

The Engineering and Physical Sciences Research Council

The University of Swansea

Cardiff University

Tata Strip Products UK

---

*Thermal Behaviour of Work Rolls  
in the Hot Mill Rolling Process*

---

Thesis submitted to Cardiff University on March 2012

For the degree of Engineering Doctorate

By Ben Wright



## Summary

This project continues the work of Daniel White which culminated in the thesis, "A Hot Strip Mill Work Roll Temperature Model", submitted to Cardiff University in 2007. The current project presents a combined thermal solution, incorporating a transient, two-dimensional model which predicts changes in the circumferentially averaged temperature profile with time, and a static, three-dimensional model which provides high-resolution results near the surface of the roll and uses the results of the two-dimensional model as boundary conditions. The material properties of both models can now exhibit temperature dependency and boundary conditions at the surface have been improved. The resolution of the static model has been increased dramatically after testing found that the temperature profile was being compromised.

The stress models use ABAQUS, a finite element software package, and have been completely redesigned. The stress solution consists of three models, each successive model concentrating more resolution on the roll bite, using the sub-modelling technique. Temperature data is imported from the thermal models, mechanical loads are applied using process data taken from the hot mill database at Port Talbot and residual loads are applied using the best currently available experimental data. Residual and thermal stresses were found to be significant, but the mechanical loads made little difference to the peak or minimum stresses per roll revolution.

A brief investigation showed that, for a high-reduction rolling schedule (40% reduction rates), increasing the work roll oxide layer thickness from  $3\mu\text{m}$  to  $10\mu\text{m}$  decreases the cyclic stress range by roughly 100MPa. Taking account of temperature variability in the material properties also made roughly 100MPa difference to the stress range. The radial temperature distribution in the roll was shown to have a strong effect on the stresses at the roll surface, with a cold roll core incurring large peak stresses for the same circumferential temperature profile.

## Declarations

### DECLARATION

This work has not previously been accepted in substance for any degree and is not concurrently submitted for candidature for any degree.

Signed.....(candidate)      Date .....

### STATEMENT 1

This thesis is being submitted in partial fulfilment of the requirements for the degree of EngD.

Signed.....(candidate)      Date .....

### STATEMENT 2

This thesis is the result of my own independent work/investigation, except where otherwise stated. Other sources are acknowledged by explicit references.

Signed.....(candidate)      Date .....

### STATEMENT 3

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed.....(candidate)      Date .....

## Acknowledgements

This project was made possible by the funding provided by the Engineering and Physical Sciences Research Council, Tata Steel UK and Swansea University, while office space and much of the necessary resources were provide by Cardiff University.

Sincere and grateful thanks must go to my academic supervisor, Prof H.P.Evans, for the many hours of advising, mentoring and proof-reading which have gone into this document and all its associated work.

Thanks also to my industrial supervisors, Ron Wigfield and Mark Veryard, for the advice, plant tours and many arranged meetings.

The RD&T department at IJmuiden have provided a lot of support over the course of the project, with especial thanks due to Harry Van-Steden, Pieter Drent, Jan De-Roo and Claas Willem-Visser, all of whom have taken considerable time out of their schedules to provide insight, data and equations.

A few individuals also deserve mention; Jeff Watkins and Terry O'Callaghan from the roll shop, for allowing me to take advantage of their experience with work rolls. Chris Littlewood, for his help with all things SQL related. Shaun Mantle, of Davy Roll, for material property data which has been invaluable. Christian Onisa of Tata RD&T, Rotherham, and Allan Mason-Jones of Cardiff University, for advice on the trickier aspects of ABAQUS usage. Prof David Smith of Bristol University, for providing residual stress data and an insight into how it could be used.

Final thanks to my parents, for a lifetime of encouragement, and to Rhi, for her continuous patience and support.

# Table of Contents

---

<b>1.</b>	<b>INTRODUCTION .....</b>	<b>1</b>
1.1.	HOT STRIP ROLLING .....	1
1.2.	PROJECT OBJECTIVES .....	5
1.3.	THESIS OUTLINE .....	6
<b>2.</b>	<b>LITERATURE REVIEW .....</b>	<b>7</b>
2.1.	GENERAL LITERATURE REVIEW.....	7
2.2.	RESEARCH IN THE ENGINEERING DOCTORATE SCHEME .....	30
2.3.	RESEARCH AT TATA STEEL.....	31
2.4.	THE WHITE MODEL <sup>3</sup> .....	32
2.4.1.	<i>General Introduction .....</i>	<i>33</i>
2.4.2.	<i>Energy Balance in a Control Volume .....</i>	<i>37</i>
2.4.3.	<i>Finite Difference Solution .....</i>	<i>42</i>
2.4.4.	<i>Application of Finite Difference to the Core Model .....</i>	<i>45</i>
2.4.5.	<i>Application of Finite Difference to the Shell Model .....</i>	<i>57</i>
2.5.	SUMMARY OF THE LITERATURE .....	63
<b>3.</b>	<b>THE THERMAL MODEL .....</b>	<b>66</b>
3.1.	INTRODUCTION.....	66
3.2.	PROCESS OUTLINE.....	67
3.3.	RECOMMENDED IMPROVEMENTS.....	70
3.3.1.	<i>Solver Efficiency Improvement .....</i>	<i>70</i>
3.3.2.	<i>Expansion of the Symmetry Condition.....</i>	<i>72</i>
3.4.	REPAIRS .....	75
3.4.1.	<i>Boundary Condition Adjustments.....</i>	<i>76</i>
3.4.2.	<i>Transiency.....</i>	<i>85</i>
3.4.3.	<i>Shell Mesh Resolution.....</i>	<i>92</i>
3.5.	RECOMMENDATIONS FROM IJMUIDEN – SEPTEMBER 2010 .....	98
3.5.1.	<i>Temperature Dependent Material Properties .....</i>	<i>98</i>
3.5.2.	<i>Roll Bite Conditions.....</i>	<i>107</i>
3.5.3.	<i>Efficiency Improvements Considered but not Implemented .....</i>	<i>109</i>
3.5.4.	<i>Surface cooling map .....</i>	<i>114</i>
3.6.	SECOND SET OF RECOMMENDATIONS FROM IJMUIDEN .....	115
3.6.1.	<i>Roll bite conditions .....</i>	<i>115</i>
3.6.2.	<i>Cooling conditions .....</i>	<i>119</i>
3.6.3.	<i>Conservative thermal models .....</i>	<i>121</i>
3.7.	CONCLUSIONS .....	154

