



UNIVERSITEIT • STELLENBOSCH • UNIVERSITY

The simulation of the flow of polymer melt in lomolding

by

Jacques Andrew Dryden Dymond

*Thesis presented at the University of Stellenbosch in partial
fulfilment of the requirements for the degree of*

Master of Mechanical Engineering

Department of Mechanical Engineering
University of Stellenbosch
Private Bag X1, 7602 Matieland, South Africa

Study leader: Prof A.H. Basson

December 2004

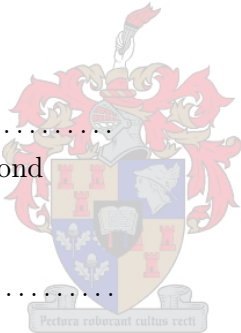
Declaration

I, the undersigned, hereby declare that the work contained in this thesis is my own original work and that I have not previously in its entirety or in part submitted it at any university for a degree.

Signature:

J.A.D. Dymond

Date:



Abstract

The simulation of the flow of polymer melt in lomolding

J.A.D. Dymond

*Department of Mechanical Engineering
University of Stellenbosch
Private Bag X1, 7602 Matieland, South Africa*

Thesis: MScEng (Mech)

December 2004

Lomolding is a new process similar to injection moulding. In this thesis a numerical model of the polymer flow into the cavity is presented. The model is used to compare the two processes with each other.

Lomolding and injection moulding were modelled numerically with the finite element method. The model was an axisymmetric model and takes phenomena such as generalised Newtonian flow, free surfaces, moving boundaries and solidification into account.

The processes' characteristics that were compared are the cavity pressure, shear rate and shear stress. Improvements in these characteristics will result in lomolding having smaller machines, less fibre breakage and a better potential for in-mould-decoration. The thesis shows that lomolding has substantially lower shear rates and shear stresses than injection moulding. The model was also used to investigate certain machine design aspects, and to gain a better understanding of the polymer flow in lomolding.

Uittreksel

The simulation of the flow of polymer melt in lomolding

J.A.D. Dymond

*Departement Meganiese Ingenieurswese
Universiteit van Stellenbosch
Privaatsak X1, 7602 Matieland, Swid-Afrika*

Tesis: MScEng (Meg)

Desember 2004

Lomoldering is 'n nuwe proses soortgelyk aan inspuitgiet. Hierdie tesis beskou 'n numeriese model van die polimeervloei soos dit in die gietholte invloei. Die model word gebruik om die twee prosesse met mekaar te vergelyk.

Lomoldering en inspuitgiet is numeries gemodeleer met die eindige-element-metode. Die model is aksiaal-simmetries en neem verskynsels soos Newtoniese vloei, vrye oppervlakke, bewegende grense en stolling in ag.

Die proses-karakteristieke wat vergelyk is, sluit die gietholte-druk, vervormingstempo en skuifspanning in. Verbeterings in hierdie karakteristieke sal meebring dat lomoldering kleiner masjiene nodig sal hê, minder veselskade sal veroorsaak en meer versoenbaar met in-gietholte-versierings sal wees, vergeleke met inspuitgiet. Die tesis toon dat lomoldering laer vervormingstempo's en skuifspanning het, in vergelyking met inspuitgiet. Die model is ook gebruik om sekere masjienontwerp-aspekte te ondersoek, en om die polimeervloei in lomoldering beter te verstaan.

Copyright © 2004 University of Stellenbosch
All rights reserved.



Acknowledgements

I would like to express my sincere gratitude to the following people who have contributed to making this work possible:

- Prof Anton Basson of the University of Stellenbosch as my study leader.
- To Lomold for financial support.
- Cobus Zietsman, Ferdi Zietsman, Brett Johnson and Jacques Booysen who helped me in the laboratory.
- Danie Els for the Latex help and support.
- My office buddies who had to put up with me for two and a half years.
- My family and Juliet for emotional support.
- My God who created an ordered universe and revealed the most important knowledge through his Son, Jesus Christ.

Dedications



To Juliet

Contents

Declaration	i
Abstract	ii
Uittreksel	iii
Acknowledgements	v
Dedications	vi
Contents	vii
List of Figures	xii
List of Tables	xv
Nomenclature	xvi
1 Introduction	1
1.1 Lomolding - A detailed description	1
1.2 Anticipated advantages of lomolding	3
1.3 Objectives	4
2 Literature review	5
2.1 General overview of polymer simulation work	5
2.2 Governing equations	6
2.3 Equations of state	7
2.3.1 Newtonian, generalised Newtonian and viscoelastic flows	7
2.3.2 Thermal properties	8
2.4 Boundary conditions	8

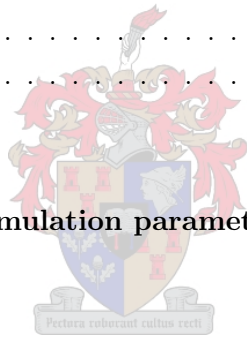


2.5	FEM, BEM and finite difference method	9
2.5.1	The finite difference method	9
2.5.2	The boundary element method	9
2.5.3	The finite element method	10
2.6	Eulerian, Lagrangian and ALE meshes	10
2.6.1	Eulerian mesh	10
2.6.2	Lagrangian mesh	10
2.6.3	Arbitrary Lagrangian-Eulerian mesh	10
2.7	Momentum equation	11
2.7.1	UVP scheme	12
2.7.2	Penalty scheme	12
2.7.3	Hele-Shaw flow	13
2.8	Free surface method - Volume Of Fluid	14
2.9	Energy equation	16
2.10	Time dependence techniques	17
2.10.1	Euler explicit integration	17
2.10.2	4th order Runge-Kutta integration	17
2.10.3	θ time stepping scheme	18
2.11	Solidification	19
3	Implementation	21
3.1	Choosing a momentum scheme	21
3.2	Sopfil flowchart	23
3.3	Momentum flowchart	25
3.4	Energy flowchart	25
3.5	Moving boundary flowchart	25
3.6	VOF flowchart	27
3.7	Modelling lomolding with Cadmould	29
4	Validation of code	31
4.1	Momentum	31
4.1.1	Newtonian flow	31
4.1.2	Generalised Newtonian flow	33
4.2	Energy-flow over a sphere	37
4.3	Time dependance	37
4.3.1	Cooling of a solid slab	38

4.3.2	Developing thermal field around a hot sphere in cold fluid stream	39
4.4	Solidification and determining the optimal solidification factor .	41
4.5	Moving boundary and ALE scheme	43
5	Experimental setup and results	46
5.1	Experimental data capturing ability	46
5.2	Experimental part and polymer	46
5.3	Cavity pressure measurements for lomolding	48
6	Simulations and results	50
6.1	Injection moulding analysis	50
6.2	Injection moulding results	51
6.3	Lomolding analysis	52
6.4	Lomolding results	53
6.5	Shear heating in hot runners analysis	55
6.6	Shear heating in hot runners results	59
6.7	Lomolding vs injection moulding	59
6.7.1	Cavity pressures	59
6.7.2	Shear rates	62
6.7.3	Shear stresses on in-mould-decorations	63
6.7.4	Filling rates at constant injection pressures	64
6.8	Lomolding piston analysis - flow patterns under the piston . . .	67
7	Investigation of discrepancies between experiments and simulations	71
7.1	Pressure sensor range	71
7.2	Process parameters, particularly melt temperature	71
7.3	Verification of viscosity data	72
7.4	Non-axisymmetric flow	72
7.5	Inadequate clamping force	73
8	Conclusion	75
8.1	Recommendations for further work	76
8.1.1	Numerical work	76
8.1.2	Experimental work	77
	List of References	79

A	Mathematics	83
A.1	Gradient operator	83
A.2	Material derivative	84
A.3	Kronecker Delta	84
A.4	Integration by parts	84
B	Finite element method	85
B.1	Nodes, elements and shape functions	85
B.2	Local elements	86
B.3	Quadratic elements - some examples	87
B.3.1	Bilinear rectangular element	88
B.3.2	Rectangular Taylor-Hood element	88
B.3.3	Rectangular Crouzeix-Raviart element	89
B.4	The finite element Galerkin method	90
B.5	System of equations	91
B.6	Numerical integration	92
B.7	Boundary conditions	93
C	Dimensional analysis of the momentum equation	95
D	FEM equations for polymer flow and other algorithms	97
D.1	Deriving the FEM equations from the governing equations	97
D.1.1	UVP scheme	97
D.1.2	Shear rate	100
D.1.3	Energy equation	100
D.1.4	Upwinding weight factor	101
D.2	Mass flow average	102
D.3	Convergence criteria	102
D.4	Error formula	102
D.5	Node sorting algorithm	102
D.6	Mesh expansion factors	103
D.7	Element search scheme	104
E	Matrix solution scheme	105
E.1	The nature of FEM matrix	105
E.2	Existing matrix libraries	106

F	Physical properties of polymers	108
F.1	Viscosity models	108
F.1.1	Power-law model	108
F.1.2	Carreau model	108
F.1.3	Cross model	109
F.1.4	Curve fitting viscosity models	109
F.2	Thermal properties of polymer	111
G	Calibration of pressure sensors	112
G.1	Weight calibrator	113
G.2	Hydraulic calibrator	116
G.3	Discussion of results	118
H	Melt temperature experiments	119
H.1	Experimental setup	119
H.2	Results	120
H.3	Conclusion	122
I	Mould filling pattern	124
J	Sensitivity study of simulation parameters	128



List of Figures

1.1	The lomolding machine - 3D CAD model	2
1.2	The lomolding machine	2
1.3	The lomolding process	3
2.1	VOF element - free surface and volume fluxes	16
3.1	Convergence rate and level for different momentum schemes	23
3.2	Sopfil flowchart	24
3.3	Momentum scheme flowchart	26
3.4	Energy flowchart	26
3.5	Moving boundary flowchart	27
3.6	VOF flowchart	28
3.7	Modelling lomolding with Cadmould	30
4.1	Sphere mesh and boundary conditions	32
4.2	Pressure distribution on the surface of a sphere	33
4.3	Pipe mesh and boundary conditions	34
4.4	Parallel plates mesh and boundary conditions	34
4.5	Pipe velocity profiles - power-law fluid	35
4.6	Parallel plates velocity profiles - power-law fluid	36
4.7	Sphere local Nusselt number vs. angle - $Pe = 300$	38
4.8	Solid slab domain	39
4.9	Cooling of a solid slab	40
4.10	Surfaced averaged Nusselt number of a hot sphere	40
4.11	Determining the value of the solidification factor	42
4.12	The effects of solidification on the velocity profiles in a pipe	42
4.13	Moving piston and moving wall domain	43
4.14	Moving piston mesh detail at the piston	43

4.15	Moving boundary validation - temperatures along the axis of symmetry	44
5.1	Experimental part geometry	47
5.2	Lomolding experimental data	49
6.1	Injection moulding domain	51
6.2	Injection moulding - part of a typical mesh	51
6.3	Injection moulding - Cadmould and Sopfil pressures	52
6.4	Solid layer thickness in injection moulding - Cadmould and Sopfil	53
6.5	Lomolding domain	54
6.6	Piston analysis domain	54
6.7	Simplified lomolding domain	55
6.8	Piston analysis mesh	56
6.9	Lomolding experimental, Cadmould and Sopfil pressures	57
6.10	Shear heating domain - hot runner and hot runner with valve rod	57
6.11	Part of the shear heating mesh	58
6.12	Temperature profiles, low volume flow rate ($Q = 58.9 \cdot 10^3 \text{ mm}^3\text{s}^{-1}$)	60
6.13	Temperature profiles, high volume flow rate ($Q = 589 \cdot 10^3 \text{ mm}^3\text{s}^{-1}$)	60
6.14	Lomolding vs. injection moulding - cavity pressures - Sopfil	61
6.15	Lomolding vs. injection moulding - centreline pressures	62
6.16	Lomolding vs. injection moulding - Shear history and maximum shear rates	64
6.17	Lomolding vs. injection moulding - Shear stresses on skins	65
6.18	Lomolding vs. injection moulding - Filling rates for 180 bar injection pressure	66
6.19	Lomolding vs. injection moulding - Cavity pressures for 180 bar injection pressure	66
6.20	Flow patterns under the piston - velocity magnitude at the cavity / moulding barrel interface	67
6.21	Flow patterns under the piston - Temperature	68
6.22	Maximum shear rate in piston analysis	69
7.1	Filling contours for non-axisymmetric flow - Cadmould	74
7.2	Non-axisymmetric fill cavity pressures - Cadmould	74
B.1	Local and global elements	86
B.2	Bilinear rectangular element	87

B.3	Rectangular Taylor-Hood element	88
B.4	Rectangular Crouzeix-Raviart element	89
D.1	Node sorting algorithm	103
E.1	Profile of a matrix for a lomolding domain	106
F.1	Viscosity model curve fit	110
G.1	Hydraulic fluid pressure on Kistler sensor	113
G.2	Melt pressure on Kistler sensor	113
G.3	Weight calibrator	114
G.4	Weight calibrator - cross section	114
G.5	Weight calibrator - %error	115
G.6	Hydraulic calibrator	116
G.7	Hydraulic calibrator - effect of o-ring and membrane	117
H.1	Position of thermocouples for measuring melt temperature	119
H.2	Melt temperatures for a typical shot	121
H.3	Melt temperatures when purging the machine	121
H.4	Interpreting the melt temperature data	122
I.1	Mould filling pattern	126
I.2	Possible reason for non-axisymmetric filling	127
J.1	Effect of mould wall temperature on pressure	129
J.2	Effect of melt freezing temperature on pressure	129
J.3	Effect of melt injection temperature on pressure	130
J.4	Effect of volume flow rate on pressure	130

List of Tables

3.1	Mass conservation of different momentum schemes	23
4.1	Simulation settings for moving boundary validation	44
5.1	Position of pressure sensors in mould	47
5.2	Parameters for simulations	48
6.1	Parameters for shear heating analysis	58
6.2	Mass flow averaged temperature rise in shear heating analysis . . .	59
6.3	Clamping force for mould filling - Lomolding vs. Injection moulding	62
6.4	Mass flow averaged shear history	64
B.1	Gauss Quadrature points	93
F.1	Cross model constants for Moplen EP301K	110
F.2	Carreau model constants for Moplen EP301K	110
F.3	Thermal properties of Moplen EP301K	111
G.1	Weight calibrator - sensor drift	115
G.2	Membrane thicknesses	117
G.3	Hydraulic calibrator - results for sensor 2	118
H.1	Melt temperature experiment parameters	120
I.1	Mould filling experimental setting	125
J.1	Base settings for sensitivity study	128

Nomenclature

Γ	boundary of a domain
Ω	domain
α	thermal diffusivity
	upwinding parameter
$\dot{\gamma}$	shear rate
Δt	time step
η	finite element local coordinate
	viscosity
θ	angle
λ	penalty number
\bar{v}	velocity
ξ	finite element local coordinate
ρ	density
τ	shear stress tensor
C	mass matrix
\bar{D}	rate of deformation tensor
F	vector function
Fo	Fourier number
F_r	fluid transfer fraction
H	characteristic element length
L	load vector
	total length
N	shape function
M	shape function of a lower order than N
Pe	Peclet number

Q	volume flow rate
R	total radius
S	stiffness matrix
	flow conductivity (Hele-Shaw flow)
T	temperature
	characteristic time through element
T_m	melting temperature of polymer
U	characteristic velocity through element
VOF	volume of fluid
V	volume
Z	dimensionless temperature
a	constant for injection moulding flow rate
	solidification factor
b	constant for injection moulding flow rate
c	specific heat
f	fraction of element that has solidified
	variable
$flops$	floating point operations
g	gravity constant
h	characteristic element dimension
k	thermal conductivity
i	matrix (row) or vector index
j	matrix (column) or vector index
m	power-law viscosity model constant
	bandwidth of matrix
n	power-law viscosity model constant
	size of matrix
\bar{n}	normal vector
p	pressure
r	radial (axisymmetric) or radius (spherical) coordinate
t	thickness
	time
u	velocity component in radial or x direction
v	velocity component in axial or y direction
w	weight function
\bar{x}	coordinate vector

z axial coordinate

Superscripts

*

non-dimensional variable

Subscripts

∞ free stream value

E fixed mesh

M moving mesh

ave average

e element

$flux$ flux across element boundary

i inside

l liquid

o outside

r radial direction or coordinate

s sphere property

solid

z axial direction or coordinate

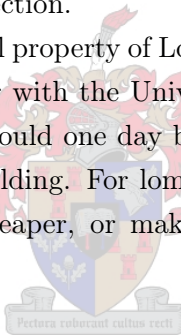


Chapter 1

Introduction

Lomolding is a process that produces finished plastic parts. The process has similarities to injection moulding and to compression moulding. The process will be described in the next section.

Lomolding is the intellectual property of Lomold. Lomold is based in Paarl, South Africa. Lomold together with the University of Stellenbosch are developing lomolding. Lomolding could one day be a process that competes with injection and compression moulding. For lomolding to become a widely-used process it has to be either cheaper, or make products that the competing processes cannot.



1.1 Lomolding - A detailed description

A 3D CAD model of the lomolding machine (used for the current research) can be seen in figure 1.1. A schematic drawing of the lomolding machine can be seen in figure 1.2. Polymer melt enters the lomolding machine from a screw extruder on the machine used for the work presented here, but could come from any other source (e.g. a twin screw compounder). The mould cooling and clamping is the same as that for injection moulding.

The major difference in the mould, compared to injection moulding, is that the piston face forms part of the mould surface. The piston face can be shaped to match the contours of the part.

Figure 1.3 shows how the lomolding machine works. A metering cylinder measures off the correct volume of melt. The measured melt, or shot, is trans-

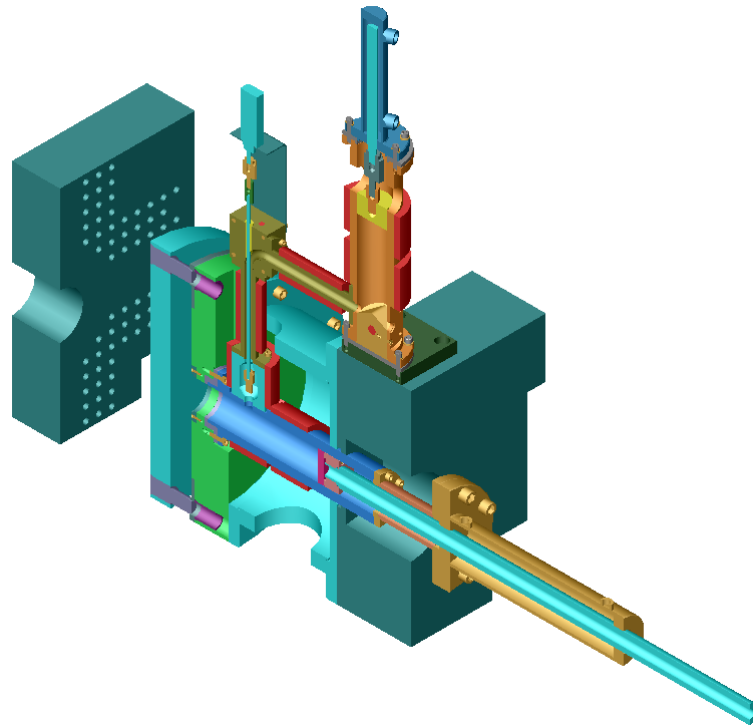


Figure 1.1: The lomolding machine - 3D CAD model

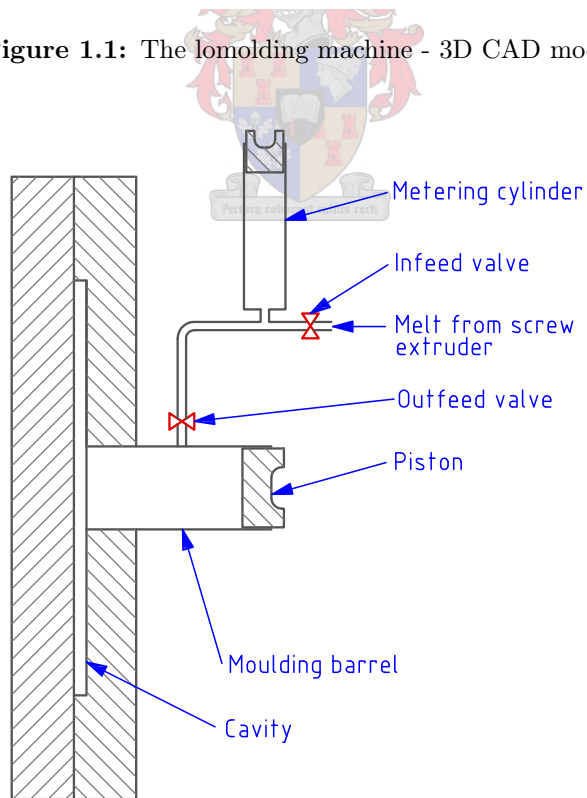


Figure 1.2: The lomolding machine

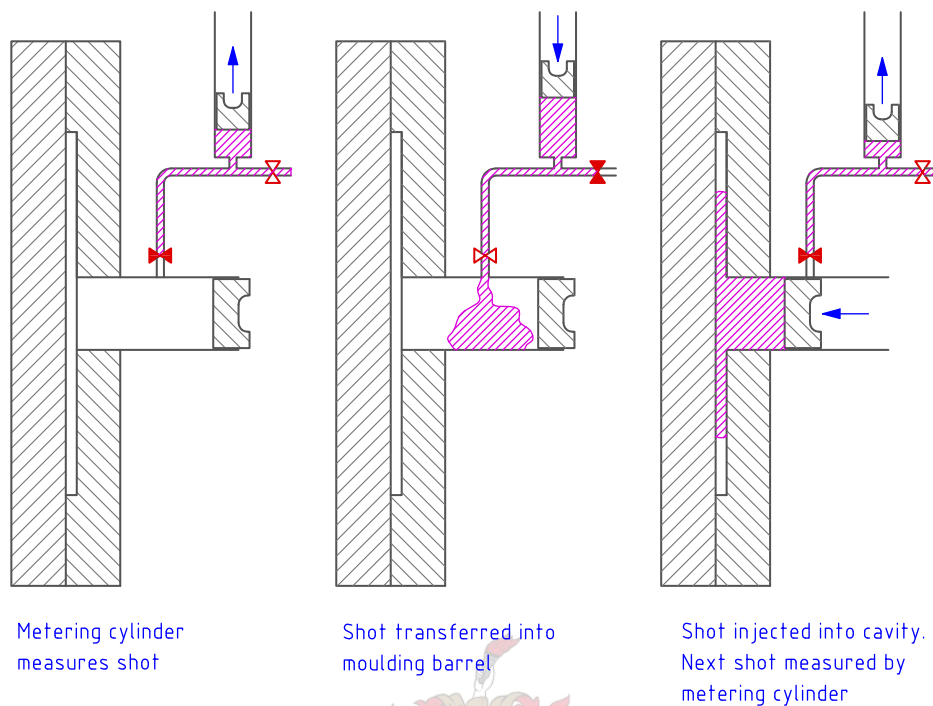


Figure 1.3: The lomolding process

ferred into the moulding barrel, where it is pushed into the cavity by the piston. The Infeed and Outfeed valves prevent the melt from flowing backwards.

1.2 Anticipated advantages of lomolding

Lomolding has the following anticipated advantages over injection moulding:

Lower cavity pressures: The melt enters the cavity through a much bigger opening, thus reducing the pressure drop. Lower cavity pressures will mean that a machine with a smaller clamp force can be used.

Lower shear rates: The large cavity entry means the shear rates will be less, compared to injection moulding where there are high shear rates around the gate.

Less fibre damage: The lower shear rates mean that there will be less fibre damage. This will allow longer fibres, which will result in better mechanical properties of the finished part.

In-mould-decoration: The lower shear rates and cavity pressure will be more compatible with in-mould-decoration. In-mould-decoration could be anything that is placed in the mould before moulding. Examples are fabrics, paint skins or skins of scratch resistant material. By using in-mould-decoration one production process of a part is cut out.

1.3 Objectives

The primary goal of this thesis is to create a numerical model of the polymer melt as it flows from the barrel into the cavity. This model can also model injection moulding, and other axisymmetric flows. This program will be referred to as Sopfil (Simulation Of Polymer Flow In Lomolding) from here on.

Sopfil is used to compare lomolding to injection moulding in this thesis. Cadmould (Simcon, 2002a) is also used in the comparison. Cadmould is a commercial code for simulating injection moulding. Cadmould can be used to model the cavity region of lomolding, but cannot be used to model the piston region. The method for modelling lomolding with Cadmould is presented later in this thesis. Experimental cavity pressure data is compared with the simulations.

Sopfil is also used to gain some insight into the flow under the piston. This information will be used to improve the design of the cavity entry area. Understanding the flow under the piston will be of great use in future projects, when new lomolding machines are designed.

Modelling lomolding with Cadmould is also investigated. It will be beneficial to lomolding if existing codes (such as Cadmould) can be used to model lomolding. This thesis investigates to what extent and accuracy lomolding can be modelled with Cadmould.

Chapter 2

Literature review

This chapter will seek to present an overview of the required theory and existing work in polymer modelling. The derivation and implementation (where required) of the equations are not discussed in this chapter, but rather in the chapter on implementation or in the appendices.

Before presenting the theory it would be helpful to state the desired characteristics of Sopfil: Sopfil must be able to solve the flow field in lomolding. This includes both a momentum and an energy solution. Sopfil should be able to simulate moving boundaries (piston) and free surfaces (melt flow front advancing through cavity). A method of simulating the solidification of the polymer against the mould wall should also be included.

Sopfil will be limited to axisymmetric problems. This will limit the parts being modelled to axisymmetric parts. An axisymmetric problem (2-dimensional problem) is a simpler problem to solve, and less computationally expensive.

2.1 General overview of polymer simulation work

The modelling of polymer flow is a mature field and various textbooks have been published on the subject (Nassehi, 2002; Tucker, 1989).

The most polymer flow simulations have been for injection moulding. The flow is approximated by the Hele-Shaw approximation (which will be discussed in section 2.7.3). Commercial injection moulding codes have been developed. An example of such a code is Cadmould (Simcon, 2002*a*). Injection moulding has also been modelled without the use of the Hele-Shaw approximation (Shen, 1998).

The work or numerical methods presented in this thesis are not new. Examples where similar numerical methods have been applied are Nassehi & Ghorishy (1998) (mixing and compounding processes), Petera & Nassehi (1995) (cone-and-plate rheometers) and Shen (1998) (injection moulding).

What makes the work presented here unique is the process that it is applied to (lomolding). Lomolding is a new process and is not well understood.

The rest of the chapter presents a more detailed literature review of the various aspects of the numerical methods used for Sopfil.

2.2 Governing equations

Polymer flow is described by the following governing equations. These equations describe a viscous incompressible flow. The equations are the mass (equation 2.2.1), momentum (equation 2.2.2) and energy conservation equations (equation 2.2.3) (Nassehi, 2002).

$$\nabla \cdot \bar{\nu} = 0 \quad (2.2.1)$$

$$\rho \frac{\partial \bar{\nu}}{\partial t} + \rho \bar{\nu} \cdot \nabla \bar{\nu} = -\nabla p \bar{\delta} + \nabla \cdot \bar{\tau} + \rho g \quad (2.2.2)$$

$$\rho c \left(\frac{\partial T}{\partial t} + \bar{\nu} \cdot \nabla T \right) = k \nabla^2 T + \bar{\tau} : \nabla \bar{\nu} + \dot{S} \quad (2.2.3)$$

The equations deserve a more thorough description, as they are the central equations of this thesis. Everything that follows from here is aimed at solving these equations.

The mass conservation equation (2.2.1) is for an incompressible substance i.e. the density of the fluid is a constant.

The momentum conservation equation (2.2.2) is the Navier-Stokes equation for incompressible flow. The equation is a force balance on an infinitesimally small control volume. The first term on the left hand side is the change of momentum with respect to time. The second term represents the momentum of the fluid entering the control volume. On the right hand side, the first term is the resultant pressure force acting on the control volume. The second term is the resultant shear stress. The last term is for body forces (in this case gravity).

After performing a dimensional analysis (see appendix C) the momentum equation can be simplified to equation 2.2.4 for lomolding, injection moulding and other polymer flows. This equation describes creeping flow. Creeping flow is dominated by a balance between the viscous and pressures forces. The other forces are much smaller and therefore neglected. The equation does not contain a time derivative (quasi steady state) and therefore the velocities change "instantly" with time.

$$0 = -\nabla p \bar{\delta} + \nabla \bar{\tau} \quad (2.2.4)$$

The energy conservation equation's (2.2.3) first term on the left hand side is the change of energy with respect to time. The second term is the energy that is convected into the infinitesimal control volume. On the right hand side, the first term is the conduction term. The second term is the viscous heat dissipation (shear heating) and the last term is for a heat source (e.g. from a chemical reaction). Thermosets are not used in lomolding at this stage, so the heat source term is neglected.

2.3 Equations of state

2.3.1 Newtonian, generalised Newtonian and viscoelastic flows



Stokes proposed that the shear stress in a fluid is a linear function of the fluid's deformation rate (White, 1974). This proposal leads to equation 2.3.1, where η is the viscosity. The elements of the rate of deformation tensor are given by equation 2.3.2. Appendix F.1 presents the viscosity models used in this thesis.

$$\bar{\tau} = 2\eta \bar{D} \quad (2.3.1)$$

Viscous fluids can be classified as one of the following three categories:

Newtonian fluids have a constant viscosity. The shear stresses are directly proportional to the deformation rates. Some examples of fluids that exhibit this behaviour are air, water and oil.

Generalised Newtonian fluids do not have a constant viscosity. The viscosity can be dependant on shear rate, temperature, pressure, etc.

Viscoelastic fluids exhibit viscous as well as elastic effects. This means that a fluid particle's condition (viscosity, stress) is dependant on its history as well

as shear rate, temperature etc. Polymers exhibit viscoelastic behaviour. The effects are usually seen around sudden changes in the geometry (such as die swell at the exit of extrusion dies). Fortunately these effects are local and dissipate quickly.

For this thesis the polymers will be modelled as generalised Newtonian fluids. This assumption is generally made when modelling polymers because viscoelastic flows are more complex to model.

$$\bar{D}_{ij} = \frac{1}{2} \left(\frac{\partial \bar{v}_i}{\partial \bar{x}_j} + \frac{\partial \bar{v}_j}{\partial \bar{x}_i} \right) \quad (2.3.2)$$

2.3.2 Thermal properties

Many of the physical parameters (such as density, thermal conductivity etc.) are also functions of temperature, pressure and other factors. To find data on how these parameters change is hard, as they are not usually supplied by the manufacturer of the polymer. The parameters' dependance on temperature, pressure, etc. is not large and for this reason, and to simplify the problem, these effects are neglected.

2.4 Boundary conditions

Boundary conditions have to be applied to the governing equations before they can be solved. Boundary conditions can either be a fixed value or a flux.

The momentum equation's boundary conditions are applied through the velocity terms. Applying pressure boundary conditions for incompressible flow is not recommended (Nassehi, 2002), as it is inconsistent with the incompressibility constraint. The three areas where the boundary conditions are of interest are:

Inlet: This area corresponds to the piston in lomolding or the gate in injection moulding. The velocities are applied to these areas.

Outlet: This area corresponds to the end of the cavity. Generally the conditions at the outlet are not known and therefore no boundary conditions are applied. This is in effect setting the normal gradients to zero at the exit, which is appropriate for fully developed flows (Nassehi, 2002).

Walls: No-slip boundary conditions are used at the walls. This is applied by specifying zero velocities along the wall nodes.

The energy equation's boundary conditions are applied through the temperature terms. Fixed values, as well as gradients (heat fluxes), can be applied. The three areas where the boundary conditions for the energy equation are of interest are:

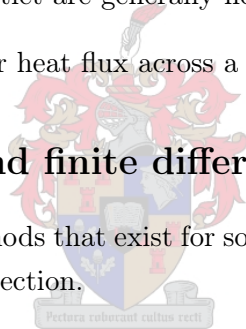
Inlet: The temperature of the melt entering the cavity is known and applied to the nodes. In the case of lomolding, it is assumed that the piston face stays at a constant temperature (see section 6.1).

Outlet: Applying boundary conditions at the outlet creates numerical difficulties because of the convective terms in the energy equation. Therefore no boundary conditions are applied at the outlet and also because the temperatures at the outlet are generally not known.

Wall: A wall temperature or heat flux across a wall can be applied.

2.5 FEM, BEM and finite difference method

The three most popular methods that exist for solving the governing equations are briefly discussed in this section.



2.5.1 The finite difference method

The finite difference method is the oldest and easiest of the three methods (Guceri, 1989). Traditionally it is limited to quite simple geometries, although with the use of grid transformations quite complex problems can be solved. Guceri (1989) gives a more detailed description of the method.

2.5.2 The boundary element method

Unlike the finite element method where the whole domain is discretized, the boundary element method only discretizes the boundary of the domain. This reduces a two-dimensional problem to a one-dimensional problem. The boundary element method is not very widely-used. Barone & Osswald (1989) gives a more detailed description of the method.

2.5.3 The finite element method

The finite element method is the most widely-used method for the solution of the governing equations. It has excellent geometric flexibility (Pittman, 1989). Traditionally triangular elements were needed to mesh arbitrary domains, but new methods are emerging which can mesh almost any domain with quadrilateral elements. Bastian & Li (2003) present such a method, and show some complex domains that were meshed with quadrilateral elements.

A summary of the finite element method is given in appendix B. For details of the method refer to Cook *et al.* (2002) and Bathe (1982). For references that use finite elements in polymer flow refer to Nassehi (2002) and Pittman (1989). The finite element was chosen for use in Sopfil because of the wide range of literature available, and because of its geometric flexibility.

2.6 Eulerian, Lagrangian and ALE meshes

There are three mesh types of importance in finite element modelling.

2.6.1 Eulerian mesh

The Eulerian mesh is a stationary mesh, i.e. the elements and nodes do not move with the flow field. Therefore governing equations have convective terms in them. The governing equations in section 2.2 were for an Eulerian mesh.

2.6.2 Lagrangian mesh

A Lagrangian mesh moves with the flow field. The governing equations for a Lagrangian mesh will not have convective terms. A Lagrangian mesh can become distorted, and then it is necessary to remesh the domain. An alternative to remeshing is mesh regeneration, which is a process of refining or coarsening the mesh where needed (Malcevic and Ghattas, 2002).

2.6.3 Arbitrary Lagrangian-Eulerian mesh

An Arbitrary Lagrangian-Eulerian (ALE) mesh is a combination of the Eulerian and Lagrangian mesh types. In the ALE mesh the nodes can move in any desired manner. The nodes can move with the fluid (as in a Lagrangian mesh), stay stationary (an Eulerian mesh) or in an arbitrary manner. Uchiyama

(2001) use an ALE mesh to model a cylinder oscillating in a fluid. Wang & McLay (1986) use an ALE mesh to model compression moulding. If the ALE mesh elements distort badly the domain has to be remeshed.

If the mesh changes with respect to time, chain rule differentiation has to be used for the partial derivative of variable f with respect to time (equation 2.6.1). M and E refer to the moving and fixed meshes respectively, where $\frac{\partial r}{\partial t}$ and $\frac{\partial z}{\partial t}$ are the mesh velocities in the radial and axial directions.

$$\left(\frac{\partial f}{\partial t}\right)_M = \frac{\partial f}{\partial r} \frac{\partial r}{\partial t} + \frac{\partial f}{\partial z} \frac{\partial z}{\partial t} + \left(\frac{\partial f}{\partial t}\right)_E \quad (2.6.1)$$

If an ALE scheme is used, the time derivative in the governing equations (for an Eulerian mesh) must be substituted with equation 2.6.2.

$$\left(\frac{\partial f}{\partial t}\right)_E = \left(\frac{\partial f}{\partial t}\right)_M - \frac{\partial f}{\partial r} \frac{\partial r}{\partial t} - \frac{\partial f}{\partial z} \frac{\partial z}{\partial t} \quad (2.6.2)$$

For this thesis an ALE scheme is used. The ALE scheme is, however, only used in the piston region, at the moving boundary. In the cavity the mesh velocity is set to zero, which then simplifies to the Eulerian formulation.

2.7 Momentum equation

The momentum equation has to be solved in conjunction with the mass conservation equation. An important aspect of a momentum solving method is that it must satisfy or circumvent the Babuska-Brezzi criterion (referred to as BB criterion from here on).

The BB criterion is necessary because of the absence of a pressure term in the mass conservation equation (Nassehi, 2002). Not satisfying the BB criterion could lead to pressure check boarding, where two solutions are found for the domain, on alternate nodes.

The two most widely-used momentum solving methods are the UVP and the penalty methods. Choosing which method and element to use is the topic of section 3.1.

For Newtonian fluids (Stokes flow) a set of linear equations is generated, but for generalised Newtonian a non-linear set of equations is generated. This is due to the viscosity term, which is a function of the shear rate. The equations are solved by an iterative process where the previous iteration's velocity solution is used to calculate the new shear rate and then the viscosity.

