<td>Anode temperature, (T_{anode})</td>
<td>3000 K</td>
</tr>
<tr>
<td>Electrode temperature, (T_{electrode})</td>
<td>4100 K</td>
</tr>
<tr>
<td>Grid resolution, (I \times J)</td>
<td>1024 x 256</td>
</tr>
<tr>
<td>Number of threads</td>
<td>4</td>
</tr>
<tr>
<td>Maximum model time</td>
<td>10 ms</td>
</tr>
</tbody>
</table>
A comment on symmetry

It is important to note that due to the left-right symmetry of the numerical model, it is possible for perfectly left-right symmetrical solutions to develop – these are physically unrealistic as they are highly sensitive to any minor perturbation away from symmetry. They are avoided by introducing a small degree of asymmetry via the boundary conditions, by biasing the location of the cathode spot in the model (that is, the spot is not exactly centered in the region). The bias very small, of the order of 0.5% of the size of the region, but is sufficient to perturb the evolution of the model away from the development of unrealistic solutions.

Temporal data sampling

The locations of the points at which temporal data is sampled from the calculation during a model run are the same for all two-dimensional models, and are positioned vertically along the centreline of the region between the graphite electrode and the anode surface.

5.2 Base case model

The Base Case model was run on an Intel Core2 Q6600 machine with four CPUs and 2GB RAM. The model required 35961 time step calculations at an average time step size of 0.28 μs, and took approximately 2.5 hours to complete.

As an indicator of the intensity of the various physical processes in the DC plasma arc model, some indicative values of the peak velocity, temperature, and electric potential are calculated from the data produced. A time average of these numbers is obtained from the last half of the simulation (5 - 10 ms), during which the arc motion is generally well established and indicative of long-term behaviour. Instantaneous values are also presented for the model at the end of the simulation time (10 ms).

Average and final peak values

For the base case model as described in the previous section, the values are presented in Table 8.
Table 8: Peak values of variables for base case model

<table>
<thead>
<tr>
<th>Variable</th>
<th>Peak value - 5 - 10ms average</th>
<th>Peak value - at 10ms</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v_x, v_y$</td>
<td>779.4 m/s</td>
<td>744.8 m/s</td>
</tr>
<tr>
<td>$Re$</td>
<td>7731</td>
<td>7390</td>
</tr>
<tr>
<td>$T$</td>
<td>15752 K</td>
<td>15491 K</td>
</tr>
<tr>
<td>Arc voltage</td>
<td>236.9 V</td>
<td>236.7 V</td>
</tr>
</tbody>
</table>

Arc voltage is defined as the maximum value of the $\phi$ field at any time. The peak velocity is seen to be below the critical limit for compressible flow considerations, and the Reynolds number (computed using the height of the domain as the characteristic length) indicates that the flow should be fully resolved at the grid resolution used.

Flow behaviour

The evolution of the arc jet flow field may be represented by the vorticity. Its behaviour in time is shown in Figures 60 - 63.

Immediately obvious is the large degree of turbulent, dynamic motion occurring. Oscillations in the vicinity of the cathode spot cause the breakdown of the stable arc column into strong, alternating vortices early on in the simulation (time $< 1$ ms). The oscillating vortex production near the cathode spot causes the arc column to take on very twisted or sinusoidal shapes.

*Figures 60 - 63: Vorticity profile for base case, scale range -150000:150000 s⁻¹*
The early evolution of the vorticity field shows the onset of the instability quite clearly, in Figures 64 - 67.

*Figures 64 - 67: Vorticity profile during onset of instability, scale range -150000:150000 s⁻¹*

**Temperature behaviour**

Temperature is an important variable, as it most closely defines the visible shape and structure of the arc. It also determines the electrically-conductive regions and hence couples tightly to the electromagnetic fields. The temperature field for the base case simulation is shown in Figures 68 - 71 at various points in time.

*Figures 68 - 71: Temperature profiles, scale range 2000:15000K*

The turbulent dynamics of the flow field produces analogous behaviour in the temperature field, due to the convective coupling between the two. The temperature distribution is seen to concentrate
near the centre of the furnace, between the cathode spot and the anode. This hot recirculation region maintains the electrical connection between the electrode and the anode surface, and sustains the arc column.

It is interesting to observe that small “parasitic” arcs form on the anode beneath the main arc jet emanating from the cathode spot. These are emergent phenomena, as the boundary conditions on the anode surface are isotropic and do nothing to encourage their formation. As can be seen in the temperature animations on the accompanying disc, the parasitic arcs are highly mobile and generally short-lived, although they can occasionally interact with one another and combine to form stronger anode arc jets.

**Electromagnetic fields**

The evolution of the electric potential field $\phi$ is shown in Figures 72 – 75. In general, there are significant voltage drops near both the cathode spot and anode surface. Within the column of the arc away from the boundaries, the voltages change in a smoother and more linear fashion.

It is also of interest to note that there is significant variation of the electric potential across the surface of the electrode near to the cathode spot. The primary cause of this variation is the choice of boundary condition for the electrode surface, namely Neumann (constant current density) as opposed to Dirichlet (constant voltage), and the perhaps somewhat artificial assumption that current only leaves the electrode surface via the cathode spot. Such variations could in practice arise from electrostatic phenomena such as charge build-up or plasma sheath effects.

*Figures 72 - 75: Electric potential, scale range 0:250V*
The current density is a function of both the electric potential and the temperature-dependent electrical conductivity. The distribution of the magnitude of the current density vector is shown in the series of Figures 76 - 79.

*Figures 76 - 79: Magnitude of current density, scale range \(0.35 \times 10^7 \text{ A/m}^2\)*

As current is closely linked with temperature via the electrical conductivity, the current density is concentrated near the centre of the furnace, and contained within the arc column. The parasitic arcs are clearly visible as intense regions of current density near the anode surface.

The current density governs the distribution of the magnetic field, which in turn feeds into the fluid flow equations via the Lorentz force source term. The magnetic field is shown at various times in Figures 80 - 83.

*Figures 80 - 83: Magnetic field, scale range -0.05 - 0.05 T*
By contrast with many of the other variables, the magnetic field does not display enormous variations as it evolves in time. This is primarily due to the concentration of the field strength near to the cathode spot (and to a lesser degree near the roots of the parasitic arcs).

**Spatial data**

In order to generate the plots shown below, the spatial field data was sampled along a straight line between the centre of the cathode spot and the centre point of the anode surface, giving a cross-section of the high-intensity arc region at various times during the simulation. The results are shown in Figure 84.

The behaviour of \( v_y \) along this line is quite erratic, and at later times even changes sign for a significant part of the flow. This results from the turbulent mixing and recirculation that occurs in the arc region.

*Figure 84: y-component of velocity at \( x = 0.1m \)  
Figure 85: Temperature at \( x = 0.1m, \text{ time} = 10ms \)*

The temperature profile is shown in Figure 85, and is reasonably constant over the length of the arc. Near to the cathode spot and the anode the concentration of current due to the primary or parasitic arcs generally causes peaks due to localised Ohmic heating, clearly visible in the 10ms temperature profile shown above and in the evolving temperature profiles near both surfaces shown in Figures 86 and 87.
The electric potential is representative of the voltage in the arc, and shows surprisingly little large scale variation over the course of the simulation once the arc column is established. As shown in Figure 88, the dependence of $\phi$ on $y$ is seen to be very linear over much of the range, with only some deviations occurring near the anode and cathode spot regions.

**Temporal data**

One of the DC plasma arc model's key aims is to understand some of the dynamic phenomena the system exhibits. A useful way to examine the dynamics is by observing how field variables change at certain locations in space over time.
In Figure 89, the evolution of the temperature at a location a few millimetres below the centre of the cathode spot is plotted. It can be seen that an initial period of smooth, stable behaviour is replaced by very rapidly changing dynamics. The temperature at this location oscillates chaotically and exhibits very large changes in value between 6000K and 20000K.

![Figure 89: Time series for temperature at x = 0.1m, y = 0.045m](image)

The initial evolution of the temperature at several locations near the centre of the region is shown in Figure 90.

![Figure 90: Time series for temperature at x = 0.1m](image)

Several phenomena may be observed here. Firstly, there is an initial drop in temperature in areas further from the cathode spot, as the plasma cools by emitting radiation before the arc jet reaches the lower portion of the region. Temperature then rises as the jet is established and reaches an early
equilibrium. This stability persists until 0.6 ms, at which point oscillatory behaviour near to the cathode spot begins - this effect takes some time to propagate down the arc column, eventually causing similar behaviour at the lower two sample locations. However, the strength of the oscillation is significantly greater, and the variations in temperature more extreme, close to the cathode spot than in the lower regions of the furnace.

The rapid variation in temperature causes similar behaviour in the electromagnetic field variables. Of particular interest is the variation in the electric potential, as the peak values of this field determine the voltage generated by the arc – when multiplied by the current, this gives the total power generated in the arc column.

The peak value of $\phi$ is estimated by tracking the value at the centre of the cathode spot. The results are shown in Figure 91.

![Figure 91: Time series for arc voltage, $x = 0.1m, y = 0.05m$](image)

The absolute value of the voltage generated by the model is quite high. The qualitative behaviour is quite interesting, as the violent oscillations of the temperature field appear to be somewhat damped by the smoothing nature of the electric potential calculation. Fluctuations are still present, and spikes can push the voltage up by up to 25% of its value in a very short time frame.

**Time-averaged field data**

Much additional study is possible when considering further interpretation of the data output by the
model. More detailed statistical and signal analysis of the local and peak field variables' behaviour can give some idea of the typical values and their variation that may be of more use in a process engineering context. By way of example of such analyses, the time-averaged current density and temperature fields based on the model data over the last 9 ms of simulation time are presented in Figures 92 and 93.

*Figures 92 – 93: Time-averaged field data over last 9ms*

As can be seen, the time-averaged fields have considerably more symmetry than the instantaneous behaviour. This lends some credence to the concept that while the arc may be undergoing extremely irregular and asymmetrical behaviour at any given moment, at the much longer time scales more typical of human observation the arc column may still appear to be a quite well defined and localised phenomenon.

**Comparison of arc behaviour with empirical correlations**

As discussed earlier, many empirical correlations have been developed during the study of DC plasma arcs. Although these correlations generally refer to steady state arcs, it is nonetheless instructive to consider the values and qualitative behaviour that they produce.

**Velocities**

Bowman\(^{11}\) provides an estimate of a DC plasma arc's velocity along its axis, ascribed to Szekely and Chang.

\[
\dot{v} = \frac{663 I_a}{40z} \tag{220}
\]

\(I_a\) is in kA, and \(z\) (distance from cathode spot) is in m. For the base case considered here, this
formula gives a value of 332 m/s at the midpoint of the region \((z = y = 0.025 \text{ m})\). Although the peak velocities in the base case model can be considerably higher than this, comparing the values of \(v_y\) along the centreline as plotted in Figure 84 shows that the velocities here are indeed of similar order, 200-300 m/s.

**Temperatures**

Bowman\(^{11}\) performs a simplified energy balance on the centreline of an axisymmetric arc, relating radiation losses to Ohmic heating given empirical models of the electrical behaviour of the arc column. This treatment results in expected peak temperatures in the range 12000-18000K for a wide range of arc parameters, values which are extensively supported by other modelling work.

As can be seen from Figures 85 and 87, the temperature values obtained in this model are similar, giving peak temperatures near the cathode spot of between 12000 and 15000K over the duration of the simulation.

**Electric potential**

The behaviour of the electric potential in a DC plasma arc column was considered by Bowman\(^{11}\), building on work studying empirical arc shape models derived from photographic data. Reynolds and Jones\(^{12}\) presented the empirical equation governing the arc voltage that results from Bowman's analysis.

For the base case of a 500A, 5cm arc, the Bowman electrical model predicts a total arc voltage of only 51 V, compared to the dynamic model's prediction of 150 - 200V (see Figure 91). Two issues contribute to this discrepancy - firstly, the Bowman model is designed using an empirical approximation to the shape and structure of a laminar flow, axisymmetric arc, and does not take into account the possibility of the “effective” length of the arc being much greater due to unsteady oscillations as observed in the dynamic model. Secondly, the value used for the electrical conductivity in the Bowman model is based on furnace process experiments - in these processes, carbon monoxide along with a mixture of metal and oxide vapours forms the plasma gas, and this may result in considerably different properties to those of the air arcs considered here.

It is however instructive to consider the normalised dependence of voltage on arc length (here taken
to mean "distance from the cathode spot surface") and its characteristics. The Bowman model displays an initial non-linear change, and tends toward linear behaviour as arc length increases. Comparing this with the electric potential data from the base case model at time = 10ms shows somewhat similar behaviour (Figure 94).

Figure 94: Comparison of normalised voltage - arc length relationships

As with the Bowman model, the base case model changes in a non-linear fashion for arc lengths less than 0.015 m, and then settles to a linear rate of change in the main body of the arc. The base case model also captures the drop in potential near to the anode, which is a subtle real phenomenon that the Bowman model does not reproduce.

Comparison of arc behaviour with existing modelling results

Unfortunately, the majority of published results relating to the modelling of DC plasma arcs focus on large industrial-scale systems running at very high currents - typically 10 kA or higher. Scale differences would render any comparisons between such work and the present results moot.

An exception to this is the early work of Ushio et al21, which considered arcs at more moderate current scales. Although their work was performed at 2.16 kA, still well above the 500A used in these tests, the results are of some interest. Peak velocities in their (steady state, axisymmetric) simulation were of the order of 1500 m/s, or roughly double the values shown in Table 8; velocity is expected to increase with increasing current, and the results do support this trend.
The variation of temperature along the region centreline (Figure 85) is also matched qualitatively and quantitatively by the Ushio model results - these show peak temperatures of 15000 K close to the cathode spot, dropping rapidly to approximately 10000 - 12000 K and then remaining roughly constant in the main body of the arc.

**Comparison of arc behaviour with photographic evidence**

The most important aspect of the DC plasma arc model is its ability to study transient and asymmetrical behaviour in the arc system. Photography is a useful tool to compare (qualitatively) model and reality.

Photography of DC plasma arcs as an experimental technique is very challenging for a number of reasons. The environment is generally a harsh one for cameras and electronics, with very high temperatures and electromagnetic fields near to the arc. Many of the phenomena of interest also occur at very short time scales, making their photographic capture difficult. Additionally, arcs of any significant size emit an enormous quantity of visible light, as well as ultraviolet and shorter wavelength radiation - this can overpower or otherwise interfere with film and digital photographic processes.

As a result of these difficulties, photographic evidence of arc behaviour is very limited. Jordan et al\textsuperscript{19} conducted some early studies using high speed film cameras. Zweben and Karasik\textsuperscript{17} photographed small (150-250A) arcs in a laboratory setting using high speed digital cameras. Jones et al\textsuperscript{20} presented photographs of larger arcs (up to 5kA) under a variety of conditions, including open air.

Photography generally captures visible light radiation from the arc, and the image obtained is closely related to the temperature profile of the plasma gas\textsuperscript{11}.

Oscillation and "hosepiping" of the arc jet is clearly visible in the photography work of Zweben and Karasik, shown in Figures 95 and 96. The tail of the arc at the anode surface is seen to move a considerable distance over this surface as the arc bends and twists - compare this with Figures 68 through 71, and the animation of temperature for the base case and other cases on the accompanying disk. The two-dimensional simulation, although conducted at a higher current and missing one dimension as compared to the experiment, captures the oscillatory nature of the jet.
motion quite well.

Figure 95: Photograph of dynamic arc behaviour\textsuperscript{44}, current \textasciitilde150A, electrode height 3.1 cm

Figure 96: Photograph of dynamic arc behaviour\textsuperscript{44}, current \textasciitilde250A, electrode height 2.7 cm
Also visible in the photographs is kinking of the arc, a phenomenon in which the arc jet direction changes abruptly at certain points along its length. This effect is also observed in the model; compare particularly Figure 70.

As current increases the unstable dynamic behaviour of the arc becomes even more extreme, as shown in Figure 96. In these photographs, large, sweeping movements of the arc column are observed. In addition, the kinking phenomenon is exaggerated at higher currents, suggesting that it may be a form of magnetic instability. Again, some similarity to the model results is seen comparing particularly Figures 70 and 71 to the centre photograph in Figure 96, and the later portions of the base case temperature animation on the accompanying disk.

The photographic investigation performed by the author with Jones et al\textsuperscript{20} was carried out at higher currents, using a pilot-scale DC arc furnace power supply. The arc was exposed to air and struck between a graphite electrode and a solid flat plate connected to the furnace anode. The arcs were photographed using standard digital still cameras equipped with welding filters to reduce the intensity of the light emitted.

*Figure 97: Temporarily stable arc; current ~ 2kA, electrode height ~ 15cm*

Figure 97 shows an axisymmetric arc of the sort studied by the majority of empirical and numerical models to date. The arc in this form, especially at higher currents, is not stable - this is analogous to the early stages of the base case (and other) model results, during which a stable symmetric arc jet is established but rapidly breaks down into unstable motion. This initial stability and subsequent breakdown can be easily seen in the early parts of the visualisation animations for the base case available on the accompanying disk.
The turbulent fluid motion in the body of the arc is clearly visible in Figure 98, and the arc structure rapidly becomes disordered and chaotic. There are many qualitative similarities to the transition to turbulent motion observed in the two-dimensional DC plasma arc model.

*Figure 98: Transition to unstable arc motion, current ~2kA, electrode height ~15cm*

Small, high-temperature spots and jets on the anode are observed in a number of the high-current arc photographs, as shown in Figure 99. The arrows indicate the locations of possible secondary parasitic arcs at the surface of the anode, a phenomenon frequently seen in the 2D models.

*Figure 99: Evidence of parasitic anode arc formation, ~2kA, electrode height ~15cm*

### 5.3 Region geometry and arc length effects

This set of results examines the impact of changing the geometry of the model region, specifically
the distance from the graphite electrode to the anode surface, with the aim of identifying any variation in the dynamic behaviour as the electrode is moved closer to or further from the anode surface.

The model parameters are identical to those used for the base case as shown in Table 9, with the exception of the region height \( y_{AL} \). The resolution of the models was also changed accordingly to preserve \( \delta x = \delta y = \delta l \). The models were run on an Intel Core2 Q6600 machine with four CPUs and 2GB RAM, and required between 34901 and 46666 time step calculations at an average time step size of between 0.21 and 0.29 \( \mu \)s. Each model run took between 0.8 and 5.2 hours to complete.