<b>4.</b>	<b>THE STRESS MODEL.....</b>	<b>156</b>
4.1.	INTRODUCTION .....	156
4.1.1.	<i>Finite Element Modelling</i> .....	156
4.1.2.	<i>The ABAQUS Package</i> .....	157
4.1.3.	<i>ABAQUS User Interface</i> .....	160
4.1.4.	<i>General Modelling Strategy</i> .....	162
4.2.	MODEL CONSTRUCTION .....	169
4.2.1.	<i>Global Model</i> .....	169
4.2.2.	<i>Surface Model</i> .....	179
4.2.3.	<i>Bite Model</i> .....	182
4.2.4.	<i>Verification Model</i> .....	184
4.3.	BOUNDARY CONDITIONS.....	188
4.3.1.	<i>Thermal Loading</i> .....	188
4.3.2.	<i>Mechanical Loading</i> .....	198
4.3.3.	<i>Residual Loading</i> .....	204
4.4.	CONCLUSIONS.....	212
<b>5.</b>	<b>VALIDATION TESTING RESULTS .....</b>	<b>213</b>
5.1.	INTRODUCTION .....	213
5.2.	THERMAL MODEL INVESTIGATIONS.....	213
5.2.1.	<i>Testing circumferential conduction</i> .....	213
5.2.2.	<i>Roll bite behaviour</i> .....	214
5.3.	STRESS MODEL VALIDATION .....	216
5.3.1.	<i>Sub-Modelling proof</i> .....	216
5.3.2.	<i>2D validation model</i> .....	232
5.3.3.	<i>Validating the 3D model</i> .....	237
5.3.4.	<i>Locating sub-model boundaries</i> .....	243
5.3.5.	<i>Mechanical loads</i> .....	246
5.3.6.	<i>Residual stresses</i> .....	251
5.4.	CONCLUSIONS.....	256
<b>6.</b>	<b>RESULTS .....</b>	<b>258</b>
6.1.	INTRODUCTION .....	258
6.2.	THE SCENARIOS.....	259
6.2.1.	<i>Early</i> .....	260
6.2.2.	<i>Normal</i> .....	261
6.2.3.	<i>Mill Stop</i> .....	262
6.2.4.	<i>Long Gap</i> .....	263
6.3.	TEMPERATURE RESULTS .....	264
6.3.1.	<i>Core model temperature profiles</i> .....	264

6.3.2.	<i>Radial graphs</i> .....	265
6.3.3.	<i>Circumferential Graphs</i> .....	268
6.4.	STRESS RESULTS.....	270
6.4.1.	<i>Comparison to literature</i> .....	275
6.4.2.	<i>Yielding study</i> .....	277
6.5.	CONCLUSIONS .....	278
<b>7.</b>	<b>CONCLUSIONS &amp; RECOMMENDATIONS</b> .....	<b>280</b>
7.1.	CONCLUSIONS .....	280
7.2.	RECOMMENDATIONS.....	281
<b>APPENDIX A.</b>	<b>SOFTWARE USERS MANUAL</b> .....	<b>283</b>
A.1.	PREREQUISITE SOFTWARE .....	283
A.2.	INPUT FILES.....	283
A.3.	THE MAIN MENU.....	284
A.4.	SQL DATA AND PROCESSING .....	287
A.5.	THERMAL MODELS .....	292
A.6.	STRESS MODELS.....	294
<b>APPENDIX B.</b>	<b>C# CODE</b> .....	<b>301</b>
B.1.	AVAILABLE CLASSES .....	301
B.2.	THERMAL MODELS .....	303
B.2.1.	<i>Core-Roll interaction</i> .....	304
B.2.2.	<i>Shell-Roll interaction</i> .....	305
B.2.3.	<i>Stand-Roll interaction</i> .....	306
B.2.4.	<i>Interface-Stand interaction</i> .....	307
B.3.	STRESS MODELS.....	308
<b>APPENDIX C.</b>	<b>ABAQUS CODE</b> .....	<b>313</b>
C.1.	PYTHON SCRIPT .....	313
C.1.1.	<i>Global model</i> .....	313
C.1.1.	<i>Surface Model</i> .....	343
C.1.2.	<i>Bite Model</i> .....	349
C.1.3.	<i>Residual Model</i> .....	352
C.2.	UTEMP .....	353
C.2.1.	<i>General Case</i> .....	353
C.2.2.	<i>2D verification</i> .....	361
C.3.	UTRAC .....	366
C.4.	DLOAD.....	367
C.5.	SIGINI .....	369
<b>APPENDIX D.</b>	<b>DATA</b> .....	<b>374</b>