2.7.1 UVP scheme

In the UVP scheme the primitive variables are discretized and the governing equations are solved directly. The scheme gets its name from the most commonly used cartesian velocity coordinates (u, v) and the pressure term (p) .

To satisfy the BB criterion it is necessary to use elements belonging to the Taylor-Hood or Crouzeix-Raviart family. These elements have higher order interpolation functions for the velocity variables than for the pressure variables. Refer to appendix B.1 for more details on the elements. Because of the lower order interpolation functions used for the pressure variables, the pressure and velocity nodes are not the same. The total number of pressure nodes is less than the total number of velocity nodes, and therefore the assembly of the stiffness matrix in the UVP scheme requires more bookkeeping than in the penalty scheme.

2.7.2 Penalty scheme

The basis of the penalty method is the assumption that the pressure is proportional to the gradient of the velocity. The approximation can be seen in equation 2.7.1. The penalty number (λ) is typically a large number. The penalty approximation can be seen as a slackening of the incompressibility constraint. Rather the flow is modelled as slightly compressible (Nassehi, 2002).

$$p = -\lambda(\nabla \cdot \nu) \quad (2.7.1)$$

The penalty approximation is substituted into the momentum equation (2.2.4). This eliminates the pressure variables from the equations that have to be solved. If the substitution takes place after discretization the discrete penalty scheme is derived. In the continuous scheme the substitution takes place before the discretization. Equation 2.7.2 shows the continuous penalty scheme.

When equation 2.7.2 is integrated in the finite element method, the penalty terms (λ) must be integrated at reduced Gauss-Legendre quadrature points. So if 3x3 quadrature is required, then 2x2 quadrature should be used.

$$0 = -\nabla(-\lambda(\nabla \cdot \nu))\bar{\delta} + \bar{\tau} \quad (2.7.2)$$

Numerical differentiation is too unstable to recover the pressure solution

directly from equation 2.7.1. Instead a variational recovery method should be used (Nassehi, 2002). The variational recovery method entails solving equation 2.7.3. The penalty terms on the right hand side should be integrated using reduced integration points.

$$\int_{\Omega_e} w \bar{p} d\Omega_e = \int_{\Omega_e} w \lambda \nabla \cdot \bar{v} d\Omega_e \quad (2.7.3)$$

2.7.3 Hele-Shaw flow

The Hele-Shaw method is based on the Reynolds lubrication approximation. It is a very powerful method that allows complex three-dimensional parts to be modelled very efficiently, providing that the parts are of a thin walled nature. For injection moulding this poses no problem, as most injection moulding parts have a thin walled nature to minimise cycle time (Boothroyd *et al.*, 1994). Most of the commercial injection moulding codes today use the Hele-Shaw method.

Two of the first people to use the Hele-Shaw method for injection moulding were Hieber & Shen (1980). A more recent formulation was published by Chiang *et al.* (1991) or Nassehi (2002).

For the Hele-Shaw method to be valid the following criteria must be met.

- The thickness dimension varies very slowly with respect to the other two dimensions, and is a lot smaller.
- The flow must be creeping flow (flow dominated by viscous and pressure terms).
- No-slip on the surfaces.

With these conditions the momentum equation can be approximated as in equations 2.7.4. The momentum equations in the x and y are now integrated and, with some mathematical simplifications, substituted into the mass conservation equation. The result can be seen in equation 2.7.5.

$$\begin{aligned} 0 &= \frac{\partial}{\partial z} \left(\eta \frac{\partial u}{\partial z} \right) - \frac{\partial p}{\partial x} \\ 0 &= \frac{\partial}{\partial z} \left(\eta \frac{\partial v}{\partial z} \right) - \frac{\partial p}{\partial y} \\ 0 &= \frac{\partial p}{\partial z} \end{aligned} \quad (2.7.4)$$

$$\frac{\partial}{\partial x} \left(S \frac{\partial p}{\partial x} \right) + \frac{\partial}{\partial y} \left(S \frac{\partial p}{\partial y} \right) = Q \quad (2.7.5)$$

$$S = \int_0^{\frac{h}{2}} \frac{z^2}{\eta} dz \quad (2.7.6)$$

Equation 2.7.5 has reduced a three-dimensional problem into a two-dimensional problem. The equation is solved for pressure. Velocity is obtained by a post processing operation.

The Hele-Shaw method cannot model lomolding, because it cannot simulate the piston region. The flow in the piston region is not flow in a thin gap between surfaces. The Hele-Shaw method can, however, model the cavity of lomolding. One of the questions that is answered in this thesis is whether Cadmould can model a simplified version of lomolding (only the cavity) and what kind of accuracy and quality can be obtained (section 6.4).

2.8 Free surface method - Volume Of Fluid

A free surface is an interface between air and melt. The free surface, in this thesis, changes with respect to time. Various methods are available which can handle free surfaces.

One of the methods uses line segments to define the flow front. Another method uses marker particles that move with the fluid through the mesh. Hirt & Nichols (1981) give a more detailed comparison of these two methods as well as the method that was used for this thesis, namely the Volume Of Fluid method (referred to as VOF from here on). The VOF method offers the greatest flexibility and is the most popular modern free surface method. The VOF method was pioneered by Hirt & Nichols (1981). More recent publications that use the VOF method were published by Shin & Lee (2000) and Nassehi & Ghoreishy (1998).

The VOF method works on an Eulerian or ALE mesh. Each element is assigned a VOF value, which varies between 0 and 1. A value of 1 represents a full element and 0 an empty element. A value between 0 and 1 represents a partially filled element (thus an element containing a free surface).

A momentum solution is calculated for the complete mesh (both full and empty elements) at each time step. Physical properties are calculated by a

rule of mixtures (equation 2.8.1). The properties of air are used for the empty cells.

$$\eta = VOF \eta_{melt} + (1 - VOF)\eta_{air} \quad (2.8.1)$$

The change of the free surface with respect to time is given by equation 2.8.2. This equation is difficult to solve numerically. One example of how to solve this equation was published by Petera & Nassehi (1996): a Lagrangian mesh is used that moves over an Eulerian mesh.

$$\frac{\partial VOF}{\partial t} + \bar{v} \cdot \nabla VOF = 0 \quad (2.8.2)$$

Hirt & Nichols (1981) and Shin & Lee (2000) avoid solving equation 2.8.2 by using a donor-acceptor scheme. This scheme is a mass conservation scheme applied to each element. At each time step the volume flux is calculated across each element boundary using equation 2.8.4. The fluid transfer factor (F_r) is best explained by referring to figure 2.1. For boundary 4, F_r has a value of 1 as the entire volume flux is composed of fluid. For boundary 2, on the other hand, F_r has a value of 0 because no fluid is transferred. Boundaries 1 and 3 will have an intermediate value. The algorithm for determining this value was published by Shin & Lee (2000).

The free surface is advanced across the domain by updating each element's VOF value (equation 2.8.3). The maximum VOF time step possible is determined by the constraint that an element cannot be over filled ($VOF > 1$) or over emptied ($VOF < 0$). The VOF time step can be quite small, but several VOF cycles can be run until the sum of the VOF time steps is equal to the time step used in the transient energy solution. The volume flux over each elemental boundary is calculated by numerical integration of equation 2.8.4.

$$VOF_{new} = VOF_{old} - \frac{\Delta t}{V_{element}} \sum V_{flux} \quad (2.8.3)$$

$$V_{flux} = F_r \int \bar{v} \cdot \bar{n} d\Gamma_e \quad (2.8.4)$$

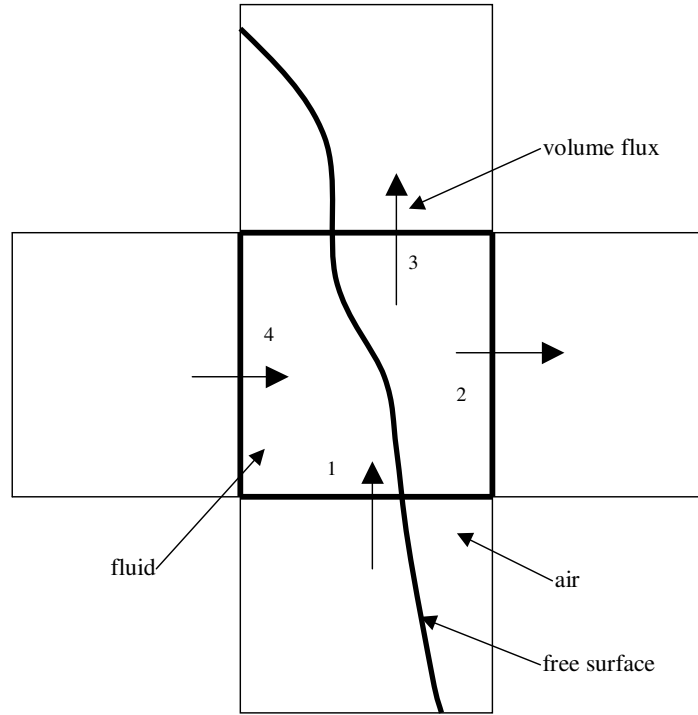


Figure 2.1: VOF element - free surface and volume fluxes

2.9 Energy equation

Polymer flows have high Peclet numbers. The Peclet number ($\frac{\rho c U H}{k}$) is a measure of the convective heat transport relative to the conductive heat transport. The high Peclet number means that the standard Galerkin method is unstable when solving the energy equation (Nassehi, 2002).

Nassehi (2002) recommends using a consistent streamline upwind Petrov-Galerkin scheme. In this scheme the convection term's weight function is replaced by equation 2.9.1

$$W_j = N_j + \alpha \frac{\partial N_j}{\partial x_i} \frac{\bar{v}_i}{|\bar{v}|^2} \quad (2.9.1)$$

where α is the upwinding parameter and N_j is the normal weighting function. Selection of this parameter is not trivial. Nassehi (2002) recommends a trial and error approach, and then maintaining α , so that the scheme is just below the threshold of stability.

2.10 Time dependence techniques

Due to the lack of time derivatives in the simplified momentum equation, the velocities change instantly with time. Therefore the momentum equation does not require a time stepping scheme. The energy equation on the other hand does contain time derivatives and these must be solved using a time dependence method.

Various methods are available for handling time dependant problems. Usually space-time is not discretized. Instead the time derivatives are handled separately. This means that equation 2.10.1 is usually solved by means of a finite difference method. One of the more popular methods is the θ time stepping technique (Nassehi, 2002). Other methods that can also be used are Euler explicit integration and Runge-Kutta 4th order integration.

$$[C]\left\{\frac{dT}{dt}\right\} + [S]\{T\} = \{L\} \quad (2.10.1)$$

2.10.1 Euler explicit integration

The new temperature after a certain time (Δt) can be expressed by a Taylor expansion (equation 2.10.2). The first order derivatives can be calculated from equation 2.10.1. The Euler explicit integration neglects the second order and higher derivatives. The new temperatures can be calculated from equation 2.10.3.

$$T(t + \Delta t) = T(t) + \Delta t \left[\frac{\partial T}{\partial t} \right]_t + \frac{\Delta t^2}{2!} \left[\frac{\partial^2 T}{\partial t^2} \right]_t + \frac{\Delta t^3}{3!} \left[\frac{\partial^3 T}{\partial t^3} \right]_t + \frac{\Delta t^4}{4!} \left[\frac{\partial^4 T}{\partial t^4} \right]_t + O(\Delta t^5) \quad (2.10.2)$$

$$T(t + \Delta t) = T(t) + \Delta t \left[\frac{\partial T}{\partial t} \right]_t \quad (2.10.3)$$

2.10.2 4th order Runge-Kutta integration

The 4th order Runge-Kutta scheme is also based on the Taylor series. It retains 4th order accuracy. The scheme can be seen in equation 2.10.4 (Hirsch, 1988). The first order derivatives can be calculated from equation 2.10.1.

$$\begin{aligned}
T_* &= T(t) + \frac{1}{4}\Delta t \left[\frac{\partial T}{\partial t} \right]_t \\
T_{**} &= T(t) + \frac{1}{3}\Delta t \left[\frac{\partial T}{\partial t} \right]_* \\
T_{***} &= T(t) + \frac{1}{2}\Delta t \left[\frac{\partial T}{\partial t} \right]_{**} \\
T_{t+\Delta t} &= T(t) + \Delta t \left[\frac{\partial T}{\partial t} \right]_{***}
\end{aligned} \tag{2.10.4}$$

2.10.3 θ time stepping scheme

The stiffness matrix, mass matrix and load vector of equation 2.10.1 change with time. An explicit scheme takes the current matrixes and vectors (at time step n) to calculate the new temperature (at time step $n + 1$). An implicit scheme takes only the matrixes and vectors at the new time step to calculate the new temperature. The θ time stepping scheme uses the matrixes and vectors at some intermediate point ($0 \leq \theta \leq 1$) between time step n and $n + 1$.

The $\frac{dT}{dt}$ term is approximated as equation 2.10.5. The stiffness matrix at time θ is found by equation 2.10.6. Other terms are found in a similar manner.

$$\left[\frac{dT}{dt} \right]_{\theta} = \frac{\{T\}_{n+1} - \{T\}_n}{\Delta t} \tag{2.10.5}$$

$$[S]_{\theta} = (1 - \theta)[S]_n + \theta[S]_{n+1} \tag{2.10.6}$$

Once these approximations are made, they can be substituted back into equation 2.10.1, which can be simplified to equation 2.10.7. This is a linear set of equations. The $n + 1$ terms are not known at time n . They can be determined by solving equation 2.10.7, using an approximation for the $n + 1$ terms. Thus it becomes an iterative process, where the old terms are used to find new terms which are closer to the solution. The iterative process continues until a specified level of convergence has been reached.

For $\theta = 1$ the scheme becomes a implicit scheme, and for $\theta = 0$ an explicit scheme. If $\theta = 0.5$ the scheme becomes a second order central difference scheme (Nassehi, 2002).

$$\begin{aligned}
([C]_{\theta} + \theta \Delta t [S]_{n+1}) \{T\}_{n+1} = \\
([C]_{\theta} - (1 - \theta) \Delta t [S]) \{T\}_n + ((1 - \theta) \{L\}_n + \theta \{L\}_{n+1}) \Delta t
\end{aligned}
\tag{2.10.7}$$

2.11 Solidification

There are two methods for modelling the solidification of melt against the wall. The first is to remesh the domain and the second is to modify the viscosity.

Shen (1998) used an ALE method to model injection moulding. At each time step the domain was remeshed: the boundary nodes were placed on the solid/liquid interface (where the temperature was equal to the melting temperature).

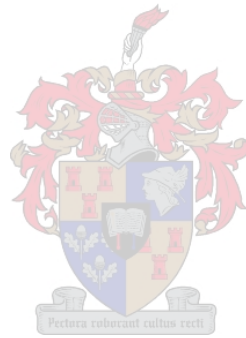
Remeshing is not an attractive option, because errors creep in due to the interpolation. Such errors can be seen in the conservation of mass (Wang and McLay, 1986) and energy. Numerical diffusion takes place because of the remeshing.

Ravindran & Lewis (1998) modelled metal casting using a stationary Eulerian mesh. They adjusted the viscosity of the melt where it solidified (equation 2.11.1). The viscosity varies between the liquid viscosity and a viscosity for the solid. The fraction of the solidified melt in the element is used as the exponential scaling value. The advantage of this method is that the same mesh can be used for each time step, and that no errors are introduced through remeshing.

$$\eta_{effective} = \eta_l \left(\frac{\eta_s}{\eta_l} \right)^f
\tag{2.11.1}$$

A slightly different version of this viscosity model was used for this thesis. The equation can be seen in 2.11.2. Here the effective viscosity ($\eta_{effective}$) is adjusted if it drops below the melting temperature (T_m). Using this model it is possible to vary the viscosity in an element itself. To determine the best solidification factor (a) a series of simulations were run and solid layer thickness is compared to other numerical results. This is the subject of appendix 4.4.

$$\begin{aligned}\eta_{effective} &= \eta_l && \text{if } T > T_m \\ \eta_{effective} &= \eta_l(1 + a(T_m - T)^2) && \text{if } T \leq T_m\end{aligned}\tag{2.11.2}$$



Chapter 3

Implementation

This chapter looks at the implementation of the theory, discussed in the literature review chapter, and the programming of Sopfil (Simulation of polymer flow in lomolding). The main algorithms will be explained by means of flowcharts and discussions.

The validation of Sopfil can be seen in chapter 4.

3.1 Choosing a momentum scheme

Various momentum solutions methods exist, which have different characteristics. Some of the methods do not enforce the mass conservation equation strongly enough to use with the VOF method, and other methods return a bad pressure field in regions where solidification occurs. This section shows how a momentum scheme was chosen.

The three schemes considered can be seen in the list below. The UVP and penalty schemes were introduced in the literature review chapter. The details of the elements can be found in section B.1.

- Penalty method (using 9 node quadratic elements)
- UVP scheme (using Taylor-Hood elements)
- UVP scheme (using Crouzeix-Raviart elements)

To evaluate the schemes a partially filled injection moulding (50x3mm, 20x11 elements) mesh with a developed thermal field was used as the starting point for each scheme. Each scheme was then used to calculate the momentum

solution and advance the flow front. The convergence rate and level, pressure field and mass conservation were compared. The pressure field is important as it will be compared with experimental data.

The mass conservation is important for the VOF method: consider an element that is completely filled, yet has a positive net inflow of mass. A zero maximum VOF time step will be calculated. This constraint can be slackened by allowing elements to be slightly over filled or over emptied. But a scheme with the best mass conservation will have the least mass being created or destroyed in the mesh, and will be easier to implement.

Figure 3.1 shows the rate and level of convergence for the different schemes. The rate of convergence for all three methods are similar, but the UVP scheme with Crouzeix-Raviart elements has a better level of convergence (2 - 4 orders of magnitude).

The mass conservation results can be seen in table 3.1. The error was determined by equation 3.1.1, which expresses the mass created as a fraction of the absolute sum of the mass crossing the element's boundaries. The mass conservation was calculated for every element. The UVP scheme using Taylor-Hood elements performed badly. The reason for this seems to be the weak enforcement of the mass conservation equation (Pittman, 1989). The Crouzeix-Raviart UVP scheme and the penalty method perform equally well.

$$mass\ created = \frac{\oint \bar{v} \cdot \bar{n} d\Gamma}{\oint |\bar{v} \cdot \bar{n}| d\Gamma} \quad (3.1.1)$$

The UVP schemes produce better pressure fields. The penalty scheme produces a good pressure field for a mesh that does not have any sharp corners or solidification in it, otherwise extreme pressure spikes occur which ruin the pressure for the whole field. The UVP scheme using Crouzeix-Raviart elements has less severe pressure spikes at the solid layer, and they are localised and do not affect the rest of the field. This is probably due to the discontinuous nature of pressure fields in Crouzeix-Raviart meshes.

In conclusion, the scheme that best satisfies the criteria is the UVP scheme using Crouzeix-Raviart elements. This scheme was used for the remainder of the thesis.

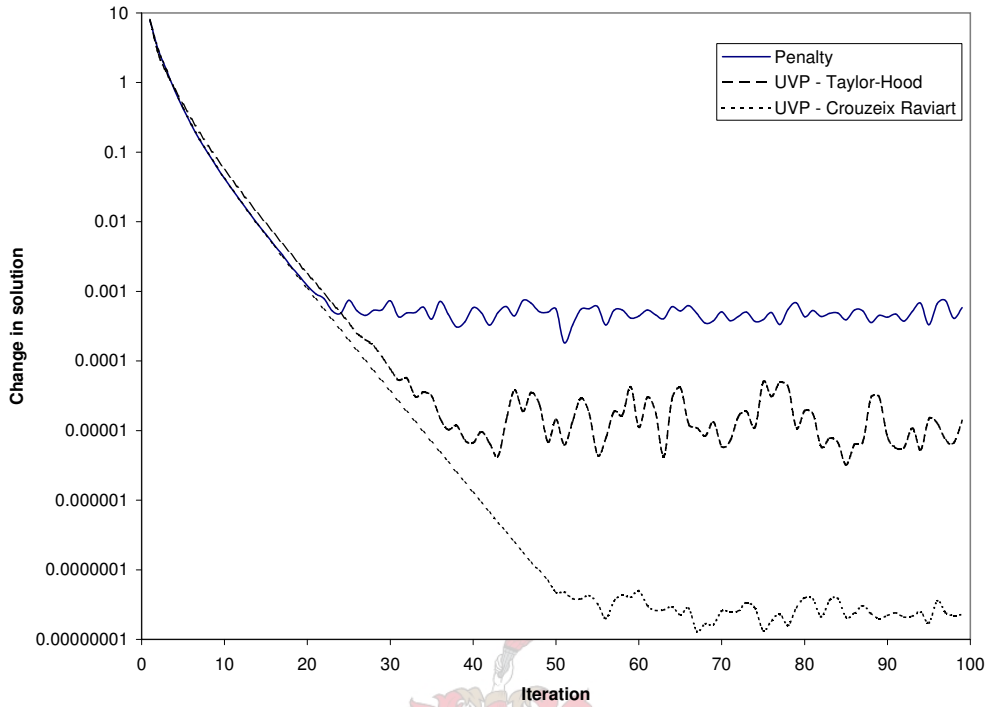


Figure 3.1: Convergence rate and level for different momentum schemes

mass created	penalty	UVP - Taylor-Hood	UVP - Crouzeix-Raviart
average (%)	0.00227	5.314	0.002094
minimum (%)	0	0.002962	0
maximum (%)	0.0509	76.67	0.04954

Table 3.1: Mass conservation of different momentum schemes

3.2 Sopfil flowchart

The main solution routine in Sopfil can be seen in figure 3.2. Various steps in the flowchart (e.g. the momentum solution) require a flowchart of their own. These flowcharts are shown in the next sections.

In the Sopfil flowchart momentum energy dependence is solved by the θ time stepping scheme. The loop for the scheme can be seen in figure 3.2. Note that in the loop the momentum solution is an iterative process itself.

Three classes of outputs can be written at every step:

- Results for the whole mesh
- Results for selected nodes

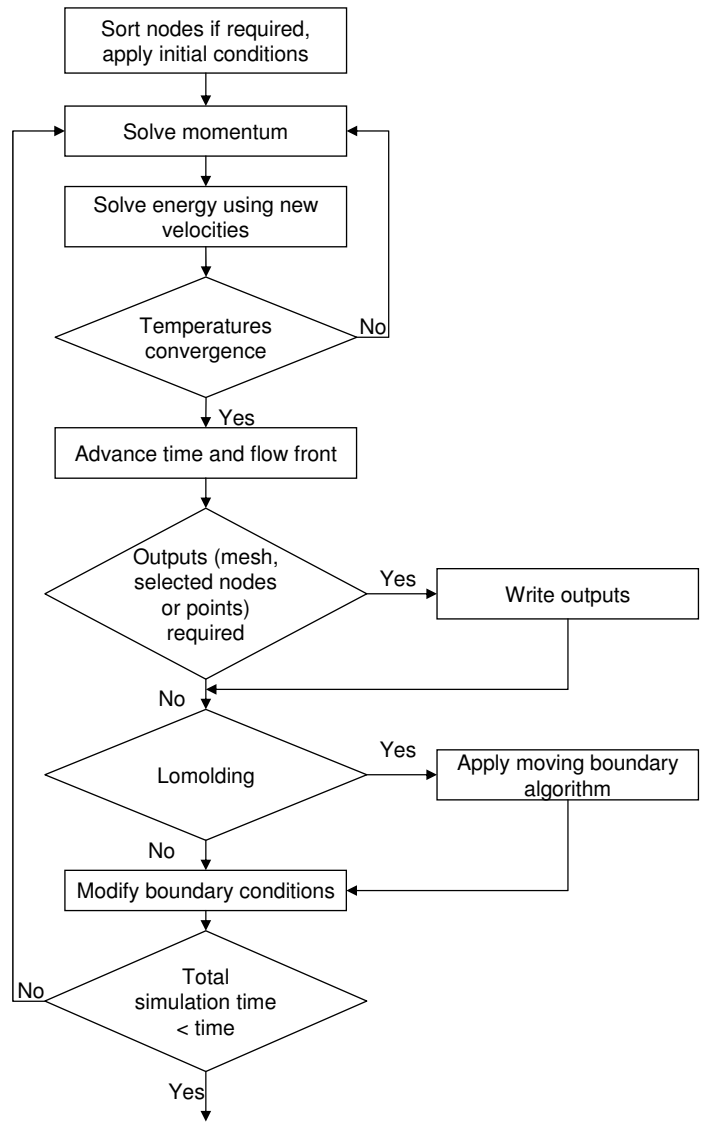


Figure 3.2: Sopfil flowchart

- Results for specified points in the mesh

The specified points are given in global coordinates. Before results can be written the local coordinates and element numbers must be found for each point. This search procedure is the topic of section D.7.

The boundary conditions can also be modified in every time step. This allows for variable injection rates and other time dependant process parameters. The moving boundary algorithm is applied after the momentum and energy solution.

3.3 Momentum flowchart

The momentum flowchart can be seen in figure 3.3. The flowchart is for the UVP scheme. The penalty scheme's flowchart will look slightly different, as the pressures are obtained in a post processing operation. The momentum solution is an iterative process, due to the generalised Newtonian nature of polymer flow.

For the first iteration (of the whole simulation) Newtonian viscosity is assumed. In every subsequent time step the previous time step's velocity solution is used as the starting value. Once a velocity solution has been found new shear rates and viscosities are calculated and thus a new solution can be calculated. The convergence criteria can be seen in section D.3.

3.4 Energy flowchart

The flowchart for the energy solution can be seen in figure 3.4. If the θ time stepping method is used, the energy equation has to be solved in the iterative loop that is in the main flowchart (figure 3.2).

3.5 Moving boundary flowchart

The moving boundary algorithm is used for the piston which advances down the moulding barrel. The moving boundary is achieved by using an ALE scheme in the moulding barrel. The nodes on the piston face are assigned a mesh velocity that is equal to the piston velocity. The nodes at the piston-cavity interface have a zero mesh velocity, as do all the cavity nodes. The nodes

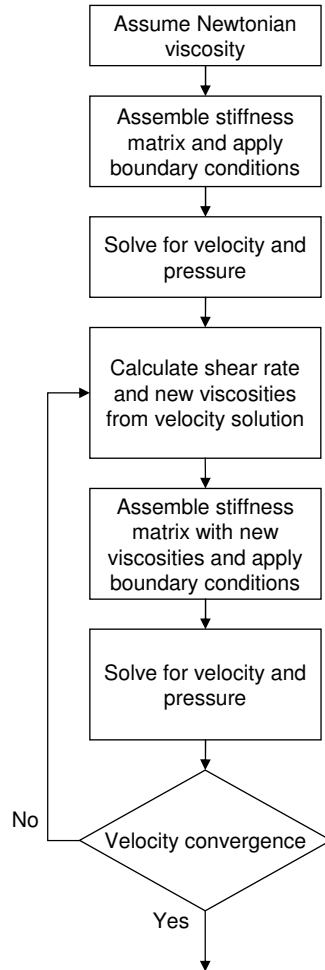


Figure 3.3: Momentum scheme flowchart

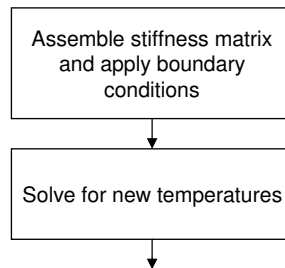


Figure 3.4: Energy flowchart

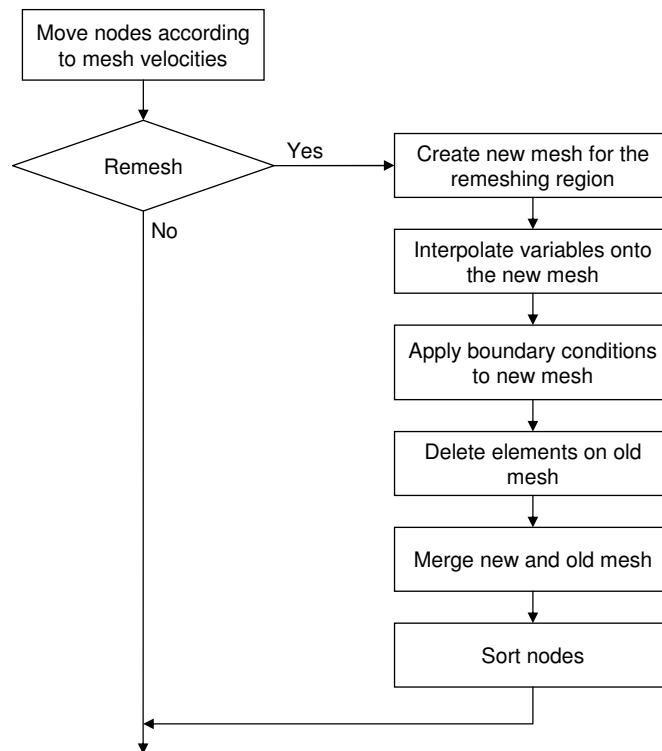


Figure 3.5: Moving boundary flowchart

in the moulding barrel have mesh velocities which are interpolated between the piston face and cavity mesh velocities.

The elements and nodes can become cramped up as the piston gets closer to the cavity. If this happens the mesh in the moulding barrel can be remeshed. A flowchart of the algorithm can be seen in figure 3.5.

3.6 VOF flowchart

The flowchart for the VOF process can be seen in figure 3.6. There is a limit to the size of a VOF time step, because an element may not be over filled or over emptied. This limit can result that the VOF time step is quite small (usually much smaller than the time steps in the energy solution scheme). However, several VOF time steps can be run consecutively until the sum of the VOF time steps is equal to the time step size specified by the energy solution scheme.

Every time the VOF of an element changes, new (F_r) values must be calculated. New volume fluxes are not calculated, as this would require a new

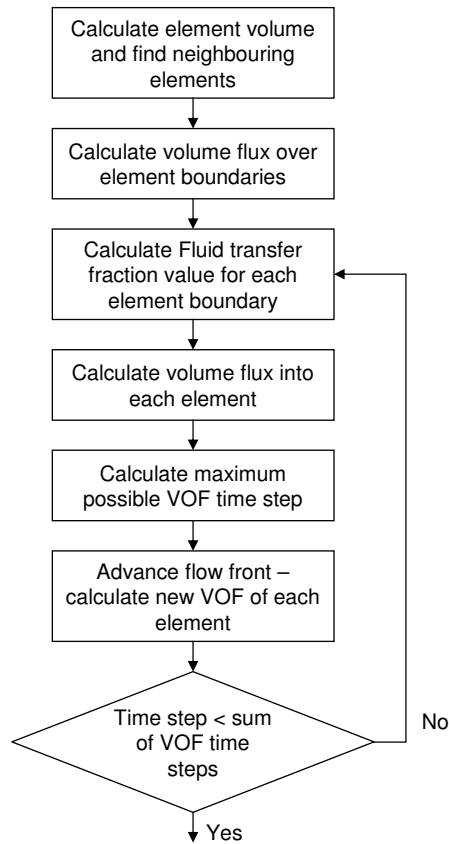


Figure 3.6: VOF flowchart

momentum solution and would be too computationally expensive. Typically there are 5 to 400 VOF steps for every time step.

Using the VOF method with a no-slip boundary condition, a layer of air is trapped between the melt and the cavity wall. The layer of elements closest to the wall are never completely filled, which means that the melt is flowing in-between air cushions. This is not realistic, and creates unrealistic pressure and temperature fields.

Applying a wall slip boundary condition to the empty and partially filled elements resolves this problem. The partially filled elements along the wall offer no resistance (shear stress) to the air flowing out of them in the direction of the wall. Therefore the no-slip boundary conditions allow the layer of trapped air to escape. At every time step the boundary element's boundary conditions are updated. If the boundary element is filled, its boundary conditions are changed to no-slip. Shen (1998) also applied wall slip boundary conditions at

the contact point (between the wall and the flow front) in his ALE analysis of injection moulding.

3.7 Modelling lomolding with Cadmould

In the literature review it was mentioned that Cadmould cannot model the piston region of lomolding. It can, however, model the cavity region. There are two possible mechanisms that could cause the pressure drop in the cavities to differ for same volume flow rate and similar cavities.

The first is the slightly different temperature profiles of the melt entering the cavity. In lomolding the moulding barrel will change the melt temperature slightly. The barrel has a large diameter, thus should not change the temperature too much. With the Cadmould simulation the melt will enter the cavity where the moulding barrel joins the cavity at the injection temperature.

The second mechanism that could change the pressure drop is the viscoelastic nature of the polymer. Viscoelastic effects should only have an effect in regions where there is an abrupt change in the geometry. Such regions would be where the cavity meets the moulding barrel in lomolding and the injection moulding gate. Both of these regions are small and the effects should dissipate quickly. The pressure drop in the rest of the cavity, away from these regions, should be the same.

The Cadmould domain can be seen in figure 3.7. The part shown was the experimental disk, which is discussed in chapter 5. Only a quarter of the disk was modelled. The region under the piston was not modelled. Melt enters the cavity by a line of injection nodes placed on the moulding barrel/piston interface. The volume flow rate of the injection nodes can be calculated from the velocity of the piston.

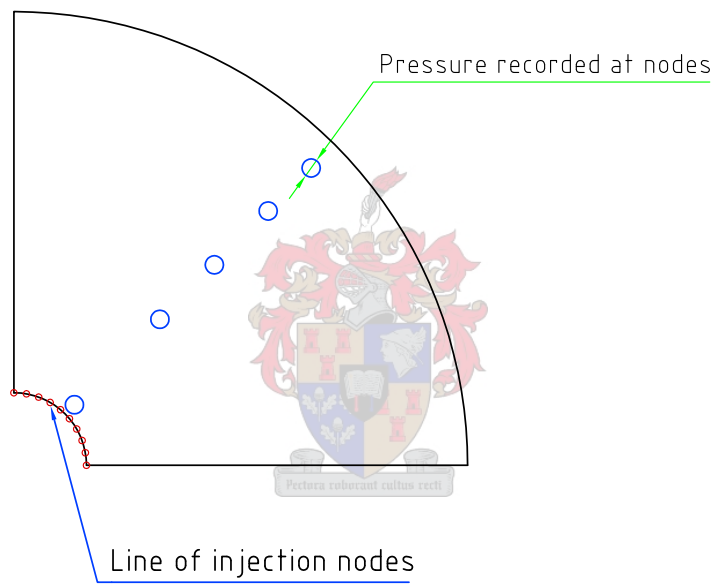


Figure 3.7: Modelling lomolding with Cadmould

Chapter 4

Validation of code

This chapter will discuss the various cases that were used to validate different aspects of Sopfil. Various aspects of the code are validated against analytical solutions or published numerical results if analytical solutions do not exist.

4.1 Momentum

4.1.1 Newtonian flow

For creeping flow (see equation 2.2.4) it is possible to derive an analytical solution for the flow over a sphere. The tangential and radial velocities can be calculated from equations 4.1.1 and 4.1.2 (Churchill, 1988). The pressure on the surface of the sphere can be calculated from equation 4.1.3. The characteristic length for the Reynolds number is the diameter of the sphere.

$$U_r = U_\infty \left(1 - \frac{3r_s}{2r} + \frac{1}{2} \left(\frac{r_s}{r} \right)^3 \right) \cos \theta \quad (4.1.1)$$

$$U_\theta = -U_\infty \left(1 - \frac{3r_s}{4r} - \frac{1}{4} \left(\frac{r_s}{r} \right)^3 \right) \sin \theta \quad (4.1.2)$$

$$\frac{2(p(\theta) - p_\infty)}{\rho U_\infty^2} = \frac{6 \cos(\theta)}{Re_D} \quad (4.1.3)$$

The computational domain is depicted in figure 4.1. On the sphere no-slip boundary conditions are applied. The velocity boundary conditions on the outer boundary are problematic; the free stream velocity cannot be applied. Because the flow is creeping flow, the velocity is affected an infinite distance

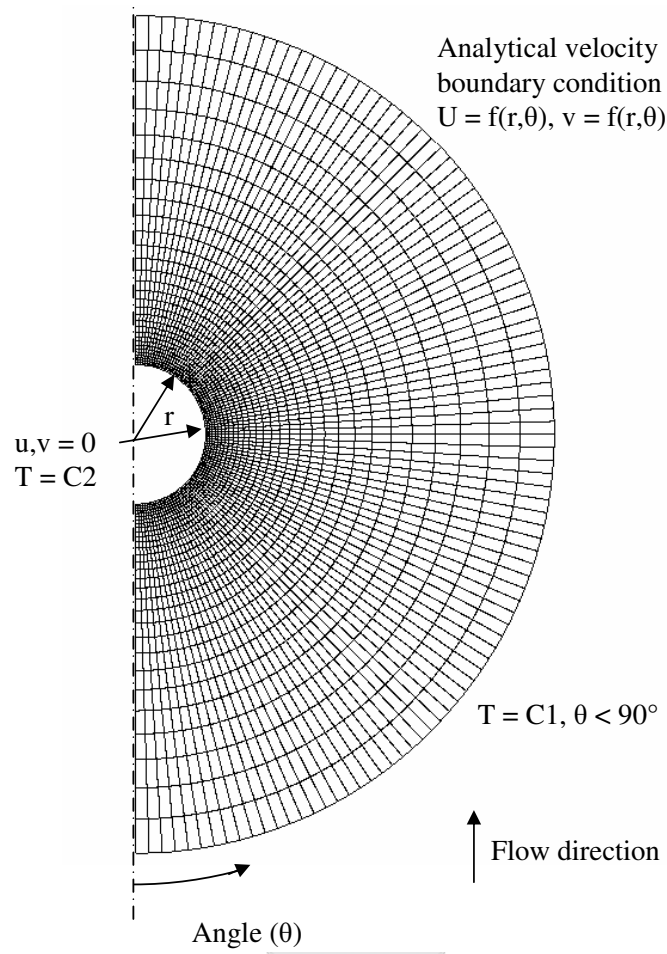


Figure 4.1: Sphere mesh and boundary conditions

away from the sphere. The velocity at a distance 10 times the sphere radius ($\theta = 90^\circ$) is only 85% of the free stream velocity. To escape having to use a very large mesh, the analytical velocity solution (equation 4.1.1 and 4.1.2) was applied as the boundary conditions. The temperature boundary conditions depicted in the figure are not used in this validation case, but are used later in section 4.2.