<table>
<thead>
<tr>
<th>Region height, ( y_{AL} )</th>
<th>Grid resolution, ( I \times J )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0125 m</td>
<td>1024 x 64</td>
</tr>
<tr>
<td>0.01875 m</td>
<td>1024 x 96</td>
</tr>
<tr>
<td>0.025 m</td>
<td>1024 x 128</td>
</tr>
<tr>
<td>0.0375 m</td>
<td>1024 x 192</td>
</tr>
<tr>
<td>0.05 m</td>
<td>1024 x 256</td>
</tr>
<tr>
<td>0.075 m</td>
<td>1024 x 384</td>
</tr>
<tr>
<td>0.1 m</td>
<td>1024 x 512</td>
</tr>
</tbody>
</table>

Average and instantaneous peak values

As for the base case, time-averaged and instantaneous peak values of velocity, temperature, and arc voltage are calculated. The values are compared over the range of region heights. The results are shown in Figures 100 - 103.

At lower values of \( y_{AL} \), the average peak values match the instantaneous peak values reasonably well, suggesting that the dynamic behaviour of the arc becomes less intense at shorter arc lengths.

The magnitude of the velocities appears to experience a maximum between 0.01 and 0.03m. Interestingly this maximum appears to correspond with a minimum in the peak temperatures.
The behaviour of the arc voltage (maximum value of \( \phi \)) as \( y_{AH} \) is changed is again seen to be largely linear, corroborating the evidence from the base case and empirical models that the bulk of the arc column away from the anode surface and cathode spot experiences linear change of \( \phi \) with length. There is some deviation from linearity at larger values, where it is possible that geometric effects due to the reduced aspect ratio of the model domain may come into play - as the aspect ratio decreases, the walls of the domain are likely to exert a stronger influence on the field variables in the vicinity of the arc.

**Figure 100: Graph of peak velocity magnitude as a function of region height**

**Figure 101: Graph of peak temperature as a function of region height**

**Figure 102: Graph of Reynolds number as a function of region height**

**Figure 103: Graph of arc voltage as a function of region height**

**Comparison of spatial data**

The temperature profiles produced by the models at various values of \( y_{AH} \) are shown in Figures
As the region height increases, the size of the high temperature recirculation zone in the immediate vicinity of the arc column increases proportionally. Higher temperatures begin to occur at greater and greater distances from the cathode spot, until by 0.075m and above it is clear that the walls are actively containing the temperatures around the arc column.

At the same time, the degree of turbulent motion increases (unsurprisingly, as the Reynolds number of the flow increases as \( y_{th} \) is changed), from a relatively smooth temperature profile at 0.0125m, to increasingly intense mixing behaviour above 0.025m.

Figures 104 - 110: Temperature plots for various \( y_{th} \) time = 10ms, scale range 2000:15000K

The differences are most strikingly visible in the animations of the temperature profiles on the accompanying disk, in the directory animations\arcmodels2d\arclength. Note particularly the very
regular, almost stable oscillations that occur at 0.0125m, compared to the very unstable turbulent dynamics of the models at larger $y_{eff}$ values.

The temperature profile along the centreline of the region for each region height is shown below in Figure 111.

*Figure 111: Temperature profiles for different values of $y_{eff}$ at $x = 0.1m$, time = 10ms*

It can be seen from this graph that in the vicinity of the arc column at the centre of the region, the average temperature rises as the height of the region decreases - the arc is "compressed" between the graphite electrode and the anode surface and becomes more localised and less diffuse as a result.

**Temporal behaviour**

As can be seen in the animations of the temperature profiles, changing the value of $y_{eff}$ has a definite impact on the dynamic behaviour of the model. To examine this further, we plot the evolution of the arc voltage (maximum value of $\phi$) as a function of time in Figure 112.

It is clear that as the region height increases and the fluid flow in the arc column becomes increasingly turbulent, both the absolute value of the arc voltage and the strength of its oscillations become larger. There is a distinct transition from regular oscillations of the arc at low values of $y_{eff}$, to more chaotic behaviour at higher values. This transition is more obvious if we plot only a subset of Figure 112, as shown in Figure 113.
Regular oscillatory behaviour at 0.0125\,m is gradually replaced by increasingly unpredictable behaviour as $y_{\text{alt}}$ rises, an indirect result of the increase in turbulence of the flow field.

At low values of region height, the period of the oscillation behaviour is also seen to increase as $y_{\text{alt}}$ rises, reflecting the fact that the arc column and the recirculation zones around it increase in size as the graphite electrode is raised further away from the anode.

5.4 Arc current effects

This set of results aims to identify any changes in quantitative and qualitative behaviour of the DC plasma arc model as the current supplied to the arc is increased. The current parameter indirectly affects the geometry of the model region, as the cathode spot width ($y_{\text{CD}}$ in Figure 59) is determined from the current and the current density specified.

The parameters used in the model are identical to those used for the base case (Table 7) with the exception of the specified current, which is changed over the range of values shown in Table 10. The effect the current has on the values of $y_{\text{CD}}$ is also shown. The models were run on an Intel Core2 Q6600 machine with four CPUs and 2GB RAM, and required between 13340 and 54233 time step calculations at an average time step size of between 0.18 and 0.75\,\mu s. Each model run took between 0.9 and 3.8 hours to complete.
Table 10: Range of arc currents tested

<table>
<thead>
<tr>
<th>Current</th>
<th>( y_{cd} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 A</td>
<td>0.00191 m</td>
</tr>
<tr>
<td>200 A</td>
<td>0.00270 m</td>
</tr>
<tr>
<td>300 A</td>
<td>0.00330 m</td>
</tr>
<tr>
<td>400 A</td>
<td>0.00381 m</td>
</tr>
<tr>
<td>500 A</td>
<td>0.00426 m</td>
</tr>
<tr>
<td>600 A</td>
<td>0.00467 m</td>
</tr>
<tr>
<td>700 A</td>
<td>0.00505 m</td>
</tr>
<tr>
<td>800 A</td>
<td>0.00539 m</td>
</tr>
<tr>
<td>900 A</td>
<td>0.00572 m</td>
</tr>
<tr>
<td>1000 A</td>
<td>0.00603 m</td>
</tr>
</tbody>
</table>

**Average and instantaneous peak values**

Time-averaged and instantaneous maximum values of the field variables in the model are calculated, the former between 5 and 10ms, and the latter at the conclusion of the model run, at 10ms. These are then compared against the specified current value. The results are shown in Figures 114 - 116.

The peak velocity of the arc jet displays a consistently increasing trend as the arc current is raised. The differences between the instantaneous values and the average values also appear to increase, suggesting that the fluid flow grows more variable with larger currents. The Reynolds number of the flow is proportional to the velocity for this set of model parameters, and varies between 2000 at 100A and 15000 at 1000A.

It should be noted that at arc current values of ~800A and higher, the magnitude of the peak velocities begins to approach and exceed the compressible flow limit of 0.3 Ma. The accuracy of the DC plasma arc model (which uses an incompressible fluid flow model) may suffer to some degree in these cases.
Trends are less obvious for the temperature and arc voltage variables. The time-averaged value of peak temperature remains largely constant over the range of currents tested, with a slight increase over the range 200 - 800A. The instantaneous peak temperature values are much more erratic and deviate from the average values considerably - this again suggests increasing instability of the flow at larger currents.

Similar effects are observed for the arc voltage. The time-averaged values suggest a slow increase in voltage over the range 100 - 700A, although at larger currents there is some anomalous behaviour - in any case, the relationship between the model arc's voltage and its current is clearly a non-Ohmic one. The instantaneous values at 10ms are more noisy, although they do support the increasing trend at lower currents. At high current values the deviation between the average and instantaneous values changes is larger, as was observed for the other variables.
Comparison of spatial data

Plots of the temperature profile for various arc currents tested are shown in Figures 117 - 121.

Evidence of the increasing strength and turbulence of the flow field is visible in the temperature profiles. The degree of mixing and turbulent transport of the temperature variable is clear in the images, as the current is increased and results in higher velocities in the arc jet. The volume of space occupied by the high temperature core of the jet also visibly increases as the arc current rises.

*Figures 117 - 121: Temperature plots for various currents, time = 10ms, scale range 2000:15000K*

The difference in flow characteristics is more obviously demonstrated with a plot of vorticity, which is related to local shear rates and fluid rotational speed. Vorticity plots for 100A and 1000A are shown in Figures 122 and 123.

At higher currents the vorticity of the fluid flow is greatly elevated, corresponding to more violent motions of the fluid. This occurs as the increased current produces larger magnetic fields, and hence forces, on the arc column.
Figures 122 - 123: Vorticity plots, time = 10ms, scale range -150000:150000 s⁻¹

The increasing velocity and turbulence of the arc jet flow are also displayed prominently in the animations of temperature and vorticity for the various cases tested. These are available on the accompanying disk, in the directory animations\arcmodels2d\current.

The values of temperature along the region centreline, between the anode surface and the cathode spot, are shown in Figures 124 and 125.

The temperature profiles through the central arc region are fairly consistent across a wide range of arc currents, showing an average arc-region temperature of 8000 - 10000K in the bulk plasma gas away from the boundaries.

It is interesting to note that at lower currents, much of the variation and oscillation of the temperature field occurs nearer to the cathode spot, whereas for larger currents, most of the variation occurs in the lower half of the region, nearer to the anode surface.
**Temporal behaviour**

The evolution of $v_y$ close to the cathode spot for the various models displays a number of interesting phenomena. The early part of this development is shown in Figure 126.

*Figure 126: Evolution of $v_y$ velocity profiles for $t < 1\text{ms}, x = 0.1\text{m}, y = 0.045\text{m}*  

In the initial stable phase of arc development, the scaling of velocity with current is very well defined - increased current produces a faster arc jet.

As time proceeds and more typical unsteady motion begins, considerable variability of the velocity begins to occur. The breakdown of the stable fluid flow and onset of the unstable behaviour is seen to occur earlier for larger currents, which is supported by the observed increase in Reynolds number and flow turbulence with current seen in the model results.

The degree of variability of the fields in the models at different currents is illustrated in Figure 127, which shows the behaviour of the arc voltage (maximum value of $\phi$) with time near the end of the simulation.

Clearly as the current is increased, the unstable behaviour of the arc voltage becomes more exaggerated. The amount of variance between successive peaks increases, and the average period of the oscillations decreases, indicating larger and more rapid changes occurring in the flow and temperature fields of the model.
5.5 Cathode spot current density effects

The aim of this set of model variations is to examine the effect of changing the cathode spot current density used in the DC plasma arc model.

This is a more academic exercise than the previous two parameters studied, as the current density is generally a property of the material used for the electrode (in the standard case, graphite) and is not easily changeable for a given furnace design and metallurgical process. However, there is a great deal of uncertainty in the literature as to what an appropriate value of \( j_k \) is - numbers from 2 up to 5 kA/cm\(^2\) have been proposed or calculated in earlier modelling studies, and it is thus prudent to examine the impact this will have on the transient dynamics of the current model.

### Table 11: Range of current densities tested

<table>
<thead>
<tr>
<th>( j_k )</th>
<th>( y_{CD \text{ at } 500A} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( 1 \times 10^7 \text{ A/m}^2 )</td>
<td>0.00798 m</td>
</tr>
<tr>
<td>( 2 \times 10^7 \text{ A/m}^2 )</td>
<td>0.00564 m</td>
</tr>
<tr>
<td>( 3 \times 10^7 \text{ A/m}^2 )</td>
<td>0.00461 m</td>
</tr>
<tr>
<td>( 4 \times 10^7 \text{ A/m}^2 )</td>
<td>0.00399 m</td>
</tr>
<tr>
<td>( 5 \times 10^7 \text{ A/m}^2 )</td>
<td>0.00357 m</td>
</tr>
<tr>
<td>( 6 \times 10^7 \text{ A/m}^2 )</td>
<td>0.00326 m</td>
</tr>
</tbody>
</table>
The parameters used in the model are identical to those used for the base case (Table 7) with the exception of the specified cathode spot current density, which is changed over the range of values shown in Table 11. Changing \( j_s \) affects both the geometry of the model - the cathode spot width \( y_{CD} \) depends on it - and the boundary conditions for \( \phi \). The models were run on an Intel Core2 Q6600 machine with four CPUs and 2GB RAM, and required between 21704 and 49694 time step calculations at an average time step size of between 0.20 and 0.48 \( \mu s \). Each model run took between 1.5 and 3.5 hours to complete.

**Average and instantaneous peak values**

Time-averaged and instantaneous peak values of the field variables in the model are calculated, the former between 5 and 10ms, and the latter at the conclusion of the model run, at 10ms. These are then compared against each other as functions of the specified current density. The results are given in Figures 128 – 130.

As the current density is increased, there is a corresponding rise in both average and instantaneous values of peak velocity magnitude. The Reynolds numbers are directly proportional to the velocities for this set of parameters, and vary between 5000 and 13000.

*Figure 128: Graph of peak velocities as a function of current density*

*Figure 129: Graph of arc voltage as a function of current density*

Peak temperature of the arc shows a marked increase as \( j_s \) is raised. This is to some degree expected, as the higher current density in the vicinity of the cathode spot leads to a larger Ohmic heating source term in the energy conservation equation, and the peak temperatures occurring in the
root of the arc will be most directly affected by the change.

Figure 130: Graph of peak temperatures as a function of current density

The total voltage generated by the arc appears to be largely unaffected by the changes in current density. Increased deviations between the instantaneous and time-averaged values at larger values of \( j_a \) suggest that the arc voltage (and presumably other field variables) exhibits more transient behaviour for higher current densities.

Comparison of spatial data

Plots of the temperature profile for various arc currents tested are shown in Figures 131 - 136.

As the current density is increased, there is a general increase in the turbulent mixing occurring in the vicinity of the arc jet. The jet itself also appears to grow hotter and more compact as the value of \( j_a \) rises.
Figures 131 - 136: Temperature plots for various $j_\alpha$, time = 10ms, scale range 2000:15000K

![Fig. 131: $j_\alpha = 1 \text{ kA/cm}^2$](image1)

![Fig. 132: $j_\alpha = 2 \text{ kA/cm}^2$](image2)

![Fig. 133: $j_\alpha = 3 \text{ kA/cm}^2$](image3)

![Fig. 134: $j_\alpha = 4 \text{ kA/cm}^2$](image4)

![Fig. 135: $j_\alpha = 5 \text{ kA/cm}^2$](image5)

![Fig. 136: $j_\alpha = 6 \text{ kA/cm}^2$](image6)

The values of temperature along the region centreline close to the cathode spot are shown in Figure 137. Note the general increase of peak temperature with increasing current density, as well as the propensity for the maximum point to move closer to the cathode surface, indicating a hotter and more concentrated arc. The 4 kA/cm$^2$ case is somewhat anomalous, but as will be seen shortly the arc behaviour in this case is quite unusual.

![Figure 137: Temperature profiles near cathode spot, time = 10ms, $x = 0.1m$](image7)

The changing value of current density produces some surprising alterations in the behaviour of the two-dimensional model. Many of these are visible in the animations on the accompanying disk, in
the directory animations\arcmodels2d\currentdensity. Two particular phenomena are highlighted here; although it is somewhat debatable whether these only occur as a result of the changing value of $j_s$ (in all likelihood similar behaviour would be observed in the standard base case model as well if it were allowed to run for a sufficient length of time), they are still of considerable interest as phenomena of the DC plasma arc model in general.

In the 4 kA/cm$^2$ case, the parasitic arcs on the anode surface combine to form a single very strong and unusually long-lived anode arc, which actually reverses the direction of the arc jet and circulation in the region for a considerable amount of time. This is shown in Figure 138.

*Figure 138: Temperature field for reverse arc flow, $j_s = 4$ kA/cm$^2$, time = 7.953ms, scale range 2000:15000K*

In the 3 kA/cm$^2$ case, a large unstable current loop forms and decays near the end of the simulation. The shapes involved are reminiscent of the higher-current unstable arc photographs obtained by Zweben and Karasik$^{17}$. The phenomenon is clearly visible in the temperature profiles plotted in Figures 139 to 142.

*Figures 139 - 142: Temperature field for formation and decay of arc loop, $j_s = 3$ kA/cm$^2$, scale range 2000:15000K*
Temporal behaviour

The initial evolution of the temperature close to the cathode spot is shown in Figure 143.

As was observed for the arc current cases, the initial stable-arc development behaviour is quite smooth, and there is a distinct and consistent dependence of temperature on \( j_i \). The onset of the typical unstable arc behaviour occurs earlier as the current density is increased - as with the current, this suggests that high-velocity flow phenomena and instabilities near to the cathode spot are driving much of the unsteady behaviour that occur in the arc model.

Figure 143: Early evolution of temperature,
\[ x = 0.1m, y = 0.045m \]

Figure 144: Early evolution of arc voltage

Figure 144 shows the evolution of the arc voltage at the early stages for selected values of \( j_i \). It is additionally interesting to observe from this graph that the oscillations of the field values grow noticeably larger as the current density is increased. This is supported by Figure 145, which shows the variation of temperature near to the cathode spot towards the end of the simulation.

The behaviour of the model at lower current densities clearly favours slower, smoother changes in the field variables once the arc jet is fully established and interacting with its surroundings. As the current density value is increased, larger and more sudden changes can occur, and the characteristic time frames of the arc system decrease considerably.
5.6 Sensitivity analysis for constant physical properties

Certain physical properties - density, heat capacity, thermal conductivity, and viscosity - in the DC plasma arc model are assumed to be constants. In reality, these properties can vary widely with temperature and (particularly) plasma gas composition.

Table 12: Property values used in sensitivity analysis

<table>
<thead>
<tr>
<th>Property</th>
<th>Value / 2</th>
<th>Value x 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu$</td>
<td>$6.535 \times 10^{-5}$ Pa.s</td>
<td>$2.614 \times 10^{-4}$ Pa.s</td>
</tr>
<tr>
<td>$\rho$</td>
<td>0.01297 kg/m$^3$</td>
<td>0.05186 kg/m$^3$</td>
</tr>
<tr>
<td>$\kappa$</td>
<td>1.534 W/mK</td>
<td>6.314 W/mK</td>
</tr>
<tr>
<td>$C_p$</td>
<td>4710 J/kgK</td>
<td>18840 J/kgK</td>
</tr>
</tbody>
</table>

In order to assess their effects, a simple sensitivity analysis is conducted by changing the values of the constant properties by a certain factor relative to the value used in the base case model (values in Table 7). Factors of 1/2 and 2 were used to indicate a likely range, as shown in Table 12. The models were run on an Intel Core2 Q6600 machine with four CPUs and 2GB RAM, and required between 27680 and 47615 time step calculations at an average time step size of between 0.21 and 0.36 µs. Each model run took between 2.0 and 3.4 hours to complete.
Variation of average peak values

To evaluate the sensitivity of the model to the various property parameters, the time-averaged maximum values of the field variables in the model are calculated between 5 and 10ms. These are then compared against each other as functions of the property multiplier to determine which have more pronounced effects.