D.1.	DEBUG DATA SET .....	374
D.2.	EXTENDED DEBUG DATA SET .....	375
D.3.	HEAVY DUTY TEST DATA SET .....	376
<b>REFERENCES</b>	<b>.....</b>	<b>387</b>

## Glossary

---

Term	Description
Backup roll	A steel roll which is significantly larger (approximately twice the diameter) of an accompanying work roll. Sits on top of the upper work roll or underneath the lower work roll and provides support to prevent unwanted work roll deflection. The loading which provides pressure in the roll bite is applied through the backup rolls.
Barrel	The larger radius section of a work roll, the surface of which is made of a hard-wearing material. The surface of the barrel is the part of the work roll which makes contact with the strip.
Bearing	Refers to a traditional bearing, i.e. An assembly which allows a stationary frame to support a rotating component.
Chock	Refers to a frame designed to fit onto a work roll journal and allow the work roll to be connected to a drive motor or bearing.
Cooling header	A series of pipes and nozzles which direct cooling spray onto the surface of the work roll barrel.
Cooling spray	Water applied to the surface of the work roll barrel for the purpose of removing excess heat.
Core	A work roll is made of two materials. The core material is the material below the surface of the barrel and in the journals.
Core model	The thermal core model is the thermal model which incorporates the entire work roll, including the area covered by the thermal shell model.
Finishing mill	A collection of rolling stands which perform successively smaller and more controlled reductions in strip thickness.
Hot Mill Database	A database which stores historical information about the rolling process. This database is used as the primary source of data for the models developed in this thesis.
Hot rolling	The process of passing a strip of metal between two rolls with the aim of reducing its thickness. The strip of metal is heated to soften it, reducing the force required to achieve deformation.
Journal	The two smaller radius sections of the work roll, which fit into the bearing chocks and never make contact with the strip.
Roll bite	The area of the work roll barrel surface which is in contact with the strip.
Roughing mill	A collection of rolling stands which perform the first few large reductions in strip thickness.
Scale	Oxide layer formed on the surface of the metal strip due to its high temperature and contact with the atmosphere.
Shell	A work roll is made of two materials. The shell material is the material near to the surface of the barrel.
Shell model	The thermal shell model is the thermal model which incorporates the area immediately below the work roll barrel surface, down to a depth where the variation in temperature in the angular direction is no longer significant. This depth need not coincide with the depth of the interface between the material shell and material core.
Stand	A stand in a hot rolling mill is a set of two work rolls and, in the case of a 4-high mill such as the hot mill in Port Talbot, two backup rolls, along with peripheral equipment for controlling rolling loads, the size of the gap between the work rolls, etc.

Term	Description
Strip	The product of a steelworks, steel strip is long and thin compared to its width, as described in the introduction. The aim of hot mill rolling is to reduce the thickness of the strip, consequently increasing its length.
Work roll	A large steel roll. The thickness of the steel strip is reduced by passing it between a carefully-controlled gap between two work rolls.
Work Roll Bending	The Work Roll Bending system imposes an opposing deflection to account for the deflection of the work roll due to the moment generated by the forces applies to the journals and the reaction from the strip on the centre of the work roll.

## Nomenclature

---

Symbols	Description	Units	Reference	Page
$\alpha$	Thermal diffusivity	$\frac{K}{s}$		
$\alpha_{bc}$	Simplifying coefficient			
$\alpha_w$	Thermal diffusivity of the work roll	$\frac{K}{s}$	19	25
$\beta$	Proportion of deformation energy dissipated as heat		11	13
$\gamma$	Simplifying coefficient			
$\gamma_0$	Simplifying coefficient			
$\gamma_1$	Simplifying coefficient			
$\lambda$	Thermal conductivity	$\frac{W}{m^2}$	11	14
$\lambda$	A simplifying coefficient			
$\lambda$	Simplifying coefficient			
$\lambda_0$	Simplifying coefficient			
$\lambda_1$	Simplifying coefficient			
$\mu$	Coefficient of friction		11	13
$\mu$	Poisson's ratio			
$\nu$	A similarity variable, combining distance and time into a single value to turn a one-dimensional transient problem into an ordinary differential		27	112
$\omega$	Angular rotational velocity of the work roll	$\frac{Rad}{s}$		
$\psi$	ratio of the product of density and specific heat capacity at the boundary in the positive i-direction to the product of density and specific heat capacity for the element as a whole			
$\rho$	Density	$\frac{Kg}{m^3}$		
$\sigma_{21}$	Yield stress at 21 degrees centigrade	$Pa$		
$\sigma_T$	Tangential stress	$Pa$	14	16