The comparison of equation 4.1.3 and the numerical pressures calculated can be seen in figure 4.2. There is a slight error at $\theta = 0$, where there is a stagnation point.

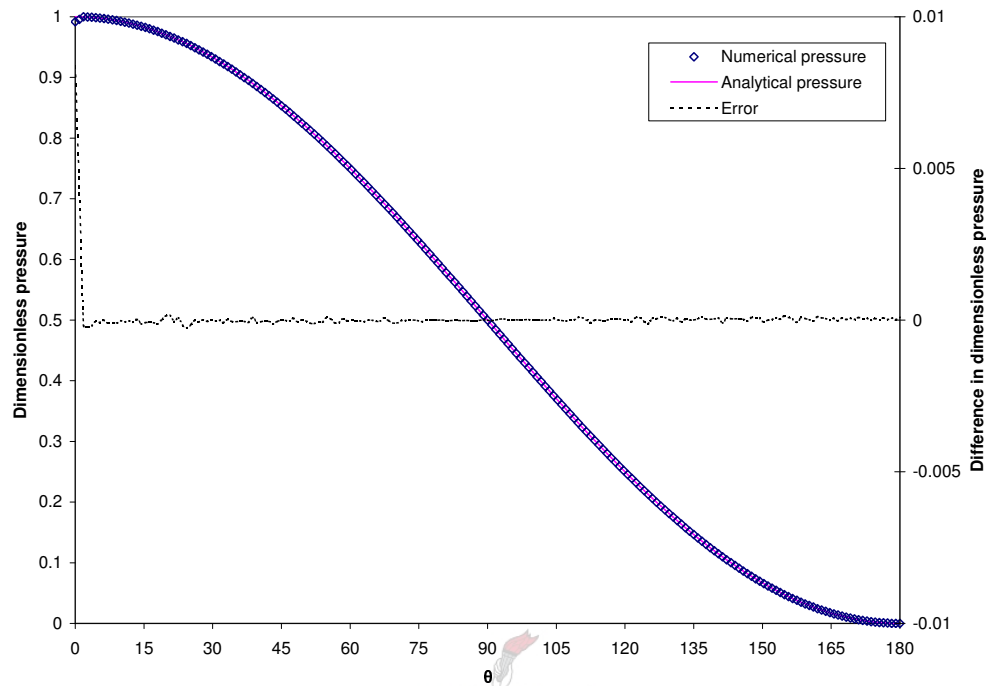


Figure 4.2: Pressure distribution on the surface of a sphere

4.1.2 Generalised Newtonian flow

Using a power-law model (see equation F.1.1) it is possible to derive some analytical solutions for simplified geometries. Examples of such geometries, for an axisymmetric coordinate system, are flow in a pipe and radial flow between parallel plates.

For a pipe the velocity distribution is given by equation 4.1.4 (Lee & Castro (1989) and Dealy & Wissbrun (1999)). The volume flow rate is given by equation 4.1.5. If the volume flow rate is known the equation can be rewritten to solve for the pressure drop along a section of pipe. The domain and mesh can be seen in figure 4.3. For flow between parallel plates the fully developed velocity profile and pressure drop are given by equations 4.1.6 and 4.1.7 (Lee & Castro (1989)). The domain and mesh can be seen in figure 4.4.

Thus two aspects can be validated from these equations. The first one is to check the shape of the velocity profile at the exit of the pipe or plates if a step input is specified ($v(r) = \text{constant}$). The second is to specify the fully developed velocity profile at the inlet and then check the pressure drop along the length of the pipe.

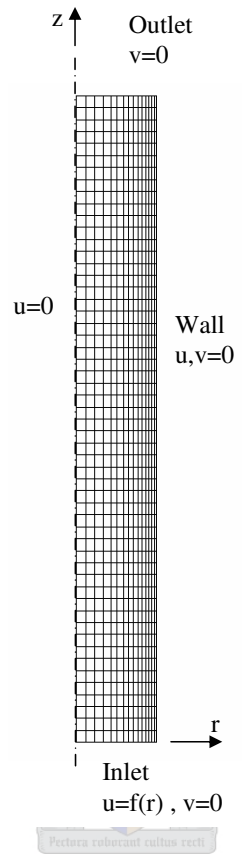


Figure 4.3: Pipe mesh and boundary conditions

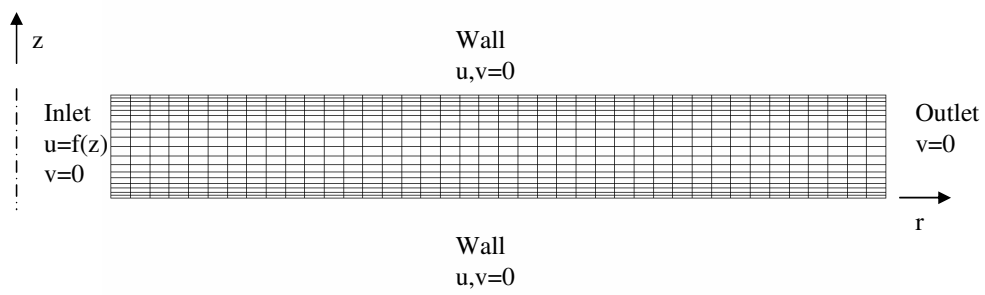


Figure 4.4: Parallel plates mesh and boundary conditions

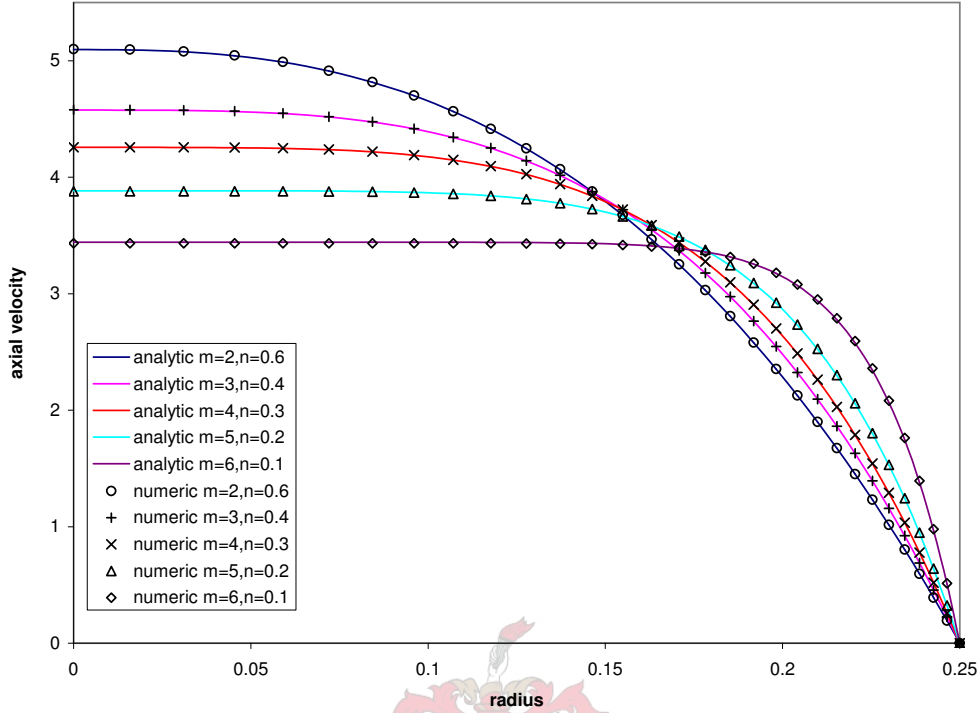


Figure 4.5: Pipe velocity profiles - power-law fluid

$$v(r) = \frac{nR}{1+n} \left(\frac{R\Delta p}{2mL} \right)^{\frac{1}{n}} \left(1 - \left(\frac{r}{R} \right)^{1+\frac{1}{n}} \right) \quad (4.1.4)$$

$$Q = \frac{n\pi R^3}{1+3n} \left(\frac{R\Delta p}{2mL} \right)^{\frac{1}{n}} \quad (4.1.5)$$

$$u(z) = \frac{1}{r} \frac{\left(\frac{t}{2} \right)^{1+\frac{1}{n}} - z^{1+\frac{1}{n}}}{1 + \frac{1}{n}} \frac{1+2n}{4\pi n \left(\frac{t}{2} \right)^{2+\frac{1}{n}}} Q \quad (4.1.6)$$

$$P_i = \frac{m}{1-n} \left(\frac{1+2n}{4\pi n \left(\frac{t}{2} \right)^{2+\frac{1}{n}}} \right)^n (R_o^{1-n} - R_i^{1-n}) Q^n \quad (4.1.7)$$

The comparison of exit velocity profiles for a pipe is shown in figure 4.5. For viscosities showing less exponential behaviour ($0.4 < n < 0.6$) the biggest error was 0.05% and for the more exponential cases ($n = 0.1$) the biggest error was 1.2%, at the region near the wall.

The pressure drop along the pipe was calculated by taking the difference of the average inlet and outlet pressure. The average pressure was calculated by

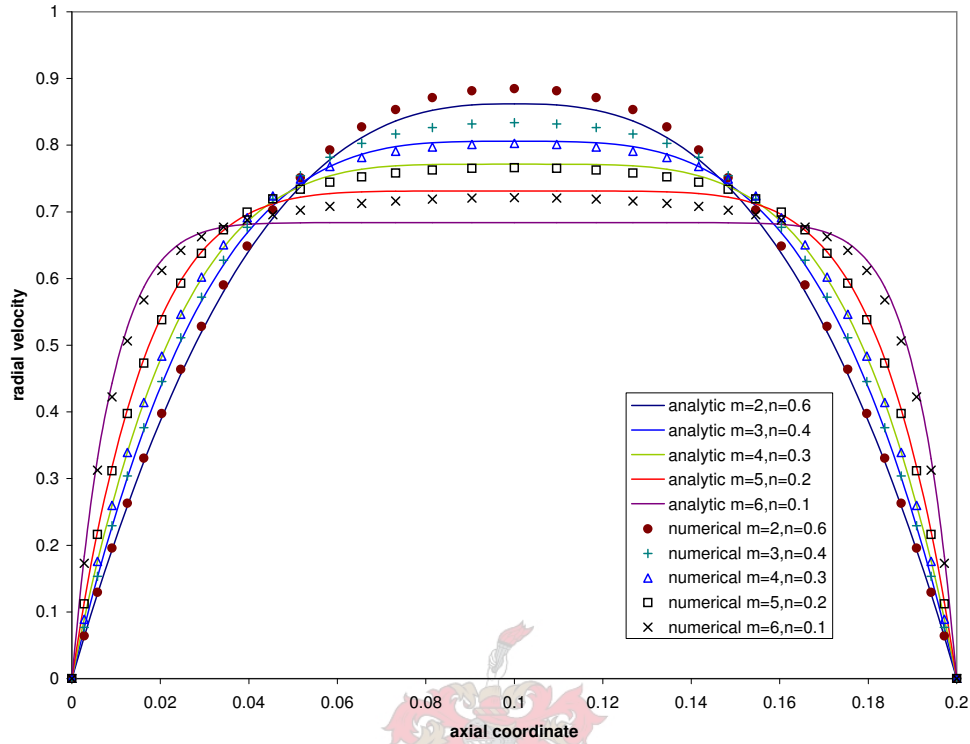


Figure 4.6: Parallel plates velocity profiles - power-law fluid

equation 4.1.8. For power-law parameters of $n = 0.6$, $m = 2$ the error in the calculated pressure was 1.4%. If the pressure drop is calculated a little away from the inlet of the pipe the error reduces to 0.019%.

$$p_{ave} = \frac{\int p \, dA}{\int dA} \quad (4.1.8)$$

The velocity profiles for radial flow between parallel plates do not compare as well as the pipe profiles. The profiles are not flat enough: the velocities near the centre are too large, and near the wall too small. The largest error is about 5%. They are, however, acceptably accurate. The profiles can be seen in figure 4.6.

The pressure drop in the flow between the parallel plates was calculated in a similar manner as the pipe case. Power-law parameters of $n = 0.6$, $m = 2$ were used. There was a 1.9% error across the total length of the plates, but if the pressure drop was calculated from a region slightly away from the entrance to the outlet, the error was reduced to 0.15%.

4.2 Energy-flow over a sphere

Abramzon & Elata (1984) published numerical results for a hot sphere (of constant temperature) submerged in cold Stokes flow. One of the results they published was the steady state local Nusselt number vs. the sphere angle.

The local Nusselt number is defined in equation 4.2.1. The dimensionless temperature gradient is calculated on the surface of the sphere (from the fluid side), and the r in this equation is the spherical coordinate system radius. The dimensionless temperature is defined in equation 4.2.2. The angle θ is defined from the forward facing part of the sphere. The computational domain that was used for the pressure distribution on the surface of a sphere is also used for this case, except that the outer boundary radius is made smaller (3 sphere radii) (figure 4.1). On the sphere a no-slip boundary condition and a constant temperature is applied. On the outside of the computational domain a constant temperature is applied for $\theta < 90^\circ$. For the rear half of the domain ($90^\circ \leq \theta \leq 180^\circ$) a zero temperature gradient is applied (which is the default boundary condition in Sopfil's FEM if no other boundary condition is applied).

$$Nu_\theta = -\frac{2}{Z_s} \left(\frac{\partial Z}{\partial r} \right)_{r=sphere} \quad (4.2.1)$$

$$Z = \frac{T - T_\infty}{T_{sphere} - T_\infty} \quad (4.2.2)$$

$$Pe = \frac{2Ur_{sphere}}{\alpha} \quad (4.2.3)$$

The numerical results can be seen in figure 4.7 for a Peclet number of 300 (equation 4.2.3). The biggest error is of the order of 3% (for the fine mesh). It should be kept in mind that although the error seems quite large, the results are gradients of the temperature and therefore will magnify any small error in the solution. Also, the Abramzon & Elata (1984) results are numerical, and therefore could also contain an error.

4.3 Time dependance

The simplified momentum equation used in Sopfil is quasi steady state, and therefore not time dependant. The energy equation on the other hand is time dependant. Two validation cases are described in this section. The first is the

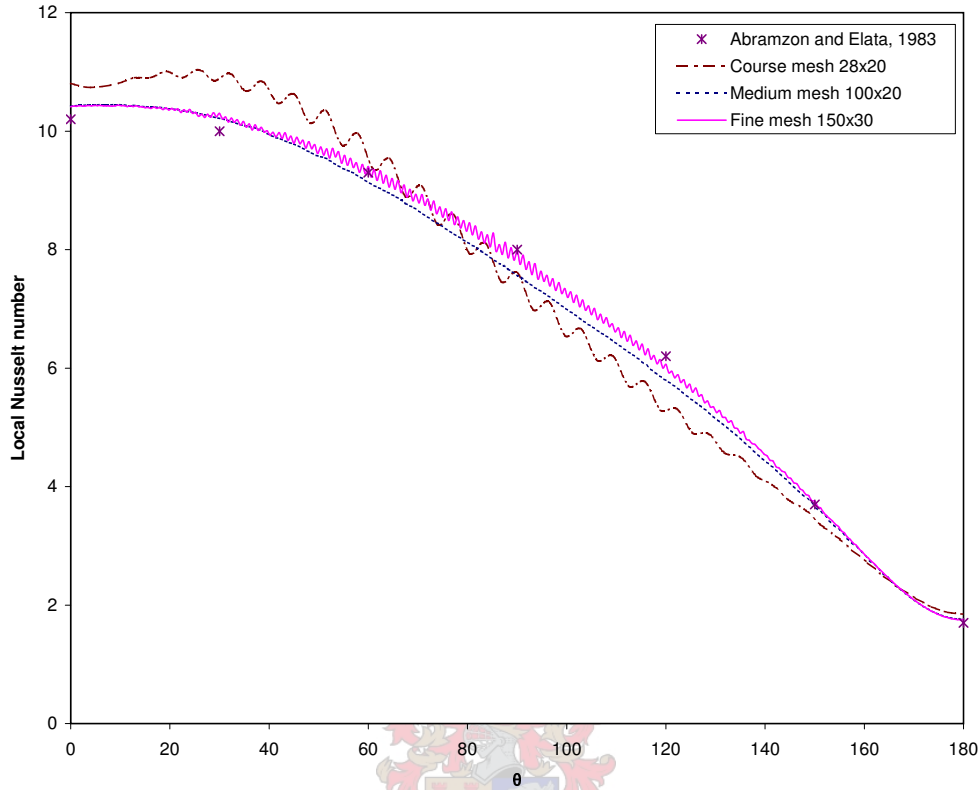


Figure 4.7: Sphere local Nusselt number vs. angle - $Pe = 300$

cooling of a solid slab and the second is the developing thermal field around a hot sphere submerged into a fluid stream.

4.3.1 Cooling of a solid slab

If a solid slab of a constant initial temperature is submerged into a cold medium, the thermal response of the slab is given by equation 4.3.1 (Mills, 1995). The domain can be seen in figure 4.8. A zero velocity field is specified for the whole domain as the slab is a solid. The surface of the slab has a constant temperature equal to that of the cooling medium. The other boundaries were given natural (zero gradient) boundary conditions. The results can be seen in figure 4.9.

$$\frac{T - T_s}{T_0 - T_s} = \sum_{n=0}^{\infty} \frac{2(-1)^n}{(n + 0.5)\pi} e^{-(n+0.5)^2\pi^2 Fo} \cos\left((n + 0.5)\pi \frac{z}{L}\right) \quad (4.3.1)$$

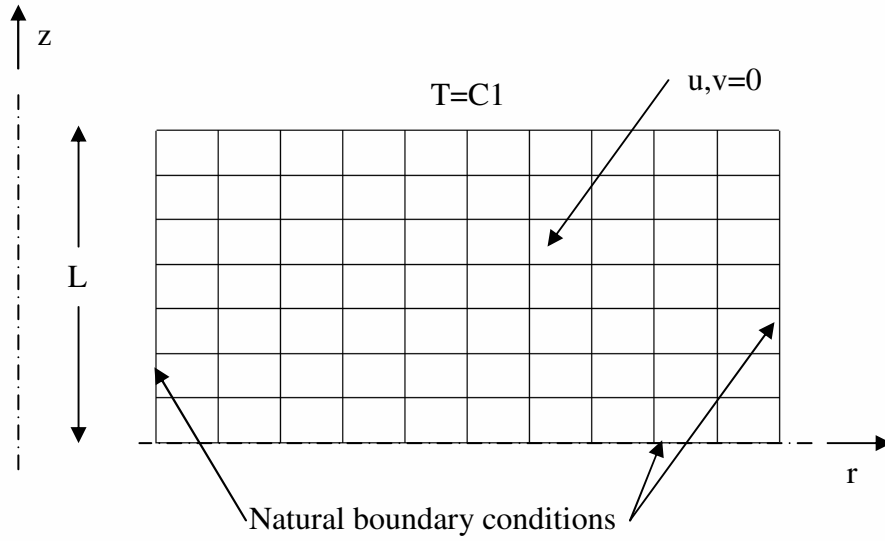


Figure 4.8: Solid slab domain

$$Fo = \frac{\alpha t}{L^2} \quad (4.3.2)$$

4.3.2 Developing thermal field around a hot sphere in cold fluid stream

This validation study uses the same domain as section 4.2; the reference calculation was also published by Abramzon & Elata (1984). In this validation case the unsteady developing thermal field is simulated. The sphere has a constant temperature throughout the calculation. Initially the fluid is at a uniform lower temperature. As the calculation starts the fluid flows over the sphere and temperature gradients form. From these gradients the Nusselt numbers can be calculated.

A surfaced averaged Nusselt number can be calculated from equation 4.3.3 (Abramzon and Elata, 1984). This value is initially very large (when the gradients are very big), but settles down to a constant steady state value after a period of time. Calculations were done for Peclet numbers of 0,10,100,300 and 1000 (as defined in equation 4.2.3).

$$Nu = -\frac{1}{Z_s} \int_0^\pi \left(\frac{\partial Z}{\partial r} \right)_{r=sphere} \sin \theta d\theta \quad (4.3.3)$$

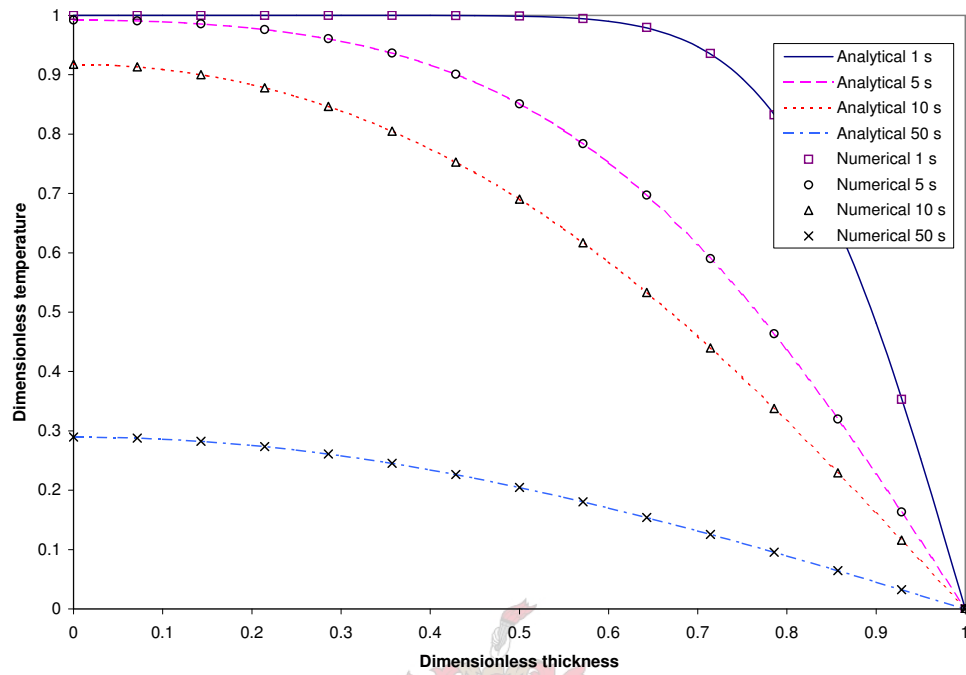


Figure 4.9: Cooling of a solid slab

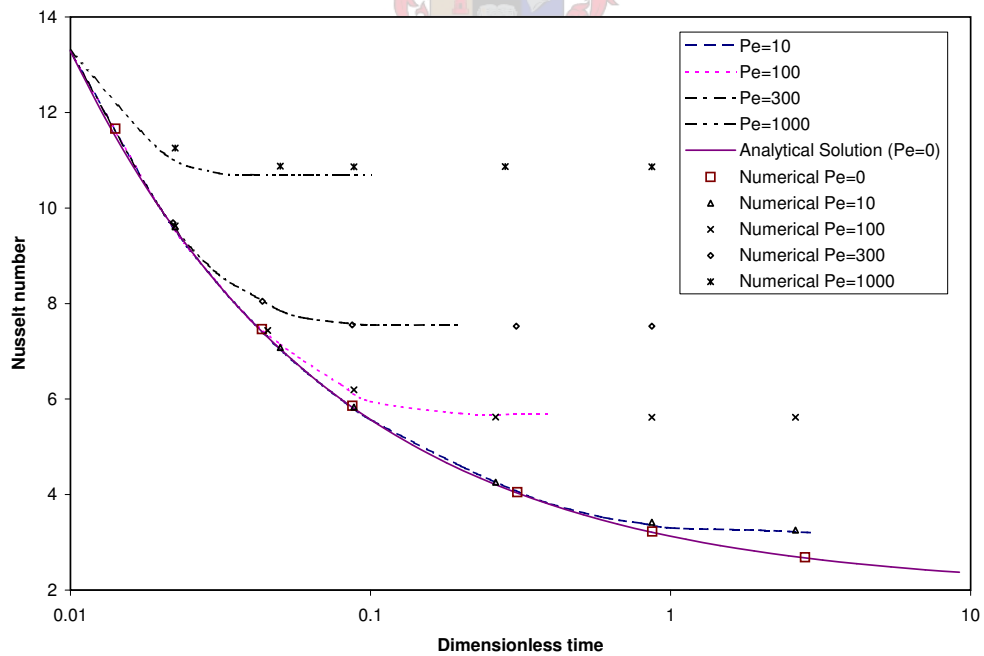


Figure 4.10: Surfaced averaged Nusselt number of a hot sphere

The results can be seen in figure 4.10. Sopfil does well at lower Peclet numbers, but starts to lose accuracy at higher Peclet numbers. This may be due to the artificial diffusion introduced into the solution by the upwinding.

4.4 Solidification and determining the optimal solidification factor

Yang *et al.* (1991) developed a steady state numerical solution for polymer flow in a tube with solidification. The polymer viscosity was modelled by a power-law model. Results for both a temperature dependant and independent viscosity were given. Yang *et al.* (1991) validated his model against Richardson (1983), who developed an analytical solution for the same problem with a temperature independent power-law viscosity.

It is possible to have a steady state solution without the whole pipe freezing over. The core of the pipe is kept hot enough by the viscous shear heating of the polymer. Thus the heat generated by shear heating must be equal to the heat conducted away through the solid layer. These were the assumptions that Yang *et al.* (1991) made for his research.

The problem was simulated on Sopfil with a transient simulation. The simulation was stopped after a time when the growth of the solid layer became very small. This was found to happen after about 50 seconds. Two mesh sizes were investigated, namely a coarse (7x66 elements) and a fine mesh (14x121 elements). Different solidification factors (a) were used on the coarse mesh to determine the optimal value. This was done by comparing the solid layer thickness to the results of Yang *et al.* (1991) and Richardson (1983). The results can be seen in figure 4.11. The optimal solidification factor was found to be 0.01.

To illustrate the effects of the solidification, figure 4.12 is included. The figure shows velocity and temperature profiles for flow in a pipe. The solidification temperature is also marked on the figure. The velocity is zero while the temperature is below the solidification temperature. When the temperature increases above the melt temperature the velocity is no longer zero. Note that as the solid layer gets thicker, the velocity in the centre of the pipe increases. This is due to mass conservation: what flows into the pipe must also flow out.

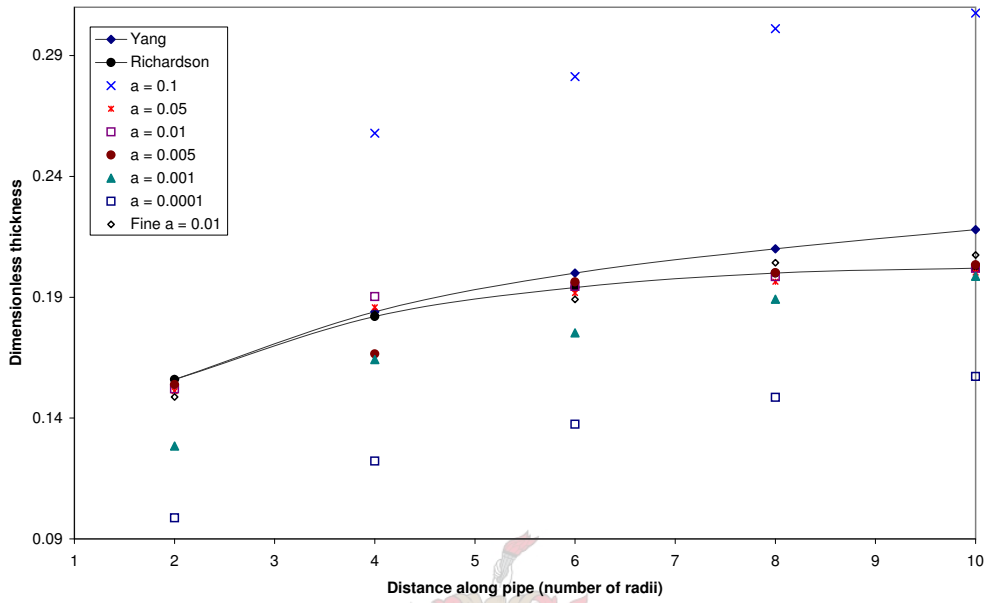


Figure 4.11: Determining the value of the solidification factor

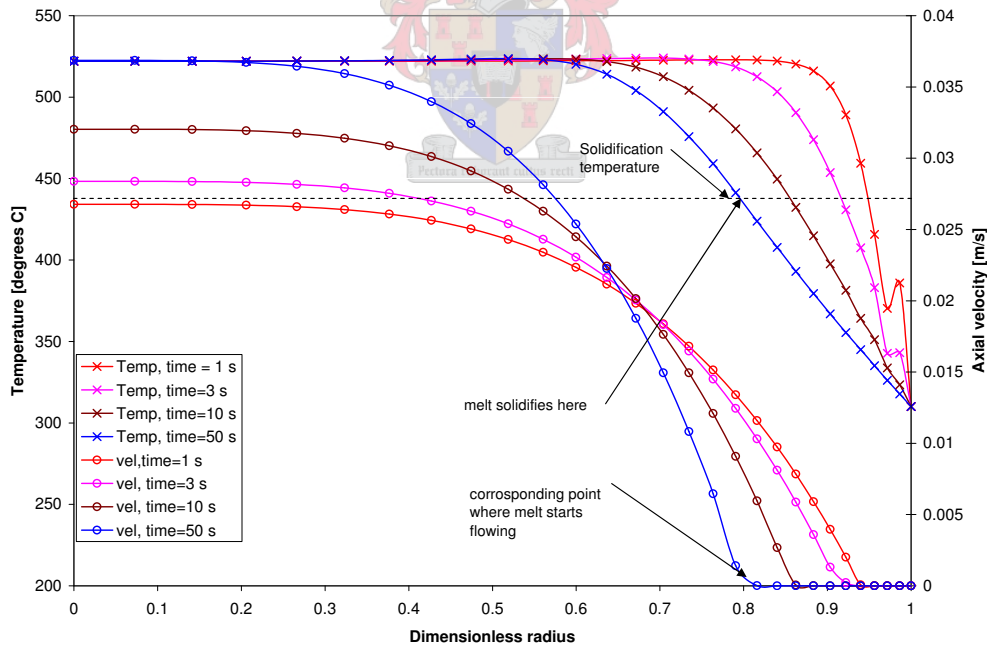


Figure 4.12: The effects of solidification on the velocity profiles in a pipe

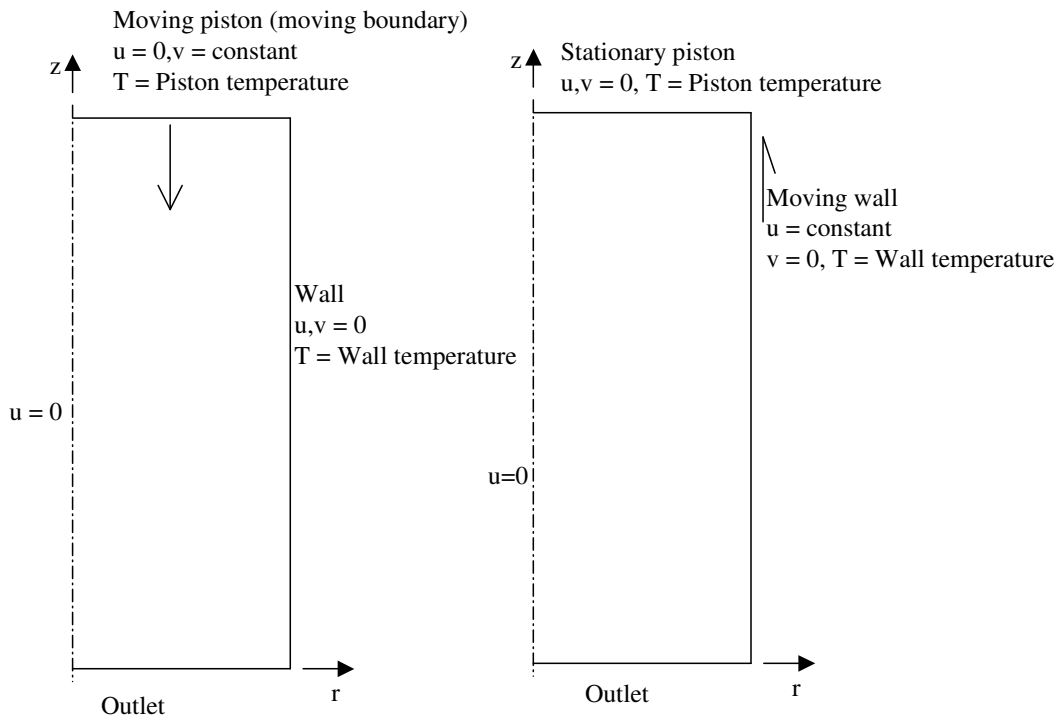


Figure 4.13: Moving piston and moving wall domain

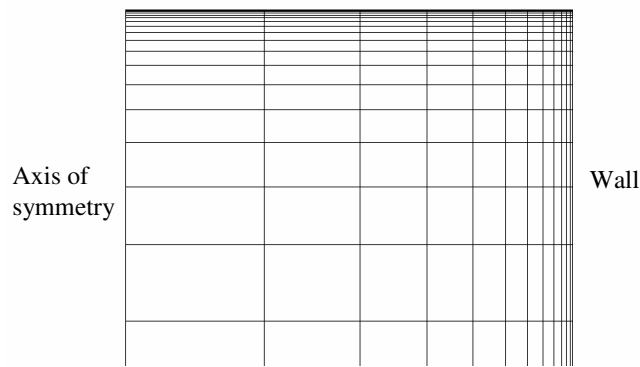


Figure 4.14: Moving piston mesh detail at the piston

4.5 Moving boundary and ALE scheme

To validate the moving boundary and the ALE moving mesh the problem of the moving piston was considered. A similar situation was created on a stationary mesh by holding the piston stationary and applying a moving wall boundary condition (in the opposite direction to the piston movement). The

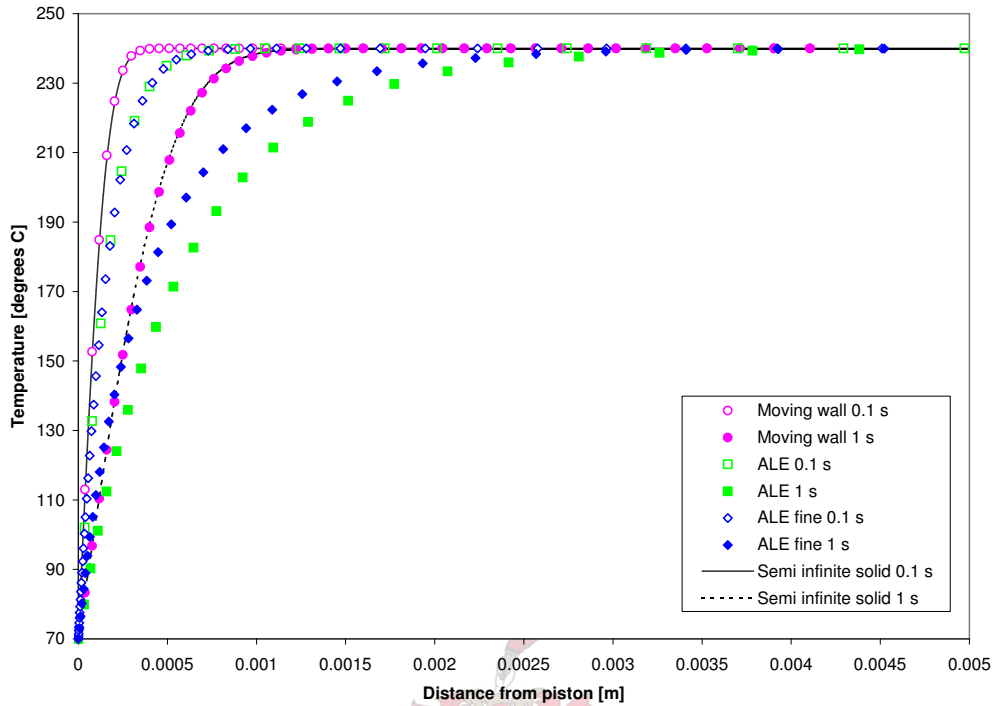


Figure 4.15: Moving boundary validation - temperatures along the axis of symmetry

Polymer	Basell Moplen EP301K
Piston velocity	$70 \text{ m}\cdot\text{s}^{-1}$
Solidification temperature	180°C
Upwinding factor	0.05

Table 4.1: Simulation settings for moving boundary validation

domains for both the moving wall and moving piston can be seen in figure 4.13. The mesh can be seen in figure 4.14. The settings for the simulation can be seen in table 4.1. The problem with the stationary wall is quite simple in that it does not require moving boundaries or remeshing. It was used as the reference calculation by which the moving boundary scheme is evaluated.