Figure 146: Graph of time-averaged peak velocities as a function of property multiplier

In Figure 146 the density and viscosity behave similarly, producing higher velocities as they are decreased. Altering the heat capacity has the least effect overall. Surprisingly changing the thermal conductivity can have a fairly strong effect, similar in magnitude to that of the density, although the dependence is inverted - peak velocities decrease with larger values of $\kappa$.

Figure 147: Graph of Reynolds number as a function of property multiplier
Figure 147 demonstrates that the largest effect influencing the Reynolds number of the flow, and hence the transient behaviour from a fluid flow point of view, is the viscosity. It is capable of changing $Re$ by nearly an order of magnitude over the range of property values considered. The density also has a significant impact as it is changed, however its impact is reduced due to the fact that $Re$ is proportional to the product of velocity (which decreases with $\rho$) and $\rho$ itself. The thermal variables both have a relatively small impact on $Re$.

*Figure 148: Graph of peak temperature as a function of property multiplier*

As shown in Figure 148, the peak temperatures generated in the arc are most sensitive to the thermal property variables, $\kappa$ and $C_\rho$. Dependence of peak temperature on heat capacity is monotonic, increasing as $C_\rho$ increases, however the thermal conductivity appears to produce a maximum in the middle of the range examined.

It is interesting to note that the range of peak temperatures is considerably narrower than that of the flow variables presented earlier - temperatures in the arc are quite insensitive to the values of the constant properties, in general.

Figure 149 shows that the arc voltage, given by the maximum value of the $\phi$ field, is also relatively insensitive to variation of the constant physical properties. The thermal variables $\kappa$ and $C_\rho$ have similar effects, both raising the arc voltage as they are increased. This is an interesting and somewhat counter-intuitive result, given that these properties generally work to increase the peak temperatures over part or all of their range - higher temperatures would lead to higher electrical
conductivities, and one would expect lower voltages, however, the opposite trend seems to occur.

Figure 149: Graph of arc voltage as a function of property multiplier

Variations in dynamic behaviour

As the DC plasma arc model is fundamentally a tool for studying the transient behaviour of arcs, the impact of the variations in physical properties on the dynamics is presented.

Beginning with heat capacity, Figures 150 to 153 show the evolution of the temperature and velocity \( v_s \) fields at a location 5mm below the centre of the cathode spot.

Figure 150: Early evolution of temperature \( (C_P) \)  
Figure 151: Developed evolution of temperature \( (C_P) \)
Lower heat capacity is seen to increase the temperatures in the arc in the initial phases, and delay the onset of transient behaviour. Higher heat capacities appear to produce somewhat more variability in the behaviour of the temperature and velocity fields, although the differences become less exaggerated later in the simulations once the transient flow is well established.

The effect of altering the thermal conductivity on the dynamic behaviour of the model is shown in Figures 154 to 157, again at a location 5mm below the centre of the cathode spot.
Higher values of $\kappa$ are seen to produce lower initial temperatures and velocities in the arc column, and considerably earlier onset of transient motion. The transient behaviour is also considerably more erratic and variable at higher values, and this trend persists in the well developed flow toward the end of the simulations. Variations in the thermal conductivity would appear to have a significant impact on the dynamics of the DC plasma arc model.

The effect of varying the density on the model’s behaviour is studied next, and the results are shown in Figures 158 to 161. As before, the location of the sample points is 5mm below the centre of the cathode spot.
As is expected, lower densities tend to produce higher initial velocities in the arc jet before the onset of turbulent transient motion. Lower densities also result in elevated temperatures during this phase, and cause the onset of variable transient motion to occur considerably earlier in the simulation. The level of fluctuation of the fields in the model, particularly the velocity, is somewhat higher at low densities, a trend that persists as the flow develops.

Finally, the effect of the plasma gas viscosity on the dynamics generated by the arc model is shown in Figures 162 through 165, again sampled at a location 5mm below the centre of the cathode spot.
Interestingly, altering the viscosity appears to have little or no effect on the early stages of the development of the arc and the onset of transient motion. As the flow becomes more fully developed however the viscosity is seen to affect the transient behaviour quite noticeably, with lower viscosities causing more turbulent and highly-variable motion as might be expected. Higher viscosities cause slower and less sudden changes in both the temperature and the velocity toward the end of the simulations.

It is clear from these limited results that the physical properties chosen for the DC plasma arc model can have a considerable impact on certain of the results it produces. In the scope of this work, this is of particular interest when considering the effect a plasma gas of different chemical composition might have on the behaviour of the arc system. This would involve changing all of the properties discussed here in some way, as well as the (non-constant) electrical conductivity and thermal radiation properties; the effects of such a change will be examined in more detail in the following section.

5.7 Argon gas

The plasmas encountered in the atmospheric pressure DC arcs used in furnace metallurgy are produced from gases in the surrounding environment, and are typically rich in either air or carbon monoxide. However, in certain specialised applications it is desirable to introduce a noble gas such as argon into the arc region, as this modifies the electrical and physical properties of the arc and additionally makes it chemically neutral. The use of inert gases in plasma arcs also finds extensive application in the welding industry, where DC arc welding equipment frequently uses argon to
improve the heat transfer properties of the arc.

With these points in mind, we examine the effect of substituting the properties of air for those of argon in the DC plasma arc model. All parameters in the model remain the same as those reported in Table 7, with the exception of the values in Table 13.

**Table 13: Argon physical properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plasma gas</td>
<td>Argon</td>
</tr>
<tr>
<td>Viscosity, $\mu$</td>
<td>$1.408 \times 10^4$ Pa.s</td>
</tr>
<tr>
<td>Density, $\rho$</td>
<td>0.04898 kg/m$^3$</td>
</tr>
<tr>
<td>Thermal conductivity, $\kappa$</td>
<td>1.665 W/mK</td>
</tr>
<tr>
<td>Heat capacity, $C_p$</td>
<td>3715 J/kgK</td>
</tr>
</tbody>
</table>

Argon also changes the temperature dependence of the electrical conductivity and thermal radiation - these values are reported in appendix 2. The model was run on an Intel Core2 Q6600 machine with four CPUs and 2GB RAM, and required 24971 time step calculations at an average time step size of between 0.40 $\mu$s. The model run took 1.8 hours to complete.

**Average and instantaneous peak values**

Time-averaged and instantaneous maximum values of the field variables in the model using argon gas are calculated, the former between 5 and 10ms, and the latter at the end of the model run, at 10ms. These are then compared against the values obtained for the base case model (air). The results are shown in Table 14.

The change to argon gas has several interesting effects on the field variables in the model. The velocity decreases (this is primarily due to the higher density and lower thermal conductivity, as seen in the sensitivity analysis), however, the Reynolds number of the arc jet increases, indicating a possibly more turbulent flow.
Table 14: Time-averaged (5 - 10ms) and instantaneous (10ms) peak values

<table>
<thead>
<tr>
<th>Variable</th>
<th>Air (averaged)</th>
<th>Argon (averaged)</th>
<th>Air (instantaneous)</th>
<th>Argon (instantaneous)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>779.4 m/s</td>
<td>578.5 m/s</td>
<td>744.8 m/s</td>
<td>694.5 m/s</td>
</tr>
<tr>
<td>Reynolds number</td>
<td>7731</td>
<td>10062</td>
<td>7390</td>
<td>12075</td>
</tr>
<tr>
<td>Temperature</td>
<td>15752 K</td>
<td>16158 K</td>
<td>15491 K</td>
<td>15057 K</td>
</tr>
<tr>
<td>Arc voltage</td>
<td>236.9 V</td>
<td>139.7 V</td>
<td>236.7 V</td>
<td>137.5 V</td>
</tr>
</tbody>
</table>

This is supported by the temperature behaviour - while the values of temperature remain quite comparable between the two gases, the difference between the time-averaged value and the instantaneous value is higher in the case of argon, indicating oscillations of increased magnitude in the field.

The arc voltage is drastically affected by the use of argon over air - it drops by nearly 100V. This is related to both the behaviour of the plasma electrical conductivity, which in the case of argon is slightly higher than air at a given temperature, and secondary effects of the other properties, which act to produce a higher-temperature and physically larger arc column in the case of argon. Clearly, modification of the electrical behaviour of a circuit containing a DC plasma arc is most easily effected by the (relatively) simple expedient of using a different plasma gas.

Comparison of spatial data

The temperature profile of the model region is plotted in Figures 166 - 169 for the two different gases at different times.

There is fairly little to choose between the turbulence levels of the temperature profiles, although the animation in the directory animations/arcmodels2d/argon on the accompanying disk does show a considerable degree of turbulent motion.

The arc column generated in argon gas is visibly larger and more spread out in space than that in air, which as mentioned is an additional contributory factor to the lower arc voltages observed for argon - increased arc width provides more area for current flow, and lowers the electric potential field correspondingly.
Examining the temperature field along the centreline of the region near to the cathode spot shows fairly similar behaviour between the two gases. This is shown in Figure 170.

*Figure 170: Temperature profile near to cathode spot, x = 0.1m, time = 10ms*

The argon arc's peak temperature occurs closer to the cathode spot surface, but the magnitude of the temperatures in this peak region are otherwise relatively similar.

The electric potential field along the centreline shown in Figure 171 demonstrates the large difference in arc voltage values generated by the two gases clearly.
It is interesting to note that the qualitative dependence of $\phi$ on $y$ is similar between the two different plasma gases, with non-linearities near to the cathode spot and anode surface, and approximately linear behaviour over the bulk of the arc column in the interior of the region.

**Temporal behaviour**

A comparison of the evolution in time of the arc voltage is shown in Figures 172 and 173.

Apart from the obvious change in magnitude, during the initial stages the air model does appear more changeable, exhibiting more frequent oscillations. However, the argon model displays much larger peak-to-peak movements in the voltage, and more erratic behaviour in general.
Figure 173 shows that toward the latter stages of the simulation, the argon model's rate of oscillations increases to match that of the air model. As the variable fields in the models evolve through time, the transient behaviour settles and becomes reasonably similar for the two gases.

5.8 Thermal boundary conditions

The boundary conditions used for the temperature field in the DC plasma arc model can have an unusually strong effect on the rest of the components of the model. This is primarily due to the strong coupling between the temperature and the electrical conductivity, which controls the behaviour of the electromagnetic fields and hence the source terms for the flow and temperature equations.

In this section, we discuss results obtained from applying variations of the boundary conditions used in the base case model. In each case, a single parameter was changed relative to the base case in Table 7, and the effects compared. The parameters altered for each test are shown in Table 15.

<table>
<thead>
<tr>
<th>Boundary condition type</th>
<th>Wall temperature</th>
<th>Anode temperature</th>
<th>Electrode temperature</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBC 1</td>
<td>1000 K</td>
<td>2000 K</td>
<td>3000 K</td>
</tr>
<tr>
<td>TBC 2</td>
<td>2000 K</td>
<td>3000 K</td>
<td>4100 K</td>
</tr>
<tr>
<td></td>
<td>3000 K</td>
<td>4000 K</td>
<td>5000 K</td>
</tr>
</tbody>
</table>

The models were run on an Intel Core2 Q6600 machine with four CPUs and 2GB RAM, and required between 23901 and 42130 time step calculations at an average time step size of between 0.24 and 0.42 μs. Each model run took between 1.7 and 3.0 hours to complete.

Average and instantaneous peak values

Time-averaged and instantaneous maxima of the field variables in the model are calculated, the former between 5 and 10ms, and the latter at the end of the model run, at 10ms. These are then compared against each other as functions of the boundary condition used.
Firstly, we consider the case of using TBC 2, corresponding to thermally insulating boundaries at all surfaces in the model. The results are shown in Table 16.

Clearly the removal of the cold-boundary influence on the DC plasma arc model has a significant impact on the peak values of the field variables, supporting the argument that the phenomena near to the boundary are influential in determining the overall behaviour of the model.

<table>
<thead>
<tr>
<th>Variable</th>
<th>TBC 1 (averaged)</th>
<th>TBC 2 (averaged)</th>
<th>TBC 1 (instantaneous)</th>
<th>TBC 2 (instantaneous)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>779.4 m/s</td>
<td>561.1 m/s</td>
<td>744.8 m/s</td>
<td>582.0 m/s</td>
</tr>
<tr>
<td>Reynolds number</td>
<td>7731</td>
<td>5565</td>
<td>7390</td>
<td>5770</td>
</tr>
<tr>
<td>Temperature</td>
<td>15752 K</td>
<td>13989 K</td>
<td>15491 K</td>
<td>13911 K</td>
</tr>
<tr>
<td>Arc voltage</td>
<td>236.9 V</td>
<td>185.2 V</td>
<td>236.7 V</td>
<td>187.7 V</td>
</tr>
</tbody>
</table>

We now turn to the specification of different values for the boundary condition temperatures, using TBC 1.

**Figure 174: Graph of peak velocities vs boundary temperature, 5 - 10ms average**

The peak velocities are presented in Figure 174. Apart from the wall temperature which exerts relatively little influence over the domain in the vicinity of the cathode spot and anode surfaces, the general trend is for increased boundary temperatures to produce lower velocities. The strength of
the gradient of the temperature near the electrically-conducting boundaries thus appears to have an indirect influence on the strength of the fluid flow.

Reynolds numbers are directly proportional to the velocity values in this case, and range between 6000 and 9000.

Figure 175: Graph of peak temperatures vs boundary temperature, 5 - 10ms average

The effect of the boundary temperatures on peak temperature is shown in Figure 175. Again, the wall temperature is seen to have a minimal impact on the overall behaviour of the temperature field. The anode and electrode boundary temperatures do exert an influence however, with both acting to decrease the peak temperatures in the region as the boundary temperature is increased. This seems somewhat counter-intuitive at first. The mechanism is believed to be as follows - lower boundary temperatures act on the electrical conductivity near to the boundary, reducing it. This causes steeper gradients of the electric potential field to develop, which in turn generate increased localised current density and hence stronger Ohmic heating of the plasma gas, producing higher temperatures. This increased local current density also produces larger local values of the magnetic field, increasing the forces on the fluid and causing the higher velocities as seen in Figure 174. It is important to note that this explanation concerns the current densities near to the cathode surface, that is, in the body of the arc and not at the surface itself; densities at the surface are specified directly by boundary conditions for the electric potential. Current densities in the body of the arc are governed by the local gradient of electric potential which is severely affected by any changes in the local temperature field.
Arc voltage is affected by the boundary conditions as shown in Figure 176. The behaviour of the maximum value of $\phi$ with changing boundary temperatures is somewhat more erratic, although some similar trends to the previous two variables can be discerned. The boundary with the most direct impact on arc voltage is the electrode (as it contains the cathode spot). Based on the previous arguments, we would expect the total voltage of the arc to decrease as the boundary temperature rises, and this is indeed observed.

\begin{figure}
\centering
\includegraphics[width=0.5\textwidth]{figure176}
\caption{Graph of arc voltage vs boundary temperature, 5 - 10ms average}
\end{figure}

\textbf{Comparison of spatial data}

Beginning with the comparison of TBC 1 and 2, plots of the temperature over the model region are presented in Figures 177 and 178.

\begin{figure}
\centering
\includegraphics[width=0.5\textwidth]{figures177-178}
\caption{Temperature plot, TBC 1 vs 2, time = 10ms, scale range 2000:15000K}
\end{figure}

Several phenomena are immediately obvious from these temperature visualisations. Firstly, the average value of the temperature in the plasma gas further from the arc column is considerably higher in the TBC 2 case. The range of temperatures is also much narrower, with a lower maximum and higher minimum over the region. The arc is more symmetric and well defined in the TBC 2
case, and there is no evidence of parasitic arcs - this phenomenon is likely dependent on strong temperature gradients near to the anode.

The turbulence of the flow also appears considerably reduced in the TBC 2 case - this may be confirmed by examining the fluid vorticity in the region, as shown in Figures 179 and 180. Although the levels of vorticity are reasonably comparable between the two models, the TBC 1 case shows considerably more mixed and asymmetric flow than that of TBC 2 - the thermal boundary conditions are clearly contributing to the transient character of the jet flow in the DC plasma arc model.

*Figures 179 - 180: Vorticity plot, TBC 1 vs 2, time = 10ms, scale range -150000:150000 s⁻¹*

![Fig. 179: TBC 2](image1)

![Fig. 180: TBC 1 (base case)](image2)

The temperature profiles along the region centreline are shown in Figure 181.

*Figure 181: Temperature profiles along centreline, time = 10ms, x = 0.1m*

![Graph showing temperature profiles](image3)

The differences in behaviour of the temperature field, particularly near to the anode surface (y = 0) and cathode spot (y = 0.05m) are striking. Peak temperature for TBC 2 occurs at the cathode spot surface, whereas the maximum in the case of TBC 1 is located in the interior of the model region. The range of temperatures produced by the TBC 1 case is also seen to be considerably larger,
supporting the observation made regarding Figures 177 and 178.

*Figure 182: Electric potential profiles along centreline, time = 10ms, x = 0.1m*

Next, we examine the behaviour of $\phi$ on the centreline of the model region - this is shown in Figure 182. The smooth behaviour of $\phi$ with distance from the anode provides additional support to the observations made earlier - the change to insulating thermal boundary conditions greatly reduces or eliminates near-boundary non-linear behaviour in many of the field variables, and this non-linearity seems to drive much of the unstable evolution of the arc system.

*Figure 183: Temperature profiles at the cathode spot for different $T_{\text{electrode}}$, x = 0.1m, time 10ms*

Changing the boundary temperatures using TBC 1 introduces small effects that cause larger deviations to propagate out into the regions further from the boundaries, due to the non-linear dynamics of the DC plasma arc model. Examining the temperature profiles along the centreline of
the region close to the cathode spot for variable $T_{\text{electrode}}$ illustrates this effect in Figure 183.

The impact of changing the values of the boundary conditions for TBC 1 can also be assessed by computing the energy flux being transferred to the wall from the gas at the boundary. This is obtained from the relationship

$$q = -\kappa \left[ \frac{\partial T}{\partial n} \right]_r,$$  \hspace{1cm} (221)

which gives the local energy flux at a point on the boundary. This is then then integrated numerically over the part of the boundary of interest to give an average value. These values are reported as functions of the boundary temperatures in Table 17.