<b>Symbols</b>	<b>Description</b>	<b>Units</b>	<b>Reference</b>	<b>Page</b>
$\sigma_{yield}$	Yield stress at temperature T	<i>Pa</i>		
$\bar{\sigma}$	Mean flow stress	<i>Pa</i>	12	14
$\tau_{i-}$	ratio of thermal conductivity at the boundary in the negative i-direction to the thermal conductivity of the element as a whole			
$\tau_{i+}$	ratio of thermal conductivity at the boundary in the positive i-direction to the thermal conductivity of the element as a whole			
$\tau_{j-}$	ratio of thermal conductivity at the boundary in the negative j-direction to the thermal conductivity of the element as a whole			
$\tau_{j+}$	ratio of thermal conductivity at the boundary in the positive j-direction to the thermal conductivity of the element as a whole			
$\tau_{k-}$	ratio of thermal conductivity at the boundary in the negative k-direction to the thermal conductivity of the element as a whole			
$\tau_{k+}$	ratio of thermal conductivity at the boundary in the positive k-direction to the thermal conductivity of the element as a whole			
$\Delta\tau$	Dimensionless time step		11	13
$\theta$	Angular coordinate	<i>Rad</i>		
$\theta$	A relaxation term to promote stability in iterative solution schemes			
$\theta$	Angular location	Degrees		
$\theta_b$	Angular size of the roll bite, i.e. The portion of the work roll which is in contact with the strip	<i>Rad</i>		
$\Delta\theta$	A small rotation in the angular dimension	<i>Rad</i>		
$\zeta$	Angular coordinate, transformed to rotate with the work roll	<i>Rad</i>	19	25
$A$	Area of a face dividing two elements	$m^2$		
$A$	Simplifying coefficient			
$a$	Half hertzian contact width	$m$		
$arc$	Arc length of the roll bite	<i>rad</i>	17	21
$B$	Simplifying coefficient			
$c$	Specific heat capacity	$\frac{J}{KgK}$	9	8
$C$	Simplifying coefficient			
$C_0$	Constant of integration		27	113
<b>Symbols</b>	<b>Description</b>	<b>Units</b>	<b>Reference</b>	<b>Page</b>

$C_1$	Constant of integration		27	113
$C_m$	proportion of frictional heating flowing into the work roll		9	8
$C_p$	Specific heat capacity	$\frac{J}{KgK}$		
$D$	Simplifying coefficient			
$E$	Simplifying coefficient			
$E$	Young's modulus (as used in Equations 4-12, 4-13 and 4-16)	$Pa$		
$E^*$	Simplifying coefficient			
$F$	Simplifying coefficient			
$F^*$	Simplifying coefficient			
$F_a$	Simplifying coefficient			
$G$	Simplifying coefficient			
$G^*$	Simplifying coefficient			
$G_0$	Simplifying coefficient			
$G_1$	Simplifying coefficient			
$G_B$	Simplifying coefficient			
$G_T$	Simplifying coefficient			
$h$	Heat transfer coefficient	$\frac{W}{m^2K}$		
$H$	Simplifying coefficient			
$h(\theta)$	heat transfer coefficient considered as a function of circumferential location	$\frac{W}{m^2K}$		
$H^*$	Simplifying coefficient			
$H_0$	Simplifying coefficient			
$h_1$	Strip thickness before rolling	$m$	11	13
$H_1$	Simplifying coefficient			
$h_2$	Strip thickness after rolling	$m$	11	13
$H_2$	Simplifying coefficient			
$H_{21}$	Vickers hardness at 21 degrees centigrade	$Pa$		
$H_3$	Simplifying coefficient			
$h_{conv}$	Heat transfer coefficient which controls heat convection from the work roll to the atmosphere	$\frac{W}{m^2K}$		
$h_{convB}$	Heat transfer coefficient which controls heat transfer from the work roll journal	$\frac{W}{m^2K}$		
$h_{spr}$	Heat transfer coefficient which controls heat convection from the work roll to the cooling sprays	$\frac{W}{m^2K}$		
$h_{spr[i,k]}$	indexed heat transfer coefficient map for the cooling applied to the surface of the work roll	$\frac{W}{m^2K}$		
$h_{sprAdjust[i]}$	Adjustment factor which matches the effect of cooling in the shell thermal model with the cooling experienced by the core thermal model			
<b>Symbols</b>	<b>Description</b>	<b>Units</b>	<b>Reference</b>	<b>Page</b>

$h_{str}$	Heat transfer coefficient which controls heat conduction from the strip to the work roll	$\frac{W}{m^2K}$		
$H_T$	Vickers hardness at temperature T	$Pa$		
$i$	location of a point in the axial direction, specified in finite difference mesh points			
$j$	location of a point in the radial direction, specified in finite difference mesh points			
$j_{0a}$	Simplifying coefficient			
$j_{0b}$	Simplifying coefficient			
$j_a$	Simplifying coefficient			
$j_b$	Simplifying coefficient			
$j_c$	Simplifying coefficient			
$j_{offset}$	An offset applied to the thermal shell model to account for index 0 not being at the roll centre line			
$k$	Thermal conductivity	$\frac{W}{m^2K}$		
$k$	subscript: location of a point in the circumferential direction, specified in finite difference mesh points			
$K$	Index of the current time step			
$k_+$	Thermal conductivity at a point on the boundary between two elements	$\frac{W}{m^2K}$		
$k_-$	Thermal conductivity at a point on the boundary between two elements	$\frac{W}{m^2K}$		
$L$	contact length	$m$		
$\Delta l$	The width of a slice of work roll	$m$	17	21
$m$	Mass	$Kg$		
$n_\theta$	total number of index points in the circumferential direction			
$P$	Mechanical power supplied to the work roll	$W$		
$p_m$	Rolling pressure	$Pa$	11	13
$P_o$	Maximum pressure within a hertzian contact	$Pa$		
$Q$	Heat flow between two control volumes	$J$		
$Q_p$	Heat flow due to deformation	$W$	11	13
$Q_r$	Heat flow due to strip contact	$W$	11	13
$\bar{Q}_{core_i}$	Total rate of heat flow from the core model for the axial index location i	$W$		
$\bar{Q}_{shell_i}$	Total rate of heat flow from the shell model for the axial index location i	$W$		
$\dot{Q}$	Heat flow between two control volumes per unit time	$W$		
$\dot{q}$	heat flow between control volumes	$W$	9	8
$\dot{q}$	Heat flow at the model boundary	$W$	11	11
$\dot{Q}_c$	Rate of change of internal energy in an element	$W$		
<b>Symbols</b>	<b>Description</b>	<b>Units</b>	<b>Reference</b>	<b>Page</b>