The simulations are compared by looking at the temperatures along the axis of symmetry. These temperature profiles are shown in figure 4.15. For the sake of comparison an analytical solution for the cooling of a semi-infinite solid slab is also included to see if the results are realistic. If the upwinding factor is too large ($\alpha = 0.5$), the temperature profiles lose accuracy. The upwinding factor was kept at the minimum value which did not cause instability ($\alpha =$

0.05). The moving boundary scheme over predicts the heat transfer. The solid layer is twice as thick as the moving wall simulation. If a finer mesh is used the accuracy is improved.



Chapter 5

Experimental setup and results

This chapter describes the experiments that were done on the lomolding machine.

5.1 Experimental data capturing ability

There are two methods of data capture from the lomolding machine. The first method is to capture digital signals from the lomolding machine's PLC. The second is to capture the data via a high speed analog card. Both sets of data are recorded on a PC.

The data that is captured from the PLC are variables that do not change very rapidly. Examples are the temperatures of the various components on the lomolding machine (including the mould). These variables are recorded at a frequency of 2 Hz.

The high speed analog card is used to capture the rapidly changing variables. Examples are the pressures in the mould, the position of the moulding piston and the hydraulic servo pressure. These values are typically recorded at a frequency of 200 Hz.

5.2 Experimental part and polymer

A flat round disk was chosen as the experimental part (see figure 5.1). Experimental measurements are taken for this part and compared to simulation results in chapter 6. The piston is located in the centre of the disk. For the injection moulding simulations the gate is also located at the centre of the

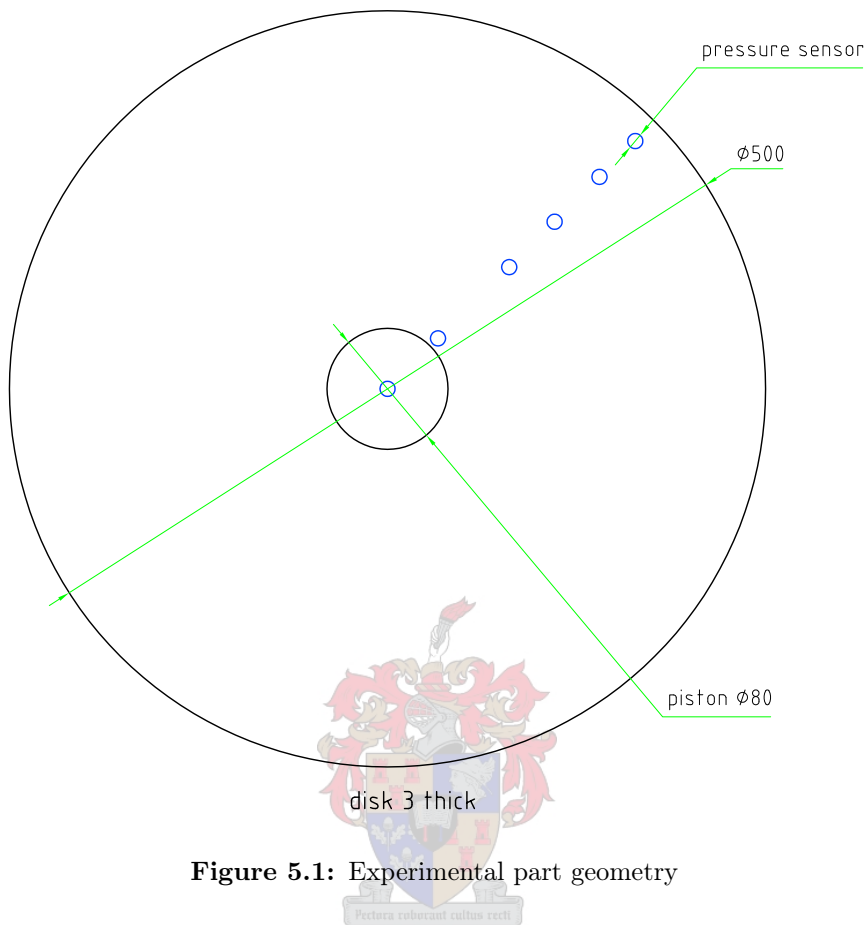


Figure 5.1: Experimental part geometry

Pressure sensor	radius [mm]
0	0
1	46.5
2	112
3	153
4	195
5	228

Table 5.1: Position of pressure sensors in mould

disk. Six pressure sensors were located in the mould. Their position from the centre of the disk can be seen in table 5.1.

The polymer that was used was a polypropylene, namely Moplen EP301K from Basell. Information about the polymer can be found at the manufacturer's website (Basell, 2004). Moplen EP301K was chosen because viscosity

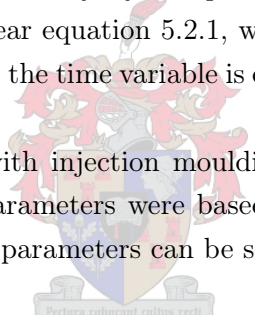
Parameter	Value
Melt temperature	240 °C
Wall temperature	45 °C
a	$-191.863 \cdot 10^3 \text{ mm}^3\text{s}^{-2}$
b	$485.805 \cdot 10^3 \text{ mm}^3\text{s}^{-1}$

Table 5.2: Parameters for simulations

data for it was available. The viscosity and thermal data can be seen in appendix F.

To determine the boundary conditions (volume flow rate or piston velocity) for the simulations, a parabola was fitted to the piston's displacement which was measured in the experiment. The piston's velocity is equal to the derivative of the displacement (see section 5.2). The volume flow rate can be calculated by multiplying the piston's velocity by the piston's area. The volume flow can then be given by the linear equation 5.2.1, where a and b are polynomial coefficients. In equation 5.2.1 the time variable is equal to zero when the piston starts moving.

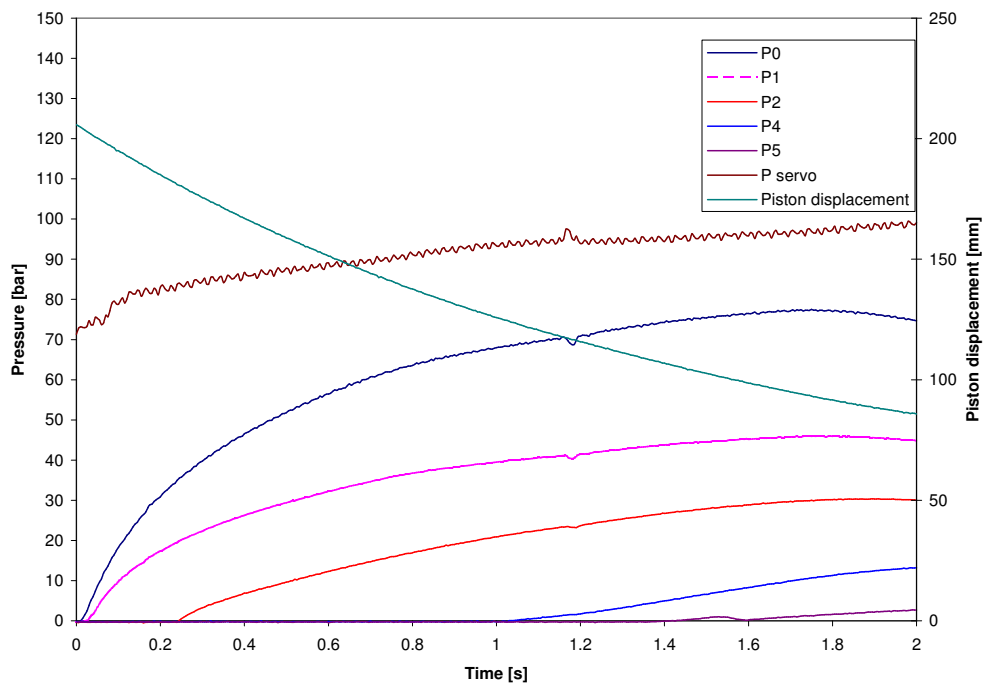
To compare lomolding with injection moulding, most of the parameters were kept the same. The parameters were based on the experimental parameters for lomolding. These parameters can be seen in table 5.2.



$$\text{Volume flow rate} = at + b \quad (5.2.1)$$

5.3 Cavity pressure measurements for lomolding

The experimental data that was captured from the lab can be seen in figure 5.2. The P servo curve is the hydraulic pressure pushing the piston forward. The piston displacement curve can be seen in the figure that was used to calculate the appropriate velocity boundary conditions for the volume flow rate.

**Figure 5.2:** Lomolding experimental data

Chapter 6

Simulations and results

In this chapter the simulations and simulation results are presented and discussed. The simulations will also be compared to the experimental measurements. Lomolding will also be compared to injection moulding. Sopfil is also used to analyse shear heating in hot runners and analyse the flow under the piston.

6.1 Injection moulding analysis

The modelling of injection moulding is explained in this section. The results from this analysis will be compared to Cadmould (Simcon, 2002*a*). The injection moulding analysis serves two purposes: firstly to give confidence in Sopfil (by comparing the results to Cadmould), and secondly as a reference point from which to evaluate lomolding. Injection moulding will be the main competitor of lomolding, and therefore it will be useful if the processes can be compared numerically.

The domain used for the injection moulding simulation can be seen in figure 6.1. A typical mesh can be seen in figure 6.2. The mesh is refined at the walls where the largest gradients (in velocity and temperature) are likely to occur and solidification takes place. The boundary conditions can be seen in figure 6.1.

The mould wall is assumed to have a constant temperature. This does not strictly happen in reality. The surface of the mould will heat up a little bit as it comes into contact with the hot melt. But because the thermal conductivity of the mould is two orders of magnitude higher than that of the

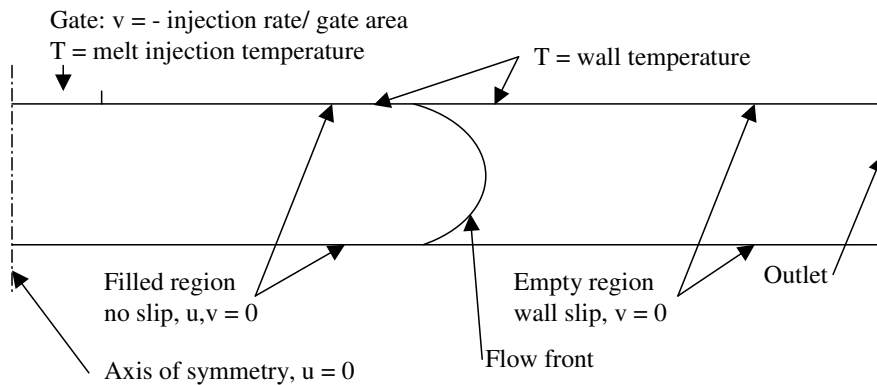


Figure 6.1: Injection moulding domain

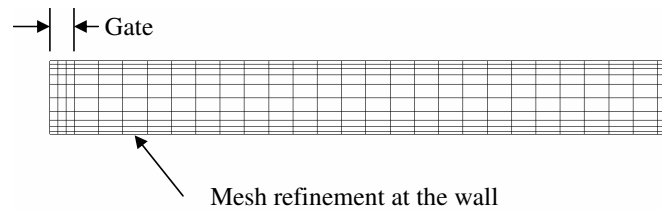


Figure 6.2: Injection moulding - part of a typical mesh

polymer, the mould surface temperature does not increase much. Furthermore a sensitivity study was conducted on the effect that the mould wall temperature has on the simulation results (see appendix J) which found that the mould wall temperature does not have a significant effect on the cavity pressures.

A non-constant injection rate could be applied at the gate by modifying the gate's boundary condition at every time step.

6.2 Injection moulding results

The simulation results of injection moulding with Cadmould and Sopfil can be seen in figure 6.3. The jumps in the Cadmould pressure curves can be explained by the fact that Cadmould can only handle an injection flow rate that is a step function. The linear injection flow rate (equation 5.2.1) therefore has to be approximated by a stepped profile. The jumps in the pressure curves occur where there is a jump in the flow rate function.

Sopfil pressures are slightly higher than Cadmould's. This can be explained by looking at the solid layer thickness in figure 6.4. The solid layer thickness in

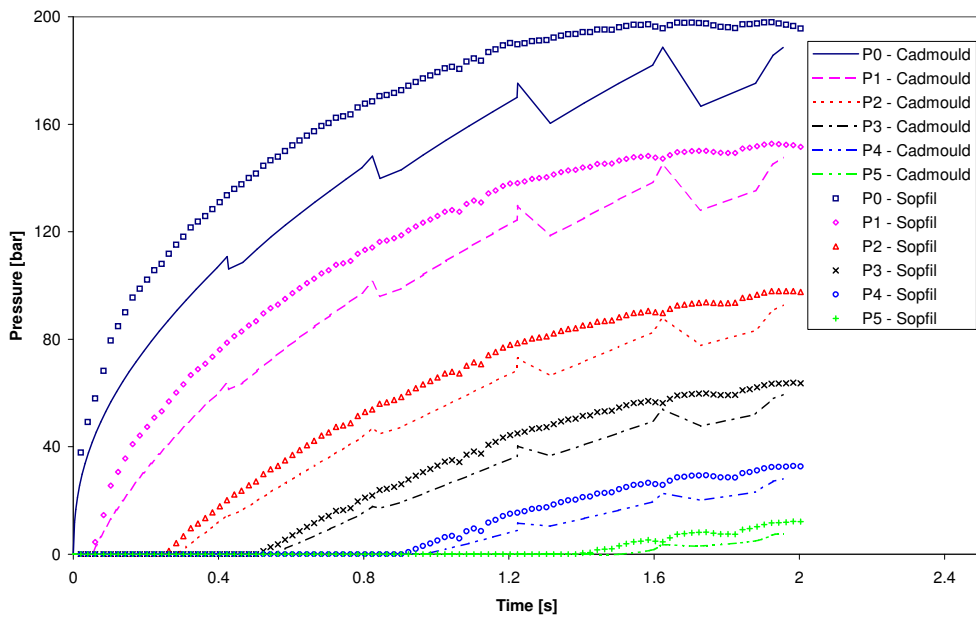


Figure 6.3: Injection moulding - Cadmould and Sopfil pressures

the figure is the sum of the solid layers on the top and bottom surfaces of the cavity. Sopfil's solid layer is slightly thicker than Cadmould's. In Newtonian flow between parallel plates the pressure drop is a function of the flow area cubed (according to the analytical solution, Lee & Castro (1989)). For a power-law fluid the pressure drop is a function of the flow area to the power of 2 or greater, depending on the fluid properties (equation 4.1.7). Therefore the difference in the solid layer thickness will definitely cause a difference in the pressures between Sopfil and Cadmould.

6.3 Lomolding analysis

This section describes how the Lomolding analysis is performed. This analysis is compared to the experimental data and is used to compare lomolding to injection moulding.

One of the major problems that was experienced was the problem of run time. To analyse the full lomolding domain would take about three weeks on the computers used (Intel Pentium 4 2.4 GHz, 512 Mb RAM). The long run times were caused by inefficient ordering of the nodes which in turn increased the bandwidth of the stiffness matrix. More details on the computation times

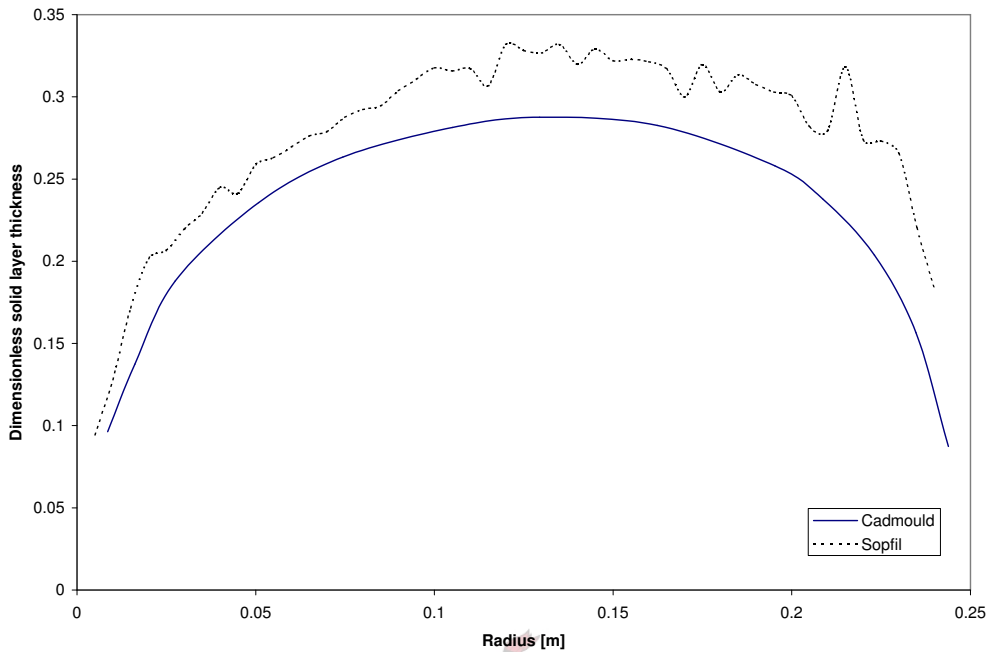


Figure 6.4: Solid layer thickness in injection moulding - Cadmould and Sopfil

and matrixes are given in appendix E. The details of the node ordering algorithm are given in appendix D.5. To solve this problem the lomolding analysis was split into two separate problems. The Lomolding domain can be seen in figure 6.5. The first analysis is the piston analysis (see figure 6.6). This analysis will give insight into the flow under the piston and to determine the boundary conditions for the simplified lomolding analysis. The simplified lomolding analysis (see figure 6.7) is used to calculate the pressures inside the cavity, shear rates, etc. By splitting the lomolding domain in such a manner the run time for each analysis was reduced to about a day. The mesh used for the piston analysis can be seen in figure 6.8.

The same slip wall/no-slip wall boundary conditions were applied as in the injection moulding analysis.

6.4 Lomolding results

Figure 6.9 shows the comparison between the experimental data of Cadmould and Sopfil. As in the injection moulding simulation Sopfil calculates a slightly higher pressure than Cadmould. The most noticeable thing is that the simulation results are a factor 2 or 3 larger than the experimental results.

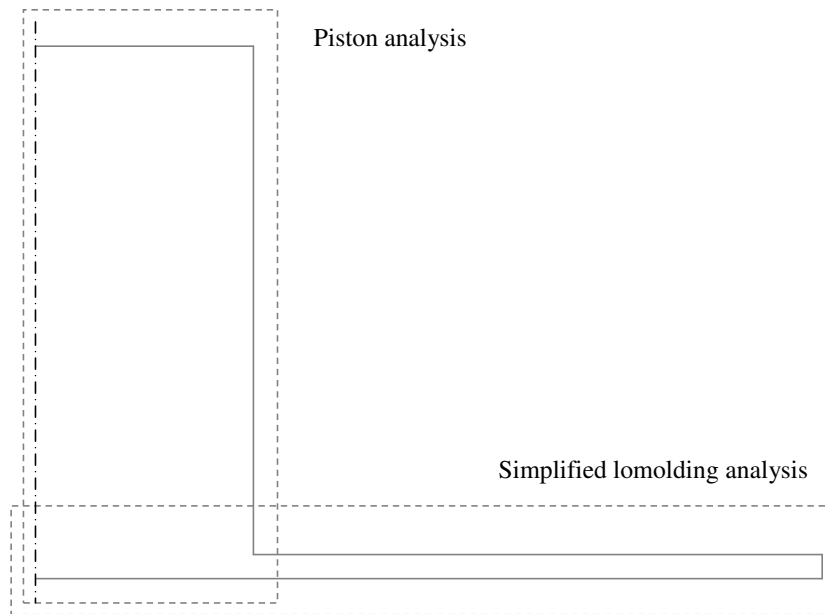


Figure 6.5: Lomolding domain

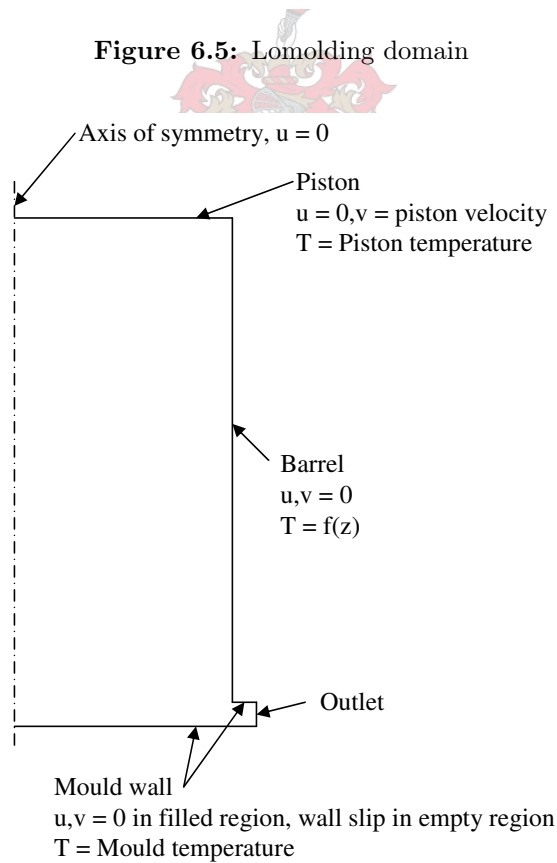


Figure 6.6: Piston analysis domain

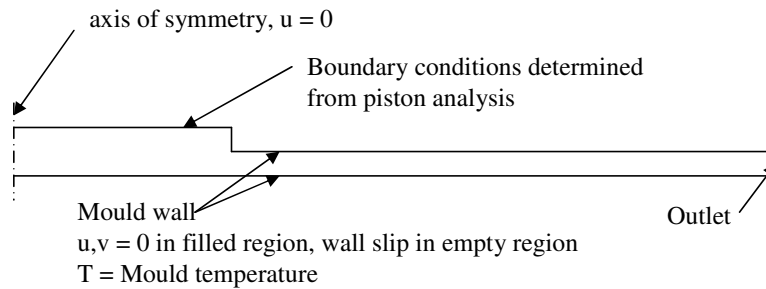


Figure 6.7: Simplified lomolding domain

Sopfil and Cadmould agree well with each other. There is reason to doubt the accuracy of the experimental results. These reasons are discussed in chapter 7.

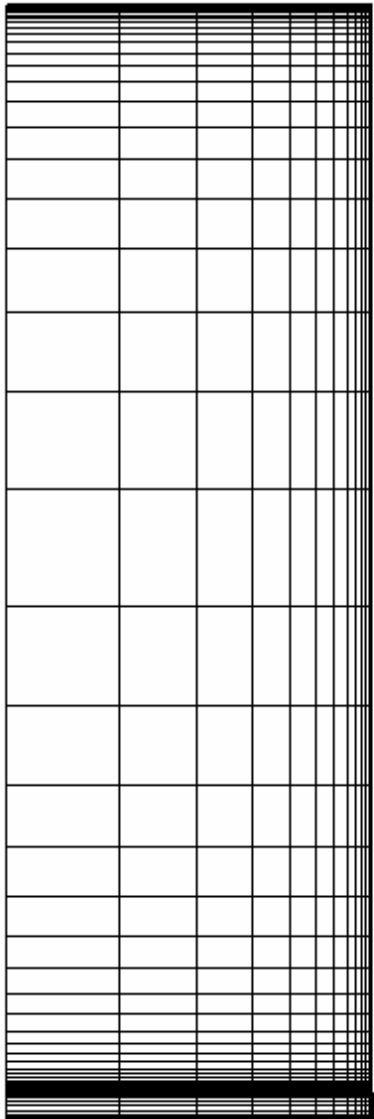
6.5 Shear heating in hot runners analysis

For machine design purposes, it is useful to know how much shear heating occurs in the hot runners on the lomolding machine. It is also important to know for the simulations at what temperature the melt enters the cavity (i.e. how much the melt temperature increases from the metering cylinder to the barrel).

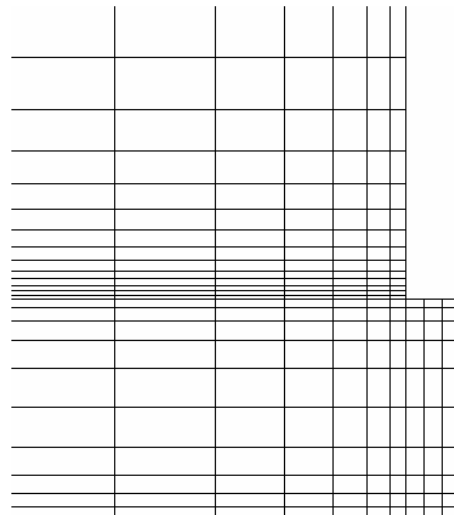
One of the hot runners has a valve connecting rod running down the middle (for the outfeed valve). It would be interesting to know what effect this has. Sopfil is well suited to run this kind of simulation.

Figure 6.10 shows the domains for the hot runner and the hot runner with the valve rod. Part of the mesh used for the calculation can be seen in figure 6.11. The domains' initial temperatures are set at the melt set temperature (240°C). The melt set temperature is also the boundary conditions of the inlet and the hot runner walls. The valve rod is given a zero temperature gradient boundary condition. The zero temperature gradient means that heat cannot be transferred to the valve rod, which is a reasonable assumption if the temperature in the valve rod has reached equilibrium. This boundary condition also means that heat conduction along the valve rod is neglected.

The simulation was run for 3 times longer than it should take for a particle to pass through the domain (based on the average velocity in the pipe) to ensure that the temperature field is fully developed.



Piston mesh



Detail of moulding barrel / cavity interface

Figure 6.8: Piston analysis mesh

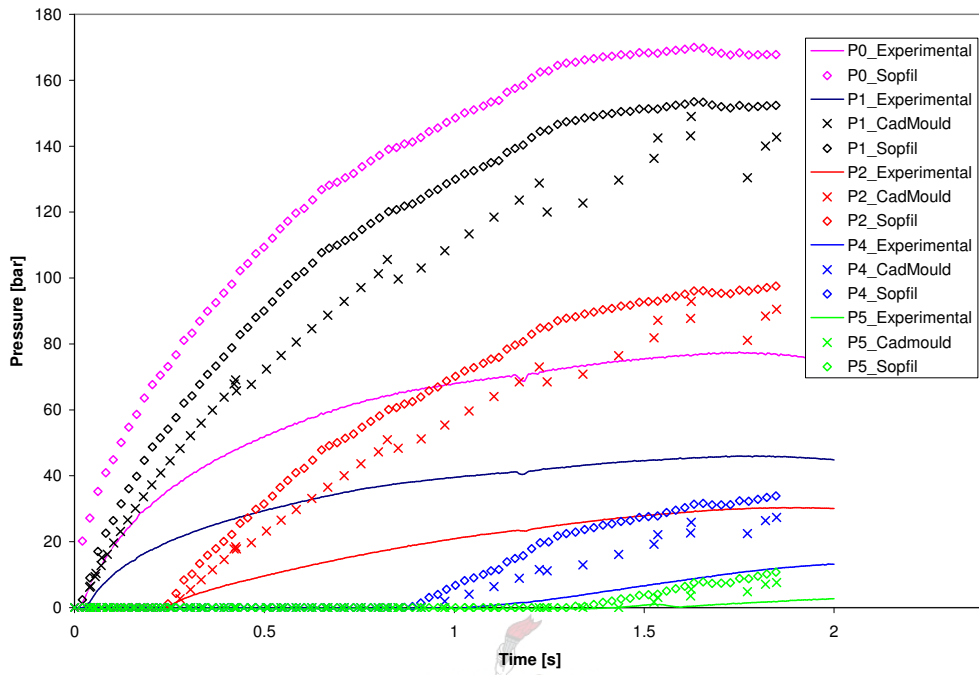


Figure 6.9: Lomolding experimental, Cadmould and Sopfil pressures

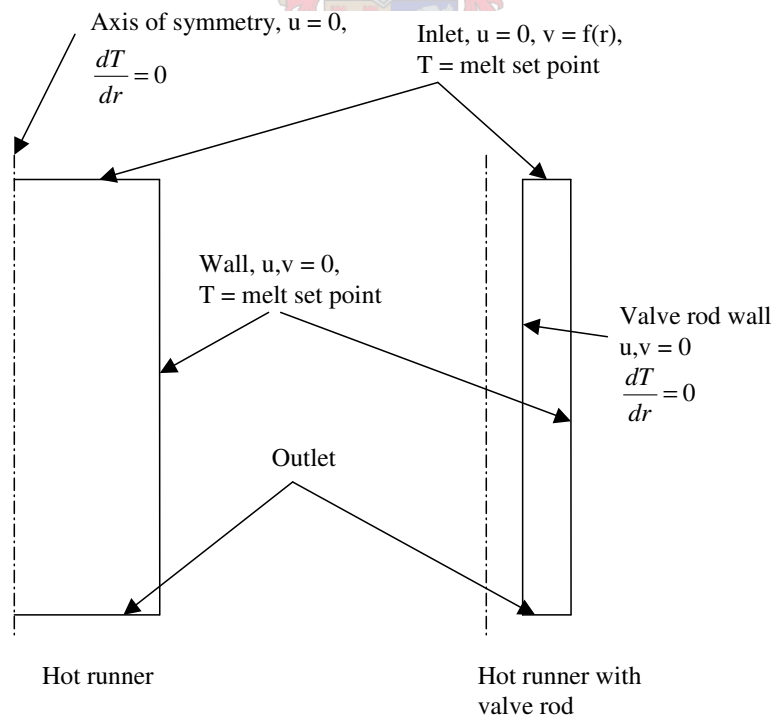


Figure 6.10: Shear heating domain - hot runner and hot runner with valve rod

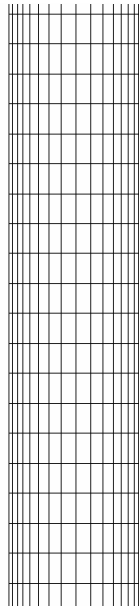


Figure 6.11: Part of the shear heating mesh

Hot runner diameter	20 mm
Valve rod diameter	8 mm
Hot runner length	224 mm
Temperature set point	240 °C
Volume flow rate 1	58.9 cm ³ s ⁻¹
Volume flow rate 2	589 cm ³ s ⁻¹

Table 6.1: Parameters for shear heating analysis

The parameters of the analysis can be seen in table 6.1. Two volume flow rates were used. The first volume flow rate (58.9 cm³s⁻¹) is the volume flow rate of the current lomolding machine. With this volume flow rate it takes about 10 seconds to transfer the shot (the melt volume for one part) for making the experimental part (see figure 5.1). A second volume flow rate was chosen that was ten times larger than the first, because on a production lomolding machine the transfer time would preferably be minimised.

	Temperature rise ($^{\circ}C$)
Hot runner ($Q = 58.9 \cdot 10^3 \text{ mm}^3\text{s}^{-1}$)	0.167
Hot runner ($Q = 589 \cdot 10^3 \text{ mm}^3\text{s}^{-1}$)	1.504
Hot runner with valve rod ($Q = 58.9 \cdot 10^3 \text{ mm}^3\text{s}^{-1}$)	0.458
Hot runner with valve rod ($Q = 589 \cdot 10^3 \text{ mm}^3\text{s}^{-1}$)	3.469

Table 6.2: Mass flow averaged temperature rise in shear heating analysis

6.6 Shear heating in hot runners results

The first volume flow rate results can be seen in figure 6.12. The second, larger volume flow rate can be seen in figure 6.13. The mass flow averaged temperature rise for the simulations can be seen in table 6.2.

The mass flow average temperature in most cases does not change enough to cause major differences in the cavity pressures. What is of concern though is the local heating of the polymer in contact with the valve rod. For the high volume flow rate this temperature exceeds the thermal degradation temperature of the polymer ($280^{\circ}C$). If a production lomolding machine is built, with a high volume transfer rate, the design of the hot runners will have to take this into account. The design change could either entail increasing the diameter of the hot runners, or not having a valve rod in the hot runner.

6.7 Lomolding vs injection moulding

In this section lomolding is compared to injection moulding. The criteria used for the comparison are cavity pressures (and clamping force), shear rates and shear stresses.

To compare the processes all the process parameters that could be kept the same were. The process parameters can be seen in table 5.2. The volume flow rate was kept the same in both processes so that the shear rates and stress could be compared objectively. No packing was taken into account.

6.7.1 Cavity pressures

Figure 6.14 shows lomolding and injection moulding pressures for the same volume flow rate, as computed by Sopfil. The figure shows that the cavity pressures are similar for the two processes. There is a difference though in the

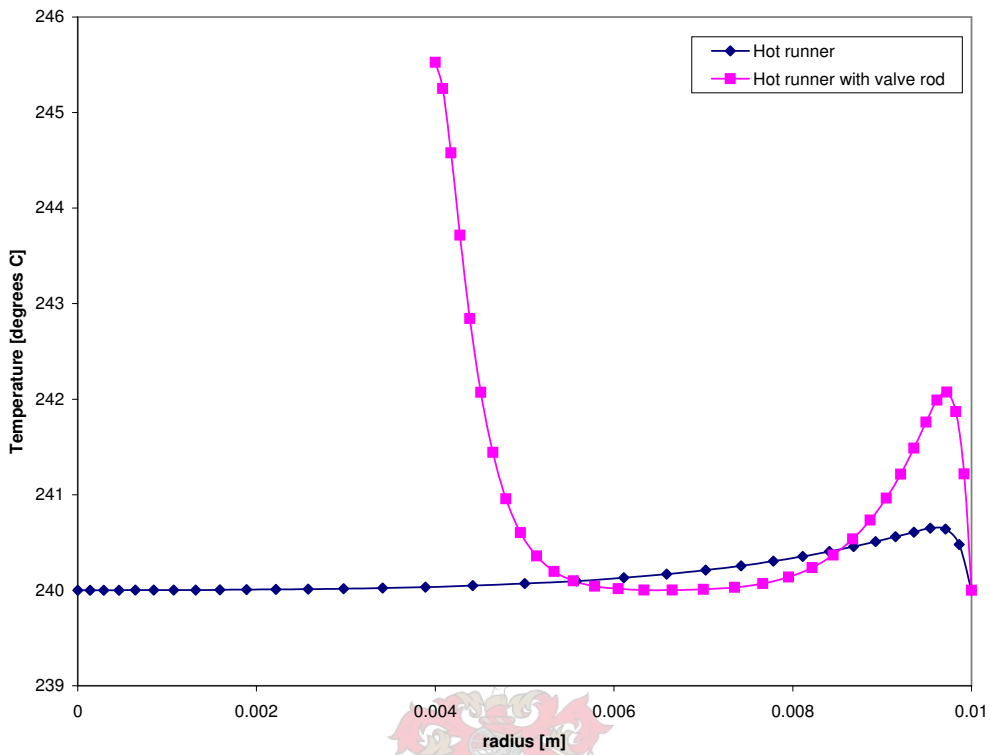


Figure 6.12: Temperature profiles, low volume flow rate ($Q = 58.9 \cdot 10^3 \text{ mm}^3\text{s}^{-1}$)

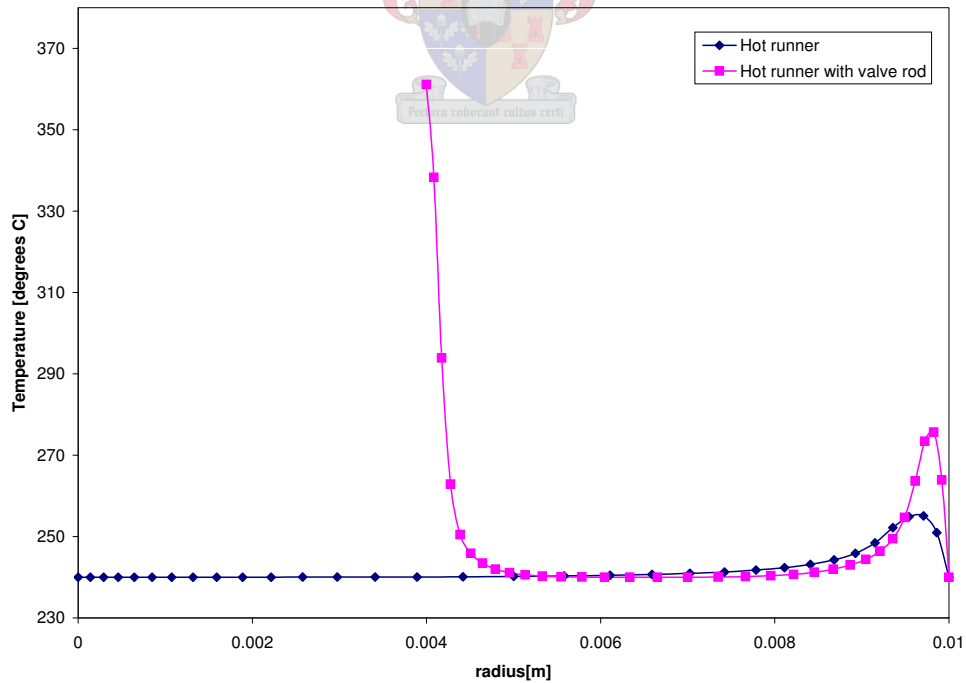


Figure 6.13: Temperature profiles, high volume flow rate ($Q = 589 \cdot 10^3 \text{ mm}^3\text{s}^{-1}$)

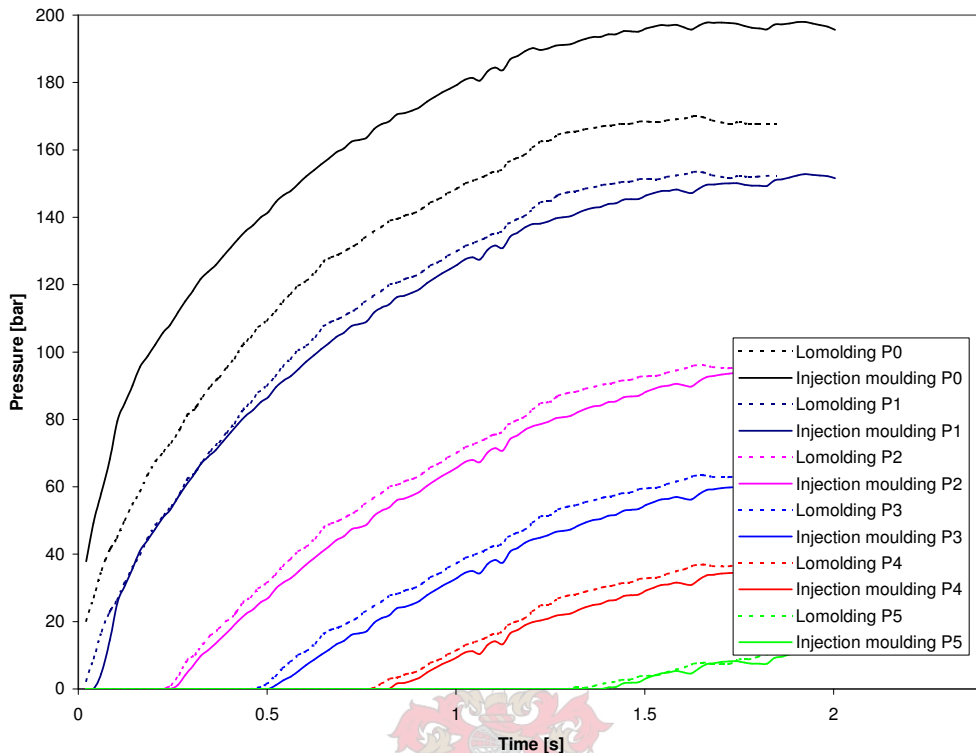


Figure 6.14: Lomolding vs. injection moulding - cavity pressures - Sopfil

piston area of the cavity: the pressure at the injection moulding gate is larger than the pressure at the centre of the lomolding piston (pressure P0).