### Table 17: Effect of changing boundary temperatures on energy fluxes, at time 10ms

<table>
<thead>
<tr>
<th>Boundary</th>
<th>Boundary temperature, $K$</th>
<th>Boundary energy flux, kW/cm²</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wall</td>
<td>1000</td>
<td>0.151</td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.105</td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.0975</td>
</tr>
<tr>
<td>Anode</td>
<td>2000</td>
<td>1.06</td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.750</td>
</tr>
<tr>
<td></td>
<td>4000</td>
<td>0.635</td>
</tr>
<tr>
<td>Electrode</td>
<td>3000</td>
<td>1.49</td>
</tr>
<tr>
<td></td>
<td>4100</td>
<td>1.14</td>
</tr>
<tr>
<td></td>
<td>5000</td>
<td>1.02</td>
</tr>
</tbody>
</table>

In all cases considered, decreasing the boundary temperature works to increase the average energy flux on the same boundary. This corresponds to an increase in the temperature gradient near to the wall as hypothesised, and this in turn drives larger differences in the electromagnetic, thermal, and flow fields.

**Temporal behaviour**

Of particular interest from a qualitative transient behaviour point of view is the comparison between
TBC 1 and TBC 2. The differences are clearly visible in a trace of the arc voltage over time as shown in Figure 184.

From this graph it is obvious that the degree of turbulence and variation in time occurring in the TBC 2 model is much lower than that seen when TBC 1 is applied, and the latter transitions far more rapidly into unsteady motion.

*Figure 184: Evolution of arc voltage for different temperature boundary conditions*

It is nonetheless interesting to note that irregular transient behaviour *does* still occur in the TBC 2 model; this is also visible in the animations for TBC 2 available on the accompanying disk, in the directory animation\temperatureBCs. Significant oscillation of the arc column and "hosepiping" behaviour causing the creation of sinusoidal, twisted arc shapes are still present in the model when thermally insulating boundary conditions are applied. This suggests that at least part of the dynamics of the arc are related to fundamental interactions between components of the model, as opposed to the choice of boundary conditions for temperature alone.

A comparison between values of the temperature field at a point a few millimetres below the cathode spot, at both the initial stages and at the end of the simulation during fully developed unsteady arc jet flow is shown in Figures 185 and 186. The rapid transition to unsteady behaviour in the TBC 1 model is in stark contrast with the evolution of the TBC 2 model, which maintains steady, smooth values of temperature until well over 1ms into the simulation. As the models move away from the initial conditions and begin to settle toward well established flow and temperature fields, the differences between the two boundary condition specifications become less well defined.
Clearly TBC 2 is still exhibiting a lesser degree of variability over time, however it is interesting to note that the typical time scales of the oscillatory behaviour occurring are very similar for both.

*Figure 185: Initial evolution of temperature, $x = 0.1m, y = 0.045m$*

*Figure 186: Evolution of temperature for established flow, $x = 0.1m, y = 0.045m$*

It appears that the phenomena governing time scales in the DC plasma arc model are properties of the fundamental equations governing the arc system, and the effect of altering the thermal boundary conditions is to enhance dynamics that are already present rather than create them from scratch.
Chapter 6 - Model Cases (3D)

In this chapter, the DC plasma arc model in three dimensions is used to study a set of problems having parameters relating to DC arc furnaces at small pilot-plant scale - in three dimensions, the focus is on the interactions and transient behaviour of multiple arc systems. As with the 2D cases, the results will begin with a detailed study of the information produced by the model using a base set of parameters. Variation of selected parameters of the base case model will then be used to examine their effect on the range of phenomena the model system may exhibit. Where possible, comparison will also be performed between the model predictions, empirical relationships, and photographic evidence.

6.1 Description of three-dimensional models and cases studied

Geometry and spatial layout

The three-dimensional DC plasma arc models all describe multiple arc systems, with a particular focus on twin arc arrangements.

The geometry defines a rectangular oblong region of space between the end of the graphite electrode(s) and the surface of the molten bath in the furnace, which serves as the anode. This is shown in Figures 187 and 188 below.

*Figure 187: Model geometry and boundaries for 3D cases*
The $x$, $y$, and $z$ dimensions start at the origin of a cartesian axis system, at corner A. The region length, width, and height are given by $x_{AB}$, $y_{AD}$, and $z_{AE}$ respectively. The anode surface is given by boundary ABCD. Surfaces $\Gamma_M$ on boundary EFGH define the location of the electrodes, and surfaces $\Gamma_N$ define the location of the cathode spots which form the root of each arc.

Surfaces $\Gamma_M$ and $\Gamma_N$ are circular, with radii either directly specified in the case of $\Gamma_M$, or indirectly calculated from a given arc current and cathode spot current density in the case of $\Gamma_N$. Any independent number of electrodes and cathode spots can be specified in the model, although for this set of results we will always assume that each cathode spot is concentric with its associated electrode. Their positions are given by specifying a series of $x$ and $y$ locations $x_{e,1}$, $x_{e,2}$, ..., and $y_{e,1}$, $y_{e,2}$, ...

**Parameters**

As for the 2D cases, values for the constant physical properties of the plasma gas ($\mu$, $\rho$, $\kappa$, $C_p$), must be given. These are specified directly by using average values over a representative range of temperatures for the particular plasma gas used.

Temperature-dependent values of $\sigma$ and $Q_h$ are also required, and are given in the same form as specified for the 2D models.
The electrical parameters of each arc in the 3D model must be given. This includes specifying the value of \( j_k \) used for all cathode spot boundaries \( \Gamma_n \), as well as a list of the (possibly different) electrical currents associated with each cathode spot. These define the radii of the cathode spots as discussed above.

Only TBC 1 will be used for the thermal boundary conditions for the three-dimensional model cases. Temperatures at the various parts of the model region boundary (anode, walls, electrode surface) must be given, as per the 2D case described in Table 6.

Finally, as for the 2D cases, parameters defining the numerics of the model must be specified. The grid dimensions \( I, J, \) and \( K \) must be given. Together with the dimensions \( x_{AE}, y_{AE}, \) and \( z_{AE} \), values for \( \delta x, \delta y, \) and \( \delta z \) can be calculated. For the 3D models, the region dimensions and grid resolution will always be chosen such that \( \delta x = \delta y = \delta z = \delta l \) in order to simplify the numerical mathematics and improve performance. The number of threads to be used, and the maximum model time to calculate until, then complete the data needed by the DC plasma arc model in three dimensions.

**A note on resolved flow models**

In general, it is relatively easy to design and run the two-dimensional DC plasma arc model to ensure that the conditions for fully-resolved fluid flow are met. Unfortunately, in three dimensions these conditions are considerably more challenging:

\[
\frac{L}{\delta l} \leq \text{Re}^{3/4} = \left( \frac{\rho |v|}{\mu} \right)^{3/4} \tag{222}
\]

where \( L \) is some characteristic length of the region or flow - in the case of the DC plasma arc model we use \( L = z_{AE} \). Due to the extremely memory-intensive nature of the three-dimensional models, practical limits on grid resolution mean that this condition is generally not met for the typical region dimensions and velocities of flows encountered when studying moderate-sized pilot plant scale DC plasma arcs.

Having said this, the three-dimensional model does display extremely good numerical stability
characteristics and smoothness of the obtained solution for even very under-resolved flows. While some information may be lost at sub-grid scale levels, the transient behaviour of the variable fields at the length scales that are captured by the model is still sufficient to offer useful insights.

It should be mentioned that with the current rate of advance in computer hardware, practical fully-resolved three-dimensional models of the DC plasma arc (at least at pilot plant furnace scales) are likely to become possible within the next two years or less.

**Initial conditions**

The initial conditions supplied to the model are equivalent to those used for the 2D case, constant-temperature impulsive start conditions. The velocity (or equivalent field) is set to zero at the start of the calculation, and the initial temperature field is set to a constant value, 10000K. From this "flash start" condition, the complete development and evolution of the arc column(s) in time may be observed.

**Boundary conditions**

The spatial boundary conditions applied to the various components of the model in three dimensions are described in Table 18 below. The intermediate variables such as the $a$ and $\theta$ fields for fluid flow and the magnetic vector potential used in the electromagnetic field calculation are supplied with derived boundary conditions, as described in chapter 2.

**Table 18: Boundary conditions for 3D plasma arc model**

<table>
<thead>
<tr>
<th>Variable</th>
<th>ADHE &amp; BCGF</th>
<th>ABFE &amp; CDHG</th>
<th>ABCD</th>
<th>$\Gamma_1$</th>
<th>$\Gamma_M$</th>
<th>$\Gamma_N$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v_x$, $v_y$</td>
<td>$v_x = v_y = 0$</td>
<td>$v_x = v_y = 0$</td>
<td>$v_x = v_y = 0$</td>
<td>$v_x = v_y = 0$</td>
<td>$v_x = v_y = 0$</td>
<td>$v_x = v_y = 0$</td>
</tr>
<tr>
<td>$T$</td>
<td>$T = T_{wall}$</td>
<td>$T = T_{wall}$</td>
<td>$T = T_{anode}$</td>
<td>$T = T_{wall}$</td>
<td>$T = T_{electrode}$</td>
<td>$T = T_{electrode}$</td>
</tr>
<tr>
<td>$\phi$</td>
<td>$\frac{\partial \phi}{\partial y} = 0$</td>
<td>$\frac{\partial \phi}{\partial y} = 0$</td>
<td>$\phi = 0$</td>
<td>$\frac{\partial \phi}{\partial z} = 0$</td>
<td>$\frac{\partial \phi}{\partial z} = 0$</td>
<td>$\frac{\partial \phi}{\partial z} = -\frac{j_k}{\sigma}$</td>
</tr>
</tbody>
</table>
Base reference case

As for the two-dimensional modelling exercise, we begin with the study of a base set of parameters, chosen to represent reasonable conditions for a twin arc DC plasma arc at laboratory or small pilot plant furnace scale. These are shown in Table 19. A limited set of model parameters are then varied independently of the others to examine their effect on the spatial and transient behaviour.

Table 19: Specification for base case 3D model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Region length, $x_{AB}$</td>
<td>0.2 m</td>
</tr>
<tr>
<td>Region width, $y_{AB}$</td>
<td>0.1 m</td>
</tr>
<tr>
<td>Region height, $z_{AE}$</td>
<td>0.05 m</td>
</tr>
<tr>
<td>Electrode diameter</td>
<td>0.05 m</td>
</tr>
<tr>
<td>Plasma gas</td>
<td>Air</td>
</tr>
<tr>
<td>Thermal radiation model</td>
<td>Optically thin</td>
</tr>
<tr>
<td>Viscosity, $\mu$</td>
<td>$1.307 \times 10^{-4}$ Pa.s</td>
</tr>
<tr>
<td>Density, $\rho$</td>
<td>0.02593 kg/m$^3$</td>
</tr>
<tr>
<td>Thermal conductivity, $\kappa$</td>
<td>3.067 W/mK</td>
</tr>
<tr>
<td>Heat capacity, $C_p$</td>
<td>9420 J/kgK</td>
</tr>
<tr>
<td>Cathode spot current density, $j_k$</td>
<td>$3.5 \times 10^7$ A/m$^2$</td>
</tr>
<tr>
<td>Number of arcs</td>
<td>2</td>
</tr>
<tr>
<td>Current, arc 1</td>
<td>250 A</td>
</tr>
<tr>
<td>Current, arc 2</td>
<td>250 A</td>
</tr>
<tr>
<td>$x_{e,1}$, $y_{e,1}$</td>
<td>0.08 m, 0.05 m</td>
</tr>
<tr>
<td>$x_{e,2}$, $y_{e,2}$</td>
<td>0.12 m, 0.05 m</td>
</tr>
<tr>
<td>Temperature boundary condition</td>
<td>TBC 1</td>
</tr>
<tr>
<td>Wall temperature, $T_{wall}$</td>
<td>2000 K</td>
</tr>
<tr>
<td>Anode temperature, $T_{anode}$</td>
<td>3000 K</td>
</tr>
<tr>
<td>Electrode temperature, $T_{electrode}$</td>
<td>4100 K</td>
</tr>
<tr>
<td>Grid resolution, $I \times J \times K$</td>
<td>$384 \times 192 \times 96$</td>
</tr>
<tr>
<td>Number of threads</td>
<td>4</td>
</tr>
<tr>
<td>Maximum model time</td>
<td>10 ms</td>
</tr>
</tbody>
</table>
A comment on symmetry

As in the 2D case, due to the symmetrical treatment of the spatial dimensions in the numerical model, it is possible for perfectly symmetrical solutions to develop – in the 3D case these too are physically unrealistic as they are highly sensitive to any minor perturbation away from symmetry. As before, they are avoided by introducing a small degree of asymmetry via the boundary conditions, by biasing the location of the cathode spots in the model (that is, the spots are not exactly symmetrical with each other in the region). The bias very small, of the order of 0.5% of the size of the region, but is sufficient to perturb the evolution of the model away from the development of unrealistic solutions.

Projected 2D visualisations

Due to the difficulties associated with visualising the variable fields in three dimensions, a method of reducing the dimensionality of the data is applied to complement the full 3D visualisations. This involves finding the maximum value of a variable at each point along a certain dimension, producing three 2D sets of data from a single 3D set - xy, showing the maximum along the z dimension, xz, showing the maximum along the y axis, and yz, showing the maximum along the x axis.

This is particularly useful for the temperature field, whose distribution in space corresponds approximately to the shape of the arc observed using visual experimental techniques.

6.2 Base case model

The Base Case model was run on an Intel Core2 Q6600 machine with four CPUs and 2GB RAM. The model required 4215 time step calculations at an average time step size of 2.4 μs, and took approximately 16.9 hours to complete.

Average and instantaneous peak values

As for the two-dimensional models, peak values of the various fields in the model are calculated from the data produced to give an indication of the intensity of the physical processes occurring. A time average of these numbers is obtained from the final half of the simulation (5 - 10 ms), at which
point the field behaviour and motion in the vicinity of the arc columns is well established. Instantaneous values are also presented for the model at the end of the simulation time (10 ms).

For the base case model as described, the values are presented in Table 20.

Table 20: Peak values of variables for base case model

<table>
<thead>
<tr>
<th>Variable</th>
<th>Peak value - 5 - 10ms average</th>
<th>Peak value - at 10ms</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v_x$, $v_y$, $v_z$</td>
<td>236.8 m/s</td>
<td>241.1 m/s</td>
</tr>
<tr>
<td>$Re$</td>
<td>2349</td>
<td>2392</td>
</tr>
<tr>
<td>$T$</td>
<td>15496 K</td>
<td>14825 K</td>
</tr>
<tr>
<td>$\phi$, arc 1</td>
<td>202.4 V</td>
<td>201.3 V</td>
</tr>
<tr>
<td>$\phi$, arc 2</td>
<td>198.0 V</td>
<td>207.7 V</td>
</tr>
</tbody>
</table>

The peak arc voltages are calculated using the maximum value of $\phi$ on each cathode spot. The voltages and temperatures agree reasonably well with the two-dimensional modelling results reported earlier.

Flow patterns

The flow field produced by the model using the base case set of parameters is shown in Figures 189 through 191 below, for times 1ms and 10ms. The contour surfaces are isovalues of the velocity magnitude, and the streamlines show the vector behaviour for part of the field.

In Figure 189, the contours are shown as partially transparent surfaces, and form two narrow and roughly concentric columns (velocity increasing toward the centre of each column) defining the shape of the two arc jets. The streamlines show the plasma being drawn into the arcs near to the cathode spots, travelling down the columns, and changing direction to recirculate into the region as the jet encounters the anode surface.
As was observed in the two-dimensional model results, the system initially forms pseudo-stable arcs as the initial conditions decay, but as the simulation proceeds the stability breaks down and transient motion begins to occur.

Figure 189: Velocity field for base case, time 1 ms, scale range 10:250 m/s

Figure 190: Velocity field for base case, time 10 ms, scale range 10:250 m/s

Figure 191: Velocity field for base case, time 10 ms, scale range 10:250 m/s
In the three-dimensional model the additional degree of freedom allows more complex motion of the fluid near to the cathode spots, and for the low values of current tested here, the motion manifests as a reasonably regular oscillation across the surface of the cathode spot. This precession of the arc root results in a rotation of the arc jet direction near to the spot, which in turn produces helical shapes in the flow and other fields further down the body of the arc.

Another phenomenon obvious in three dimensions is the magnetic interaction of the two arc columns - as conductors of direct current electricity in close proximity, each arc column experiences an attractive magnetic force due to the other. This force causes the arc jets to bend toward each other and begin to connect electrically and physically into a single larger jet directed at the anode.

**Temperature field**

The temperature field generated by the three-dimensional model using the base case parameters is shown in Figures 192 and 193. These show the values at the end of the simulation time, 10ms.

Several things are immediately visible in the temperature fields. Firstly, the degree of recirculation in the region as compared with the 2D models is considerably lower - many of the outlying reaches of space far from the arc columns appear to be cooling exponentially via conduction from the cold walls and gradual losses by thermal radiation, and have not yet come completely into equilibrium. This is partly due to the lower currents used for this case, but principally because of the extra dimensionality of the problem - the 3D model more accurately represents the arc columns and their interactions with the space in their vicinity.

The temperature plots also show the spiral structure of the arc columns generated by the transient behaviour in the model near to the cathode spot surfaces. The thermal structure of the arcs appears relatively similar to that observed in the 2D cases, with a very hot arc root region near to the cathode spot feeding high temperature gases down the arc jet to form the characteristic columnar shape.

The magnetic attraction between the arc columns is also clearly visible in the temperature profiles, with the arcs joining briefly near to the anode surface.
Electromagnetic fields

The electric potential field in the model region for the base case is shown in Figures 194 and 195. Data for the end of the simulation, at model time 10ms, are presented.

The electric potential field is concentrated near to the cathode spot surfaces, which are the highest-voltage points in the region. The gradient of the field also grows larger near to the lower surface, where the colder anode cools the nearby plasma gas and reduces its electrical conductivity. Near to the centre of the region the joining of the arc columns causes a region of elevated temperature, which in turn results in the electric potential field gradient being highly compressed against the anode surface near the point at which the arcs approach each other.
Figure 194: Electric potential field for base case, time 10ms, scale range 0:200V

Figure 195: Electric potential field for base case, time 10ms, scale range 0:200V

The magnitude of the current density vector field is shown in Figures 196 and 197.

Figure 196: Current density magnitude for base case, time 10ms, scale range 3500:3500000A/m²
As with the 2D cases current density closely tracks the temperature field, as its distribution in space is controlled by the temperature-dependent electrical conductivity. It is interesting to observe that near to the anode surface there are several "hot spots" of current density, suggesting the formation of parasitic arcs as observed extensively in the two-dimensional modelling results.