$\dot{q}_{cond}$	The rate of heat flow into the work roll from the strip, per unit area	$\frac{W}{m^2}$		
$\dot{q}_{conv}$	The rate of heat flow from the work roll to the cooling sprays, per unit area	$\frac{W}{m^2}$		
$\dot{q}_{friction(\theta)}$	circumferentially-indexed array of heat generation terms	$\frac{W}{m^2}$		
$\dot{q}_g$	The rate of heat flow into the surface per unit area which is due to heat generation effects	$\frac{W}{m^2}$		
$\dot{Q}_{net}$	Balance of heat flow into and out of an element per unit time	$W$		
$\dot{Q}_r$	Rate of heat flow across an element boundary in the radial-direction	$W$		
$\dot{Q}_{Spr}$	Rate of heat flow extracted by the cooling water	$W$		
$\dot{Q}_{Str}$	Rate of heat flow between the strip and work roll	$W$		
$\dot{Q}_\theta$	Rate of heat flow across an element boundary in the angular-direction	$W$		
$\dot{Q}_x$	Rate of heat flow across an element boundary in the x-direction	$W$		
$\dot{Q}_y$	Rate of heat flow across an element boundary in the y-direction	$W$		
$\dot{Q}_z$	Rate of heat flow across an element boundary in the z-direction	$W$		
$r$	Radial coordinate	$m$		
$\Delta r *$	Dimensionless radial mesh spacing		11	12
$\Delta r$	A small distance in the radial dimension	$m$		
$R$	Work roll radius	$m$	11	12
$R$	Effective contact radius	$m$		
$S$	Area of an element that is exposed to the surface	$m^2$	9	8
$t$	Time	$s$		
$\Delta t$	A small change in time	$s$		
$\Delta t_{rm,n}^b$	temperature rise from contact with the strip	$K$	8	7
$\Delta t_{rm,n}^c$	temperature rise from radial heat diffusion	$K$	8	7
$\Delta t_{rm,n}^f$	temperature rise from frictional heating	$K$	8	7
$T(\theta, t)$	Temperature considered as a function of both circumferential location and time	$^\circ C$		
$T_{ambient}$	Temperature of the ambient environment	$^\circ C$		
$T_B$	Temperature of the chock, which is in contact with the work roll journal	$^\circ C$		
$T_f$	Temperature of the ambient environment	$^\circ C$		
$t_{rm,n}$	temperature at position (rm,n)	$K$	8	7
$t_s$	The index of the current time step			
$T_s$	temperature of the surface of the body	$K$	27	112
$T_{Spr}$	The temperature of the water applied to the work roll by the cooling sprays	$^\circ C$		
<b>Symbols</b>	<b>Description</b>	<b>Units</b>	<b>Reference</b>	<b>Page</b>

$T_{str}$	Temperature of the strip	$^{\circ}C$		
$T_{str[i,k]}$	indexed contact temperature used to apply heating to the surface of the work roll	$^{\circ}C$		
$\Delta T$	Change of temperature in an element	$K$		
$u$	velocity of material flowing through a control volume	$\frac{m}{s}$		
$u$	dummy variable for $\eta$		27	113
$\Delta U$	the change in the internal energy of a control volume	$J$		
$V$	Volume	$m^3$		
$\bar{V}_r$	Average relative velocity between the work roll and strip	$\frac{m}{s}$	11	13
$\Delta V$	A small change in the volume of an element	$m^3$		
$w$	Represents $\frac{dT}{d\eta}$		27	113
$w$	total load between contacting bodies	$N$		
$w(x)$	An arbitrary function using x as its only dependent variable			
$x$	Axial coordinate	$m$		
$\Delta x$	A small distance in the axial dimension	$m$		
$y$	The second dimension in rectangular coordinates	$m$		
$\Delta y$	A small distance in the second dimension in rectangular coordinates	$m$		
$z$	The third dimension in rectangular coordinates	$m$		
$\Delta z$	A small distance in the third dimension in rectangular coordinates	$m$		



## 1. Introduction

### 1.1. Hot Strip Rolling

The hot rolling process is generally an intermediate stage in the process route of steel in its solid form. The aim of a general rolling process is to transform the relatively short and thick slabs of steel produced by a steelworks casting facility into a very long, thin sheet. The hot rolling process uses elevated temperature to soften the material to perform the bulk of the thickness reduction. If a very thin strip of material is required then the process must be concluded in a cold rolling mill for the extra stability that rolling at lower temperatures provides. This sheet can be coiled or chopped and stacked for easy transport and will be the correct thickness for the customer to punch / cut / press / roll / etc. into their end product.