Figure 6.15 shows the pressure distribution in the cavity (pressure vs. radius) at the instant when filling is complete. Both Sopfil and Cadmould show that the pressure drops in the cavities are similar. The difference between the processes' pressure distribution in the piston area is clearly seen. In injection moulding the pressure continues to rise to the gate, whereas in lomolding the pressure almost levels off under the piston. The maximum cavity pressure in lomolding is 14.3% lower than in injection moulding.

The clamping force was calculated by integrating the pressure (from Sopfil simulation) across the cavity. The resulting clamping forces can be seen in table 6.3.

It should be noted that this comparison was done with the same volume flow rate for both processes, and that packing was not taken into account.

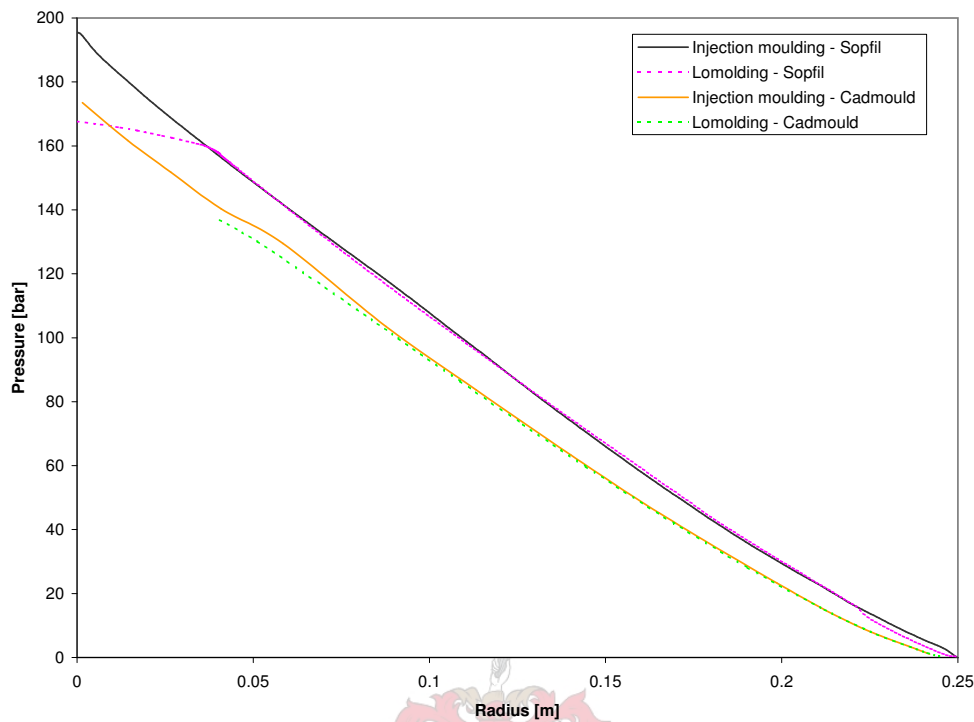


Figure 6.15: Lomolding vs. injection moulding - centreline pressures

	Clamping force [ton]
Lomolding	112.769
Injection moulding	113.238
Difference	-0.42%

Table 6.3: Clamping force for mould filling - Lomolding vs. Injection moulding

6.7.2 Shear rates

Injection moulding and lomolding shear rates were compared by examining the shear rates along streamlines. Two criteria were chosen to compare the processes. The first is the maximum shear rate that occurs along a streamline. The second criterion is the shear rate history that a particle experiences travelling down a streamline. The shear rate history was calculated by equation 6.7.1. A series of streamlines were chosen with the end points at regular

intervals (across the thickness) at the end of the cavity.

$$\text{Shear history} = \int \dot{\gamma} dt \quad (6.7.1)$$

The situation chosen to compare the processes was a filled domain, with the maximum volume flow rate boundary conditions applied. The simulations were run using Sopfil. No solidification was used. The wall temperatures were set to the melting temperature of the polymer melt. The reason for not choosing any solidification was that some of the streamlines moved through the solidified elements (very low velocities and very long times) which is not physically realistic. The position of the interface between the solid layer and the flow is also not well defined. Should it be taken where the temperature drops below the solidification temperature, the maximum shear rate or where the velocity is suitably close to zero? These points should physically all coincide, but do not with the solidification model. Not using solidification resolves this problem as all of these points occur at the wall. The position of the interface is also important for the comparison of the shear stresses, which will be discussed in the next section.

The shear history and maximum shear rates can be seen in figure 6.16. A mass flow average of the shear history was taken and can be seen in table 6.4.

Lomolding has a 15.13% lower mass flow averaged shear history than injection moulding. The big difference between the processes is the maximum shear rate. The maximum shear rates in injection moulding are an order of magnitude bigger than those of lomolding.

6.7.3 Shear stresses on in-mould-decorations

Using the same simulations as used for the shear rate comparison, the shear stresses along the mould wall of the processes were compared. The shear stresses that are of interest are the ones that an in-mould-decoration would experience. The in-mould-decoration is situated on the cavity wall on the opposite site to the piston or gate.

Figure 6.17 shows the shear stress plotted versus the radial coordinate. There are small oscillations near the outlet of the domain but these can be ignored as they are due to effects at the outlet. The shear stresses in the cavities are almost identical for both processes. However, in lomolding the shear stress decreases under the piston, as opposed to injection moulding where

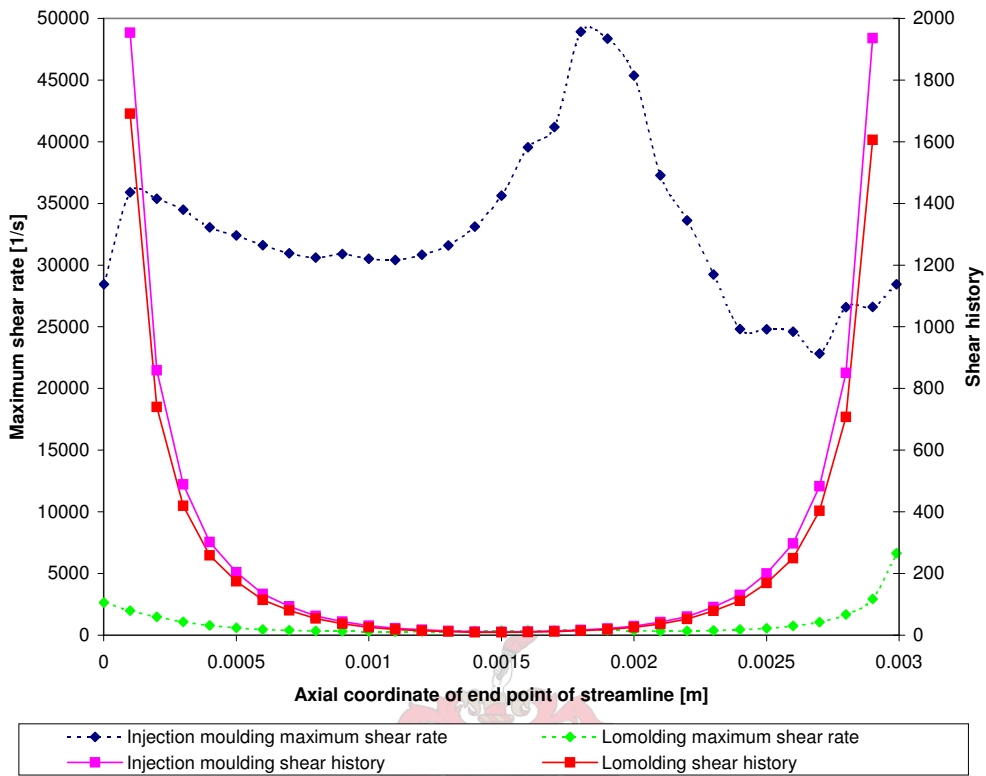


Figure 6.16: Lomolding vs. injection moulding - Shear history and maximum shear rates

Lomolding	31.69
Injection moulding	37.34

Table 6.4: Mass flow averaged shear history

the shear stress carries on rising to the gate. The maximum shear stress in lomolding was 48% smaller than in injection moulding.

6.7.4 Filling rates at constant injection pressures

All the comparisons up to now have been conducted at a prescribed volume flow rate. In this analysis Cadmould is used to analyse the filling time and cavity pressures at a prescribed injection pressure.

For lomolding the injection pressure is specified at the nodes at the moulding barrel / cavity interface because there is little pressure drop in the piston and Cadmould cannot model the piston region.

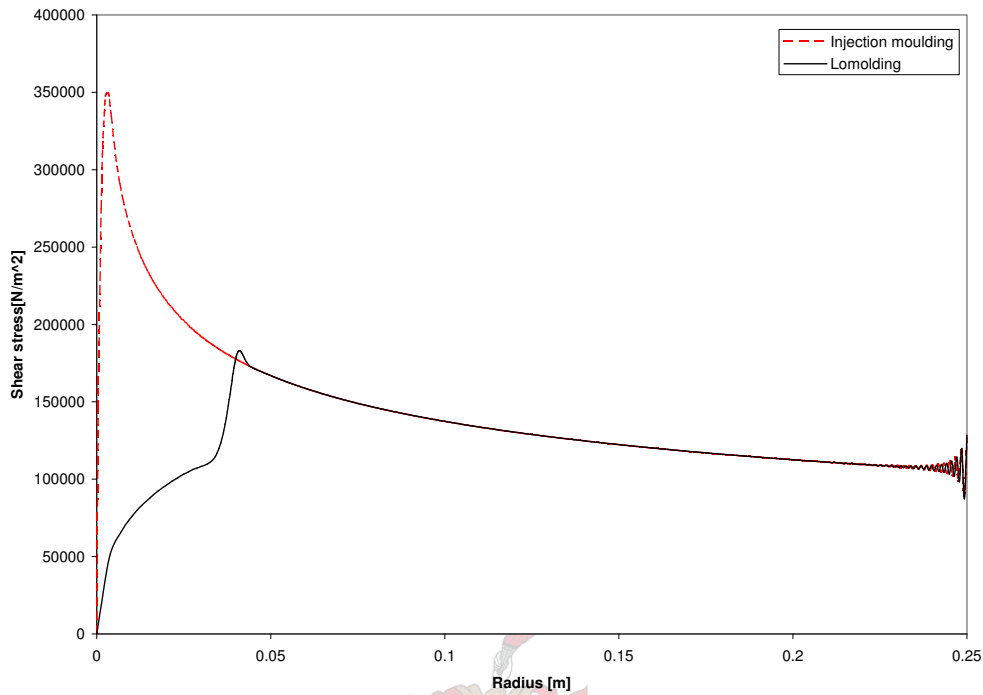


Figure 6.17: Lomolding vs. injection moulding - Shear stresses on skins

The lomolding and injection moulding filling rates can be seen in figure 6.18. The cavity pressures at the end of cavity filling can be seen in figure 6.19.

Figure 6.18 shows that lomolding fills the cavity about 5 times faster than injection moulding. This is because the cavity pressure at the moulding barrel / cavity interface is higher than the corresponding point in the injection moulding cavity. There is a bigger pressure difference driving the volume flow rate in lomolding.

The pressure drop is relatively insensitive to volume flow rate; a small increase in pressure causes a large increase in volume flow rate. This phenomena can clearly be seen in the sensitivity study in appendix J (figure J.4). Therefore the slight increase in the pressure in lomolding causes a large increase in volume flow rate.

If the injection pressure for injection moulding is increased so that the cavity filling time is equal to that of lomolding the cavity pressures are identical (see figure 6.19). The injection pressure was 34% larger than lomolding's injection pressure.

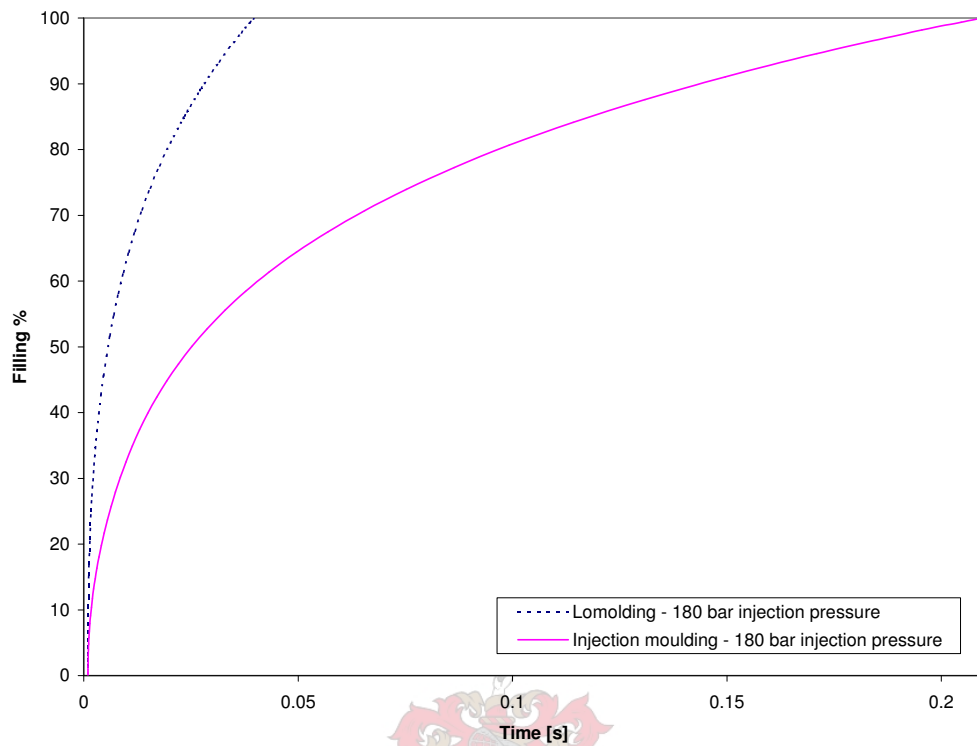


Figure 6.18: Lomolding vs. injection moulding - Filling rates for 180 bar injection pressure

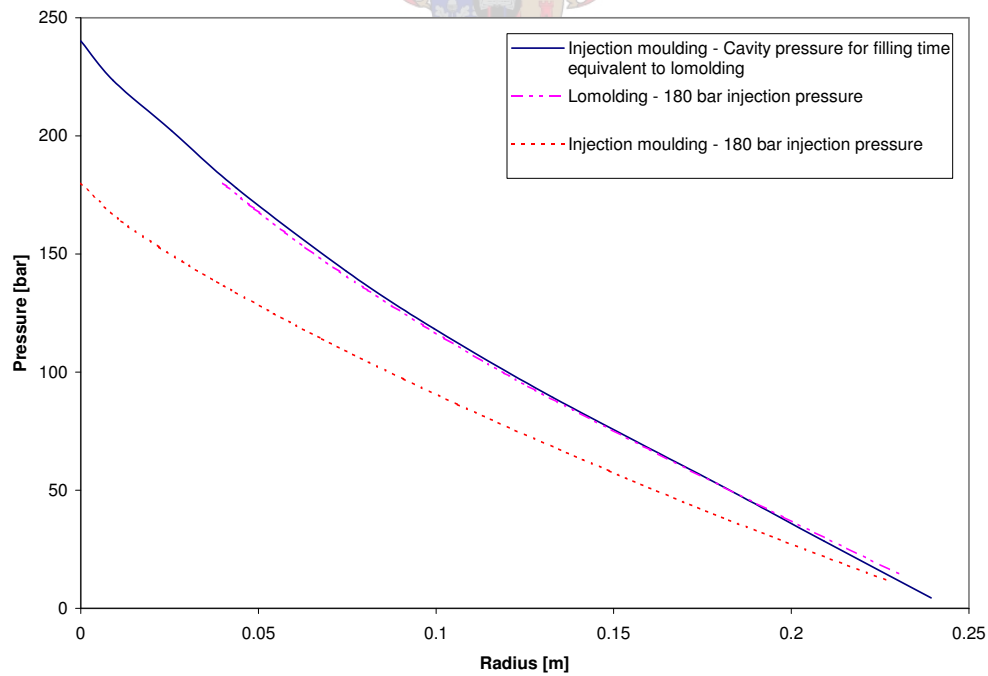


Figure 6.19: Lomolding vs. injection moulding - Cavity pressures for 180 bar injection pressure

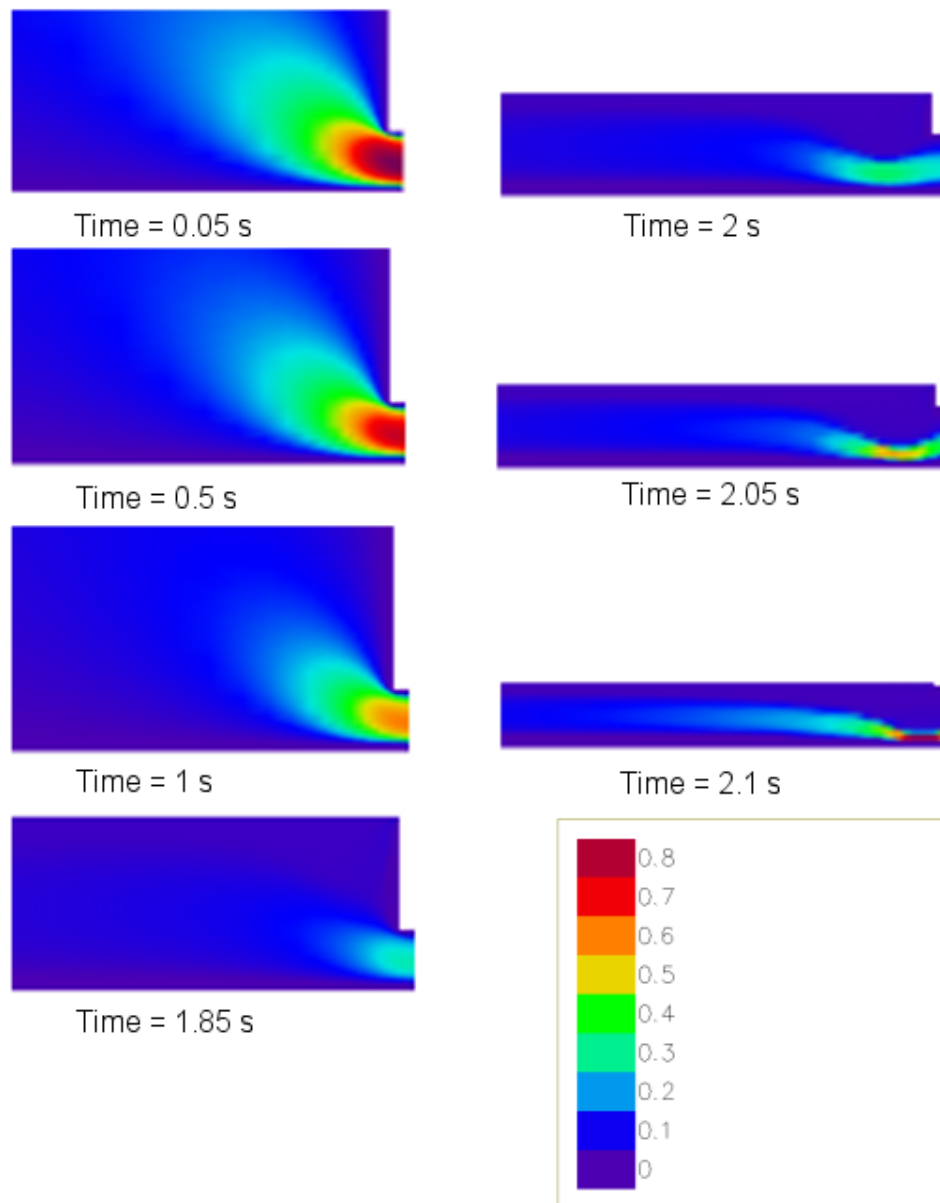


Figure 6.20: Flow patterns under the piston - velocity magnitude at the cavity / moulding barrel interface

6.8 Lomolding piston analysis - flow patterns under the piston

The flow under the piston was analysed to gain insight into the flow and solidification patterns. The analysis was performed using the experimental

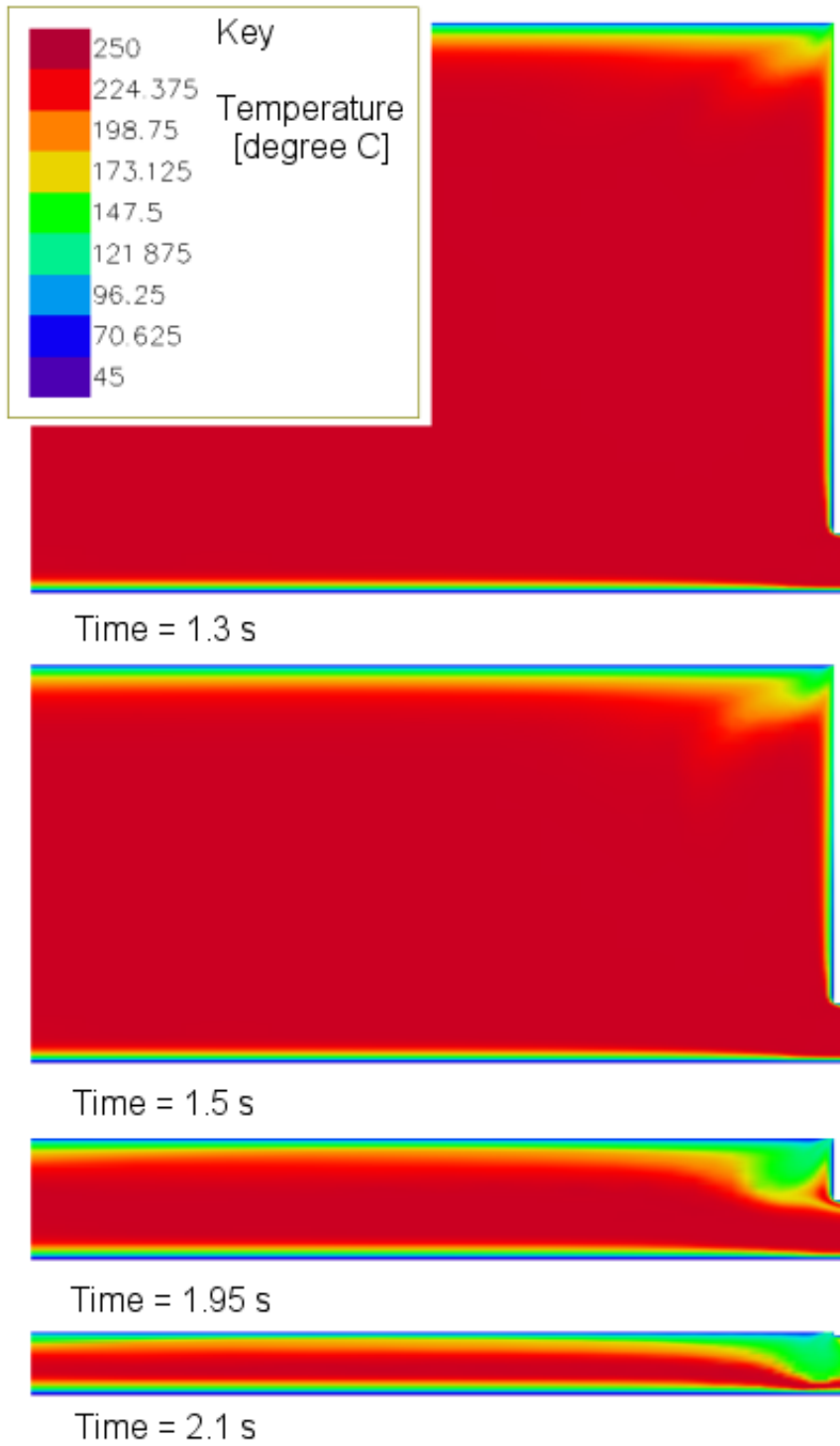


Figure 6.21: Flow patterns under the piston - Temperature

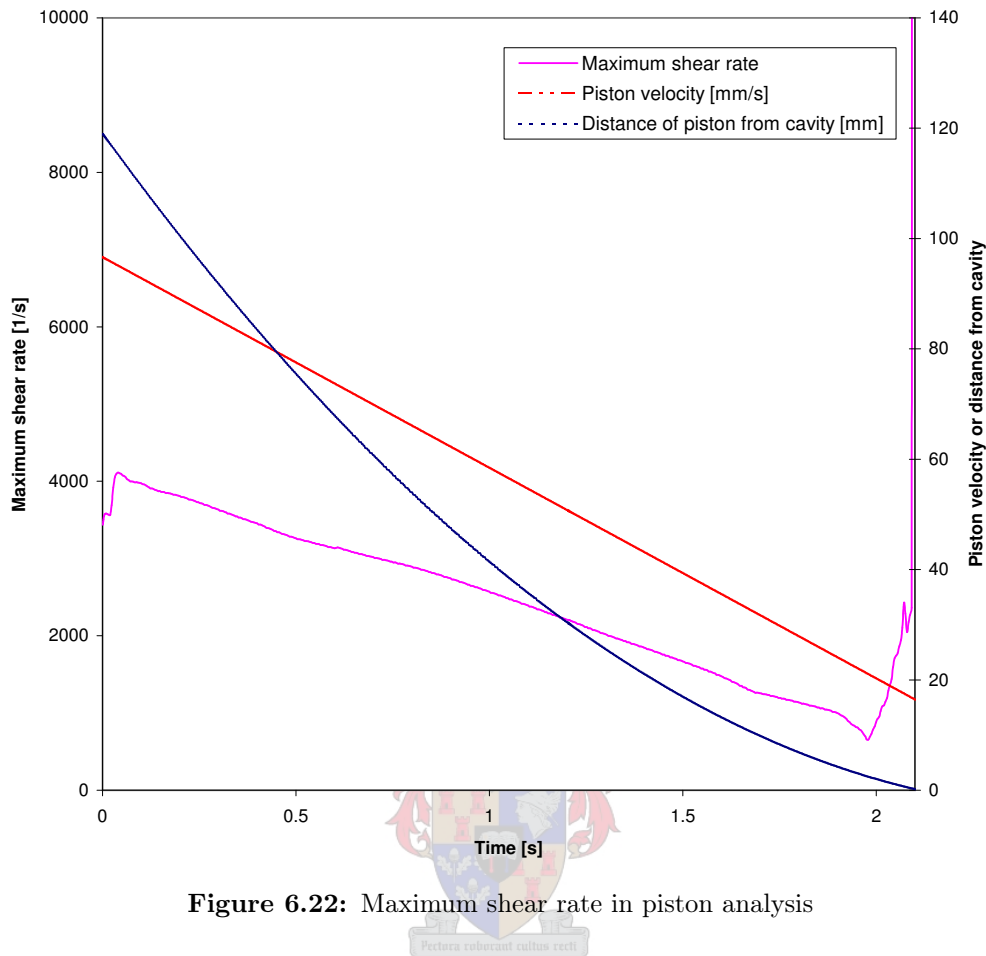


Figure 6.22: Maximum shear rate in piston analysis

process parameters (section 5.2).

Figure 6.20 shows magnitude of velocity contour plots for various time steps through the simulation. Figure 6.21 shows temperature contour plots at various time steps through the simulation.

The velocity contour plots shows how the maximum velocity decreases as the piston's velocity decreases. The piston velocity decreases as it approaches the cavity (see table 5.2). However, as the piston gets close to the end of the moulding barrel (from about 2 s onwards) the maximum velocity increases. This is due to solidified polymer build-up that is pushed into the cavity, restricting the flow.

The temperature contour plots show the build-up of the solidified polymer. The polymer solidifies at a temperature of 180 °C, which in figure 6.21 is represented by the colour yellow.

Polymer solidifies on the barrel close to the cavity, where the barrel's tem-

perature drops below the melting temperature of the polymer. The temperature contour plot shows how this solidified polymer is scraped up onto the piston. The solidified polymer build-up is pushed into the cavity at the end of the piston stroke.

Figure 6.22 shows the maximum shear rate (per time step) for the simulation. The maximum shear rate decreases as the piston velocity decreases. However, when the solid layer build-up is pushed into the cavity the maximum shear rate increases because the flow into the cavity is restricted. The drastic increase of shear rate is not realistic. The solid layer build-up in reality will be quite elastic (especially at its high temperature). In the analysis the solid layer build up's elasticity is not taken into account. In reality it should deform a bit, and not cause as much as a blockage as it does in the simulation.

The increase in shear rate is undesirable because it will mean more fibre breakage and larger shear stresses on the in-mould-decoration. There are two possible methods to prevent the increase in shear rate. One would be to change the piston's control strategy. The piston's velocity at the end of the stroke should be small. Using a parabolic velocity function (instead of the current linear function) would be one way of doing this. The second way would be to redesign the moulding barrel so that the solid build-up is either not so large, or that it is not pushed into the flow path of the cavity. The solid layer build-up could be captured in a step cut out between the barrel and the cavity.

Chapter 7

Investigation of discrepancies between experiments and simulations

The simulation results and the experimental results did not correlate well (see chapter 6.4). This chapter discusses the reasons why the accuracy of the experimental results is doubtful.

7.1 Pressure sensor range

The pressure sensors have a range of 0 - 2000 bar. The highest pressure measurements in lomolding are 70 bar, and this is directly under the piston. Most of the pressure sensors have measurements under 40 bar. This is 2% of the sensors' full range. Appendix G shows work that was done to calibrate the pressure sensors. The best calibration results found that there was a 5% to 10% error at the range from 0 to 40 bar. The errors get progressively worse as the pressures get lower. Therefore it is probable that some errors occurred in the pressure measurement.

7.2 Process parameters, particularly melt temperature

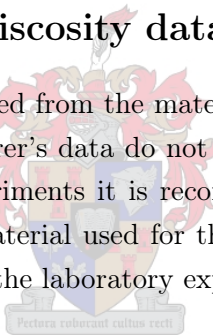
Appendix J shows a sensitivity study that was done on the parameters in the simulations. The parameters were wall temperature, freezing temperature,

initial melt temperature and the piston speed. The sensitivity study found that freezing temperature, piston speed and wall temperature do not have a significant influence on the cavity pressures. The initial melt temperature has the biggest influence: a temperature difference of 60 °C results in a factor 2 difference in cavity pressures.

Appendix H shows experiments that were done to verify the temperature of the melt entering the barrel and the cavity. The shot does not have a uniform temperature: some parts of the shot are below the set temperature and others above. The conclusion of the experiments was that over-all the melt enters the barrel about 10 °C too cold. The lower melt temperature should cause higher cavity pressures, which is an error in the wrong direction. Nevertheless the inaccuracies in melt temperature introduce an uncertainty into the experiments.

7.3 Verification of viscosity data

The viscosity data were obtained from the material manufacturer. There is a possibility that the manufacturer's data do not match the actual viscosity of the material. For future experiments it is recommended that the material's viscosity is calibrated. The material used for the calibration should be from the batch that will be used in the laboratory experiment.



7.4 Non-axisymmetric flow

The flow into the cavity is not axisymmetric. Appendix I shows the filling pattern and the experimental procedure by which it was obtained. The filling pattern was determined by a series of short shots. The filling pattern shows that the melt flows to the top of the cavity first, and the pressure sensors lie in the bottom half of the cavity. Therefore the pressures in the top of the cavity will be greater than the pressures in the bottom.

A simulation was run in Cadmould to determine the effects of non-axisymmetric filling. The non-axisymmetric filling was induced by specifying more injection nodes at the top of the piston than at the bottom. The same volume flow rate was specified for each injection node, and therefore more melt enters the top than the bottom. The filling contours can be seen in figure 7.1. Figure 7.2 shows cavity pressures for nodes located at the top and bottom of the cav-

ity. The figure shows that the pressures in the top half are larger than in the bottom half, and the difference increases the more non-axisymmetric the flow is.

The jumps in the pressure curves in figure 7.2 at the end of the filling time (1.6 s - 1.8 s) occur when the melt has reached the end of the cavity in the upwards direction. When this happens the volume flow rate in the upwards direction is very close to zero, and therefore there is almost no pressure drop in that direction.

The Cadmould simulation was not run to try and model what happens in the laboratory, but rather to determine what influence non-axisymmetric filling has on cavity pressures.

7.5 Inadequate clamping force

The theoretical clamping forces, from the simulations (calculated in section 6.7.1), were found to exceed the clamping force of the machine (110 ton). The inadequate clamping force could mean that the mould is opening slightly during moulding. If the mould opens the flow area will increase and the pressure drop will decrease.

In injection moulding when the mould opens flashing can be seen. In lomolding the flashing will be less because a fix volume of melt is pushed into the cavity. Consider that the mould does open in mould filling: when the piston reaches the cavity wall the melt would not yet have reached the ends of the cavity because the cavity's thickness (and thus volume) has increased due to the mould opening. When the piston stops, the cavity pressures will drop and the mould will be able to close in a similar manner to compression moulding. Very little flashing was observed in the experiments.