2D temperature projections

The temperature images shown below were generated as described in the previous section, by plotting two of the three dimensions showing the maximum value of temperature along the third. Figure 198 shows the projected temperature field at the end of the simulation.

The shape and structure of the arc columns is visible in this plot, as well as the twisting of the jets due to the transient instability at the cathode spots. This visualisation method is particularly useful for showing the initial process of arc establishment and breakdown into unstable motion - this is shown in the sequence of images from the early stages of the simulation, in Figures 199 to 202.
Figures 199 - 202: Temperature field (xz plane) for base case, scale range 2000:15000K

Fig. 199: 0.5 ms

Fig. 200: 1 ms

Fig. 201: 1.5 ms

Fig. 202: 2 ms

The arc jet formation is largely complete by 1 ms, and it is shortly after this point that the oscillatory behaviour begins, causing the arc column structure to change from simple cylindrical jets into helically twisted shapes. This process of arc formation and transition to unstable motion is also visible in the various animations on the accompanying disk, in the directory animations/arcmodels3d/basecase.

Spatial data

A more quantitative view of the field data presented above is possible if we fix two of the three dimensions and sample along the remaining one. This is particularly useful when performed in the x-direction along the plane in which the two arcs lie.

Figure 203: Various temperature profiles showing arc attraction, y = 0.05m, time 10ms
Sampling the temperature field at different heights above the anode surface gives an indication of the curved shape of the arc columns, shown in Figure 203. As we move from the upper surface to the lower, the peak values of temperature give an estimate of the location of the centreline of the high-temperature arc jet. The peak temperatures are most elevated (and most separated) near to the cathode spots. They approach each other and eventually begin to merge at the midpoint of the region, closer to the anode surface.

In Figure 204, sampling the z-component of the velocity field at different times during the simulation shows the change produced by the transition to unsteady state motion.

![Figure 204: $v_z$ profiles at different times, $y = 0.05m$, $z = 0.025m$](image)

It can be seen that as the transient behaviour of the arcs becomes increasingly dynamic, the peak velocity of the jets in the direction of the anode drops significantly as more energy is diverted into the helical precession motion.

**Temporal data**

The transient behaviour of the variable fields at certain locations in the three-dimensional plasma arc model is presented. We begin with the temperature variable, examining the evolution in time of a location a few millimetres below the surface of each of the two cathode spots. This is shown in Figure 205.
It is interesting to contrast these results with those for the 2D models - the latter generally display far more chaotic behaviour of temperature and the other field variables in time, whereas the three-dimensional model appears to develop highly regular oscillations.

This is borne out in the animations of the temperature field for the 3D base case model on the accompanying disk - it can be seen that the nature of the arc instability occurring here is associated with a regular precession of the arc jet around each cathode spot surface. This gives rise to both the helical shapes of the arcs as well as the regular oscillatory behaviour observed in the temperature variable close to the cathode spots.

The early evolution of the temperature profiles appears to be quite different for the two arcs in the base case model, with the second arc located at $x = 0.12\text{m}$ reaching higher initial temperatures. The initiation of transient behaviour is visible in the temperature traces, beginning at roughly the same time for each arc at shortly after 1ms as discussed earlier.

If we zoom in on the data shown in Figure 205, the regularity of the oscillations becomes more obvious. This is shown in Figure 206. It is interesting to note that the oscillations of the arc columns in the vicinity of the cathode spots are almost perfectly out of phase with each other. The character of the oscillations of each arc is also rather different - the arc at $x = 0.08\text{m}$ is exhibiting multi-modal oscillations, and considerably lower peak-to-peak values of temperature than the arc at $x = 0.12\text{m}$. The period of the oscillation for both arcs is similar, at approximately 0.185ms. This is similar to the time scales observed in the two-dimensional models, suggesting a degree of consistency as far
as the dynamics of the system are concerned.

*Figure 206: Regular oscillations in temperature values, \( y = 0.05m, z = 0.045m \)*

![Figure 206: Regular oscillations in temperature values](image)

Studying the arc voltage traces reveals similar behaviour. The arc voltages are defined as the local maxima of the electric potential field at each cathode spot surface, and are shown in Figure 207.

*Figure 207: Evolution of arc voltages*

![Figure 207: Evolution of arc voltages](image)

As was seen in the two-dimensional modelling exercise, large variations in temperature (and hence electrical conductivity) are somewhat damped by the electric potential field calculation. Oscillations are still visible, however, relative to the absolute value of the arc voltages their magnitude is small. This is even more exaggerated in the 3D model, which exhibits only around 5% of the total voltage in transient variation.
Examining the final stages of the simulation in Figure 208, the oscillatory behaviour of voltage is clear. As with the temperature variable, the oscillations are highly regular and well defined and operate on the characteristic 0.18 - 0.2ms time scales.

**Comparison of arc behaviour with photographic evidence**

As has been discussed previously, there is a lack of visual, qualitative experimental data in the field of DC plasma arc studies. This is further exacerbated in the case of multiple-electrode, multiple-arc systems, which are a somewhat specialised and comparatively recent invention.

To date, the only photographic work on small DC plasma arc furnaces using two electrodes has been performed by the author and colleagues, and is documented in Jones and Reynolds\textsuperscript{18}. This work used an actual pilot-scale DC arc furnace vessel containing a molten bath made from a synthetic slag. The furnace roof was removed to allow the arcs to be visually observed. The scale of the furnace prevented the use of high currents or arc lengths, typically limiting the values to below 1 kA and 15cm respectively.

Due to physical limitations of the furnace design, the electrode separation (20cm) used was considerably larger than that considered in the three-dimensional base case model, which typically leads to the arc columns remaining separate rather than joining together as is observed in the model.

As shown in Figures 209 and 210, the qualitative behaviour of the twin arc system as observed
experimentally has some similarities with the three-dimensional modelling results. The individual arc columns are well defined, and the temperatures concentrated in them appear to drop from peak values close to the cathode spots to lower temperatures closer to the anode.

*Figure 209: Twin arcs interacting magnetically, current ~380A each, arc lengths 6-9cm*

![Image of twin arcs interacting magnetically](image1)

*Figure 210: Twin arcs interacting magnetically, current ~470A each, arc lengths 14-15cm*

![Image of twin arcs interacting magnetically](image2)

The arc columns bend toward each other as they are attracted magnetically, smearing out the zone of high heat transfer on the anode surface and forming a region of convective coupling near the middle of the furnace in between the two electrodes.

In Figure 211, the arc on the right has entered an unstable operation regime. Although the instantaneous shape of the arc column is not visible in this photograph (the exposure time is far too long, at around 1/50 s), it is clear that the time-averaged shape of the column is much more diffuse
when compared with the arc on the left. Temperature appears highest in the vicinity of the electrode and cathode spot, and drops off rapidly as we move down towards the anode surface.

*Figure 211: Unstable twin arcs, current ~500A each, arc lengths 5-7cm*

This diffuse shape and temperature profile are quite similar to what is observed in the 3D base case model after the onset of the helical oscillations - compare the arc on the left with the shapes that develop in the early stages of the simulation, eg. Figure 200, and the arc on the right with the shapes that develop later, eg. Figures 198 and 202. With the onset of transient behaviour, the body of each arc is seen to grow larger and cooler in both the model and the photographic evidence.

### 6.3 Electrode separation effects

This set of results examines the effect of changing the electrode separation in the three-dimensional DC plasma arc model. The electrode separation is an important design variable when constructing furnaces with multiple electrodes, as it typically cannot be changed easily once the furnace vessel is constructed. An understanding of how the positioning of the electrodes can affect the behaviour of the arcs they generate is therefore of some importance.

The model parameters used are identical to those shown in Table 19, with the exception of the locations of the cathode spot and electrode surfaces. These are changed as shown in Table 21.

The models were run on an Intel Core2 Q6600 machine with four CPUs and 2GB RAM, and required between 3720 and 4400 time step calculations at an average time step size of between 2.3 and 2.7 μs. Each model run took between 14.9 and 17.6 hours to complete.
Table 21: Range of variables tested for electrode separation

<table>
<thead>
<tr>
<th>Electrode separation</th>
<th>$x_{c1}, y_{c1}$</th>
<th>$x_{c2}, y_{c2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 cm</td>
<td>0.09 m, 0.05 m</td>
<td>0.11 m, 0.05m</td>
</tr>
<tr>
<td>4 cm</td>
<td>0.08 m, 0.05 m</td>
<td>0.12 m, 0.05m</td>
</tr>
<tr>
<td>6 cm</td>
<td>0.07 m, 0.05 m</td>
<td>0.13 m, 0.05m</td>
</tr>
<tr>
<td>8 cm</td>
<td>0.06 m, 0.05 m</td>
<td>0.14 m, 0.05m</td>
</tr>
<tr>
<td>10 cm</td>
<td>0.05 m, 0.05 m</td>
<td>0.15 m, 0.05m</td>
</tr>
</tbody>
</table>

Average and instantaneous peak values

As for the base case, time-averaged and instantaneous maximum values of velocity, temperature, and arc voltage (for each arc) are calculated. These values are compared over the range of electrode separations tested.

In Figure 212, the maximum values of velocity generally rise and then level out as the separation between the arcs increases. This is not unexpected, as the degree of interaction between the arc columns is expected to increase as the separation decreases, causing large changes in behaviour at low separation values.

![Figure 212: Graph of peak velocity magnitude as a function of electrode separation](image)

Counter-intuitively, the difference between the instantaneous and averaged peak values grows larger
as separation distance increases, suggesting that the transient flow behaviour is more variable the further apart the two arcs are positioned. This will be examined and discussed in more detail using the temporal data later in the section.

The Reynolds numbers for this model are directly proportional to the velocity, and vary between 2000 and 2500.

*Figure 213: Graph of peak temperature as a function of electrode separation*

The behaviour of the peak temperatures with electrode separation is more erratic, as shown in Figure 213. The values all lie in a narrow range within 1000K of each other. In contrast to the velocity results, the difference between the average and instantaneous peak temperatures appears to grow larger with lower separation values.

*Figure 214: Graph of arc voltages as a function of electrode separation*
The time-averaged voltages of each arc are shown in Figure 214. The voltages are defined as the maximum value of the $\phi$ field at each cathode spot surface.

Again, this graph shows an expected effect - the voltages of each arc tend towards the same constant value as the separation between them is increased. At increased separation values, the arcs are expected to interact minimally and behave largely as two independent, identical systems - this is reflected in the results at 8 and 10cm. At lower separations, interaction between the arc columns is expected to increase, causing more turbulent behaviour in the flow and temperature field and producing asymmetries. This trend is very visible in Figure 214, with the difference between the voltages calculated on each cathode spot increasing significantly between 6 and 2cm.

**Comparison of spatial data**

The change in nature of the interaction between the two arc columns in the model as the electrode separation distance is reduced is clearly visible in the behaviour of the temperature field, shown in Figures 215 to 219.

*Figure 215: Temperature field for separation 10cm, time 10ms, scale range 5000:14000K*
Figure 216: Temperature field for separation 8cm, time 10ms, scale range 5000:14000K

Figure 217: Temperature field for separation 6cm, time 10ms, scale range 5000:14000K

Figure 218: Temperature field for separation 4cm, time 10ms, scale range 5000:14000K
As the electrode separation distance decreases, the magnetic attraction between the two arc columns increases, and they begin to bend more toward each other. At 6cm, the arcs are still (barely) two distinct entities, but by 4cm, the flow and temperature fields begin to merge along at least part of their length, and by 2cm, the arcs have joined quite near to the cathode spots to form a single, larger arc column. Interestingly, the arc tail flame caused by the change of direction as the jet encounters the anode surface is considerably more contained in the immediate vicinity of the arcs at intermediate separations - the 5000K isotherm is spread out over a large area in Figure 219 as compared to Figures 217 and 218.

The helical patterns formed by the transient motion of the jet near the cathode spots is most clearly visible in the arc structures at large separation distances. As the arcs move closer together, the interaction between them begins to affect the structure, producing more irregular shapes.

This behaviour is also apparent if we examine the 2D projected temperature profiles in the xz plane. These are shown in Figures 220 - 224.

As the arcs are placed closer together, the temperatures near to the centre of the model region in between the two electrodes rise significantly, both due to the proximity of the arcs and their increased interaction.
Figures 220 - 224: Temperature field (xz plane) for various separations, time 10ms, scale range 2000:15000K

Fig. 220: 10cm

Fig. 221: 8cm

Fig. 222: 6cm

Fig. 223: 4cm

Fig. 224: 2cm

At a separation of 2cm, the two arc jets are joined into a single arc column for most of their length, and the temperatures in the central region begin to approach those seen in single arc systems. This can be seen more clearly in Figure 225.

*Figure 225: Vertical temperature profiles at centre of region, x = 0.1m, y = 0.05m*

As the arcs begin to merge at lower separation distances, the temperature profile along the vertical centreline of the region increases in magnitude and begins to become much more uneven, indicative of the fact that a single large arc jet is now occupying the space at the centre of the region.
The increasing deflection of the arc columns toward each other can be quantified by measuring the distance between the local peak values of temperature near to the anode surface. Performing this calculation at time 10 ms gives the relationship shown in Figure 226. This shows the amount that the arc jets have moved toward each other from their original starting points at the cathode spots.

Figure 226: Distance between centrelines of arc jets near to anode vs original separation

![Graph showing the relationship between arc column separation and electrode separation.](image)

The behaviour at 2 cm is not included on this graph, since at this point the arcs have already merged together (arc column separation = 0). It is clear that bringing two arcs closer together causes them to bend toward each other increasingly strongly.

It is interesting to compare this result with the empirical model of arc deflection in twin-arc furnaces suggested by Reynolds and Jones. Their model describes the shape of the arc as a circular section in the plane defined by the two electrodes, with radius of curvature proportional to the distance between the electrodes:

\[ R_a = A_a s_e \]  

(223)

With this definition, the arc column separation at the anode surface may be calculated as shown in (224), where \( L_a \) is the distance between electrode surface and anode.

\[ a_s = s_e - 2 \left[ A_a s_e - \sqrt{A_a s_e^2 - L_a^2} \right] \]  

(224)

As the model is empirical, it requires fitting (usually from furnace data) of the proportionality
parameter $A_o$. Using this parameter to fit (224) to the data is seen to give relatively good agreement with $A_o = 2.453$. This is shown in Figure 227.

**Figure 227: Comparison of empirical deflection relationship with current model**

![Graph showing comparison between 3D plasma arc model and empirical deflection model](image)

**Temporal behaviour**

The transient behaviour of the three-dimensional models changes character considerably as the electrode separation is changed. This is shown in Figure 228 below.

**Figure 228: Evolution of temperature 5mm below cathode spot 1, $y = 0.05m$, $z = 0.045m$**

![Graph showing temperature evolution over time](image)

The behaviour of the temperature at a point close to the first cathode spot surface shows that the system at very close arc separations is experiencing far more irregular, turbulent motion. The 2cm temperature trace undergoes a number of transitions between different types of transient behaviour -
initially smooth, stable variation as the arcs develop is replaced by rapidly changing chaotic behaviour. At about 3ms this settles into quasi-periodic oscillations, but these begin to break down around 6ms and are replaced with further chaotic behaviour. By contrast, the unsteady behaviour of the equivalent temperature value at 10cm separation is limited to regular oscillations.

*Figure 229: Evolution of temperature 5mm below cathode spot 2, y = 0.05m, z = 0.045m*

If we similarly examine the temperature evolution just beneath cathode spot 2 at the end of the simulation for all arc separations (Figure 229), there is a clear procession from simple, regular oscillations at 10cm and 8cm, through to multi-modal oscillations of increasing strength at 6 and 4cm, and finally irregular oscillations at 2cm. Clearly the degree of irregularity of the motion of the arc is governed to some degree by the level of interaction between the two arc columns as they approach each other and their fields begin to couple together more tightly.

*Figure 230: Evolution of arc voltage for arc 1, y = 0.05m, z = 0.045m*
Examining the evolution of the voltage of the first arc over time reveals some additional phenomena. This is shown in Figure 230. Again, the process of regular oscillations being gradually replaced by more chaotic behaviour as the electrode separation decreases is observed. The frequency of the oscillations for electrode separations above 4cm also displays a slight tendency to increase as separation increases, even for well separated arcs.

Studying the velocity in the vicinity of the cathode spots shows similar behaviour to the arc voltage and temperature. Figure 231 shows the evolution of the z-component of the velocity vector at a location 5mm below the first cathode spot, during the last 1ms of the simulation.

Figure 231: Evolution of $v_z$ 5mm below cathode spot 1, $y = 0.05m$, $z = 0.045m$

The velocities display distinctly exponential behaviour with regard to increasing electrode separation, tending towards identical regular oscillations as the separation grows large. Deviations from this behaviour occur most significantly at 4cm (increase in magnitude of regular oscillations) and 2cm (breakdown into turbulent, irregular motion).

The early evolution of the $x$-component of the velocity in the vicinity of the cathode spot is shown in Figure 232, and demonstrates how the increased arc deflection behaviour at smaller separations is driven by the jet velocities.
The initial tangential velocity in the root of the arc near to the cathode spot increases rapidly as the separation distance is decreased - this gives rise to a more angled arc jet, and produces the characteristic deflection of the arc columns toward each other. Although less obvious in the later parts of the transient traces, this tendency persists during the unsteady state regime of the fluid flow, maintaining the arcs' deflection.

6.4 Arc current asymmetry effects

The aim of this set of results is to examine the impact of changing the electrical symmetry of the 3D plasma arc model. In real furnaces, the currents supplied to each electrode in a multiple-arc system are seldom exactly equal at any given time, and may remain significantly out of balance for a long time relative to the speed of the arc dynamics. The effects of such asymmetry on the behaviour of the model are thus of some interest.

The model parameters used are again identical to those in Table 19, with the exception of the current supplied to the first cathode spot, which is changed according to Table 22. This affects the furnace geometry by changing the diameter of the cathode spot boundary, \( \Gamma_{N,1} \).

The models were run on an Intel Core2 Q6600 machine with four CPUs and 2GB RAM, and required between 4090 and 4239 time step calculations at an average time step size of between 2.3 and 2.4 \( \mu \)s. Each model run took between 16.4 and 17.0 hours to complete.
Table 22: Range of variables tested for current asymmetry

<table>
<thead>
<tr>
<th>Current, cathode spot 1</th>
<th>Diameter of $\Gamma_{N,1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>50 A</td>
<td>0.00135 m</td>
</tr>
<tr>
<td>100 A</td>
<td>0.00191 m</td>
</tr>
<tr>
<td>150 A</td>
<td>0.00234 m</td>
</tr>
<tr>
<td>200 A</td>
<td>0.00270 m</td>
</tr>
<tr>
<td>250 A</td>
<td>0.00302 m</td>
</tr>
</tbody>
</table>

**Average and instantaneous peak values**

As for the previous cases examined, time-averaged and instantaneous maximum values of velocity, temperature, and arc voltage (for each arc) are calculated. These values are compared over the range of currents tested, with the results shown in Figures 233 to 235.