At Port Talbot liquid metal is solidified at the ConCast (Continuous Caster) into slabs approximately 234mm<sup>1</sup> thick and of varying widths (.650-1.88m) and lengths (7-10m) depending on customer requirements. The slabs are transported to the Hot Mill where their thickness is reduced to within a range<sup>2</sup> of 1.4mm to 18mm. The finished product (now called strip) is either sent out as a finished HR (Hot Rolled) product or for further processing e.g. skin pass, pickling, cold reduction, annealing, coating, etc.

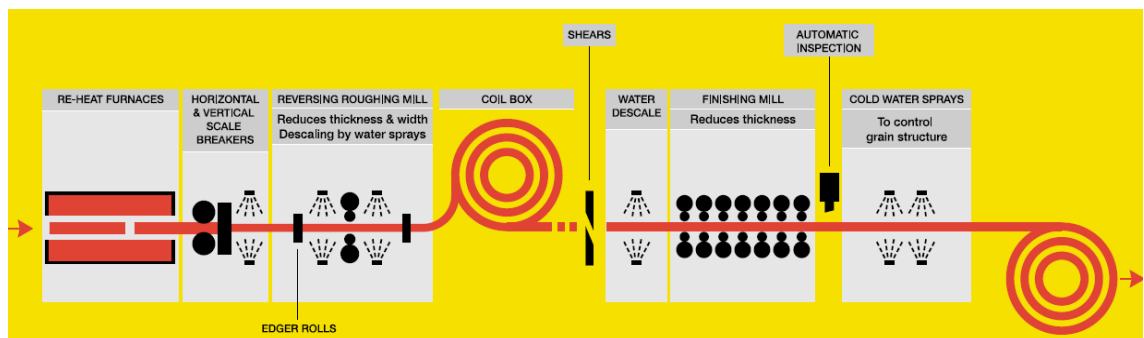


Figure 1-1: Layout of the hot mill at Port Talbot<sup>1</sup>

The layout of the hot mill is illustrated in Figure 1-1. The slabs received by the Hot Mill must first pass through the re-heat furnace. This furnace provides the heat necessary to soften the metal to reduce the force required to achieve the target thickness reduction, and to meet the product-dependent temperature requirements that will give the desired material properties. The slabs are transferred to roller-tables which transport them to the horizontal scale breakers (HSB). The scale breakers remove the surface oxide layer formed on the slab during the heating cycle. An 8-12% reduction in the slab's thickness takes place in the HSB. The

Reversing Roughing Mill then reduces the gauge (thickness) to 28-45mm. The Roughing Mill also reduces the width by up to 80mm. At this point the steel is referred to as a 'transfer bar'.

The distance between the Roughing Mill and the first stand of the Finishing Mill is shorter than the length of most transfer bars. To allow the Roughing Mill and Finishing Mill a limited amount of independence in processing speed a "coil box" coils the transfer bar from the Roughing Mill before uncoiling it into the Finishing Mill, acting as a buffer. There are several other advantages to this approach; heat conservation, even temperature distribution through the strip thickness, coils can be held for short periods and still be rolled if there is a line stoppage and the uneven head and tail of the strip can be cropped more conveniently.