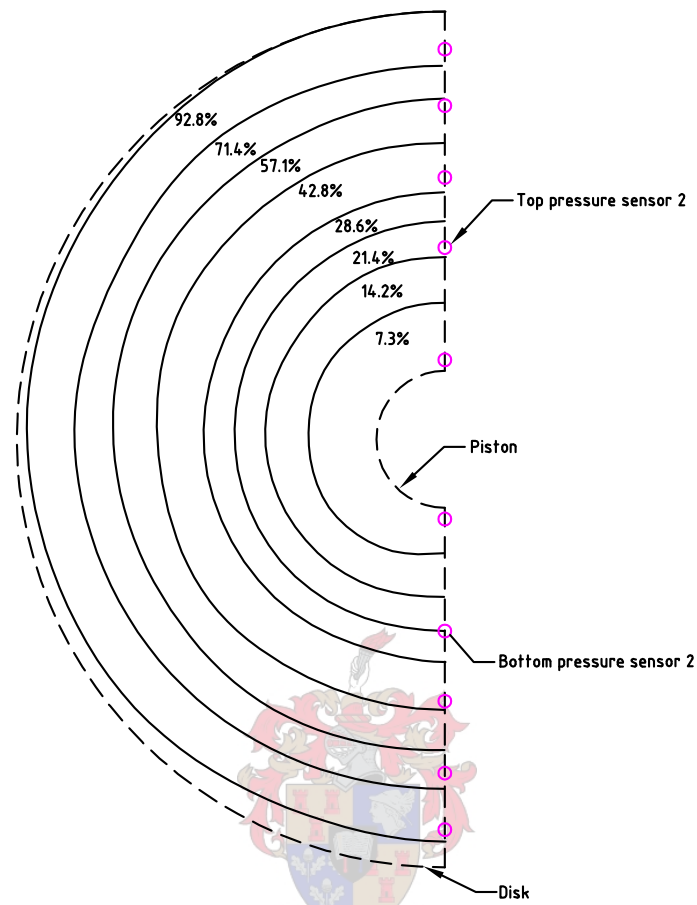


Figure 7.1: Filling contours for non-axisymmetric flow - Cadmould

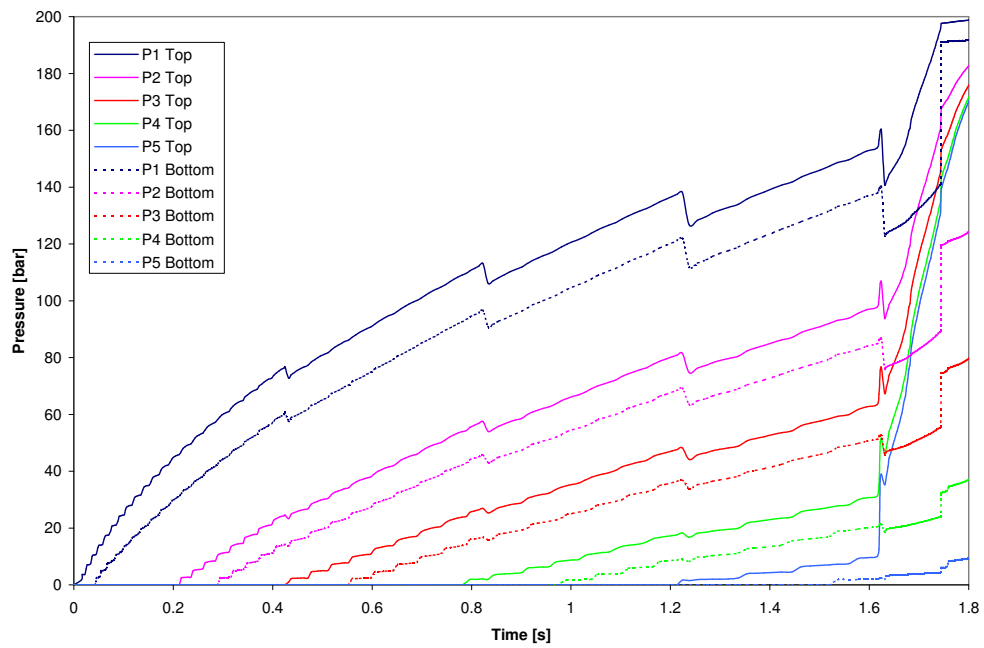


Figure 7.2: Non-axisymmetric fill cavity pressures - Cadmould

Chapter 8

Conclusion

A program Sopfil was written to create an axisymmetric model of lomolding and injection moulding. Sopfil models the flow of polymer melt from the barrel into the cavity. Various aspects of Sopfil (momentum, energy scheme) were validated against analytical solutions that exist for certain simplified problems. Sopfil performs adequately in the validation cases. Sopfil and Cadmould (a commercial injection moulding program) were used to compare lomolding to injection moulding. It was found that Sopfil and Cadmould give similar results (within 10%) for the cavity regions of lomolding and injection moulding. Because of the similar results it was concluded that the cavity of lomolding can be modelled with Cadmould. Therefore Cadmould can be used to analyse complex 3D parts to be made with lomolding.

The simulations programs did not compare well to the experimental lomolding data. There are reasons to doubt the accuracy of the experimental results which were stated in the main body of the thesis.

Sopfil and Cadmould were used to compare lomolding to injection moulding in the filling phase. The processes were compared using the same geometry and process parameters (most notably the same volume flow rate). Lomolding was found to have the following advantages over injection moulding:

- 14% lower maximum cavity pressure during filling.
- 0.5% lower clamping force.
- Maximum shear rates that are an order of magnitude lower.
- 48% lower maximum shear stresses on in-mould-decorations.

During the filling phase, lomolding mould's clamping force was 0.42% lower than injection moulding, which is not significantly lower. The cavity pressures are lower, but also not enough to be a major advantage over injection moulding. The lower shear rates and stresses of lomolding are substantial advantages. The lower shear rates will mean that there is substantially less fibre breakage. Less fibre breakage will mean that parts made by lomolding will be stronger than parts made by injection moulding. The lower shear stresses of lomolding will mean that lomolding has a better potential for in-mould-decoration than injection moulding.

Sopfil was used to simulate the flow of polymer melt from under the piston into the cavity. It was found that the piston pushed solidified polymer build-up (on the moulding barrel) into the cavity, which resulted in a restriction of the flow and an increase in the shear rates.

Overall: Sopfil is able to model the mould filling phase of lomolding and injection moulding. It was used to compare the two processes with each other as well as gain insight into the flow under the piston.

8.1 Recommendations for further work

There is scope for more work and improvements on this project. In this section some aspects will be discussed, which the author feels will considerably improve the quality of the work, and provide more insight into lomolding.

8.1.1 Numerical work

Nassehi (2002) recommends doing one momentum iteration followed by an energy iteration, etc. This is different to Sopfil where the momentum solution is iterated to convergence before an energy iteration is done. The energy and momentum equations are strongly coupled through the viscosity term, and therefore structuring Sopfil as Nassehi (2002) recommends might result in a more efficient program.

The current solidification scheme could be improved. The current method (adjusting the viscosity) does not give good results in the region of the solid layer. Typical results that are of interest in this region are the shear stress, shear rate and the thickness of the solid layer. Improvements can either be made to the solidification model function, or by using an ALE method. The ALE mesh could be moved at each time step so that the mesh boundary is

at the solid layer interface. Therefore the solidification model will not be necessary.

Using an ALE scheme in the cavity region will also have the advantage of being able to model the moving flow front. This will do away with the need for the VOF method. The elements will, however, become distorted with an ALE scheme, and therefore a remeshing scheme will have to be used. The domain would not be a simple rectangular domain anymore like the piston, and it would probably be necessary to use a scheme that can mesh complex domains. Examples of such schemes can be found in Lee *et al.* (2003) and Bastian & Li (2003).

More work can be done on characterising and optimising lomolding. Aspects that could be examined are the piston velocity profile, the piston to part diameter ratio and the cavity entrance geometry (where the piston meets the cavity).

8.1.2 Experimental work

The pressure sensors all currently lie on one radial line. Experiments have shown that the flow is not axisymmetric. A Cadmould simulation showed that non-axisymmetric filling has a large influence on the pressures. To record the effects of non-axisymmetric filling it is recommended that some pressure sensors are placed at regular intervals around the cavity: for example 3 sensors could be placed at 120° intervals on a constant radius.

The viscosity of a polymer can vary from batch to batch. To ensure that the correct viscosity is used in the simulations a sample of the polymer from the batch used for the experiments should be calibrated.

Simulations showed that the clamping force of the lomolding machine is not adequate for the current experimental part. A smaller part should be used in future simulations, which would fall within the clamping force capacity of the lomolding machine.

The lomolding machine was built by modifying an injection moulding machine. If the experimental part could also be made as an injection moulded part, it would be possible to compare the two processes experimentally. Having an experimental comparison would allow further comparisons between the two processes and the numerical work. When choosing the part for injection moulding, it should be checked whether the part can be made with the cur-

rent screw extruder. An important parameter to check is the maximum shot volume.

The current pressure sensors' ranges are far too big. Sensors with a smaller range would give better measurements. From the simulations for the current experimental part it is recommended that the full scale range should be 200 bar.



List of References

- Abramzon, B. and Elata, C. (1984). Unsteady heat transfer from a single sphere in stokes flow. *Int. J. Heat Mass Transfer*, vol. 27, no. 5, pp. 687–695. (Cited on pages 37 and 39.)
- Barone, M.R. and Osswald, T.A. (1989). *Boundary element solution of field problems in Tucker, C (eds.).Fundamentals of computer modeling for polymer processing.* Hanser, New York. (Cited on page 9.)
- Basell (2004). Webpage of basell polyolefins. Available at: <http://www.basell.com>, [April 2004]. (Cited on page 47.)
- Bastian, M. and Li, B.Q. (2003). An efficient automatic mesh generator for quadrilateral elements implemented using c++. *Finite elements in analysis and design*, vol. 39, pp. 905–930. (Cited on pages 10 and 77.)
- Bathe, K.J. (1982). *Finite element procedures in engineering analysis.* Prentice-Hall International, London. (Cited on page 10.)
- Bathe, K.J. and Wilson, E.L. (1976). *Numerical methods in finite element analysis.* Prentice-Hall, Inc., New Jersey. (Cited on page 106.)
- Bhate, A. (1996). C-mold viscosity models [Online]. Available at: http://www.menet.umn.edu/~bhate/temp/TA_web/mat1/tutorials/visc.html, [July2003]. (Cited on page 109.)
- Boothroyd, G., Dewhurst, P. and Knight, W. (1994). *Product design for manufacturing and assembly.* Marcel Dekker, Inc., New York. (Cited on page 13.)
- Chiang, H.H., Hieber, C.A. and Wang, K.K. (1991). A unified simulation of the filling and postfilling stages in injection molding. part 1: Formulation. *Polymer Engineering and Science*, vol. 31, no. 2, pp. 116–124. (Cited on page 13.)
- Churchill, S. (1988). *Viscous flow - The practical use of theory.* Butterworths, London. (Cited on page 31.)

- Cook, R.C., Malkus, D.S., Plesha, M.E. and Witt, R.J. (2002). *Concepts and applications of finite element analysis*. 4th edn. John Wiley & Sons, New York. (Cited on pages 10 and 88.)
- Davies, R. (2004). Newmat c++ matrix library [Online]. Available at: http://www.robertnz.net/nm_intro.htm, [March 2004]. (Cited on page 107.)
- Dealy, J. and Wissbrun, K. (1999). *Melt rheology and its role in plastics processing*. Kluwer Academic Publishers, London. (Cited on page 33.)
- Demmel, J.W. (1997). *Applied numerical linear algebra*. SIAM, Philadelphia. (Cited on page 106.)
- Dongarra, J. (2003). Freely available software for linear algebra on the web [Online]. Available at: <http://www.netlib.org/utk/people/JackDongarra/la-sw.html>, [March 2004]. (Cited on page 107.)
- Gerald, C.F. and Wheatley, P.O. (1999). *Applied numerical analysis*. 6th edn. Addison-Wesley, Reading, Massachusetts. (Cited on page 92.)
- Golub, G.H. and Loan, C.F. (1989). *Matrix computations*. 2nd edn. The Johns Hopkins University Press, London. (Cited on page 105.)
- Guceri, S.I. (1989). *Finite difference solution of field problems in Tucker, C (eds.). Fundamentals of computer modeling for polymer processing*. Hanser, New York. (Cited on page 9.)
- Hieber, C.A. and Shen, S.F. (1980). A finite-element / finite difference simulation of the injection-molding process. *Journal of Non-Newtonian Fluid Mechanics*, vol. 7, pp. 1–32. (Cited on page 13.)
- Hirsch, C. (1988). *Numerical computation of internal and external flows - Fundamentals of numerical discretization*, vol. 1. John Wiley & Sons Ltd., London. (Cited on page 17.)
- Hirt, C.W. and Nichols, B.D. (1981). Volume of fluid (vof) method for the dynamics of free boundaries. *J. Computational Physics*, vol. 39, pp. 201–225. (Cited on pages 14 and 15.)
- Lee, C. and Castro, J. (1989). *Model simplification in Tucker, C (eds.). Fundamentals of computer modeling for polymer processing*. Hanser, New York. (Cited on pages 33, 52, and 89.)
- Lee, K., Kim, I., Cho, D. and Kim, T. (2003). An algorithm for automatic 2d quadrilateral mesh generation with line constraints. *Computer-Aided Design*, vol. 35, pp. 1055–1068. (Cited on page 77.)

- Malcevic, I. and Ghattas, O. (2002). Dynamic-mesh finite element method for lagrangian computational fluid dynamics. *Finite Elements in Analysis and Design*, vol. 38, pp. 965–982. (Cited on page 10.)
- Mills, A.F. (1995). *Heat and mass transfer*. Irwin, London. (Cited on page 38.)
- Nassehi, V. (2002). *Practical aspects of finite element modelling of polymer processing*. John Wiley & Sons, LTD, London. (Cited on pages 5, 6, 8, 10, 11, 12, 13, 16, 17, 18, 76, 88, 89, 97, 100, and 101.)
- Nassehi, V. and Ghoreishy, M.H.R. (1998). Finite element analysis of mixing in partially filled twin blade internal mixers. *Intern. Polymer processing*, vol. XIII, pp. 231–238. (Cited on pages 6 and 14.)
- Osswald, T. and Menges, G. (1995). *Materials science of polymers for engineers*. Hanser Publishers, Munich. (Cited on page 108.)
- Papalambros, P.Y. and Wilde, D.J. (2000). *Principles of optimal design - Modeling and computation*. 2nd edn. Cambridge University Press, Cambridge. (Cited on page 104.)
- Petera, J. and Nassehi, V. (1995). Use of the finite element modelling technique for the improvement of viscometry results obtained by cone-and-plate rheometers. *Journal of non-Newtonian Fluid Mechanics*, vol. 58, pp. 1–24. (Cited on page 6.)
- Petera, J. and Nassehi, V. (1996). Finite element modelling of free surface viscoelastic flows with particular application to rubber mixing. *International Journal for Numerical Methods in Fluids*, vol. 23, pp. 1117–1132. (Cited on page 15.)
- Pittman, J.F.T. (1989). *Finite elements for field problems in Tucker, C (eds.). Fundamentals of computer modeling for polymer processing*. Hanser, New York. (Cited on pages 10, 22, 84, and 92.)
- Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P. (1999). *Numerical recipes in C*. 2nd edn. Cambridge University Press, Cambridge. (Cited on page 103.)
- Ravindran, K. and Lewis, R.W. (1998). Finite element modelling of solidification effects in mould filling. *Finite elements in analysis and design*, vol. 31, pp. 99–116. (Cited on page 19.)
- Richardson, S.M. (1983). Injection moulding of thermoplastics: freezing during mould filling. *Rheol. Acta*, vol. 22, pp. 223–236. (Cited on page 41.)
- Shen, Y.K. (1998). Study on moving boundary problems of injection moulding. *Int. Comm. Heat Mass Transfer*, vol. 25, pp. 701–710. (Cited on pages 5, 6, 19, and 28.)

- Shin, S. and Lee, W.I. (2000). Finite element analysis of incompressible flows with moving free surface by selective volume of fluid method. *Int. J. Heat and fluid flow*, vol. 21, pp. 197–206. (Cited on pages 14 and 15.)
- Simcon (2002a). Cadmould version 6. (Cited on pages 4, 5, and 50.)
- Simcon (2002b). *Cadmould version 6 - Program discription - Theory*. Simcon, IKV. (Cited on page 108.)
- Thiart, G.D. (2000). Numerical fluid dynamics 844. Class Notes, University of Stellenbosch. (Cited on page 104.)
- Tucker, C.L. (ed.) (1989). *Fundamentals of computer modeling for polymer processing*. Hanser, New York. (Cited on page 5.)
- Uchiyama, T. (2001). Ale finite element method for gas-liquid two-phase flow including moving boundary based on an incompressible two-fluid model. *Nuclear Engineering and Design*, vol. 205, pp. 69–82. (Cited on page 10.)
- Wang, H.P. and McLay, R.T. (1986). Automactic remeshing scheme for modelling hot forming process. *Journal of Fluids Engineering*, vol. 108, pp. 465–469. (Cited on pages 11 and 19.)
- White, F. (1974). *Viscous fluid flow*. McGraw-Hill, Inc. (Cited on page 7.)
- Wylie, C.R. and Barrett, L.C. (1995). *Advanced engineering mathematics*. 6th edn. McGraw-Hill, New York. (Cited on page 84.)
- Yang, L., Chen, S. and Charmchi, M. (1991). Steady solidification of non-newtonian fluid flowing in a round tube. *Polym. Eng. Sci*, vol. 31, no. 3, pp. 191–196. (Cited on page 41.)

Appendix A

Mathematics

Vector notation is used in this document. A vector is represented by a bar over the symbol of the vector (e.g. \bar{v}).

$$\bar{v} = \nu_x \mathbf{i} + \nu_y \mathbf{j} + \nu_z \mathbf{k} \quad (\text{A.0.1})$$

A.1 Gradient operator

The gradient operator is represented by ∇

$$\nabla = \frac{\partial}{\partial x} \cdot + \frac{\partial}{\partial y} \cdot + \frac{\partial}{\partial z} \cdot \quad (\text{A.1.1})$$

When applied to a vector (\bar{F}):

$$\nabla \cdot \bar{F} = \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} + \frac{\partial F_z}{\partial z} \quad (\text{A.1.2})$$

The axisymmetric version of the gradient operator can be seen in equation A.1.3.

$$\nabla = \frac{1}{r} \frac{\partial}{\partial r} (r \cdot) + \frac{\partial}{\partial z} \cdot \quad (\text{A.1.3})$$

A.2 Material derivative

The material derivative is represented by $\frac{D}{Dt}$, it contains the time derivative as well as the convective effects.

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + \bar{v} \cdot \nabla \quad (\text{A.2.1})$$

A.3 Kronecker Delta

The Kronecker Delta is a tensor that is defined as follows Pittman (1989).

$$\bar{\delta} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{A.3.1})$$

A.4 Integration by parts

Integration by parts is used to reduce the order of derivatives, and it is also useful for applying natural boundary conditions (gradient boundary conditions e.g. $\frac{dT}{dx}$.)

First of all consider the chain rule applied using the ∇ operator

$$\nabla \cdot (PQ) \equiv (\nabla \cdot P)Q + P(\nabla \cdot Q) \quad (\text{A.4.1})$$

where P and Q are functions of the coordinates. Integrating across the domain (Ω) and rearranging gives

$$\int_{\Omega} P(\nabla \cdot Q)d\Omega = - \int_{\Omega} (\nabla \cdot P)Qd\Omega + \int_{\Omega} \nabla \cdot (PQ)d\Omega \quad (\text{A.4.2})$$

Green's divergence theorem is stated in equation A.4.3 (Wylie and Barrett, 1995) where \bar{F} is a vector function. Applying Greens divergence theorem to A.4.2, (taking $\bar{F} = PQ$) we obtain the integration by parts formula (A.4.4). It can be used for one, two or three dimensions.

$$\int_{\Omega} \nabla \cdot \bar{F}d\Omega = \int_{\Gamma} \bar{F} \cdot \bar{n}d\Gamma \quad (\text{A.4.3})$$

$$\int_{\Omega} P(\nabla \cdot Q)d\Omega = - \int_{\Omega} (\nabla \cdot P)Qd\Omega + \int_{\Gamma} PQ \cdot \bar{n}d\Gamma \quad (\text{A.4.4})$$

Appendix B

Finite element method

This appendix will give a brief description of the finite element method. The FEM is part of the weighted residual method family. For the sake of explaining the process, the steady state heat conduction equation will be used as an example. The governing equation for heat conduction is:

$$\nabla^2 T + \dot{S} = 0 \quad (\text{B.0.1})$$

B.1 Nodes, elements and shape functions

The domain is divided up into elements and nodes. Every point in the domain must be covered by no more and no less than one element. Elements may not overlap. Various element shapes are possible. The most popular shapes, for a two-dimensional domain, are triangular and quadrilateral elements.

At any a point in the element the temperature can be given by the nodal temperatures and the shape functions (see equation B.1.1). Each node has its own shape function. A shape function has the properties that at its node it has the value of one, at any other node it has the value of zero. At any point in the domain the sum of the shape functions is equal to one. A shape function is also defined across all the other elements (and nodes) in the domain, although it has a value of zero outside its own element.

$$\bar{T}(x, y) = \sum_j N(x, y)_j T_j \quad (\text{B.1.1})$$

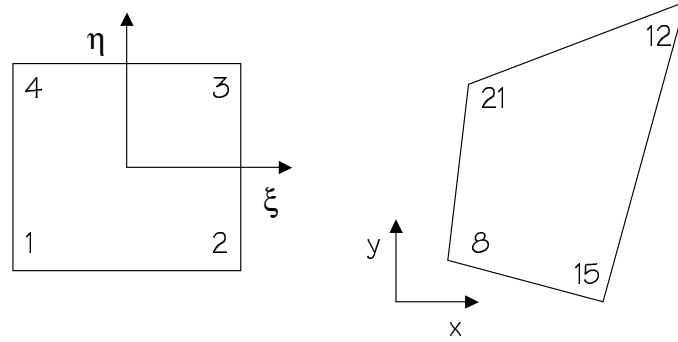


Figure B.1: Local and global elements

B.2 Local elements

For every node (in the whole domain) there will be a shape function with different constants. This is not desirable. A solution is found by mapping the elements to one master local element. The local coordinates are ξ and η . The local shape functions are used to map the global coordinates (equation B.2.1), where i refers to the element's nodes.

$$\begin{aligned} x &= \sum_i N(\xi, \eta)_i x_i \\ y &= \sum_i N(\xi, \eta)_i y_i \end{aligned} \quad (\text{B.2.1})$$

Quite often it is necessary to have an expression for the global shape functions differentiated with respect to the global coordinates. Consider the chain rule differentiation shown below.

$$\begin{aligned} \frac{\partial N(\xi, \eta)_i}{\partial \xi} &= \frac{\partial N(\xi, \eta)_i}{\partial x} \frac{\partial x}{\partial \xi} + \frac{\partial N(\xi, \eta)_i}{\partial y} \frac{\partial y}{\partial \xi} \\ \frac{\partial N(\xi, \eta)_i}{\partial \eta} &= \frac{\partial N(\xi, \eta)_i}{\partial x} \frac{\partial x}{\partial \eta} + \frac{\partial N(\xi, \eta)_i}{\partial y} \frac{\partial y}{\partial \eta} \end{aligned} \quad (\text{B.2.2})$$

Written in matrix form

$$\begin{bmatrix} \frac{\partial N(\xi, \eta)_i}{\partial \xi} \\ \frac{\partial N(\xi, \eta)_i}{\partial \eta} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} \begin{bmatrix} \frac{\partial N(\xi, \eta)_i}{\partial x} \\ \frac{\partial N(\xi, \eta)_i}{\partial y} \end{bmatrix} \quad (\text{B.2.3})$$

The Jacobian is defined in equation B.2.4. The Jacobian is a function of ξ

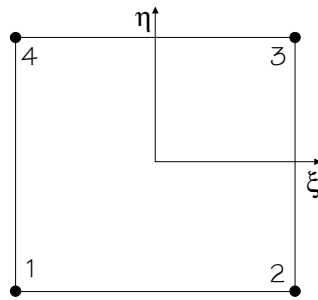


Figure B.2: Bilinear rectangular element

and η . It is a measure of how the global coordinates change with respect to the local coordinates.

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} \quad (\text{B.2.4})$$

The global derivatives can now be expressed in terms of the local derivatives and the inverse of the Jacobian (equation B.2.5).

$$\begin{bmatrix} \frac{\partial N(\xi, \eta)_i}{\partial x} \\ \frac{\partial N(\xi, \eta)_i}{\partial y} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \frac{\partial N(\xi, \eta)_i}{\partial \xi} \\ \frac{\partial N(\xi, \eta)_i}{\partial \eta} \end{bmatrix} \quad (\text{B.2.5})$$

The integration transformation from global to local coordinates is given by equation B.2.6.

$$dx dy = \det(\mathbf{J}) d\xi d\eta \quad (\text{B.2.6})$$

The advantages of using local elements is that only the shape functions for one local element need to be programmed. The integration of local shape functions also has attractive advantages that will be discussed in the section on numerical integration.

B.3 Quadratic elements - some examples

Some of the most common quadratic elements are presented here. The elements used for Sopfil are also presented.

B.3.1 Bilinear rectangular element

The bilinear element is a very versatile element. The element can be seen in figure B.2. Its shape functions are (Cook *et al.*, 2002):

$$\begin{aligned} N_1 &= \frac{1}{4}(1 - \xi)(1 - \eta) \\ N_2 &= \frac{1}{4}(1 + \xi)(1 - \eta) \\ N_3 &= \frac{1}{4}(1 + \xi)(1 + \eta) \\ N_4 &= \frac{1}{4}(1 - \xi)(1 + \eta) \end{aligned} \quad (\text{B.3.1})$$

B.3.2 Rectangular Taylor-Hood element

The rectangular Taylor-Hood element (see figure B.3) satisfies the Babuska-Brezzi condition. There are nine velocity nodes. These have a quadratic shape function, which can be seen in equation B.3.2 (Nassehi, 2002). There are four pressure nodes. Their shape functions are the same as the bilinear rectangular elements' (equation B.3.1).

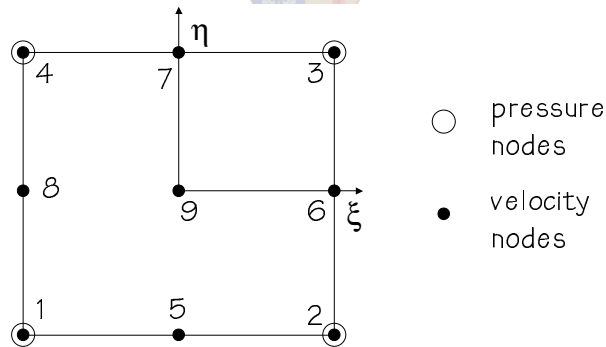


Figure B.3: Rectangular Taylor-Hood element

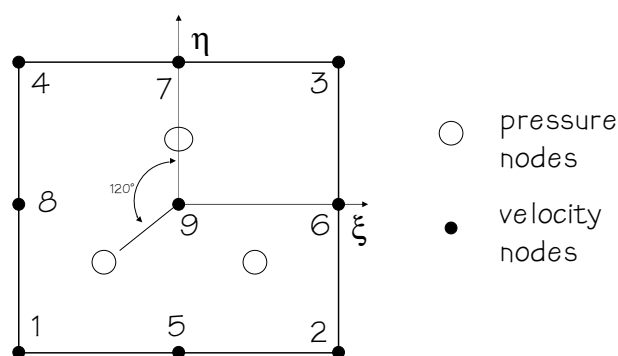


Figure B.4: Rectangular Crouzeix-Raviart element

$$\begin{aligned}
 N_1 &= \frac{1}{4}\xi(1-\xi)\eta(1-\eta) \\
 N_2 &= -\frac{1}{4}\xi(1+\xi)\eta(1-\eta) \\
 N_3 &= \frac{1}{4}\xi(1+\xi)\eta(1+\eta) \\
 N_4 &= -\frac{1}{4}\xi(1-\xi)\eta(1+\eta) \\
 N_5 &= -\frac{1}{2}(1-\xi^2)\eta(1-\eta) \\
 N_6 &= \frac{1}{2}\xi(1+\xi)(1-\eta^2) \\
 N_7 &= \frac{1}{2}(1-\xi^2)\eta(1+\eta) \\
 N_8 &= -\frac{1}{2}\xi(1-\xi)(1-\eta^2) \\
 N_9 &= (1-\xi^2)(1-\eta^2)
 \end{aligned} \tag{B.3.2}$$

B.3.3 Rectangular Crouzeix-Raviart element

The rectangular Crouzeix-Raviart element can be seen in figure B.4. The velocity nodes' shape functions are the same as those for the Taylor-Hood element (equation B.3.2). The pressure nodes' shape functions can be seen in equation B.3.3 (Nassehi, 2002). The Crouzeix-Raviart element also satisfies the Babuska-Brezzi condition (Lee and Castro, 1989).

$$\begin{aligned}
N_1 &= \frac{4}{3}\eta + \frac{1}{3} \\
N_2 &= 1.1547005 \xi - \frac{2}{3}\eta + \frac{1}{3} \\
N_3 &= -1.1547005 \xi - \frac{2}{3}\eta + \frac{1}{3}
\end{aligned}
\tag{B.3.3}$$

B.4 The finite element Galerkin method

Returning to the example of the heat equation, the approximation for the temperature (equation B.1.1) can now be substituted back into the governing equation (equation B.0.1). Because the temperature is an approximate solution, there will be an error, and the equations will no longer be equal to zero. This error is called the residual (equation B.4.1).

$$r = \nabla^2 \bar{T} + \dot{S} \tag{B.4.1}$$

To find the best solution, the residual must be minimized in some sense. This is done by multiplying it with a weighting function, and integrating this product across the domain and setting the answer equal to zero (equation B.4.2).

$$\int_{\Omega} wr = \int_{\Omega} w(\nabla^2 \bar{T} + \dot{S}) = 0 \tag{B.4.2}$$

In the Galerkin method the weighting function is equal to the shape function. To reduce the order of the derivatives, integration by parts (see A.4) is applied (equation B.4.4). This also presents a convenient way to introduce natural boundary conditions, which will be discussed further in section B.7.

$$\int_{\Omega} N_i(\nabla^2 \bar{T} + \dot{S}) = 0 \tag{B.4.3}$$

$$\int_{\Omega} (\nabla N_i)(\nabla \bar{T}) = \int_{\Omega} N_i \dot{S} + \int_{\Gamma} N_i \bar{n}(\nabla \bar{T}) d\Gamma \tag{B.4.4}$$

B.5 System of equations

When the residuals have been integrated the end product is a set of linear equations B.5.1. When the boundary conditions have been applied (discussed in section B.7), and if the temperature gradients are known, the set of equations can be solved. In equation B.5.1 the stiffness matrix is represented by \overline{S} , the load vector by \overline{L} and the vector of unknowns by $\overline{\phi}$ (in this case the nodal temperatures). The set of linear equations are solved by a matrix solver (see appendix E for details).

$$\overline{S}\overline{\phi} = \overline{L} \quad (\text{B.5.1})$$

The stiffness matrix and load vector are assembled as follows: Let there be n nodes and s elements. The terms on the right hand side of equation B.4.4 do not contain any unknowns, and therefore are part of the load vector.

Now the weighting function (N_i) consists of n different shape functions. N_1 will be the first row of the stiffness matrix, N_2 will be the second, and so on. The index j is for the columns of the stiffness matrix.

The domain integral can be split up into elemental integrals. The integral across one element results in an elemental stiffness matrix. Equation B.5.2 shows the splitting up procedure, the approximation for the temperature (equation B.1.1) has been substituted in.

$$\int_{\Omega} (\nabla N_i)(\nabla \overline{T}) = \sum_{e=1}^s \int_{\Omega_e} (\nabla N_i)(\nabla N_j) T_j d\Omega_e \quad (\text{B.5.2})$$

The global stiffness matrix is made up of the sum of the elemental stiffness matrixes. Consider the situation depicted in figure B.1. Let the local nodes (1,2,3 and 4) correspond to the global nodes (8,15,12 and 21). The local stiffness matrix will then look like equation B.5.3. The shape functions are only defined for the local element domain. The global shape functions (defined for the whole domain) will be present in the global stiffness matrix, and are

built up from elemental contributions.

$$\bar{S}_e = \begin{bmatrix} \int(\nabla N_8)(\nabla N_8) & \int(\nabla N_8)(\nabla N_{15}) & \int(\nabla N_8)(\nabla N_{12}) & \int(\nabla N_8)(\nabla N_{21}) \\ \int(\nabla N_{15})(\nabla N_8) & \int(\nabla N_{15})(\nabla N_{15}) & \int(\nabla N_{15})(\nabla N_{12}) & \int(\nabla N_{15})(\nabla N_{21}) \\ \int(\nabla N_{12})(\nabla N_8) & \int(\nabla N_{12})(\nabla N_{15}) & \int(\nabla N_{12})(\nabla N_{12}) & \int(\nabla N_{12})(\nabla N_{21}) \\ \int(\nabla N_{21})(\nabla N_8) & \int(\nabla N_{21})(\nabla N_{15}) & \int(\nabla N_{21})(\nabla N_{12}) & \int(\nabla N_{21})(\nabla N_{21}) \end{bmatrix} \quad (\text{B.5.3})$$

When the global stiffness matrix is built up, matrix element (1,1) from equation B.5.3 will be added to global matrix element (8,8). Element (2,1) will be added to global element (15,8), etc.

The Load vector is built in a similar method.

B.6 Numerical integration

The FEM integrals must be evaluated using numerical methods. Well established numerical integration routines are available. Examples of such routines are Trapezoidal and Simpson $\frac{3}{8}$ rule (Gerald and Wheatley, 1999). The problem with these routines is that for good accuracy a large number of points must be sampled. This becomes computationally expensive.

A better solution is found by using Gauss-Legendre quadrature (equation B.6.1). Using this method, with n sampling points, an exact answer is obtained for a polynomial up to the order of $2n - 1$ (Pittman, 1989). The limitation of this method is that the integration must be performed between the bounds of $(-1, 1)$. But if local elements coordinates are used this is not a problem, as the element boundaries are between -1 and 1 .

$$\int_{-1}^1 f(\xi) d\xi \approx \sum_{l=1}^n W_l f(\xi_l) \quad (\text{B.6.1})$$

Each sampling point is multiplied by the appropriate weighting factor. A table of the first few sampling points and their weighting factors can be seen in table B.1.

Integration can also be done in two dimensions (equation B.6.2)

$$\int_{-1}^1 \int_{-1}^1 f(\xi, \eta) d\xi d\eta \approx \sum_{l=1}^n \sum_{m=1}^n W_l W_m f(\xi_l \eta_m) \quad (\text{B.6.2})$$

n	Weighting factor	Sampling point
1	2	0
2	1	± 0.5773502692
3	0.5555555556 0.8888888889	± 0.7745966692 0

Table B.1: Gauss Quadrature points

B.7 Boundary conditions

Two types of boundary conditions are of primary concern, namely Dirichlet (also referred to as essential boundary conditions) and Von Neuman (also referred to as natural boundary conditions).

Dirichlet boundary conditions are values that are fixed. For instance the temperature at one node could be specified to equal a fixed value. These boundary conditions can be applied to the system of equations in two ways.

The first way is to substitute the nodal value in, take all the known values across to the Load vector, and then solve the system of equations. The matrix size is reduced by one row and column. A lot of matrix ordering and reshuffling work can be saved if the reduction is done while the stiffness matrix is being assembled.

The second way is to replace the diagonal element of the stiffness matrix with a large number. The corresponding load vector element is replaced with the desired value multiplied by the large number. When the system of equations is solved, the diagonal term will dominate that nodal value.

The first method results in a smaller matrix to solve. The bandwidth of the first matrix is usually smaller than that of the second method. The solution is also more accurate. These factors result in a faster solution, and for this reason the first method is used.

Von Neuman boundary conditions are gradient boundary conditions. An example would be to specify a heat flux into the domain ($\frac{dT}{dx}$). These are applied via the boundary integral (after integration by parts, see equation B.4.4).

A common Von Neuman boundary condition is a flux value of zero. In the temperature case this represents an insulated boundary. For flow simulations the outlet boundary conditions are usually of this type. This boundary condition is the easiest to apply, as all that has to be done is to set the surface

integral to zero.

It should be noted that, for elemental boundaries that do not lie on the global domain boundaries, the elemental surface integrals cancel each other out when the global system of equations are assembled and therefore do not need to be calculated.

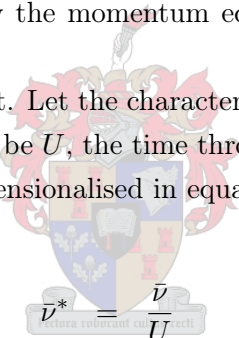


Appendix C

Dimensional analysis of the momentum equation

This appendix will show how the momentum equation was simplified by dimensional analysis.