*Figure 233: Graph of peak velocity magnitude as a function of current at cathode spot 1*

*Figure 234: Graph of peak temperature as a function of current at cathode spot 1*

The variation of peak velocity magnitude and temperature with changing current is fairly random. This is to be expected in this case, as while the peak values for lower currents would be expected to be lower, the presence of the second arc at 250A current for all tests masks any discrepancies.

Trends are more clearly visible when the arc voltage of each arc is considered, as shown in Figure 235.
The voltage calculated at the first cathode spot is distinctly lower at lower currents, while the voltages recorded at the second cathode spot (constant current) remain approximately constant. This is expected - although the arc is a non-Ohmic conductor it does display some dependence of voltage on current, and the voltage of the first arc is expected to drop as its current is decreased.

*Figure 235: Graph of arc voltages as a function of current at cathode spot 1*

**Comparison of spatial data**

The behaviour of the individual arc columns and their interaction alters significantly as the asymmetry between the currents carried by each is increased, as can be seen in the temperature profiles shown in Figures 236 - 240.

*Figure 236: Temperature field for currents 50A and 250A, time 10ms, scale range 5000:14000K*
Figure 237: Temperature field for currents 100A and 250A, time 10ms, scale range 5000:14000K

Figure 238: Temperature field for currents 150A and 250A, time 10ms, scale range 5000:14000K

Figure 239: Temperature field for currents 200A and 250A, time 10ms, scale range 5000:14000K
Several interesting phenomena are seen as the current at the first cathode spot is changed. Firstly, the magnetic attraction causes the first arc's deflection toward the second to increase as its current decreases. At 50A, the first arc is drawn across the region to join with the second in an almost horizontal fashion. At 100A and 150A the first arc's jet is angled more steeply toward the lower surface, although it is still completely merging with the second before they encounter the anode. As 250A and symmetry is approached, the arcs' interactions begin to become equal and maintain more distinct columns all the way to the anode surface.

As the current supplied to the first arc increases, so does the intensity of the arc jet. The temperatures in the jet increase, as do the velocities. Between 100A and 150A there appears to be a transition from a steady state, stable arc jet into the transient helical motion observed previously. The transient behaviour strengthens as the current is raised further.

In the simulation at 200A, a phenomenon familiar from the two-dimensional modelling results is observed - the formation of mobile parasitic arcs at the anode surface. This behaviour translates to the three-dimensional model, although it appears that the conditions for initial formation of these arcs are more demanding in 3D (at least at the lower currents tested), as they are only observed in one out of the nine cases studied so far. It is interesting to note that once the anode arc is established in the 200A case it appears quite stable and persistent, suggesting that the appearance of parasitic arcs in the other models is only a matter of time.

Examining the 2D projected temperature profiles shows much of the same behaviour - these are shown in Figures 241 - 245.
Figures 241 - 245: Temperature field (xz plane) for various currents, time 10ms, scale range 2000:15000K

**Fig. 241:** currents 50A and 250A

**Fig. 242:** currents 100A and 250A

**Fig. 243:** currents 150A and 250A

**Fig. 244:** currents 200A and 250A

**Fig. 245:** both currents 250A

The transition between steady and unsteady behaviour of the first arc in Figures 242 and 243 is visible, as are the quite different flow and temperature patterns associated with the parasitic arc system in Figure 244.

The increase in arc intensity can also be seen in the vertical velocity profiles through the region from the centre of the first cathode spot down to the anode surface below, shown in Figure 246.

*Figure 246: Vertical profile of \( v_z \) beneath cathode spot 1, \( x = 0.08m, y = 0.05m \)*

The scaling of velocity with the current applied to the first arc is clearly visible, particularly in the
bulk of the region further from the boundaries. There is some anomalous behaviour of the profile at 200A, however recall that this is the simulation which produced parasitic anode arcs, and these may be expected to interfere with the flow profiles to some degree. In the vicinity of the cathode spot at $z = 0.05m$, the velocities peak at the constricted root of the arc. The transition to unsteady flow above 100A is seen to cause considerable oscillation and irregularity of the velocity profile.

**Temporal behaviour**

The changing current applied to the first electrode/arc has a marked effect on the transient nature of the variable fields. The difference is most pronounced close to the first cathode spot, and is shown in Figure 247.

![Figure 247: Evolution of temperature field near cathode spot 1, $x = 0.08m, y = 0.05m, z = 0.045m$](image)

The onset of instability in the first arc occurs later and later in the simulation as the current decreases from 250A, until at below 150A this arc remains stable for the duration of the model time. As current informs both the dimensions of the cathode spot and the electrical and magnetic field behaviour close to it, the source terms for the flow and temperature fields are largely controlled by the current of the arc alone. The value of the source terms near to the cathode spot have been seen to be of considerable importance in driving the unsteady dynamics of the arc column, and thus this dependence of transient behaviour on current flow is not unexpected.

In Figure 248, the last 1ms of model evolution time shows the oscillatory behaviour of the temperature field near to cathode spot 1.
As can be seen, the models at 50A and 100A remain very stable in time even very late in the simulation, whereas regular oscillatory behaviour occurs for higher currents. It is interesting to note that as the current flow to the first arc increases the period of the oscillations grows shorter, indicating a more rapid precession of the arc jet around the surface of the cathode spot.

The different character of the temporal behaviour of the two arcs in an asymmetric condition is most obvious in the 50A model. The arc voltage traces for each arc in the simulation are plotted in Figure 249.

The rapid rise in voltages as the flow and temperature fields establish themselves already show
distinct asymmetry between the two arcs. This is then further exacerbated by the onset of instability in the second arc, distinct at 1.5ms.

6.5 High resolution cases - triple arc

The aim of the high resolution cases is to provide examples of the type of systems that can be effectively studied with the DC plasma arc model using more powerful computer hardware. This case was run on an IBM P690 machine with 32 processors and 32 GB of memory, required 3000 time step calculations at an average time step size of 1.7 µs, and took approximately 2 days to complete. Modern high performance SMP computers should be very capable of performing these and larger simulations in reasonable time.

The triple arc system studied here is a hypothetical one. Three electrode furnaces with the electrodes positioned equidistantly apart forming an equilateral triangle do exist in industrial applications, however, they invariably use alternating rather than direct current, and seldom operate with the arcs exposed to the atmosphere in the furnace. Having said this, there is a natural progression from large twin electrode DC furnaces to adding further electrodes when necessary, and so the scenario is not solely of academic interest.

The parameters used in the model are identical to those shown in Table 19, with the changes shown in Table 23.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Region length, $x_{AB}$</td>
<td>0.2 m</td>
</tr>
<tr>
<td>Region width, $y_{AD}$</td>
<td>0.2 m</td>
</tr>
<tr>
<td>Region height, $z_{AE}$</td>
<td>0.05 m</td>
</tr>
<tr>
<td>Number of arcs</td>
<td>3</td>
</tr>
<tr>
<td>Current, arc 1</td>
<td>250 A</td>
</tr>
<tr>
<td>Current, arc 2</td>
<td>250 A</td>
</tr>
<tr>
<td>Current, arc 3</td>
<td>250 A</td>
</tr>
<tr>
<td>$x_{e,1}$, $y_{e,1}$</td>
<td>0.05 m, 0.0711 m</td>
</tr>
<tr>
<td>$x_{e,2}$, $y_{e,2}$</td>
<td>0.15 m, 0.0711 m</td>
</tr>
<tr>
<td>Parameter, cont'd</td>
<td>Value, cont'd</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>$x_{c3}, y_{c3}$</td>
<td>0.1 m, 0.158 m</td>
</tr>
<tr>
<td>Grid resolution, $J \times J \times K$</td>
<td>512 x 512 x 128</td>
</tr>
<tr>
<td>Number of threads</td>
<td>32</td>
</tr>
<tr>
<td>Maximum model time</td>
<td>5 ms</td>
</tr>
</tbody>
</table>

**Average and instantaneous peak values**

As for the previous cases examined, the time-averaged and instantaneous peak values of velocity, temperature, and arc voltage (for each of the three arcs) are calculated. For this high resolution case, the time-averaged value is calculated over the last half of the simulation, from 2.5ms to 5ms.

As before, the arc voltage is defined as the maximum value of the $\phi$ field on the cathode spot surface related to each arc.

*Table 24: Average and instantaneous peak field values*

<table>
<thead>
<tr>
<th>Variable</th>
<th>Average value</th>
<th>Value at 5 ms</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v_x, v_{y}, v_z$</td>
<td>257.3 m/s</td>
<td>272.9 m/s</td>
</tr>
<tr>
<td>$Re$</td>
<td>2553</td>
<td>2707</td>
</tr>
<tr>
<td>$T$</td>
<td>15992 K</td>
<td>16232 K</td>
</tr>
<tr>
<td>Arc voltage, arc 1</td>
<td>186.21 V</td>
<td>196.49 V</td>
</tr>
<tr>
<td>Arc voltage, arc 2</td>
<td>187.61 V</td>
<td>194.04 V</td>
</tr>
<tr>
<td>Arc voltage, arc 3</td>
<td>189.23 V</td>
<td>195.85 V</td>
</tr>
</tbody>
</table>

The values are presented in Table 24, and are seen to be of the same order as those seen in the earlier 3D simulations. This is due to the fact that the same current is supplied to each individual arc in the triple arc simulation, and the electrodes are spaced well enough apart that each arc is largely independent of the others. We can therefore expect the behaviour to be similar to the various twin-arc cases considered earlier.

227
Comparison of spatial data

The temperature fields in Figures 250 to 254 show the evolution of the arc shapes and interactions.

*Figure 250: Temperature field, triple arc system, time 1 ms, scale range 6000:14000K*

![Temperature field, triple arc system, time 1 ms, scale range 6000:14000K](image1)

*Figure 251: Temperature field, triple arc system, time 2 ms, scale range 6000:14000K*

![Temperature field, triple arc system, time 2 ms, scale range 6000:14000K](image2)

*Figure 252: Temperature field, triple arc system, time 4 ms, scale range 6000:14000K*

![Temperature field, triple arc system, time 4 ms, scale range 6000:14000K](image3)
Much as for the twin arc systems, the triple arc model shows various phenomena relating to unsteady behaviour and magnetic interactions.

With three arcs of similar current originating on electrodes spaced equally far apart, we expect the arc columns to be attracted toward their common centre point. It can be seen in the top view of the temperature field at 5ms, Figure 254, that this is indeed happening.

As with the twin arc models, the breakdown from stable arc columns into transient helical shapes occurs at between 1 and 2 ms simulation time. In this model, the arcs are sufficiently far apart that they do not interact appreciably apart from being deflected toward each other.
Parasitic anode arcs form near the base of each arc between 3 and 4ms. They are fairly short-lived, and are seen to be rapidly pushed towards the centre of the region by the angled jets of the main arc columns before they can become fully established. It is possible that if the electrodes are spaced closer together, conditions at the centre of the region would favour the formation of a single strong anode arc connecting electrically with the three primary arcs.

The interaction of the arc jets creates a complex three-dimensional recirculation pattern in the centre of the region which can be seen in a plot of the developing velocity field, as shown in Figures 255 and 256.

*Figure 255: Velocity field for triple arc system, time 5 ms, scale range 10:250 m/s*

![Velocity field for triple arc system, time 5 ms, scale range 10:250 m/s](image)

*Figure 256: Velocity field for triple arc system, time 5 ms, scale range 10:250 m/s*

![Velocity field for triple arc system, time 5 ms, scale range 10:250 m/s](image)
It can be seen that the peak velocities are very localised, and occur only in the immediate vicinity of each arc column. The velocity streamlines show that the interaction of the three arcs produces a upflow pattern in the central part of the region. The evolution of this central flow is shown in Figure 257.

The flowrate of the central upwelling grows rapidly with time as the flow profiles of the three arcs develop and stabilise. At 5ms it still appears to be rising exponentially, indicating that the simulation has not reached a fully-developed state. A simulation to determine the longer-term behaviour of the flow profile would be of some value in this case.

*Figure 257: Development of $v_z$ along vertical centreline of region over time, $x = 0.1m$, $y = 0.1m$*

**Temporal behaviour**

The evolution of the arc voltages in time is shown in Figure 258.

As with the earlier models, voltage of each arc rises rapidly as the region cools from the initial conditions and the arc shapes begin to be established. Regular oscillations begin at 1.5ms, as the arcs make the transition to helical-precession transient behaviour.
Further interesting trends are visible in the behaviour of the temperatures near to the cathode spots, as shown in Figures 259 and 260.

The general behaviour of the second arc at these sample points is quite different to that of the other two - the initial temperatures calculated do not rise as high, and the amplitude of the oscillations in temperature that it exhibits are considerably larger. This may be an artefact of imprecise sample point location, as one would expect (and this is supported qualitatively by the spatial data) the arcs to behave very similarly, at least during the initial period of arc formation.

The onset of transient motion in this system is clearly visible at 1.5ms. It is also interesting to note that the impact of the unsteady behaviour on the absolute value of the temperature at this location is
significant - as the arc columns begins to move and rotate, the arcs become much more diffuse and spread out over a larger area, lowering the average values of temperature and velocity.

*Figure 260: Evolution of temperature field, each arc sampled 5mm below the related cathode spot location*

![Graph showing temperature evolution](image)

The last 1ms of the evolution of the temperatures near to the root of each arc reveals an additional fact - the regular oscillations shown by each arc are approximately 120° out of phase with each other. This supports a similar observation made in the twin-arc cases, where the oscillations of each arc in a two arc system also tend to be completely out of phase. Out-of-phase motion would seem to be a stable periodic behaviour for multiple arc systems.

### 6.6 High resolution cases - twin arc

The second of the high resolution cases returns to the twin arc system considered earlier. In this case, “high resolution” refers to the increase in the size of the numerical grid, doubling the resolution in each direction and increasing the computational and memory loads by approximately a factor of 8. The case was run on an IBM P690 machine with 32 processors and 32 GB of memory, required 8806 time step calculations at an average time step size of 0.53 μs, and took approximately 4 days to complete.

At higher resolution, higher arc currents (and hence higher velocities and temperatures) can be used in the three-dimensional DC plasma arc model without causing numerical instability. This is demonstrated by increasing the current applied at each cathode spot boundary to 500A, to enable useful comparisons between both the twin arc 3D models and the single arc 2D models considered
earlier.

The parameters used in the model are identical to those shown in Table 19, with the exceptions shown in Table 25.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current, arc 1</td>
<td>500 A</td>
</tr>
<tr>
<td>Current, arc 2</td>
<td>500 A</td>
</tr>
<tr>
<td>$x_{e,1}, y_{e,1}$</td>
<td>0.05 m, 0.05 m</td>
</tr>
<tr>
<td>$x_{e,2}, y_{e,2}$</td>
<td>0.15 m, 0.05 m</td>
</tr>
<tr>
<td>Grid resolution, $I \times J \times K$</td>
<td>768 x 384 x 192</td>
</tr>
<tr>
<td>Number of threads</td>
<td>32</td>
</tr>
<tr>
<td>Maximum model time</td>
<td>4.8 ms</td>
</tr>
</tbody>
</table>

**Table 25: Model parameters for high resolution twin arc case**

Average and instantaneous peak values

The time-averaged and instantaneous maximum values of velocity, temperature, and arc voltages are calculated in the usual fashion. For this high resolution case, the time-averaged value is calculated over the all times after 2.5ms, and the peak value is recorded at the end of the simulation, at 4.8ms. The results are shown in Table 26.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Average value</th>
<th>Value at 4.8 ms</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v_x, v_y, v_z$</td>
<td>534.4 m/s</td>
<td>549.8 m/s</td>
</tr>
<tr>
<td>$Re$</td>
<td>5300</td>
<td>5454</td>
</tr>
<tr>
<td>$T$</td>
<td>21423 K</td>
<td>21746 K</td>
</tr>
<tr>
<td>Arc voltage, arc 1</td>
<td>222.75 V</td>
<td>224.65 V</td>
</tr>
<tr>
<td>Arc voltage, arc 2</td>
<td>229.57 V</td>
<td>244.56 V</td>
</tr>
</tbody>
</table>

**Table 26: Average and instantaneous peak field values**

It is interesting to compare these values with those for the base case model in the two-dimensional
simulation results in Table 8 - although the latter models only a single arc and simulates twice as long a period of time, the current per arc and various other parameters are similar.

The velocities are of similar order of magnitude, although somewhat elevated in the 2D model. The change in velocities that occurs as a result of the current increase from 250A and 500A in the three-dimensional twin-arc cases is approximately proportional to current - this behaviour was also observed in the 2D models.

The peak temperatures are considerably higher in the high resolution twin arc model as compared to those generated by the two-dimensional models and lower-current three-dimensional models, however, it is quite likely that given the short time this simulation was run for it may not yet have had a chance to come fully into equilibrium.

The magnitude of the arc voltages compare well with the peak values of $\phi$ measured in the 2D cases, and are somewhat higher than the values calculated for the twin arc system at lower currents. This is consistent with the trends observed in the 2D cases.

**Comparison of spatial data**

The flow fields in the high resolution twin arc model are considerably more turbulent. This is principally due to the higher currents input to each cathode spot and arc, creating higher velocities and causing a transition toward more chaotic behaviour of the fields. The temperature field at various times is shown in Figures 261 to 265.

*Figure 261: Temperature field, high resolution twin arc system, time 1ms, scale range 6000:14000K*
Figure 262: Temperature field, high resolution twin arc system, time 2ms, scale range 6000:14000K

![Figure 262](image)

Figure 263: Temperature field, high resolution twin arc system, time 4ms, scale range 6000:14000K

![Figure 263](image)

Figure 264: Temperature field, high resolution twin arc system, time 4.8ms, scale range 6000:14000K

![Figure 264](image)
The onset of transient motion occurs earlier in the 500A twin arc case than it does in simulations at lower arc currents; by 1ms model time, the flow is already in unsteady state motion, with the arc columns beginning to twist violently.

Parasitic anode arc formation is much stronger at the higher currents used for this simulation. Multiple parasitic arc jets, up to five at a time, are seen to develop on the outskirts of the main arc jets as they contact the anode surface. These secondary arcs are again highly mobile and interact with each other to a great degree, much as was observed in many of the 2D case results.