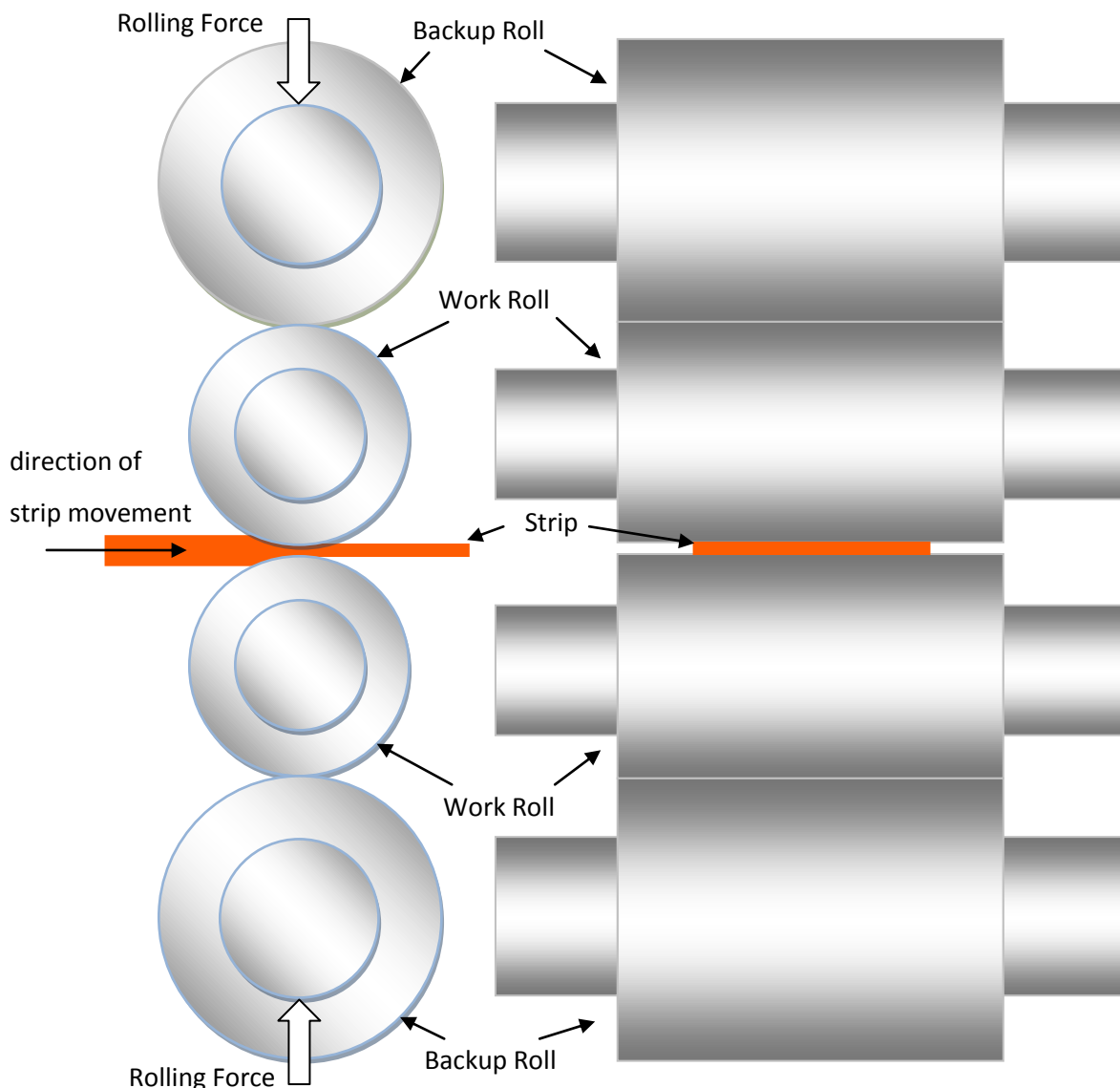


Figure 1-2: Basic roll layout in a single hot rolling stand; left – side view, right – front view.

The Finishing Mill is a sequence of seven rolling stands, each within a short distance of the last to preserve heat in the strip and keep the footprint of the rolling line to a minimum.

Each stand reduces the gauge until the desired thickness within the range of 1.4mm to 18mm is achieved. The finished strip is then coiled for transfer to its next stage of processing or for transport to the customer.

All of the stands in the Finishing Mill are of a four-high configuration, as shown in Figure 1-2, meaning that each stand holds four rolls. The middle two rolls are known as “work rolls” and make direct contact with the strip. The outer two rolls are the “backup rolls” and their role is to apply the rolling force and counter work roll deflection under the high forces that they must exert on the strip as it passes through the stand.



Figure 1-3: A work roll

All the work rolls at Port Talbot roughly conform to the shape shown in Figure 1-3 and are made of spun-cast steel<sup>3</sup>. The first step of this process is to rotate the work roll mould around its centre axis at a high rate while injecting the first batch of liquid steel. This steel forms the outer layer of the cylindrical roll and is held in that position by the centrifugal effect caused by rotation of the mould, cooling to form the working surface of the new roll. Once this layer has hardened a softer steel grade is injected to fill the remaining space. This process combined with additional surface hardening heat treatment produces a work roll with a very hard wearing surface and improved elastic behaviour below the surface.

The thermal boundary conditions of the hot mill rolling problem are complicated. Part of this is due to the number of interactions occurring around the work roll surface. Another part comes from the constant alternation between rolling and idle behaviour (each stand in the Finishing Mill takes 50-120 seconds to roll an individual strip with an idle period of 20-150 seconds between strips, depending on the processing considerations for that product) and due to the highly aggressive environment it has proven very difficult to take any kind of reliable measurements of transient behaviour. Figure 1-4 gives a simplified representation of the zones to be considered when attempting to create a thermal model of a work roll.

Heat input is primarily from the strip, the heat conducting from the strip to the work roll through the roll bite where the two are in direct physical contact. Through the roll bite there will be high contact pressures and some relative velocity between the strip and the roll, so heat generation through friction must also be considered. Where the two surfaces are not in

contact but are within line of sight there will be some heat exchange through radiation. The bearings which support the work roll will also generate a certain amount of heat which should be incorporated.