Consider a typical element. Let the characteristic length of the element be H , the characteristic velocity be U , the time through the element be T . Then the variables can be non-dimensionalised in equation C.0.1


$$\begin{aligned}\bar{v}^* &= \frac{\bar{v}}{U} \\ \bar{x}^* &= \frac{\bar{x}}{H} \\ \nabla^* &= H\nabla \\ t^* &= \frac{t}{T} \\ p^* &= \frac{pH}{\eta U} \\ \bar{D}^* &= \frac{H}{U}\bar{D}\end{aligned}$$

(C.0.1)

Now substituting back into the momentum equation 2.2.2 and rearranging, the dimensionless equation can be found (equation C.0.3).

$$\frac{\rho U}{T} \frac{\partial \bar{v}^*}{\partial t^*} + \frac{\rho U^2}{H} (\bar{v}^* \cdot \nabla^* \bar{v}^*) = -\frac{\eta U}{H^2} (\nabla^* p^* \bar{\delta}) + \frac{2\eta U}{H^2} (\nabla^* \bar{D}^*) + \rho g \quad (\text{C.0.2})$$

$$\frac{\rho H^2}{T\eta} \frac{\partial \bar{v}^*}{\partial t^*} + \frac{\rho U H}{\eta} (\bar{v}^* \cdot \nabla^* \bar{v}^*) = -(\nabla^* p^* \bar{\delta}) + 2(\nabla^* \bar{D}^*) + \frac{\rho g H^2}{\eta U} \quad (\text{C.0.3})$$

Taking the typical values for polymer flow, shown below, the magnitude of the coefficients of equation C.0.3 can be calculated. The coefficients of equation C.0.5 are much smaller than the others (2 orders of magnitude). Therefore they can be dropped from equation C.0.3, which is then written as equation C.0.6. This can then be changed back into dimensional form (equation C.0.7)

$$\begin{aligned} \rho &= 1000 \frac{kg}{m^3} \\ \eta &= 100 Pa \cdot s \\ H &= 0.005 m \\ U &= 0.1 m/s \\ T &= \frac{0.005}{0.1} = 0.05 s \\ g &= 9.81 m/s^2 \end{aligned} \quad (\text{C.0.4})$$

$$\begin{aligned} \frac{\rho U H}{\eta} &= 0.005 \\ \frac{\rho g H^2}{\eta U} &= 0.00245 \\ \frac{\rho H^2}{T\eta} &= 0.005 \end{aligned} \quad (\text{C.0.5})$$

$$0 = -\nabla^* p^* \bar{\delta} + 2\nabla^* \bar{D}^* \quad (\text{C.0.6})$$

$$0 = -\nabla p \bar{\delta} + 2\eta \nabla \bar{D} \quad (\text{C.0.7})$$

Appendix D

FEM equations for polymer flow and other algorithms

This appendix will briefly present the derivation of the FEM equations. Other important algorithms used in this thesis are also discussed and presented.

D.1 Deriving the FEM equations from the governing equations

D.1.1 UVP scheme

This section will show how to get from the governing equations to the FEM equations. This formulation comes from Nassehi (2002). Note that for all the equations in this section an axisymmetric coordinate system will be used.

The expansion of the mass conservation equation (equation 2.2.1), can be seen in equation D.1.1. The expansion of the rate of deformation tensor (equation 2.3.2) can be seen in equation D.1.2. The expansion (in the r and z directions) of the momentum equation (equation 2.2.4) can be seen in equation D.1.3.

$$\frac{\partial u}{\partial r} + \frac{u}{r} + \frac{\partial v}{\partial z} = 0 \quad (\text{D.1.1})$$

$$\bar{D} = \begin{bmatrix} \frac{\partial u}{\partial r} & \frac{1}{2} \left(\frac{\partial u}{\partial z} + \frac{\partial v}{\partial r} \right) \\ \frac{1}{2} \left(\frac{\partial u}{\partial z} + \frac{\partial v}{\partial r} \right) & \frac{\partial v}{\partial z} \end{bmatrix} \quad (\text{D.1.2})$$

$$\begin{aligned}
& -\frac{\partial p}{\partial r} + \frac{\partial}{\partial r} \left(2\eta \frac{\partial u}{\partial r} \right) + \frac{2\eta}{r} \frac{\partial u}{\partial r} - \frac{2\eta}{r} \frac{u}{r} + \frac{\partial}{\partial z} \left[\eta \left(\frac{\partial v}{\partial r} + \frac{\partial u}{\partial z} \right) \right] = 0 \\
& -\frac{\partial p}{\partial z} + \frac{\partial}{\partial z} \left(2\eta \frac{\partial v}{\partial z} \right) + \frac{\eta}{r} \left(\frac{\partial v}{\partial r} + \frac{\partial u}{\partial z} \right) + \frac{\partial}{\partial r} \left[\eta \left(\frac{\partial v}{\partial r} + \frac{\partial u}{\partial z} \right) \right] = 0 \quad (\text{D.1.3})
\end{aligned}$$

The local approximation (equation D.1.4) for velocity and pressure can be substituted into equation D.1.3 and D.1.1. Note that pressure has a lower order shape function (M) than velocity (N). The Galerkin FEM can also be applied and the results can be seen in equation D.1.5

$$\begin{aligned}
\bar{u} &= \sum_{j=1}^n N_j u_j \\
\bar{v} &= \sum_{j=1}^n N_j v_j \\
\bar{p} &= \sum_{l=1}^m M_l p_l \quad (\text{D.1.4})
\end{aligned}$$

$$\begin{aligned}
& \int_{\Omega} N_i \left\{ -\frac{\partial \bar{p}}{\partial r} + \frac{\partial}{\partial r} \left(2\eta \frac{\partial \bar{u}}{\partial r} \right) + \frac{2\eta}{r} \frac{\partial \bar{u}}{\partial r} - \frac{2\eta}{r} \frac{\bar{u}}{r} + \frac{\partial}{\partial z} \left[\eta \left(\frac{\partial \bar{v}}{\partial r} + \frac{\partial \bar{u}}{\partial z} \right) \right] \right\} r dr dz = 0 \\
& \int_{\Omega} N_i \left\{ -\frac{\partial \bar{p}}{\partial z} + \frac{\partial}{\partial z} \left(2\eta \frac{\partial \bar{v}}{\partial z} \right) + \frac{\eta}{r} \left(\frac{\partial \bar{v}}{\partial r} + \frac{\partial \bar{u}}{\partial z} \right) + \frac{\partial}{\partial r} \left[\eta \left(\frac{\partial \bar{v}}{\partial r} + \frac{\partial \bar{u}}{\partial z} \right) \right] \right\} r dr dz = 0 \\
& \int_{\Omega} -M_l \left\{ \frac{\partial \bar{u}}{\partial r} + \frac{\bar{u}}{r} + \frac{\partial \bar{v}}{\partial z} \right\} r dr dz = 0 \quad (\text{D.1.5})
\end{aligned}$$

Using integration by parts and simplifying, the equations above can be written as equations D.1.6. These are the working equations that were programmed into Sopfil. Note that i and j are indexes of the velocity nodes, and l is an index of the pressures nodes.

$$\begin{bmatrix} A_{ij}^{11} & A_{ij}^{12} & A_{il}^{13} \\ A_{ij}^{21} & A_{ij}^{22} & A_{il}^{23} \\ A_{lj}^{31} & A_{lj}^{32} & A_{ll}^{33} \end{bmatrix} \begin{bmatrix} u_j \\ v_j \\ p_l \end{bmatrix} = \begin{bmatrix} B_j^1 \\ B_j^2 \\ B_l^3 \end{bmatrix} \quad (\text{D.1.6})$$

$$A_{ij}^{11} = \int_{\Omega_e} \left(2\eta \frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} + \eta \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} \right) dx dy \quad (\text{D.1.7})$$

$$A_{ij}^{12} = \int_{\Omega_e} \left(\eta \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial x} \right) dx dy \quad (\text{D.1.8})$$

$$A_{il}^{13} = - \int_{\Omega_e} \left(M_l \frac{\partial N_i}{\partial x} \right) dx dy \quad (\text{D.1.9})$$

$$A_{ij}^{21} = \int_{\Omega_e} \left(\eta \frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial y} \right) dx dy \quad (\text{D.1.10})$$

$$A_{ij}^{22} = \int_{\Omega_e} \left(\eta \frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} + 2\eta \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} \right) dx dy \quad (\text{D.1.11})$$

$$A_{il}^{23} = - \int_{\Omega_e} \left(M_l \frac{\partial N_i}{\partial y} \right) dx dy \quad (\text{D.1.12})$$

$$A_{lj}^{31} = - \int_{\Omega_e} \left(M_l \frac{\partial N_j}{\partial x} \right) dx dy \quad (\text{D.1.13})$$

$$A_{lj}^{32} = - \int_{\Omega_e} \left(M_l \frac{\partial N_j}{\partial y} \right) dx dy \quad (\text{D.1.14})$$

$$A_{ll}^{33} = 0 \quad (\text{D.1.15})$$

$$B_j^1 = \int_{\Gamma_e} N_i \left[\left(-p + 2\eta \frac{\partial u}{\partial x} \right) n_x + \eta \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) n_y \right] d\Gamma_e \quad (\text{D.1.16})$$

$$B_j^2 = \int_{\Gamma_e} N_i \left[\eta \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) n_x + \left(-p + 2\eta \frac{\partial v}{\partial y} \right) n_y \right] d\Gamma_e \quad (\text{D.1.17})$$

$$B_l^3 = 0 \quad (\text{D.1.18})$$

D.1.2 Shear rate

The shear rate is required by the viscosity functions. For axisymmetric flow the shear rate can be calculated by equation D.1.19 (Nassehi, 2002).

$$\dot{\gamma} = \left[2 \left(\frac{\partial u}{\partial r} \right)^2 + 2 \left(\frac{u}{r} \right)^2 + 2 \left(\frac{\partial v}{\partial z} \right)^2 + \left(\frac{\partial u}{\partial z} + \frac{\partial v}{\partial r} \right)^2 \right]^{\frac{1}{2}} \quad (\text{D.1.19})$$

D.1.3 Energy equation

The finite element formulation of the energy equations comes from Nassehi (2002). When the energy equation (2.2.3) is expanded for axisymmetric flow and the ALE terms substituted in, equation D.1.20 is obtained.

$$\begin{aligned} & \rho c \frac{dT}{dt} + \rho c \left[\left(u - \frac{\partial r}{\partial t} \right) \frac{\partial T}{\partial r} + \left(v - \frac{\partial z}{\partial t} \right) \frac{\partial T}{\partial z} \right] \\ & = \frac{1}{r} \frac{\partial}{\partial r} \left(r k \frac{\partial T}{\partial r} \right) + \frac{\partial}{\partial z} \left(k \frac{\partial T}{\partial z} \right) + \eta \dot{\gamma}^2 \end{aligned} \quad (\text{D.1.20})$$

Substituting in the finite element approximation for temperature (equation D.1.21) and applying the streamlined upwinding, the FEM equation (D.1.22) is obtained. Determining the upwinding weight factor is the subject of the next section. Note that the finite element approximation is not used for the time derivative, which is handled by the θ time stepping method.

$$\bar{T} = \sum_{j=1}^n N_j T_j \quad (\text{D.1.21})$$

$$\begin{aligned} & \int_{\Omega} N_i \rho c \frac{dT}{dt} r dr dz + \int_{\Omega} N_i^* \rho c \left(u \frac{\partial N_j}{\partial r} \bar{T} + v \frac{\partial N_j}{\partial z} \bar{T} \right) r dr dz \\ & - \int_{\Omega} N_i \rho c \left(\frac{\partial r}{\partial t} \frac{\partial N_j}{\partial r} \bar{T} + \frac{\partial z}{\partial t} \frac{\partial N_j}{\partial z} \bar{T} \right) r dr dz \\ & = \int_{\Omega} N_i \frac{1}{r} \frac{\partial}{\partial r} \left(r k \frac{\partial N_j}{\partial r} \bar{T} \right) + N_i \frac{\partial}{\partial z} \left(k \frac{\partial N_j}{\partial z} \bar{T} \right) r dr dz + \int_{\Omega} N_i \eta \dot{\gamma}^2 r dr dz \end{aligned} \quad (\text{D.1.22})$$

Using integration by parts and simplifying the equations, the working form of the equation is obtained in equation D.1.23. If a steady state energy equation is required, the unsteady term is set to zero and the term disappears from the equation. The elements of the matrixes and load vectors can be found in equations D.1.24, D.1.25 and D.1.26. The load vector contains the boundary integrals for boundary condition heat fluxes.

$$[C]\left\{\frac{dT}{dt}\right\} + [S]\{T\} = \{L\} \quad (\text{D.1.23})$$

$$C_{ij} = \int_{\Omega} N_i \rho c r dr dz \quad (\text{D.1.24})$$

$$\begin{aligned} S_{ij} = & \int_{\Omega} \rho c N_i^* \left(u \frac{\partial N_j}{\partial r} + v \frac{\partial N_j}{\partial z} \right) \\ & - \rho c N_i \left(\frac{\partial r}{\partial t} \frac{\partial N_j}{\partial r} + \frac{\partial z}{\partial t} \frac{\partial N_j}{\partial z} \right) + k \left(\frac{\partial N_i}{\partial r} \frac{\partial N_j}{\partial r} + \frac{\partial N_i}{\partial z} \frac{\partial N_j}{\partial z} \right) r dr dz \end{aligned} \quad (\text{D.1.25})$$

$$L_i = \int_{\Omega} N_i \eta \dot{\gamma}^2 r dr dz + \int_{\Gamma} k N_i \left(\frac{\partial T}{\partial r} \bar{n}_r + \frac{\partial T}{\partial z} \bar{n}_z \right) r d\Gamma \quad (\text{D.1.26})$$

D.1.4 Upwinding weight factor

The streamlined upwind weight factor is calculated from equation D.1.27. N_i is the normal shape function. Nassehi (2002) recommends choosing an upwinding parameter (α) so that the scheme borders on instability. The characteristic element lengths (h_r and h_z) can be calculated from equation D.1.28

$$N_i^* = N_i + \alpha \frac{|h_r u + h_z v|}{2|\bar{v}|^2} \left(u \frac{\partial N_i}{\partial r} + v \frac{\partial N_i}{\partial z} \right) \quad (\text{D.1.27})$$

$$\begin{aligned} h_r &= 2 \frac{\partial r}{\partial \xi} + 2 \frac{\partial r}{\partial \eta} \\ h_z &= 2 \frac{\partial z}{\partial \xi} + 2 \frac{\partial z}{\partial \eta} \end{aligned} \quad (\text{D.1.28})$$

D.2 Mass flow average

Mass flow averaging is a way of averaging a field value across a section of the domain. An example would be the temperature in a pipe. More weight is given to the areas of the flow with a higher normal velocity. The mass flow average formula, for a field variable f , can be seen in equation D.2.1.

$$\text{mass flow average} = \frac{\int f \rho (\bar{v} \cdot \bar{n}) dA}{\int \rho (\bar{v} \cdot \bar{n}) dA} \quad (\text{D.2.1})$$

D.3 Convergence criteria

For the momentum scheme and the θ time stepping scheme an iterative process is used. The convergence criteria for a vector of variables f can be seen in equation D.3.1. For the momentum scheme the vector of velocities (both u and v components) is used.

$$\text{tolerance} > \sum_{i=1}^n (f_i^{\text{new}} - f_i^{\text{old}})^2 \quad (\text{D.3.1})$$

D.4 Error formula

At various places in this thesis the percentage error of a computed value was calculated. The formula used to do this was equation D.4.1. The reference could either be an analytical solution, or an experimental measurement.

$$\%error = 100 \frac{\text{value} - \text{reference}}{\text{reference}} \quad (\text{D.4.1})$$

D.5 Node sorting algorithm

Sorting the nodes into a certain order can significantly reduce the bandwidth and thus computation time of the FEM solution (discussed further in appendix E). Equation D.5.1 shows how the bandwidth of one element is calculated. The bandwidth for the whole system is the maximum bandwidth of all the

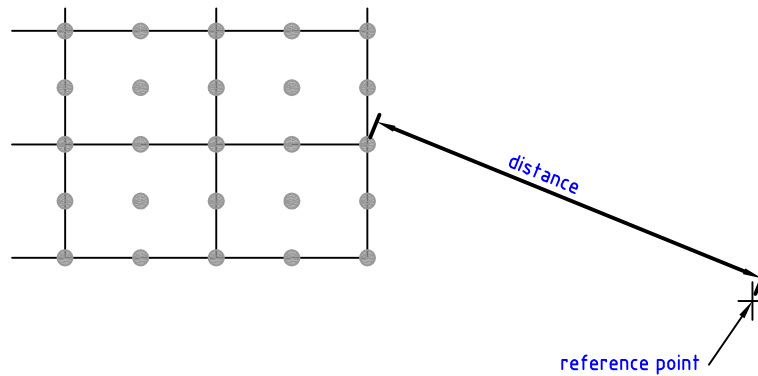


Figure D.1: Node sorting algorithm

elements. In other words the worst scenario would be to have the first node lie in the same element as the last node.

$$band\ width_{element} = \max(node_{number}) - \min(node_{number}) \quad (D.5.1)$$

The algorithm that was used in Sopfil is described as follows:

- Choose a reference point.
- Calculate the distance of each node to the reference point
- Sort the nodes using distance as the sorting criteria.

It was found that the smallest bandwidth was calculated when the reference point was chosen to lie far away, either in the x direction or the y direction. If the domain is longer in the x direction then it is better to offset the reference point in the x direction, and visa versa.

The sorting algorithm used was a heap sort algorithm. A bubble sort is simpler to program, but becomes computationally expensive when more than a thousand nodes are used. Details of the heap sort algorithm can be found in Press *et al.* (1999).

D.6 Mesh expansion factors

Mesh expansion was used in various grid generation algorithms (lomolding, injection moulding). Mesh expansion allows the user to place smaller elements by the wall than in the centre of the disk.

The formulas that were used can be seen in equations D.6.1 and D.6.2. They were taken from Thiart (2000).

$$x_i = x_{i-1} + \chi^{i-2}(\delta x)_{1,2} \quad (\text{D.6.1})$$

$$(\delta x)_{1,2} = x_2 - x_1 = \frac{1 - \chi}{1 - \chi^{N-1}} L \quad (\text{D.6.2})$$

D.7 Element search scheme

In the FEM it is easy to get the global coordinates of a point if the local coordinates and element number are known (see equation B.1.1). Sometimes, however, the local coordinates and element number are desired for a global point. An example of such a case is if the results of a specific global point in the mesh are required.

To find the local coordinates a search scheme is used. The error function is defined in equation D.7.1. The target point is given by x_{global} , y_{global} and x_k, y_k is the current point (k refers to the iteration). This equation is differentiable, which is important for the search scheme.

$$error = (x_{global} - x_k)^2 + (y_{global} - y_k)^2 \quad (\text{D.7.1})$$

The next is to determine which element to search. This is done by cycling through the elements, and only searching in those that have a node / nodes that are north, south, east and west of the global point. More than one potential element could satisfy this condition. All potential elements are searched, the correct element should return local coordinates that satisfy equation D.7.2.

$$-1 \leq \xi, \eta \leq 1 \quad (\text{D.7.2})$$

The best search scheme was found to be Newton's method, which is a second order search scheme (see equation D.7.3). For more details refer to Papalambros & Wilde (2000). Newton's method converges in a couple of iterations.

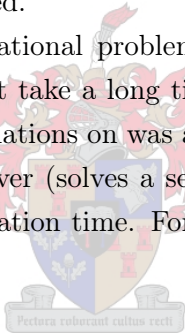
$$\begin{bmatrix} \xi \\ \eta \end{bmatrix}_{k+1} = \begin{bmatrix} \xi \\ \eta \end{bmatrix}_k - \begin{bmatrix} \frac{\partial^2 error}{\partial \xi^2} & \frac{\partial^2 error}{\partial \xi \partial \eta} \\ \frac{\partial^2 error}{\partial \xi \partial \eta} & \frac{\partial^2 error}{\partial \eta^2} \end{bmatrix}_k \begin{bmatrix} \frac{\partial error}{\partial \xi} \\ \frac{\partial error}{\partial \eta} \end{bmatrix}_k \quad (\text{D.7.3})$$

Appendix E

Matrix solution scheme

This appendix will briefly describe the matrix solver that was used to solve the FEM equations. Existing matrix libraries for C++ and the structure of the matrix will also be discussed.

The nature of the computational problem (large matrixes, several time steps) leads to simulations that take a long time (in the order of days). The computer used to run the simulations on was a 2.4 GHz Intel Pentium 4 with 512 Mb RAM. The matrix solver (solves a set of linear equations) takes the largest percentage of the simulation time. For this reason finding an efficient solver is important.



E.1 The nature of FEM matrix

A banded matrix will be formed if the nodes numbers are chosen wisely: if nodes of a similar number lie close to each other. The nodes are ordered in such manner by a sorting algorithm. An example of a sorting algorithm can be seen in appendix D.5.

Figure E.1 shows the profile of the matrix for the momentum solution of a lomolding domain (6.5). The matrix has a symmetrical structure, but the values are not symmetrical. Note that although the matrix has a size of 16890, its bandwidth is only 449. This has considerable advantages from a storage and computation time point of view.

To illustrate the saving made in computation time consider equations E.1.1 and E.1.2 (Golub and Loan, 1989). These equations give the *flops* for LU

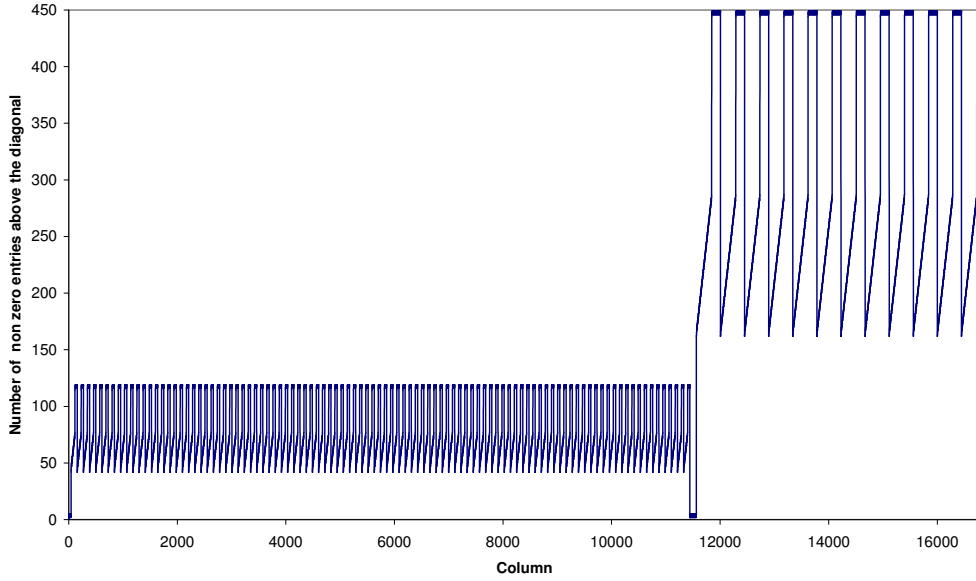


Figure E.1: Profile of a matrix for a lomolding domain

decomposition. The LU decomposition for the banded matrix in figure E.1 is 472 times faster than that of the full matrix.

It is possible to exploit the structure of the matrix further. Consider the matrix structure in figure E.1: there are many zero entries even in the banded section. These values do not have to be eliminated as they are already zero. A solution scheme that takes this into account is a skyline solver (Bathe and Wilson, 1976).

$$flops_{full\ matrix} \approx \frac{2}{3}n^3 \quad (\text{E.1.1})$$

$$flops_{banded\ matrix} \approx 2nm^2 \quad (\text{E.1.2})$$

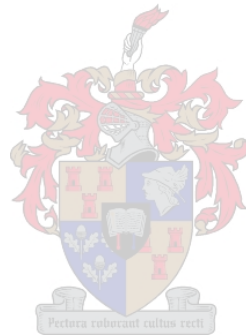
E.2 Existing matrix libraries

The solution of linear sets of equations (matrix solvers) is a current field of research. To decrease the computation time many tricks can be performed, for example the computer architecture exploited (Demmel, 1997). For this reason it is silly to program one's own solver. In the author's case a LU decomposition routine took 4 times longer to run than a matrix library's routine that was

downloaded from the internet.

Dongarra (2003) gives a summary of the freely available matrix libraries on the internet. Most of these libraries are written to run on Unix and Linux systems, in either Fortran or C.

For this thesis the Newmat matrix library was used (Davies, 2004). Although it is not the most efficient matrix library and did not have all the features (skyline solver), it did compile in C++ on a Microsoft Windows operating system and supported banded matrixes.



Appendix F

Physical properties of polymers

This appendix will present the models used to model generalised Newtonian viscosity. A method for calibrating one model to another will also be discussed. The actual model constants as well as the thermal properties of the polymer used for the experiments are also presented.

F.1 Viscosity models

F.1.1 Power-law model

The power-law model is one of the simplest viscosity models for a generalised Newtonian fluid. Model parameters are m and n . There is no direct temperature influence in the model, although m can be modified to introduce a crude temperature dependence.

$$\eta = m|\dot{\gamma}|^{n-1} \quad (\text{F.1.1})$$

F.1.2 Carreau model

The Carreau viscosity model is given by equation F.1.2 (Simcon, 2002*b*). The model is also referred to as the Bird-Carreau-Yasuda model (Osswald and Menges, 1995). The viscosity's dependence on temperature is introduced into the model by the WLF relationship (see equation F.1.3). Note that temperature is in $^{\circ}\text{C}$. The model constants are P_1 , P_2 , P_3 , T_B and T_0 .

$$\eta = \frac{P_1 a_T}{(1 + \dot{\gamma} P_2 a_T)^{P_3}} \quad (\text{F.1.2})$$

$$\ln(a_T) = \frac{8.86(T_B - T_S)}{101.6^\circ C + T_B - T_S} - \frac{8.86(T - T_S)}{101.6^\circ C + T - T_S} \quad (\text{F.1.3})$$

F.1.3 Cross model

The Cross viscosity model is given by equation F.1.4 (Bhate, 1996). Temperature and pressure dependence are introduced by η_0 term which can be calculated from F.1.5. The model constants are n , τ^* , D_1 , D_2 , D_3 , A_1 and $A_2 \sim$.

$$\eta = \frac{\eta_0}{1 + \left(\frac{\eta_0 \dot{\gamma}}{\tau^*}\right)^{1-n}} \quad (\text{F.1.4})$$

$$\begin{aligned} T \geq T \sim & \quad \eta_0 = D_1 e^{\frac{-A_1(T-T \sim)}{A_2+(T-T \sim)}} \\ T < T \sim & \quad \eta_0 = \infty \end{aligned} \quad (\text{F.1.5})$$

$$T \sim = D_2 + D_3 p \quad (\text{F.1.6})$$

$$A_2 = A_2 \sim + D_3 p \quad (\text{F.1.7})$$

F.1.4 Curve fitting viscosity models

Injection moulding simulation programs do not all use the same viscosity models. Therefore it is sometimes necessary to change between models. This was done by a curve fitting one model to another. The same can be done to fit a model to experimental viscosity data.

The Cross model constants were supplied as the viscosity data for Moplen EP301K, and can be seen in table F.1. Cadmould uses the Carreau model. Microsoft Excel's solver was used to fit the Carreau model to the Cross model. The error (equation F.1.8) was minimised over a range of shear rates and temperatures by varying the model constants. The graphical result can be seen in figure F.1 and the calculated constants in table F.2.

$$\sum [\eta_{cross}(\dot{\gamma}, T) - \eta_{carreau}(\dot{\gamma}, T)]^2 \quad (\text{F.1.8})$$

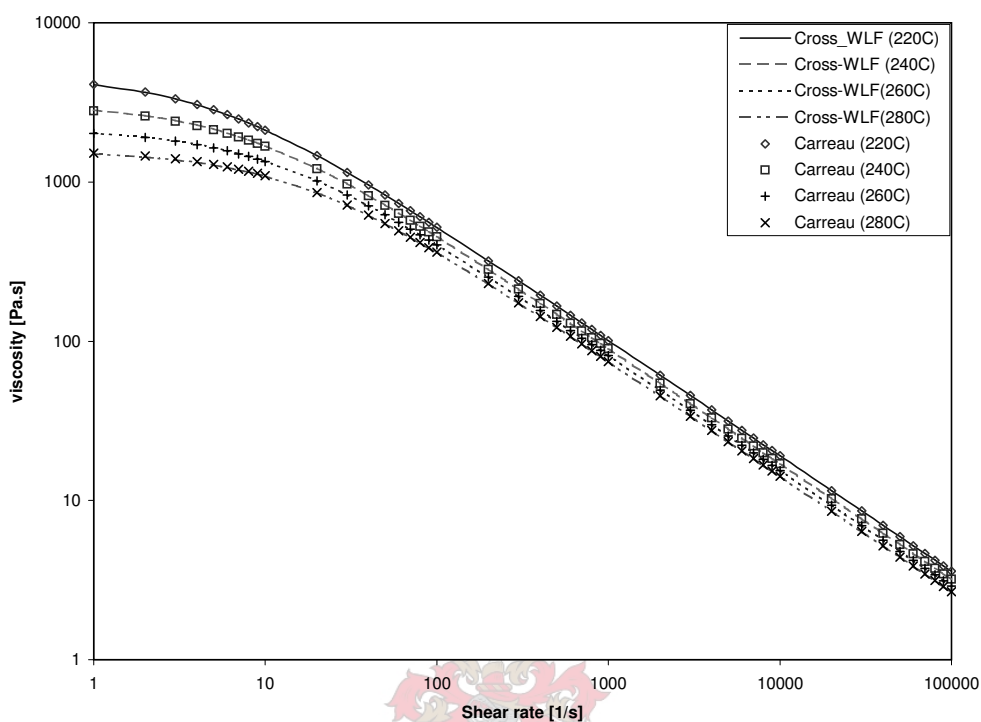


Figure F.1: Viscosity model curve fit

n	0.275
τ^*	23709.6
D_1	4.2407e15
D_2	263.15
D_3	0
A_1	33.717
$A_2 \sim$	51.6

Table F.1: Cross model constants for Moplen EP301K

P_1	4670.304
P_2	0.196908
P_3	0.725212
T_B	220
T_S	125.3876

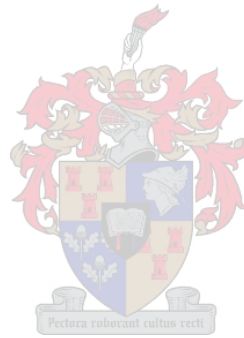
Table F.2: Carreau model constants for Moplen EP301K

F.2 Thermal properties of polymer

The thermal properties of the polymer can be seen in table F.3.

k	0.16 W/(m·K)
c	1960 J·kg/K
ρ	735.023 kg·m ⁻³

Table F.3: Thermal properties of Moplen EP301K



Appendix G

Calibration of pressure sensors

This appendix will show how the Kistler pressure sensors were calibrated. The sensors were calibrated so that there could be confidence in their readings because there were large discrepancies between the simulation pressure results and the experimental pressure measurements.

The Kistler sensors are made specifically for measuring the pressures inside the cavities of injection moulding. The sensors can measure pressure over a large range of temperatures. The sensors cannot be used to measure gas or fluid pressure (as mentioned in the operating manual). The reason for this is not given. This was verified by measuring fluid pressure with and without a membrane over the sensor; the results can be seen in figure G.7.

There are two possible explanations for the sensors not being able to measure fluid pressure. Figure G.1 and G.2 show one of the sensors in a fluid and melt environment. The first explanation could be that fluid enters the gap in-between the mould and the sensor and builds up a force on the back face of the sensor. This reduces the total force that the sensor experiences and thus the measured pressure. Polymer melt will not enter the gap, because of its high viscosity. The second possible explanation is that the melt forms a skin (a layer of solidified polymer, see figure G.2) as it comes into contact with the cold mould. The skin also prevents polymer melt from getting into the gap and onto the back face.

Two methods were tried for calibrating the sensors, namely a weight calibrator (direct force) and a hydraulic calibrator (fluid pressure, with a membrane covering the sensor). These two methods will be discussed in more detail in the following sections.

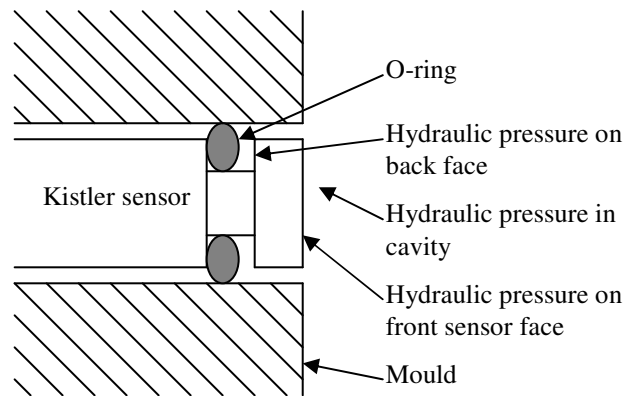


Figure G.1: Hydraulic fluid pressure on Kistler sensor

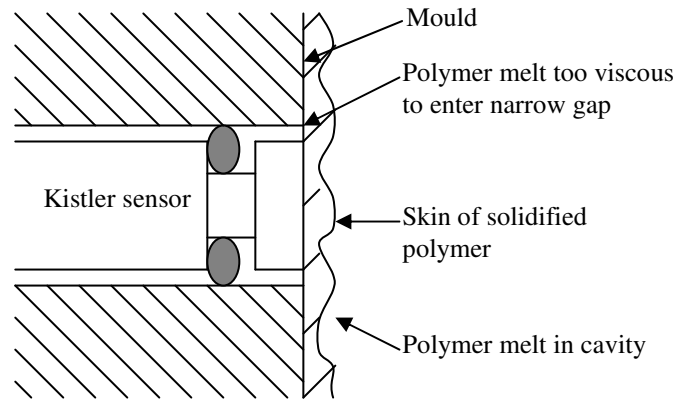


Figure G.2: Melt pressure on Kistler sensor

G.1 Weight calibrator

The idea behind the weight calibrator is to apply a force to the sensor rather than a pressure (figures G.3 and G.4). The force can be converted into an equivalent pressure (equation G.1.1). The force was applied by hanging weights on the sensor. The sensor was clamped in the sensor housing, which was attached to the mounting arm. The mounting arm is clamped onto a flat surface. The weight bracket is the interface between the sensor and the weight cradle (figure G.3). The sensor readings were recorded on a PC by means of an Eagle card (high speed analog capture card).

$$pressure_{mass}[Pa] = \frac{force[N]}{sensor\ area[m^2]} = \frac{mass[kg] \cdot g[m \cdot s^{-2}]}{sensor\ area[m^2]} \quad (G.1.1)$$

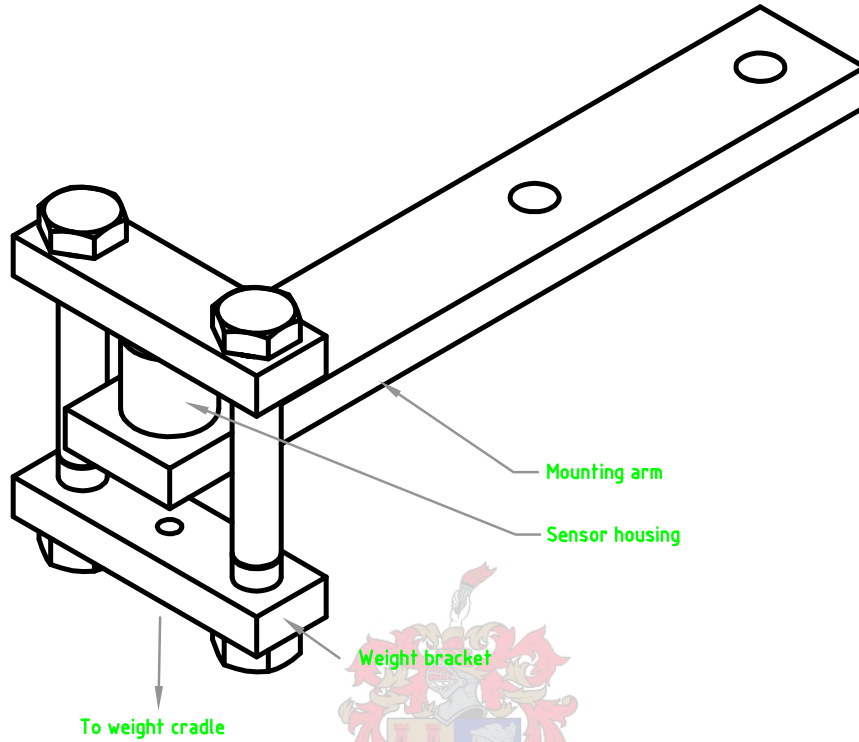


Figure G.3: Weight calibrator

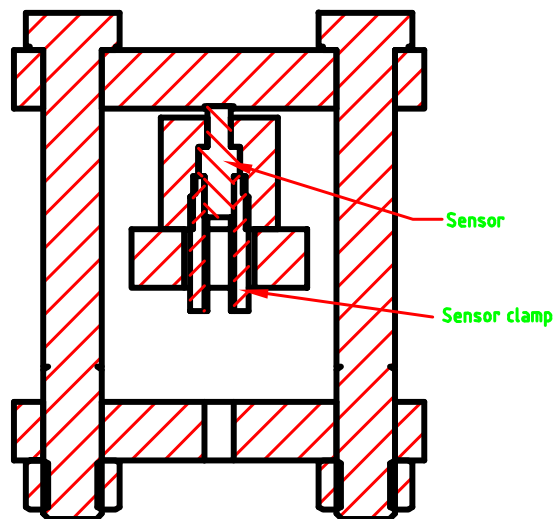


Figure G.4: Weight calibrator - cross section

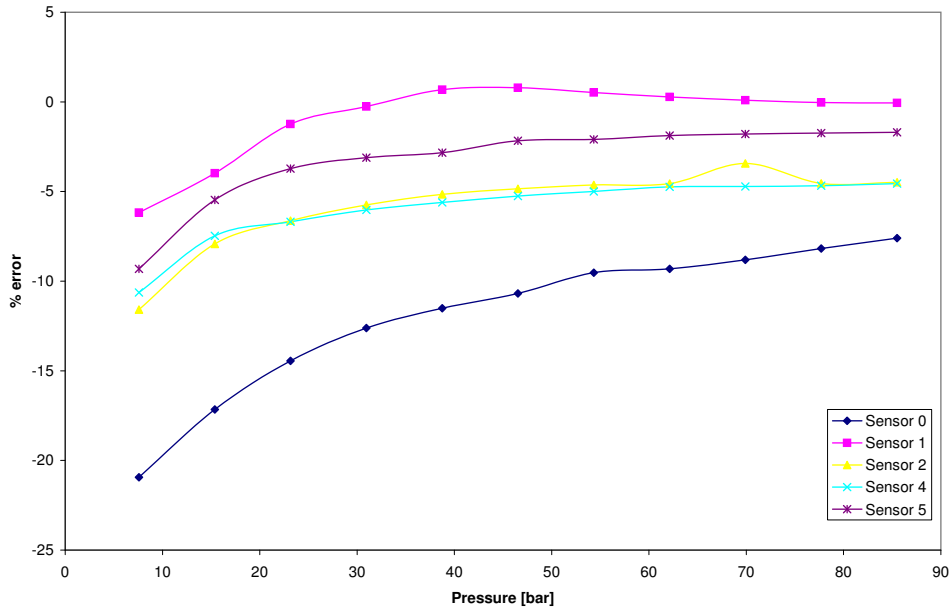


Figure G.5: Weight calibrator - %error

$$drift[bar \cdot minute^{-1}] = \frac{\Delta pressure[bar]}{time[minute]} \quad (G.1.2)$$

Two sets of readings were taken for each sensor. In the first reading a series of 1 kg (up to 10 kg) weights were packed on the weight cradle. This provided a series of readings across the desired pressure range. The error in the readings were determined using equation D.4.1. In the second set of readings the weights were packed on in 5 kg steps (up to 15 kg), and were left on for about 60 seconds, to determine the drift in the sensor readings. The drift was calculated using equation G.1.2. The results can be seen in figure G.5 and table G.1.