There is some evidence of arc deflection toward the centre point between the two electrodes, however, the increased turbulence of the fluid flow causes the arc columns to be more spread out in space at 500A each, and the magnetic interactions are largely masked by local oscillations in the flow fields around each arc. This can be seen quite clearly when the temperature values along a line through the middle of the furnace are plotted as shown in Figure 266.

It is instructive to compare this graph with that shown in Figure 203, for lower-current arcs. The increased irregularity of the profiles here is obvious, with only rough trends rather than distinct peaks serving to indicate the position of the arc columns.
The turbulent behaviour of the fluid flow in and around the arc columns is also visible in the velocity fields, shown in Figures 267 and 268.
As with the previous three-dimensional cases, it is seen that the high velocities are tightly contained within the space occupied by each arc column. The behaviour of the streamlines is now considerably less smooth however, with a high degree of tangling being caused by the turbulent flow and interactions with the parasitic arc jets.

**Temporal behaviour**

The time-dependent variation of the fields in the high resolution twin arc model is of some interest, as the behaviour of the arc system appears qualitatively more disordered in the results so far.

*Figure 269: Evolution of temperature values, y = 0.05m, z = 0.045m*

*Figure 270: Evolution of v, velocity, y = 0.05m, z = 0.045m*

As can be seen from Figures 269 and 270, it is clear that the velocity and temperature fields are undergoing far more erratic motion as compared with the twin-arc models at lower currents considered earlier. The chaotic oscillations of the field variables are reminiscent of similar behaviour observed in the two-dimensional single arc models.

Examining the temperatures close to the cathode spots over the first 1ms of simulation time (Figure 271) shows that the onset of transient behaviour for both arcs in the high resolution twin arc model occurs at close to 0.6ms. This is in remarkably good agreement with the two-dimensional single-arc models, which predict the breakdown of the initial stability to occur at a similar time for similar individual arc currents. This suggests that the dynamics of the DC plasma arc system are being captured reasonably well by the simpler 2D models.

In Figure 272, toward the end of the simulation, the patterns displayed by the temperature fields in
the vicinity of each cathode spot show quite clearly the more disorganised, irregular behaviour that is occurring.

Figure 271: Early evolution of temperature field, 
\[ y = 0.05m, z = 0.045m \]

Figure 272: Established transient behaviour, 
temperature field, \( y = 0.05m, z = 0.045m \)

It is interesting to compare these results with the temporal behaviour of both the two- and three-dimensional base case models - at increased currents, more turbulent motion of the arc column is definitely observed, and this behaviour seems to originate in the intense region close to the root of the arc on the electrode.

For the three-dimensional models, a distinct transition in behaviour from steady state, through to regular oscillatory motion, and finally to more disorganised, chaotic motion is seen to occur as the current flow to the arc or arcs in the model is raised.
Chapter 7 - Discussion of results, conclusions and recommendations

Model development

The development of a mathematical and numerical model to describe the coupled multi-physics problem of the direct-current plasma arc as found in modern DC arc furnaces has been largely successful.

The mathematical formulation of the fundamental physical processes occurring in the DC plasma arc has been presented. This formulation incorporates sub-models of various transport and field phenomena of relevance to the DC plasma arc problem, including fluid flow, thermal energy transfer, and electromagnetism. As the fluid dynamics is arguably the most difficult aspect of the model, a variety of alternative formulations of the Navier Stokes equations were presented and discussed in terms of their ease of use and generality. Aspects of the various coupling effects between sub-models were identified and discussed, demonstrating how they act in concert to form and sustain the plasma arc jet.

Numerical approximations to the mathematical model were then discussed. The aims of designing a model suitable for operation at high spatial resolutions as well as focusing on the transient behaviour of the DC plasma arc system were used to guide the choice of the numerical methods used. Finite difference schemes utilising a variety of staggered and non-staggered grid geometries were used to convert the fundamental partial differential equations governing the system into discrete versions suitable for computational solution.

Recent advances in methods for transient equations using explicit Runge-Kutta algorithms to advance the discrete equations through time were applied to the convection-diffusion parts of the model in order to improve the numerical stability. These were combined with operator splitting and implicit methods where additional stability properties were desirable. Overall the numerical model preserves approximately second-order accuracy in space and at least first-order accuracy in time.

Various parts of the model require the solution of elliptic equations at every time step (or Runge-Kutta step). These solutions were performed using direct spectral transform methods where the set of difference equations has constant coefficients, and a generic multigrid solver where they do not.
Aspects of the computational algorithms used for the model and their effective treatment on parallel computers were then presented. Arguments for and against distributed- and shared-memory parallelism were made, with SMP being selected as the parallelisation model of choice. Handling of the enormous amounts of data generated by the models was addressed using various sampling schemes to capture only the data of interest for a given run. Finally, a brief discussion of various algorithms used for visual presentation of the data in 2D and 3D was presented.

Overall there are several novel points associated with the development and implementation of the DC plasma arc model in this work. The first is the particular combination of formulations and methods that were chosen to make up the model. The second is the use of cartesian grids and approximated magnetic fields for modelling of the dynamics of the system in two dimensions. A third is the strong focus on unsteady-state equations and methods to capture the dynamics and qualitative behaviour of the DC plasma arc system on its characteristic very short time scales.

**Two-dimensional modelling results**

The two-dimensional modelling results began with a set of test cases, designed to study various aspects of the sub-models comprising the DC plasma arc model.

In the first pair of tests, the evolution of both steady and unsteady state jet flow problems were considered. In the steady state case, it was seen that the solutions given by all three Navier Stokes formulations discussed in the development phase of the model were in reasonable agreement with one another both quantitatively and qualitatively. The spatial accuracy of the gauge formulation was evaluated and confirmed to be of second order or higher. The unsteady state test demonstrated some clear differences between the stability of the methods as well as the quality of the solution they produced, with the gauge method appearing to be the best option.

The following test focused on the performance of the two-dimensional magnetic and electric field solvers. A series of error comparisons between model and analytical solution was performed to evaluate an optimal choice for the effective domain thickness, an empirical variable needed for the approximate magnetic field solver, in relation to the width of the current path at the cathode spot boundary. This was found to be twice the radius of the cathode spot diameter for a given current. The convergence rate of the multigrid solver used for the electric potential field solution was then
examined, and a number of different algorithm design parameters were investigated to determine their effectiveness. The optimised multigrid algorithm was shown to have an acceptably rapid convergence rate for the kind of problems arising in the DC plasma arc model.

The final test case examined the performance of the full DC plasma arc model with regards to parallelisation. The thread overhead for the model was studied on a single-processor computer system, and the overall speed increase of the model on multiprocessor systems was presented. It was seen that for relatively low numbers of processors the gain in performance is appreciable, although there is likely to be a break-even point beyond which adding more processors to a given problem may affect the performance detrimentally.

The two-dimensional DC plasma arc modelling effort was focused on the study of the dynamics and behaviour of single arc systems. A base case set of parameters was chosen to be representative of a small to medium scale DC arc furnace pilot plant, and the model run to produce time-dependent data describing the velocity, temperature, electrical, and magnetic fields. Representative values of the field data were calculated and presented, showing the transport phenomena in the arc to be occurring at high velocities and temperatures.

The dynamic, asymmetric, turbulent nature of the jet flow in and around the arc was demonstrated with visualisations of the various fields computed by the model. Several transient phenomena were observed, including the initial development of a stable, symmetric arc jet and its subsequent breakdown into transient motion, and the spontaneous evolution of parasitic arc jets on the anode surface. The latter have important implications for the mechanisms of heat transfer to the molten bath in a metallurgical furnace - in traditional axisymmetric arc models, the heat transfer is shown to be quite uniform and smooth over the anode surface, whereas this work indicates that localised and highly mobile “hot spots” can form on the anode in the vicinity of the roots of the parasitic arcs. These hot spots can result excessive vaporisation of the molten material in the furnace, and the process chemistry (particularly in the gas phase) may be affected as a result.

The arc column temperatures were seen to exhibit high values of the order of 15000K or higher near to the cathode spot, dropping to lower values of the order of 9000K in the bulk jet flow further from the boundaries, with large changes possible over relatively short periods of time (of the order of 0.1ms). This very short time scale has implications for the design of the power control electronics used in modern DC arc furnace power supplies - the electrical measurements and control actions
may need to operate at greater (possibly much greater, for high-current arcs) than 10kHz in order to perform optimally. Spatially, the electric potential was observed to obey approximately linear behaviour over the length of the arc column.

Comparison of the base case model with a variety of empirical estimates of the velocity, temperature, and electrical properties of the DC plasma arc found a reasonable degree of qualitative and quantitative agreement. Likewise, a comparison between experimental photographs of arc behaviour and the model results showed a number of compelling qualitative similarities, particularly as regards the turbulent and unsteady nature of the fluid flow and temperature profiles, and the asymmetrical shapes attained by the arc column as it evolves in time.

Following the analysis of the base case model several of the parameters in the model were varied relative to their base case value, and their effects on the resulting behaviour studied.

The first set of results examined changes in the geometry of the region in which the arc operates, by altering the distance from the anode surface to the electrode tip. This was seen to have a significant impact on the Reynolds number of the arc jet flow as well as the total voltage produced by the arc, both of which increased significantly with region height. Examining images of the temperature field showed that at lower region heights, the arc column became more compact and localised, and the degree of turbulence in the region decreased. Most striking was the change in transient behaviour of the fields in the model, exhibiting regular oscillations at lower region heights and moving to increasingly irregular, chaotic behaviour as the the height was increased. This transition suggests that there is a change in modes between so-called “brush arc” operation, in which the electrode is positioned very close to the anode surface, and the more usual “open arc” operation, in which the electrode is positioned well away from the anode. Brush arc operation in DC arc furnaces is traditionally associated with more stable operation and is often used during recovery phases - these modelling results provide some explanation as to why this may be the case.

The electric current supplied to the model was the next parameter to be varied. Larger currents produced a distinct trend in the peak velocities of the model, increasing almost linearly. Study of the initial transient behaviour of the velocity field also indicated a strong, increasing dependence of local velocities in the arc column on current. This produced a corresponding increase in the degree of turbulent mixing of the fluid flow, which was confirmed qualitatively by examining visualisations of the temperature and vorticity fields at various current levels. This has important
implications for the design of DC arc furnaces, as scale-up calculations for the current are generally highly non-linear and poorly understood in terms of their effects on the fluid flow in both the freeboard gas space and the molten bath. In particular, the direct observation of the increase of turbulence and the change in time scale of the transient arc behaviour with the increase in current are original in regard to earlier modelling work.

The effect produced by altering the cathode spot current density was then examined. Less a design variable and more a physical property of a particular electrode material, the range of values for this parameter as reported in other literature is large, and it is therefore a source of some uncertainty in the DC plasma arc model. Higher values of current density were seen to give increased peak temperatures and velocities in the arc system, generally corresponding to a more localised, intense region close to the root of the arc at the cathode spot. This was confirmed in the transient behaviour of the local temperature values close to the cathode spot boundary. Images of the temperature profile for various current densities also confirmed this, with the arc growing clearly more compact and the turbulence of the region greater at higher current densities. Several unusual and interesting transient behaviour patterns were observed in these models close to the theoretical value for graphite, 3 - 4 kA/cm², including reversed arc jets and the formation of very large unstable current loops. This serves to indicate that traditional axisymmetric or even three-dimensional models, if performed at steady-state, are greatly limited. Unsteady calculations as performed here are of great importance in order to fully capture the range of dynamic behaviour possible in the arc system.

A brief sensitivity analysis of the model to various constant physical property parameters was conducted next. Variation of these properties is of interest primarily because (although treated as constant in the model) they are in general functions of both temperature and the composition of the plasma gas. Fluid viscosity, density, and thermal conductivity were found to have the largest effect on peak velocities in the simulation, with viscosity having the most pronounced effect on the Reynolds number of the arc jet flow. Maximum temperatures in the arc column were most significantly affected by variations in thermal conductivity and heat capacity. The voltages generated by the arc in the model were seen to be fairly insensitive to changes in any of the properties, with maximum variations within 10% of the base case value.

Following on from the examination of physical properties, a practical case of this was studied by changing the constituent gas of the plasma in the model from air to argon. This additionally affected the dependence of electrical conductivity and thermal radiation on temperature. Argon was
observed to decrease the peak velocities in the arc jet, but also raise the Reynolds number of the flow. The temperatures calculated by the model were quite similar, however, the voltage of the arc was nearly 100V lower in the case of argon. Qualitative data was presented in the form of temperature visualisations for the two different gases, which showed that the arc column formed in argon was considerably wider and the average temperature along its length was higher, both contributing to the large decrease in voltage seen. This has implications for advanced process design, control, and safety of DC arc furnaces, as the partial or complete replacement of the atmosphere in the furnace freeboard with an alternative gas may be able to affect useful changes in the electrical behaviour of the arc and furnace as a whole.

The last set of parameters examined for the two-dimensional plasma arc cases was the boundary conditions specified for the temperature field in the model. The constant temperature (TBC 1) boundary conditions are largely process and material dependent, and can potentially take on different values accordingly. Their influence on the model was examined by additionally examining the case of thermally-insulating boundary conditions (TBC 2) at all surfaces in the model. The latter had a marked impact on the peak values of velocity and the arc voltage, lowering both by a significant amount. Of the constant temperature cases, varying the temperature at the wall surfaces was seen to have the least impact on the model. The temperature at the anode and electrode surfaces were both seen to decrease both peak velocity and temperature as they increased - this has implications on the process design for DC arc furnaces, as the materials used for both the electrode and the molten bath anode will determine these temperatures and hence (to some degree) the level of turbulence in the arc jet. A qualitative comparison between the temperature and vorticity profiles of the TBC 1 and TBC 2 cases showed that insulating boundary conditions produced a more symmetric and less turbulent arc jet, with a lower spread of temperatures over the region.

Qualitatively, behaviour of the temperature profiles close to the cathode spot as well as the electric potential dependence on the length of the arc were both seen to be quite different. For the TBC 1 case, it was shown using temperature profiles perpendicular to the cathode spot that small changes at the boundary could grow into considerably larger changes in the bulk region, lending credence to the hypothesis that the model is quite sensitive to the values used for boundary conditions in certain important areas. Finally, an examination of the transient behaviour of the arc voltages and local temperatures over the duration of the model run showed that while significant differences were present between the TBC 1 and TBC 2 cases, irregular oscillatory behaviour on similar time scales was still present in both. This suggests that the short-term dynamics of the arc system is an inherent feature of the fundamental equations and not merely a boundary condition artefact.
Three-dimensional modelling results

The three-dimensional DC plasma arc modelling work focused on the dynamics and behaviour of multiple arc systems. As with the 2D work, a base set of parameters was selected first selected, representing a small scale twin arc furnace. The base case model was run to produce time-dependent data of velocity, temperature, and electromagnetic fields describing the evolution of the twin arc system in three dimensions. Maximum values of velocity, temperature, and arc voltages were calculated and compared favourably to earlier simulations. Analysis of imagery of the three-dimensional velocity and temperature fields showed that the arc jets initially manifest as straight, steady cylindrical columns, but instabilities near the cathode spot lead first to oscillations, and ultimately precession of the arc jet around the cathode spot, creating a characteristic transient spiral or helical structure. While such behaviour has previously been postulated from experimental work and empirical models, this is an original result for numerical modelling studies of the DC plasma arc - this phenomenon does not appear in any steady-state or axisymmetric models to date, and appears to be a stable and persistent mode of operation for arcs at intermediate currents.

The two arc columns were also observed to attract and interact with each other, as expected from electromagnetic theory. The results of the time dependence of the local temperature close to the cathode spots and the arc voltages for each arc were presented, and revealed that the nature of the transient behaviour for the lower-current arcs used in the 3D base case model was very regular when compared to the 2D modelling results, with the oscillations at each arc tending to be out of phase with each other. Time scales observed were similar to the 2D models, with the period of the oscillations being of the order of 0.18 - 0.2 ms. A qualitative comparison was carried out between photographic evidence of twin arc behaviour at similar electrical and dimensional parameters to those used in the model, and a number of similarities regarding the temperature profiles of the arcs and the magnetic attraction between them were observed.

Following the detailed analysis of the model results using the base case parameters, several of the parameters were varied to examine their effects on the dynamic behaviour produced.

The separation distance between the electrodes (and therefore arcs) was the first to be considered. Reducing the electrode separation produced noticeable effects on the peak values of the velocity field, and the voltage measured at each arc, with the values indicating that at larger separations, the
interactions between the arc columns were less pronounced. This was confirmed qualitatively and quantitatively by examining visualisations of the three-dimensional temperature field at varying degrees of separation, and using the local maxima in the field to determine the degree of deflection of the arc columns toward each other. This was observed to increase steadily as electrode separation was decreased, until the arcs began to merge together into a single larger column. A comparison between the arc deflections measured in the model results and those predicted by a simple empirical model was conducted, and showed reasonable agreement in the behaviour. More irregular behaviour of the helical structure of the arcs was observed as the arcs grew closer together and their interaction increased, supported by evidence of more chaotic behaviour at close separation distances seen in the transient behaviour of local temperatures close to the cathode spot surfaces. Study of the evolution of the arc voltages over time also revealed a slight tendency for the frequency of the regular oscillations exhibited by the models at larger electrode separations to increase with electrode separation.

The nature of the interaction and turbulence of the arc columns at varying electrode separations has several design and operation implications for DC arc furnaces using twin electrodes. The introduction of a second electrode and arc breaks the approximate axisymmetry of the furnace vessel and flow and temperature fields - as a result, optimal placing of the feed ports in the roof of the furnace, as well as design of the cooling elements in different parts of the furnace sidewalls and roof may be affected by the choice of electrode separation. Identification of the point at which the two arc columns join before meeting the anode surface is also important design information, and is potentially a desirable feature to generate a high-intensity heat transfer zone on the anode surface. Knowledge of the relationship between arc proximity and the degree and nature of the resulting unstable behaviour is valuable for the design of the furnace's electrical control systems, since greater instability and interaction between the arcs poses more difficulties for the power control electronics and can potentially lead to increased process downtime.

The effect of altering the degree of symmetry between the two currents imposed at each electrode was then examined. Little effect was seen when examining the maximum values of the velocity and temperature fields, although asymmetric behaviour of the arc voltages for each arc was observed as the disparity between the currents increased. Examining the visualisation images generated from the three-dimensional temperature profiles at various asymmetrical current conditions showed that the magnetic deflection of the lower-current arc was more severe than that at higher current, and that the low-current arc column underwent a transition from steady state to dynamic behaviour between
100 and 150A. This was also confirmed in the transient local temperature data measured close to each cathode spot in the model. Parasitic arc phenomena akin to those seen in the two-dimensional models were also observed in certain of these results.

The current asymmetry results suggest that tight control over the electrical symmetry between the two arcs in a twin electrode DC arc furnace is of some importance - due to the extremely short time scales of the arc system's transient behaviour, the effects of current asymmetry can become established very rapidly. Any pronounced difference in the currents supplied to the arcs generally leads to highly biased flow and temperature patterns in the central region of the furnace, and as a result, it is possible for poor electrical control have a negative impact on the chemical and thermal performance of the process.

The final two sets of results considered models of increased size and resolution. The first such case was a model of three equidistantly-spaced arcs in close proximity. The evolution of the arc columns and their interaction in three dimensions displayed a number of similarities with the twin arc models considered earlier. Despite localised transient behaviour occurring in the vicinity of each arc, the velocity and temperature fields were seen to be largely symmetric, with the three arc columns bending toward the central point between them. The jets formed by each arc were seen to interact to produce a stable upflow at the centre of the region. Close examination of the local transient temperature behaviour at positions close to the cathode spots showed that the regular oscillations exhibited by each arc column were equally out of phase with each other, supporting a similar observation made in the twin arc cases.

The final case examined for the three-dimensional DC plasma arc model examined the behaviour of the twin arc system at high resolution and elevated currents. With the parameters used, more direct comparisons were possible between this model and the two-dimensional results. Representative peak values of the temperature and velocity fields as well as the arc voltage were calculated, and the agreement with the 2D base case model was seen to be reasonable for all variables with the exception of temperature, which was considerably higher in the three-dimensional case. Various visualisations of the temperature and velocity fields revealed considerably more turbulent motion occurring in the higher-current twin arc model, as well as copious formation of parasitic arcs at the anode surface. Comparisons between the evolution of local temperature close to the cathode spots over time identified a number of similarities with the two-dimensional model, including the time of onset of instability at 0.6 ms, and the typical time frames of the transient behaviour, at 0.1 ms. The
Transient behaviour also showed good qualitative agreement between the 2D and 3D models at similar arc currents.

**Advantages of dynamic modelling**

For the study of DC plasma arc furnaces on small to medium pilot plant scale, the model developed in this work has been successful in demonstrating a wide range of interesting effects and behaviours, and displays reasonable agreement with empirical correlations and visual evidence.

Due to the extremely short time scales on which the arc system works, many of the phenomena that result from the coupled multi-physics system that defines the arc are not easily measured or otherwise quantified experimentally. In the absence of measurements, computational models such as this can greatly assist users of technology incorporating DC plasma arcs in their understanding of how the arc functions as a coupled, electromagnetically-driven fluid flow phenomenon.

In contrast with the majority of the work performed to date on the numerical study of DC plasma arcs, the results of this work are unique in that they demonstrate beyond doubt the fundamentally non-linear and dynamic characteristics of the arc system, whether it is a single arc in isolation, or a number interacting together. The nature of the dynamics are seen to be critically dependent on certain parameters of the model, and in cases where these parameters are used as key control or design variables for real world systems, knowledge of this sort has great relevance to the design of the control and operations infrastructure for such systems.

**Two-dimensional vs three-dimensional modelling considerations**

The current state of the art in computer hardware unfortunately places limitations on the modelling effort that can be realistically applied to the study of the dynamic behaviour in the DC plasma arc problem. Two-dimensional models are able to capture all effects down to the viscous length scales relatively easily, but sacrifice accuracy and physical realism by removing a dimension from the problem. Three-dimensional models, which remove many of these limitations, are unable to run at the very highest resolutions required to resolve all scales of motion in the fluid flow due to limitations in computer power and data storage capacity.

However, the results of the current work have demonstrated that a great deal of the qualitative
behaviour of the DC plasma arc system can be captured adequately by approximate
two-dimensional cartesian-geometry models, provided these are designed based on a sound
interpretation of the fundamental physical processes at work and the limitations of the
two-dimensional interpretation are taken into account.

For the study of multiple arc systems, or the particular details related to the dynamics of the
structure of the arc column in three-dimensional space, the 3D models are required, and due to their
excellent numerical stability properties can still be usefully applied even in cases where they are not
able to fully resolve the fluid flow fields.

**Recommendations for future work**

There are a number of enhancements and extensions that could be applied to the DC plasma arc
model in order to increase its value enormously.

Further work on correlating and comparing the two- and three-dimensional arc models with
experimental data is vital. To this end, the development, implementation, and testing of high speed
electrical and photographic measurement techniques in the DC plasma arc industry is deemed to be
of considerable importance, both in its own right and as a tool to verify models such as those
developed here.

Further work on refinement and optimisation of the algorithmic aspects of the model are of some
importance. Strongly-coupled multi-physics problems such as these are very demanding on
computing resources, and optimisation of the computer code that implements them is critical. A
large degree of optimisation was performed as part of this work, however, there is no doubt that
more can be done, depending on the parallel computing resources available.

As the power of computer hardware increases, more generalised extensions to the models can be
contemplated. Inclusion of temperature-dependent physical properties of the plasma fluid would be
valuable and relatively easy to perform, assuming such data is available from LTE simulations. For
the study of larger arcs at higher currents, both compressible fluid flow models and induced current
effects in the electromagnetism model would need to be considered.

Finally, the numerical methods used to discretise the fundamental equations for the DC plasma arc
model could be examined and revised once sufficient computer power becomes available. Unstructured finite volume or finite element methods coupled with intelligent grid adaptation methods would potentially be able to investigate a range of additional phenomena related to the dynamics of plasma arcs in DC furnaces, including electrode geometry effects and interaction of the arc jet with a liquid anode surface.
Appendices

Appendix 1 - Induced current calculation

In order to estimate the influence of the induced current on the electric field, the arc is approximated by the Bowman shape model, given by (1) and (2). The peak value of magnetic field is approximated by the value at the surface of a long, cylindrical conductor of constant conductivity:

\[ B_z \approx \frac{\mu_0 I_a}{2\pi r_a} \tag{225} \]

The magnitude of the electric field is given by the approximate potential gradient in the arc column:

\[ E_a \approx \frac{\partial \phi}{\partial L_A} = \frac{I_a}{\sigma_a \pi r_a^2} \tag{226} \]

With cylindrical symmetry of the velocity and magnetic fields, we have:

\[ |\mathbf{v} \times \mathbf{B}| = B_z |\mathbf{v}| \tag{227} \]

Comparing the size of (226) and (227) will give an estimate of how significant the induced current effect is in the DC plasma arc model for the typical values of currents and velocities expected, as discussed in section 2.3.

Table 37: Values used in induced current calculation

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arc current, I_a</td>
<td>1000 A</td>
</tr>
<tr>
<td>Maximum fluid velocity,</td>
<td></td>
</tr>
<tr>
<td>Average plasma conductivity, \sigma_a</td>
<td>5714 \Omega \cdot m^{-1}</td>
</tr>
</tbody>
</table>

The value of \sigma_a used corresponds to a plasma resistivity of 0.0175 \Omega cm, which is a typical value measured in arc characteristic tests on pilot scale furnaces as described in Reynolds and Jones\textsuperscript{12}.  

253
The values of (226) and (227) at the root \( (L_a = 0) \) and in the main body \( (L_a >> r_a) \) of the arc are compared in Table 38.

| Position | \( E_a \)     | \( B|v| \)  |
|----------|---------------|-------------|
| \( L_a = 0 \) | 6125 V/m     | 66 V/m     |
| \( L_a >> r_a \) | 598 V/m     | 21 V/m     |

In neither case does the induced current exceed a few percent of the electrostatic current, and it may be safely neglected for the DC plasma arc model at these scales.

**Appendix 2 - Discrete temperature-dependent data used for air and argon gases**

The values used for the electrical conductivity and thermal radiation energy loss are dependent on the temperature of the plasma and its composition. Values for the two materials considered in this work are presented in the list over page. The units for \( \sigma \) are \( \Omega^{-1}m^{-1} \), and the units for \( Q \), are \( W/m^3 \). Values of \( \sigma \) smaller than 0.1 are cut off at 0.1 to limit any unstable numerical behaviour in the electric potential field solver.

Data sources are Boulos et al\(^{45} \), and Naghizadeh-Kashani et al\(^{43} \).
ity

ve

U

n

QR, argon
0.32
1.36
5.75
24.34
103.13
436.93
1851.18
7842.94
33228.42
140779.88
596446.54
2526983.89
9703689.83
33331896.47
99469283.09
271430809.77
671321453.74
1478315065.76
3103050982.5
5991546601.25
10845936276.66
18016197151.47
28279199788.63
41302752843.8
55929589104.57
71314791323.03
85174731763.47
97112920947.94
107191753354.87
115350785401.8
124806119825.27
137716575049.26
154306871372.37
172994351661.03
194308182206.29
219504176443.87
254641697287.28
299478825041.14
355136500137.83
424038553127.68
510842144327.91
625296520583.66
772217349609.59
948412210514.14
1160983790332.42
1421285392724.03
1739948640447.7
2130058668648.83
2607634401621.54
3192286331171.07
3908021774005.07
4784230674099.67
5856892429629.16
7170053299886.66
8777635058335.99
10745649159755.5
13154907340889.5
16104339958864.6
19715058326906.1
24135327857343.2
29546656221983.4
36171246525431.3
44281121537877.4
54209293651897.1
66363439230506.7

C

ap

e

To
w

QR, air
0.48
3.35
23.5
164.61
1153.24
8079.37
56602.2
137944.24
350027.05
936965.32
3343898.31
14987677.03
60379923.32
207023170.64
585588287.65
1393844188.36
2939030498.55
5797309671.61
10441348653.6
17627949374.46
26968147772.97
40598897800.58
57085603436.15
76716019915.31
96319848089.34
118843129019
136478543844.66
156745655519.53
171783828273.53
188153196812.79
203284985094.04
217595106399.55
234640862833.32
256443993104.22
281689715248.4
318501476440.87
360081315154.62
407786242830.34
461706351564.66
530256434763.76
612220560464.98
703918856495.41
802222869218.37
912608355798.97
1025409899930.61
1152267051402.09
1285915490913.14
1426292379520.94
1576527268895.05
1720519512108.32
1877789562786.05
2020981863435.53
2173906144301.9
2312561759121.09
2455389382321.18
2609284205690.2
2775905145774.9
2927387381084.95
3087136061539.72
3255602290301.2
3433261787408.28
3620616233129.43
3818194684622.65
4026555069903.07
4246285763336.99

of

σ, argon
0.1
0.1
0.1
0.1
0.1
0.1
0.1
0.72
5.16
23.65
76.82
182.9
358.08
624.4
990.34
1437.8
1935.8
2458.4
2990
3523.1
4054.4
4582.2
5104.9
5619.9
6124.3
6613.4
7084.4
7532.2
7956.1
8356.5
8736.1
9098.5
9447
9784.3
10111
10428
10728
11014
11275
11502
11687
11819
11893
11912
11882
11813
11735
11657
11579
11501
11423
11345
11267
11189
11111
11033
10955
10877
10799
10721
10643
10565
10487
10409
10409

rs

σ, air
0.1
0.1
0.1
0.1
0.1
0.1
1.35
5.56
16.33
38.71
79.18
147.46
264.1
467
779.37
1185.2
1656.5
2169.3
2705.5
3251.7
3799.5
4343.3
4879.8
5405.8
5919.5
6417.1
6897.9
7357.7
7796.4
8214.2
8612.7
8994.4
9362.1
9718.7
10067
10407
10743
11069
11391
11707
12013
12306
12582
12834
13048
13234
13379
13524
13669
13814
13959
14104
14249
14394
14539
14684
14829
14974
15119
15264
15409
15554
15699
15844
15989

ni

T, K
1000
1500
2000
2500
3000
3500
4000
4500
5000
5500
6000
6500
7000
7500
8000
8500
9000
9500
10000
10500
11000
11500
12000
12500
13000
13500
14000
14500
15000
15500
16000
16500
17000
17500
18000
18500
19000
19500
20000
20500
21000
21500
22000
22500
23000
23500
24000
24500
25000
25500
26000
26500
27000
27500
28000
28500
29000
29500
30000
30500
31000
31500
32000
32500
33000

255


Appendix 3 – Numerical testing of 2D Base Case model

In order to verify that the nature of the transient behaviour predicted by the DC plasma arc model is not an artefact of the solution technique, several tests are performed to analyse the model's response to a variety of numerical parameters.

The conditions used for the tests are as outlined in Section 5.2 (specifically Table 7) unless otherwise stated.

Grid independence

The first test concerns the level of spatial discretisation required in order to achieve approximately repeatable behaviour. The 2D Base Case model is run at three different resolutions, 512 x 128, 1024 x 256 (the default), and 1536 x 384. The transient behaviour of the temperature field is used as an indicator of the model behaviour as a whole – the variation of temperature at a location 5mm below the cathode spot surface is presented in Figures 273 and 274.

![Figure 273: Local temperature behaviour at different resolutions](image1)

![Figure 274: Local temperature behaviour at different resolutions – early evolution](image2)

It can be seen from these graphs, particularly in the development of the field variables in the first 1 ms of the simulation, that a grid resolution of 512 x 128 is inadequate. The development of the temperature profile near to the cathode spot is considerably different to the others at this lowest resolution, and more importantly the onset of transient motion is delayed until nearly 1 ms into the model run. Both higher resolution models predict similar early behaviour of the temperature field, and indicate the onset of chaotic transient motion at similar times (0.6 ms). The variability and time
constants associated with the erratic oscillatory motion of the arc column in both higher resolution models also appear similar. 1024 x 256 is a suitable resolution in order to achieve grid independent solutions.

Effect of time step size

As the model makes use of adaptive time-stepping in the explicit Runge-Kutta algorithms, it is not possible to define a fixed size for the time steps used. However, an examination of the effect of the time step can be conducted by varying the $C_1$ and $C_2$ multipliers in the time step size constraints (76). Taking $C_1 = C_2 = CFL$, the variation of temperature at a location 5mm below the cathode spot surface is shown in Figures 275 and 276. $CFL = 1$ is the default case as used by all simulations in Chapters 5 and 6.

![Figure 275: Local temperature behaviour at different time step multipliers](image1)

![Figure 276: Local temperature behaviour at different time step multipliers – early evolution](image2)

It can be seen that altering the size of the time steps used in the DC plasma arc model has little to no effect on the quantitative or qualitative behaviour of the fields in the early part of the model. The onset of erratic transient motion is predicted to occur at similar times (0.6 ms). The nature of the models’ behaviour in fully developed transient motion is also qualitatively very repeatable - both the spread of temperatures calculated and the frequency of the oscillatory motion appear similar. A CFL value of 1 is seen to be a reasonable choice.

Effect of initial conditions

It is desirable that the initial conditions chosen should not significantly affect the nature of the
behaviour predicted by the model. In order to examine the effect of the initial conditions used, two cases are considered in which they are altered slightly – one uses an initial temperature of 10100K (instead of 10000K), and the other uses an initial velocity field with $v_y = -10$ m/s (instead of zero). The resulting temporal evolution of the temperature field at a location 5mm below the cathode spot surface in each case is shown in Figures 277 and 278.

**Figure 277: Local temperature behaviour with different initial conditions**

![Temperature Behaviour](image1)

**Figure 278: Local temperature behaviour with different initial conditions - early evolution**

![Temperature Behaviour](image2)

As can be seen from the early evolution of the temperature profiles, the solutions at different initial conditions track each other closely at first and then begin to move apart rapidly once the transient oscillation of the arc column begins. Importantly however the time of onset of the transient behaviour is consistently predicted despite changes in the initial conditions, and the qualitative characteristics of the temporal behaviour of the temperature field appear similar, even though the instantaneous values begin to diverge from one another once the erratic motion begins. This divergence despite very small changes (of the order of 1%) in the initial conditions in the model suggests the presence of sensitive dependence and mathematically chaotic behaviour in the DC plasma arc model, although further work would be required to verify this.
References


2. H Lagendijk and RT Jones: Production of ferronickel from nickel laterites in a DC-arc furnace, Proceedings of Nickel-Cobalt 97 at the 36th Annual Conference of Metallurgists (WC Cooper, I Mihaylov, CA Levac, and RA Berryman, editors), Canadian Institute of Mining, Metallurgy and Petroleum, Montreal Quebec, 1997, pp 151-162


5. RT Jones: Recovery of cobalt, nickel, and copper from slags using DC-arc furnace technology, Cobalt News, Issue 98/2, April 1998, pp 3-6


of Mining and Metallurgy, Volume 102, no. 1, 2002, pp 5-9


15. MW Meng and GA Irons: Comparison of electric arc models with industrial data, 58th Electric Furnace Conference and 17th Process Technology Conference proceedings (DL Kanagy, editor), Iron and Steel Society, Warrendale PA, 2000, pp 183-193

260


26. F Qian, B Farouk and R Mutharasan: Modeling of fluid flow and heat transfer in the plasma
region of the DC electric arc furnace, Metallurgical and Materials Transactions B, Volume 26B, October 1995, pp 1057-1067


30. L Gu, R Jensen and JA Bakken: Metal vapour infiltration in argon arcs used for heating liquid metal, 51st Electric Furnace Conference proceedings (Iron and Steel Society - Electric Furnace Division, editors), Iron and Steel Society, Warrendale PA, 1994, pp 193-199

31. NJ Holt and JA Bakken: Transferred argon arc reactor with emphasis on radiation heat transfer, 51st Electric Furnace Conference proceedings (Iron and Steel Society - Electric Furnace Division, editors), Iron and Steel Society, Warrendale PA, 1994, pp 177-186


35. W E and JG Liu: Vorticity boundary condition and related issues for finite difference


45. MI Boulos, P Fauchais and E Pfender: Thermal plasmas: fundamentals and applications,

47. WH Press, SA Teukolsky, WT Vetterling and BP Flannery: Numerical recipes in C, 2<sup>nd</sup>, Cambridge University Press, 1992


49. P Tackley, Multigrid for viscous flow: part 1, lecture material available at http://www.gfd.ethz.ch/education/research_material

50. FFTW numerical spectral transform library, http://www.fftw.org

51. FFmpeg animation software, http://ffmpeg.mplayerhq.hu

52. The Visualisation Toolkit (VTK), http://www.vtk.org


55. WM Pun and DB Spalding: A general computer program for two-dimensional elliptic flows, Imperial College London Report HTS 76/2, 1976


40, 2007, pp 7357-7366