	$drift [bar \cdot minute^{-1}]$
sensor 0	0.3872
sensor 1	0.0094
sensor 2	-0.0159
sensor 4	-0.0172
sensor 5	-0.0441

Table G.1: Weight calibrator - sensor drift

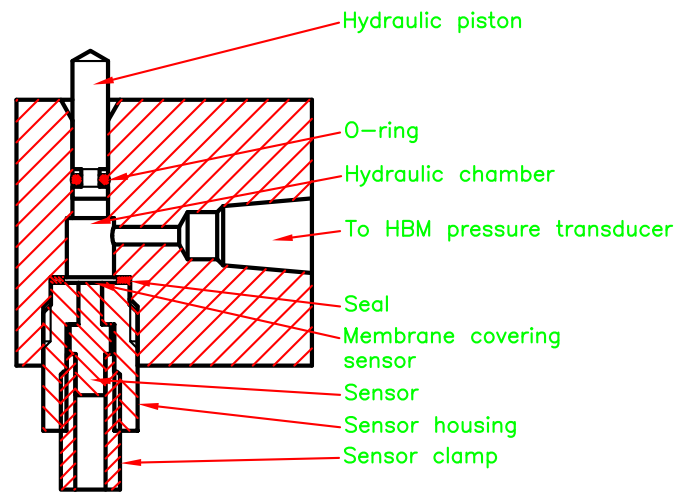


Figure G.6: Hydraulic calibrator

G.2 Hydraulic calibrator

The hydraulic calibrator attempts to create a situation that is physically similar to what happens in the mould. The sensor face is covered with a thin membrane (creating a similar situation to figure G.2). A hydrostatic pressure is built up on the other side of the membrane. The hydraulic calibrator can be seen in figure G.6. Membranes of different thicknesses and materials were tried (see table G.2).

Pressure was built up in the hydraulic chamber by hanging weights on the hydraulic calibrator's piston, in a similar method to the weight calibrator. If the force on the piston is known, then it should be possible to determine the pressure inside the chamber. A problem arises with the o-ring: there is considerable friction between the o-ring and the wall. Theoretically equation G.2.1 could be used, but no way could be found to calculate the o-ring frictional force. Figure G.7 shows the effects of the o-ring: pressure is built up by packing three series of weights on the piston. The weights are then taken off in the same order. The measured pressure does not return to the same values in the unloading stage compared to the loading stage. The figure also shows the effect of the membrane: without a membrane (of film) the pressure is lower than with a membrane, which backs up the claims made in the beginning of the appendix.

To overcome the problem of the o-ring friction a HBM pressure transducer

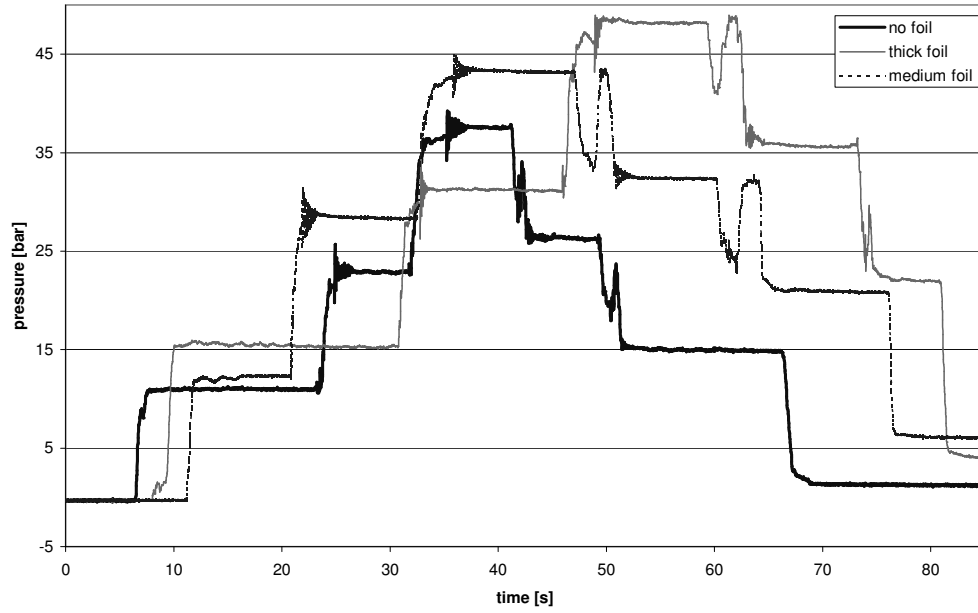


Figure G.7: Hydraulic calibrator - effect of o-ring and membrane

membrane	membrane thickness [mm]
thin foil	0.015
medium foil	0.02
thick foil	0.06
sticky plastic	0.04
plastic bag	0.015

Table G.2: Membrane thicknesses

was used to measure the pressure in the hydraulic chamber. Although it might not be good practice to calibrate one sensor with another, the HBM transducer is used exactly as it was designed to be used, and therefore should give accurate answers.

$$pressure_{hydraulic\ chamber}[Pa] = \frac{force_{mass}[N] - force_{o-ring\ friction}[N]}{sensor\ area[m^2]} \quad (G.2.1)$$

Table G.3 gives selected results for sensor 2. The second column is the pressure calculated from equation G.2.1, neglecting the friction force. The % error was calculated using the HBM transducer reading as the reference.

membrane	pressure-mass	HBM transducer	Kistler sensor	% error
no membrane	20.75	19.00	13.57	-28.57
	38.09	34.74	27.69	-20.29
	55.44	50.74	42.06	-17.11
plastic bag	20.75	21.60	16.50	-23.61
	38.09	35.50	28.75	-19.00
	55.44	54.70	48.45	-11.42
sticky plastic	20.75	19.34	16.24	-16.03
	38.09	35.70	31.52	-11.71
	55.44	53.44	48.19	-9.82

Table G.3: Hydraulic calibrator - results for sensor 2

G.3 Discussion of results

The results from the weight calibrator show that the sensors have, on average, an error of about 5%. The results from the hydraulic calibrator are harder to interpret. The errors are larger than those of the weight calibrator. Some of the difficulties encountered with the hydraulic calibrator are the membrane and the friction in the o-ring.

The problem of the o-ring friction was largely solved by using the HBM sensor to record the pressure in the hydraulic chamber, which gives a good reference pressure.

The effect of the membrane can be seen on the measurements. If a membrane tears, the errors in the pressure readings are similar to those where no membrane was used.

The hydraulic calibrator with a membrane does not accurately model what happens inside the mould, because the errors are larger than those obtained from the weight calibrator, which should be very similar.

The best indication of the accuracy of the pressure sensors comes from the weight calibrator. The sensors give readings with large errors at low pressures (up to 20% below at 20 bar).

Appendix H

Melt temperature experiments

An experiment was conducted to verify the temperature of the melt in the lomolding machine, because melt temperature has a significant effect on cavity pressures (see appendix J).

H.1 Experimental setup

Thermocouples were inserted into the hot runner system in three places. The position of the thermocouples can be seen in figure H.1.

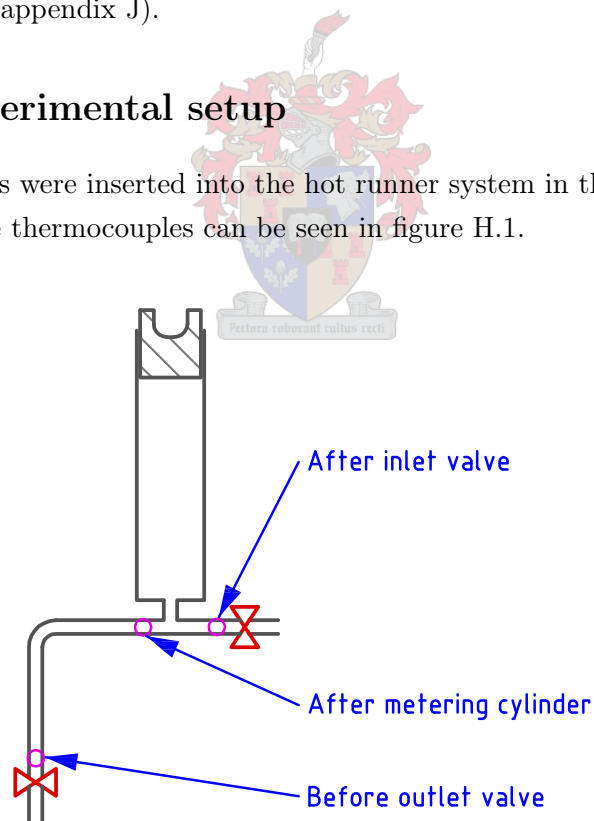


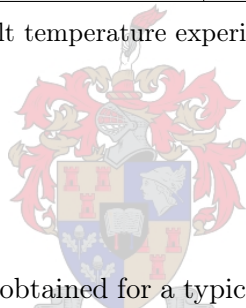
Figure H.1: Position of thermocouples for measuring melt temperature

The thermocouples have a diameter of 1.6 mm. The small diameter means they have a quick response time. The tips of the thermocouples were not placed at the centre of the pipes. They were placed so that the tips of the thermocouples protrude about 2 to 3 mm from the wall. This was done because it was feared that the thermocouples might break off due to the high drag forces of the viscous flow.

The parameters and settings of the machine can be seen in table H.1.

Parameter	Value
Melt temperature set point	240 °C
Shot size	600 cm ³
Time for cooling and demoulding	60 s
Material	Sasol Novolen 1100N
Screw speed	50%
Back pressure	30 bar

Table H.1: Melt temperature experiment parameters



H.2 Results

Figure H.2 shows the results obtained for a typical shot. Figure H.3 shows the temperature profiles obtained when purging the machine. When the machine is purged, the outlet and inlet valves are opened, and melt from the extruder flows straight through the machine.

Figure H.4 shows how the data can be interpreted. When the temperature curve of the polymer shows rapid change, polymer is flowing past the thermocouple. When the curve rises in a gradual exponential manner, there is no flow.

By looking at the results for purging of the machine, it appears that the screw extruder cannot deliver melt at the set point: all the readings tend to a value of 20°C below the set point.

Once the melt is stationary in the lomolding machine its temperature increases to the set point. The first bit of flow past the thermocouples is the melt that has been lying in the lomolding machine, and this melt is close to the set point because it has warmed up in the lomolding machine. But the remainder of the melt is below the set point.

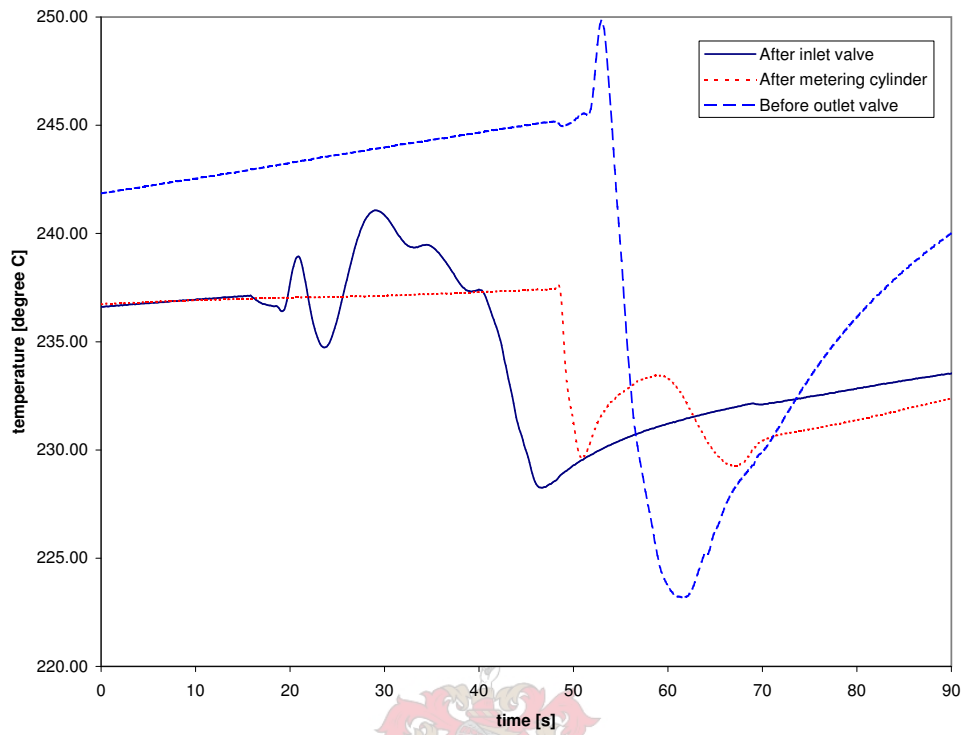


Figure H.2: Melt temperatures for a typical shot

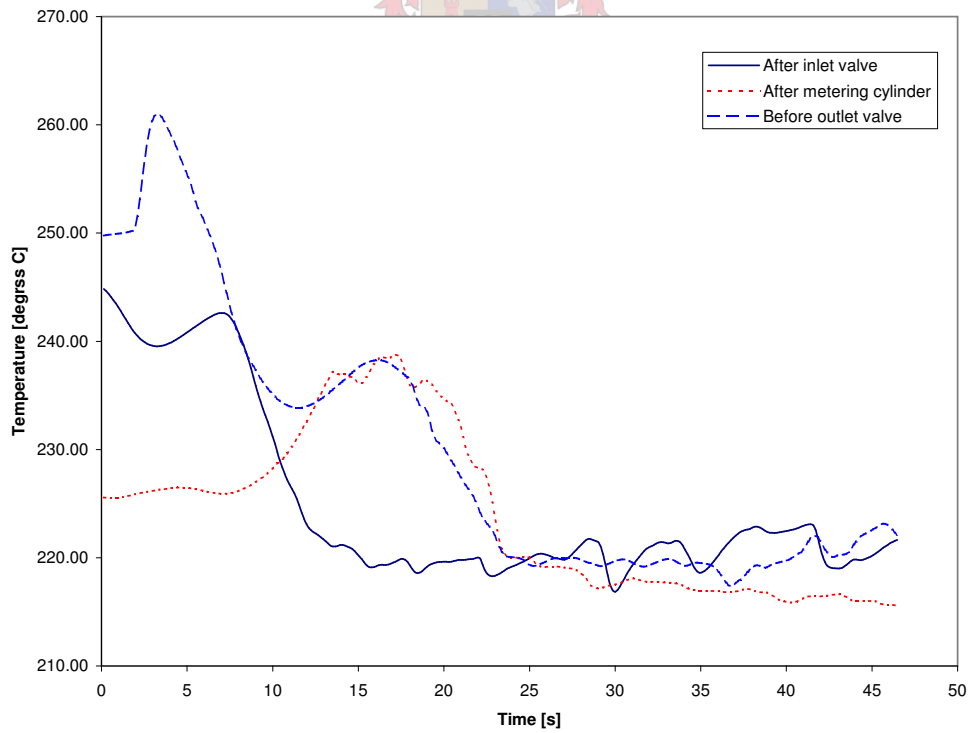


Figure H.3: Melt temperatures when purging the machine

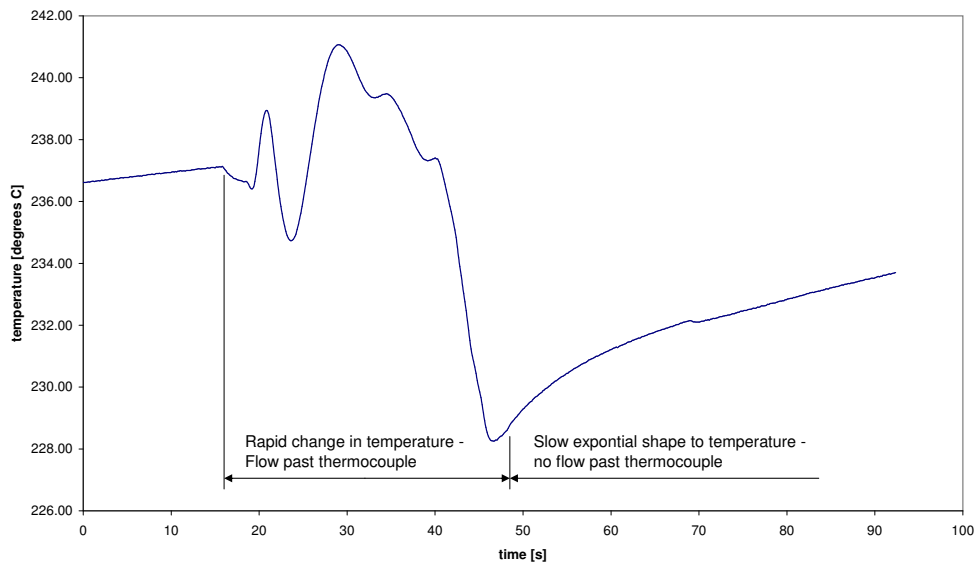


Figure H.4: Interpreting the melt temperature data

The temperature at the outlet valve seems to be set to high. What is interesting is that as melt flows past, the temperature drops below the temperature of the other two thermocouples. This could be attributed to the mixing that occurs upstream in the bend and as the melt flows past the valve rod. The melt in the centre of the hot runner will be colder than the melt next to the hot runner walls. The hot runner will heat the melt next to the walls. When the mixing occurs past the bend and valve rod, colder melt could flow over thermocouple at the outlet valve. The valve rod lies in the middle of the vertical hot runner.

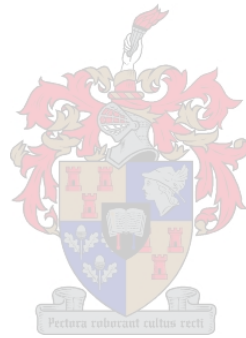
H.3 Conclusion

The melt temperature is not constant in a given shot. It varies from about 20°C below and 10°C above the set point.

To obtain a better melt temperature, and one that is more uniform, a bigger screw extruder should be used. Another possibility would be to set the screw extruder's temperature higher than the set point, or to adjust the rate at which melt leaves the screw extruder.

Smaller shot sizes will mean that the melt spends longer in the lomolding machine. Therefore it can be expected that melt temperature for smaller shots

will be closer to the set point.



Appendix I

Mould filling pattern

A series of short shots were run to determine the filling pattern of the cavity. The same shot size was used every time, and the piston's stop position was adjusted to set the percentage of mould filling. This was done, instead of using different shot sizes, to keep any effect that the barrel had on the melt the same. The settings for the experiment can be seen in table I.1.

The filling pattern was drawn on AutoCAD by placing the short shots on a printed radial grid. Points were measured at 15° intervals, and recorded in AutoCAD. A spline was drawn through these points.

Figure I.1 shows the filling pattern of the mould. The solid lines are the measured flow boundaries. The dotted lines are the theoretical axisymmetric equivalent flow boundaries. These boundaries are found by taking the volume of the experimental shot and calculating a disk with the same volume.

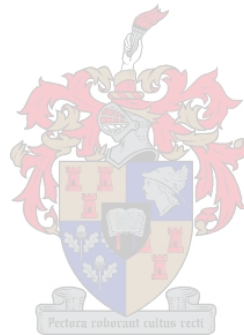
Figure I.1 shows that the melt flows to the top of the mould first (the pressure sensors lie directly down). This is especially evident in the early stages of mould filling. One of the reasons for this could be that the melt forms a skin at the bottom of the barrel before the piston injects it into the mould (see figure I.2). The barrel is heated, so this should not occur. But at the front end of the barrel, where it meets the mould, the barrel is cold. A skin (solidified layer of polymer) forms on the cold part of the barrel. The skin creates a temporary blockage at the bottom of the barrel when the melt starts entering the mould, and therefore the melt flows more in the upwards direction.

The skin could also form if the melt flows out of the barrel (while the shot is being transferred from the metering cylinder), and comes into contact with

Parameter	Value
Polymer	Sasol Novolen 1100N
Melt temperature	240 °C
Disk size (cavity geometry)	$\phi 500 \times 3 \text{ mm}$
Piston injection speed	$70 \text{ mm}\cdot\text{s}^{-1}$

Table I.1: Mould filling experimental setting

the mould wall. The melt will lie at the bottom of the barrel, and thus the skin will also block the flow downwards.



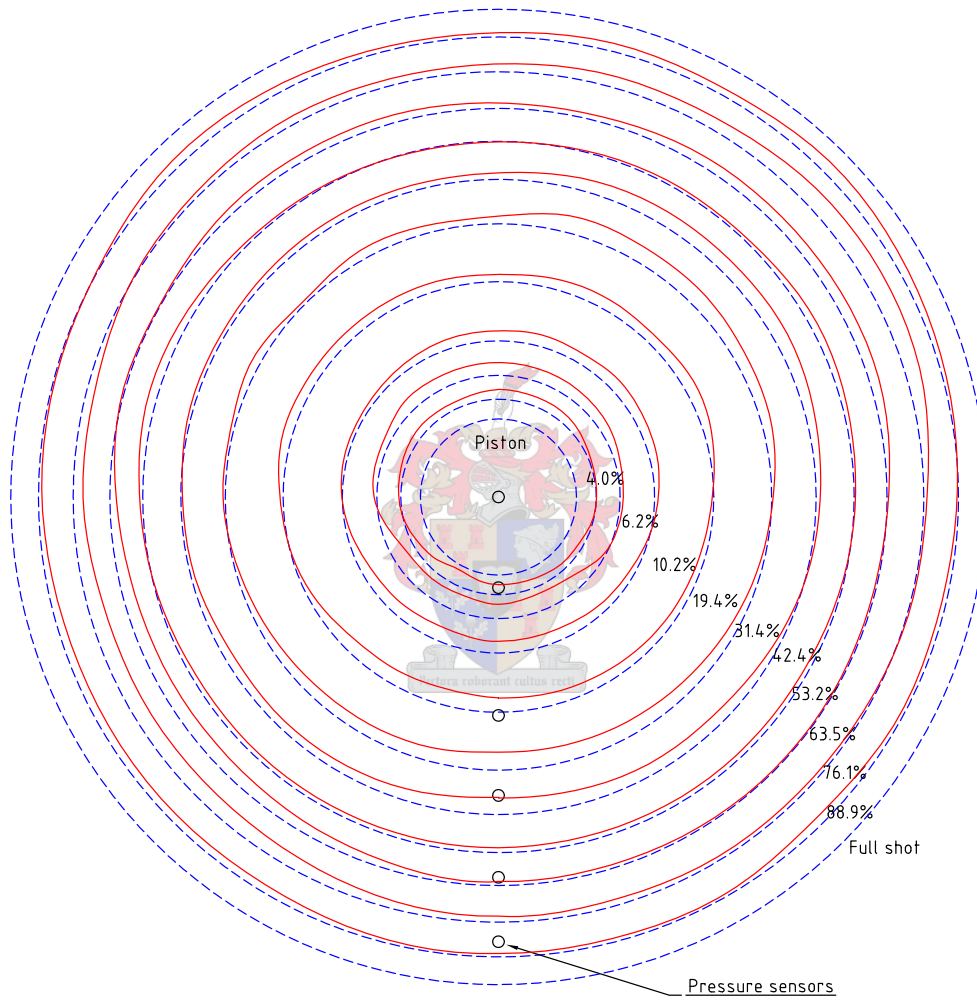


Figure I.1: Mould filling pattern

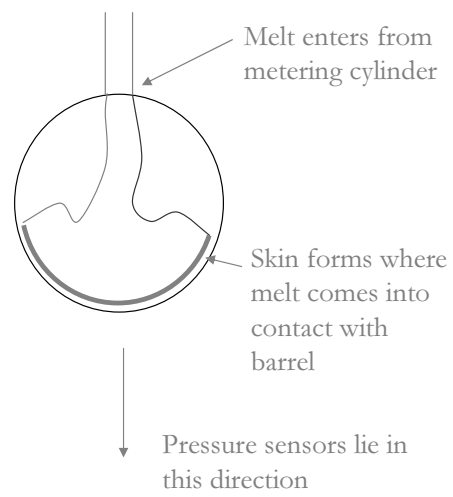


Figure I.2: Possible reason for non-axisymmetric filling

Appendix J

Sensitivity study of simulation parameters

A sensitivity study was done to determine the effect of the simulation parameters on the cavity pressures. Lomolding was simulated with Cadmould (as described in section 3.7). The study was performed by keeping all the parameters the same except one. The effect of varying this parameter could be quantified by looking at the changes in pressure of a node (corresponding to pressure sensor 1's position, see table 5.1). The base setting for the simulation can be seen in table J.1. The parameters that were varied were the wall temperature, the freezing temperature of the polymer, the polymer injection temperature and the volume flow rate. The results were plotted and can be seen in the following figures.

From the figures the following can be concluded:

- The freezing temperature of the polymer and the volume flow rate do not greatly affect the cavity pressures.

Parameter	Value
Polymer	Basell Moplen EP301K
Melt temperature	240 °C
Wall temperature	45 °C
Volume flow rate	485.805 cm ³ ·s ⁻¹

Table J.1: Base settings for sensitivity study

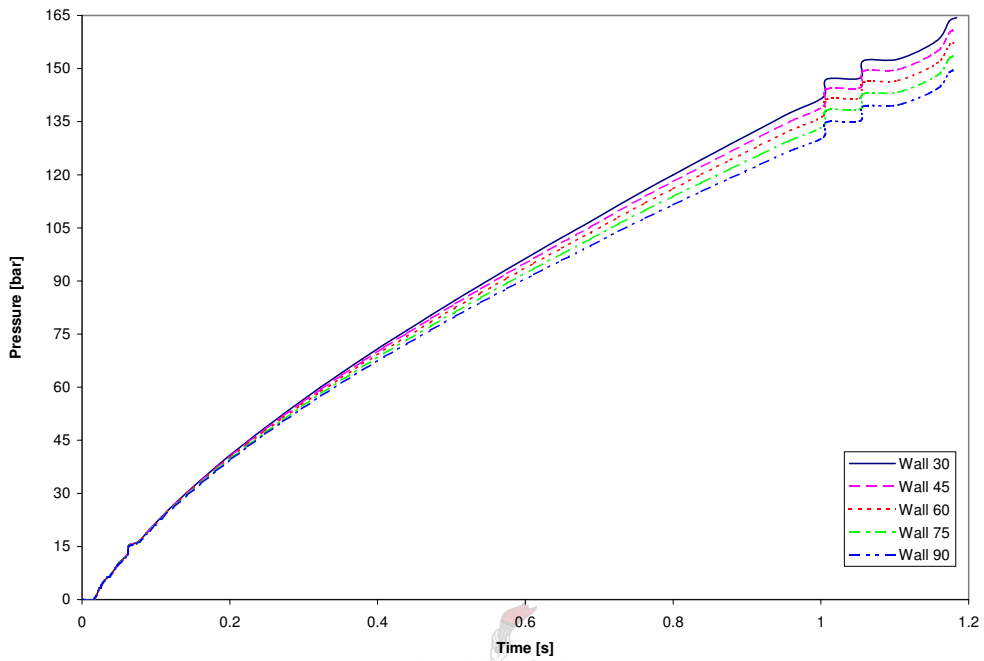


Figure J.1: Effect of mould wall temperature on pressure

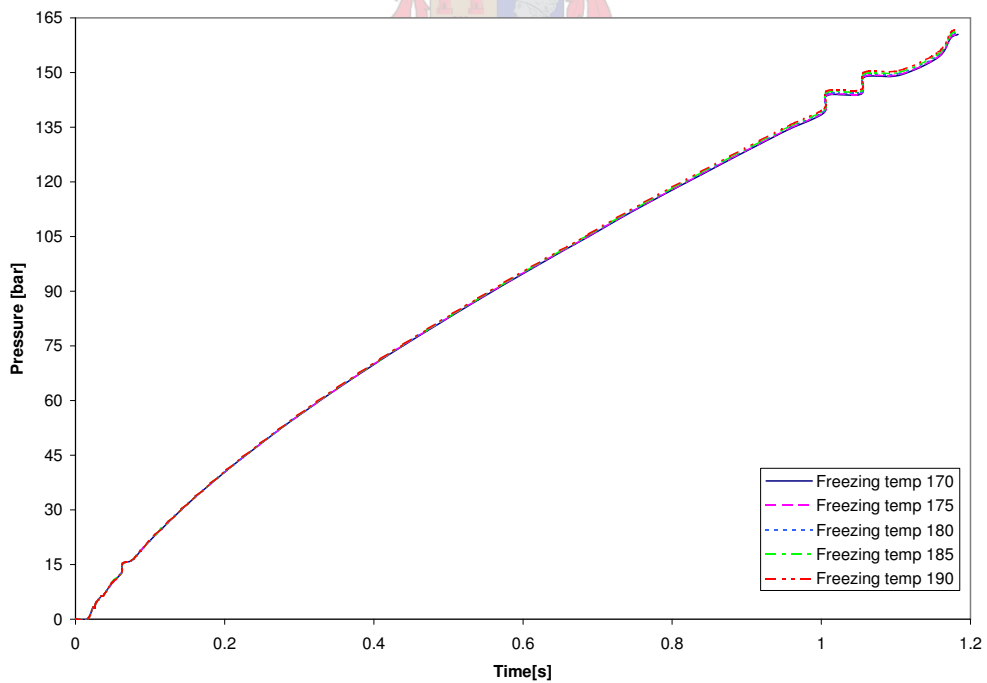


Figure J.2: Effect of melt freezing temperature on pressure

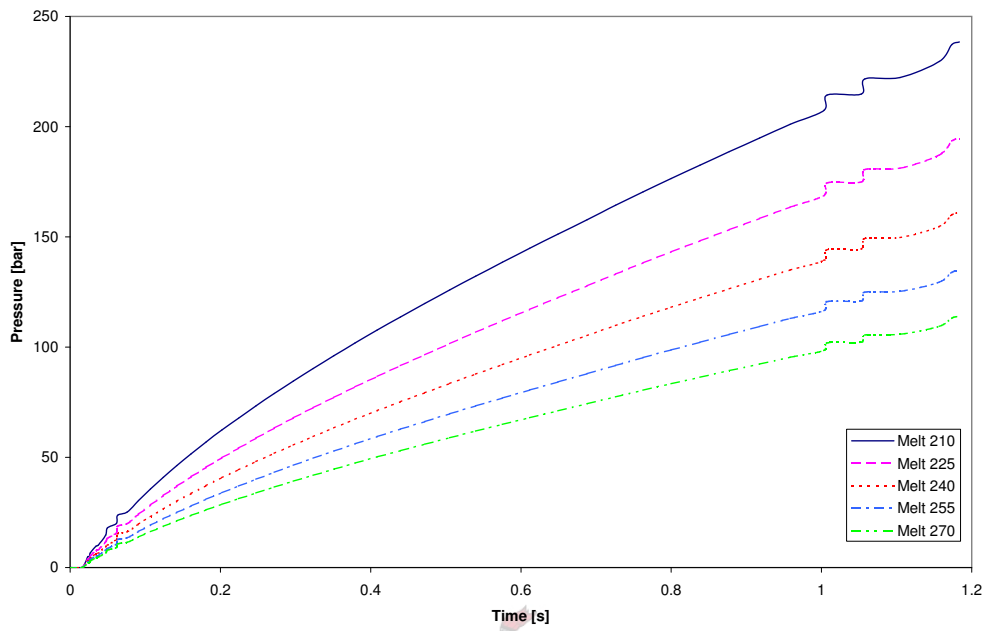


Figure J.3: Effect of melt injection temperature on pressure

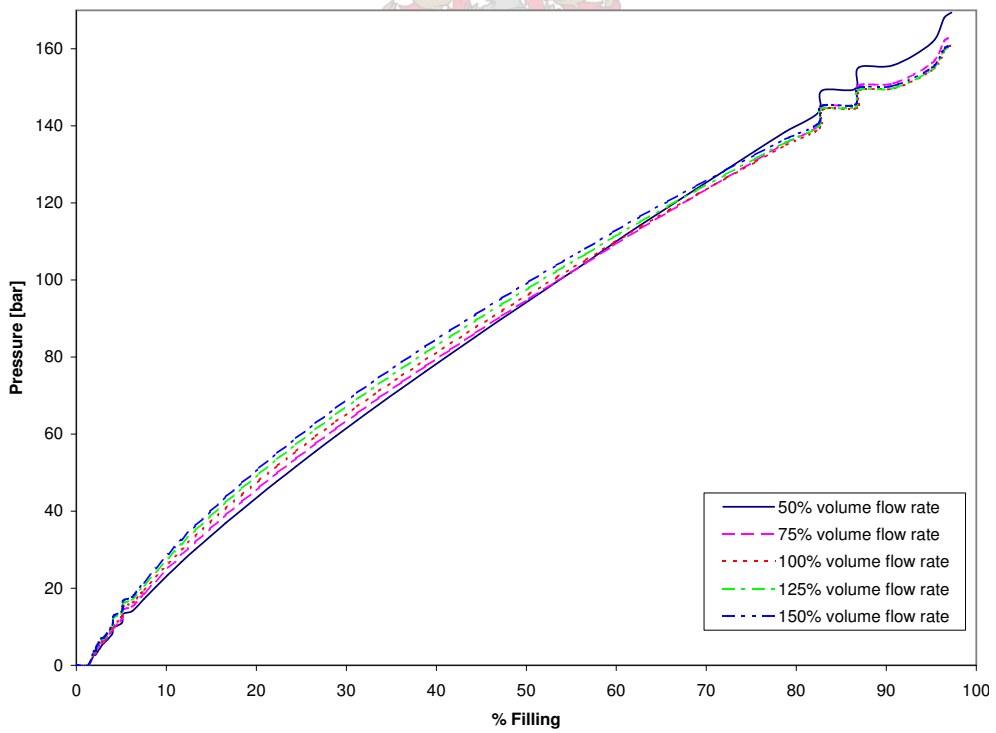


Figure J.4: Effect of volume flow rate on pressure

- The wall temperature has a slight effect. A difference in wall temperature of 60°C results in a 15 bar pressure difference.
- By far the biggest influence on the cavity pressures is the temperature at which the melt is injected.

The volume flow rate figure shows the optimal volume rate. At about 125% volume flow rate the lowest cavity pressures are recorded. This is due to the balance between two factors. The first factor is the increase of the pressure drop as the volume flow rate increases. The second factor is the thicker solid layer that develops at a lower volume flow rate, which also results in a higher pressure drop.