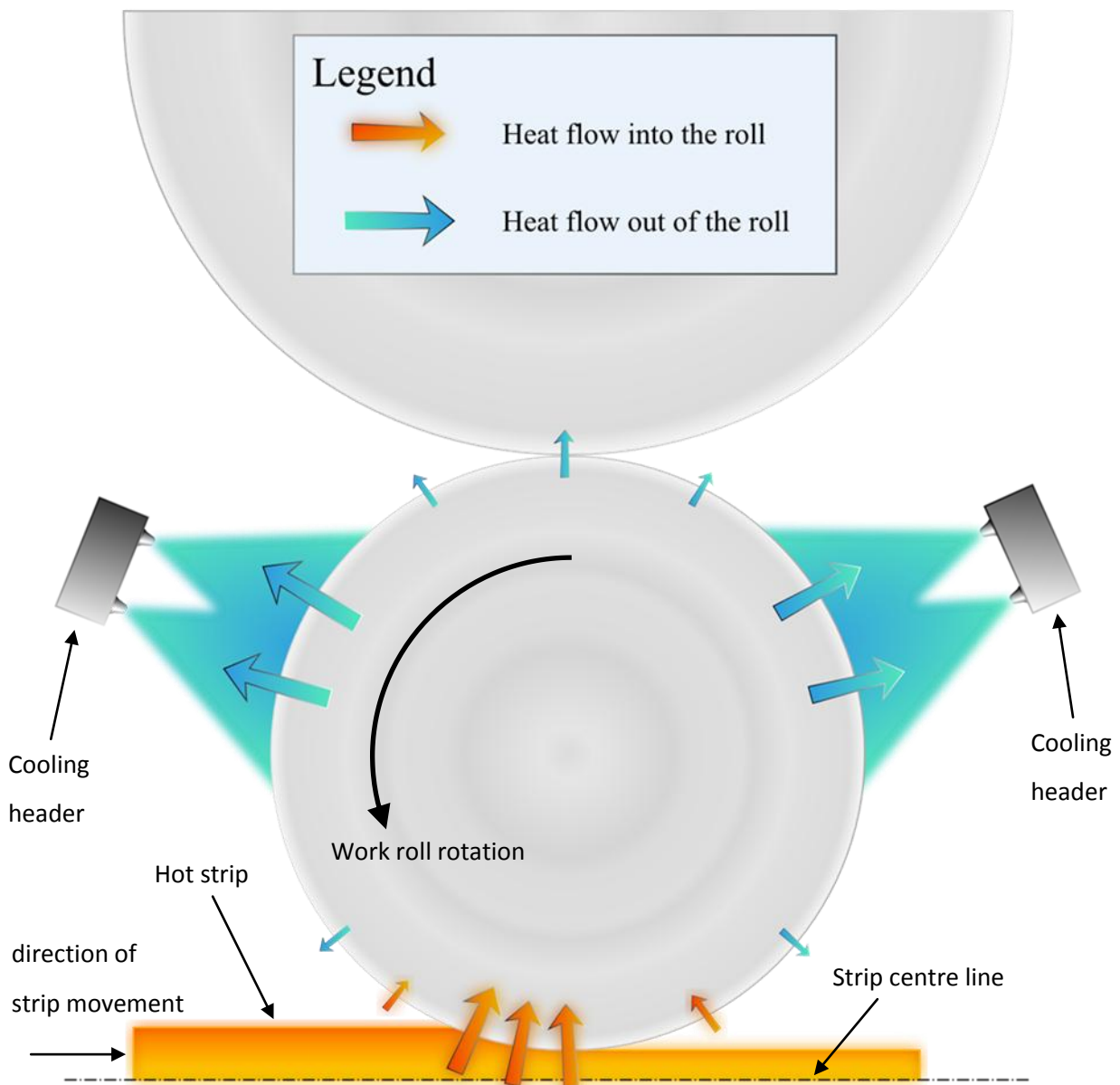


Figure 1-4: Heat flux into and out of a work roll

The bulk of the heat transferred to the work roll is almost immediately removed from the roll surface by the cooling water sprays. These sprays consist of banks of water jets which ideally provide a smoothly consistent barrage of cooling water to absorb and transport heat away from the work roll. The aim of an ideal cooling system is perfectly homogenous or specifically varied heat extraction, depending on the design<sup>4</sup>. The exact configuration of these cooling sprays along with coolant temperatures and flow-rates are important factors in dictating the heat flow from the surface of the work roll. The modelling of this aspect has

been a research project topic in its own right. Heat conduction to the backup rolls will remove further heat energy while all the remaining areas of the work roll are subject to convective heat loss to the ambient environment.

The incorporation of these boundary conditions into a work roll thermal model will be treated in detail in section 2.4.

## 1.2. Project Objectives

---

This project acts as a continuation of the work carried out by White<sup>3</sup>, an Engineering Doctorate student who worked on a project entitled “A hot strip mill work roll temperature model” from September 2003 – 2007. The original project brief was very broad in scope, encompassing work roll temperature prediction, expansion, strip profile, roll wear and lubrication. This was later focused to the prediction of work roll temperature profile, prediction of stresses and expansion and roll wear. An additional aim of the project and the reason for its introduction was to investigate stress effects at the material interface in the backup roll. However, this objective was effectively removed by the discovery of a mechanical setup error in the mill. The error was corrected and all occurrences of the backup roll failure mode which had prompted the initial curiosity halted. The project continued under the more general objectives of extending work roll life and improving strip profile.

At the culmination of the project White had produced a computer program capable of directly linking with the hot mill database, launching many kinds of simulation from the acquired data and displaying the results. Two linked thermal models gave a comprehensive three dimensional temperature field, an ABAQUS script generator launched customised stress analyses in both two and three dimensional forms and a simple wear model had been incorporated.

The current project began with similarly broad aspirations;

- To further develop the program into a tool for...
  - dynamically predicting work roll shape.
  - estimating the probability of work roll failure given its thermal history.
- If practical, convert the tool to run in-line with the hot mill, either as an element of the control system or as a diagnostic aid or a warning system for the mill operatives

Through the discoveries made throughout the research period along with a visit to the Tata Steel research department in IJmuiden, Holland, to discuss the unification of Tata Steel’s research efforts in this field, the project objectives became;

- To improve the accuracy and validation of the temperature models
- To expand the symmetry condition applied to the temperature models to enable investigation of asymmetrical effects
- To create a stress model capable of resolving the stress field
- To incorporate some of the expertise available at the technical centre in IJmuiden, particularly the heat transfer coefficient generation facility available in the TEX model
- To use the temperature and stress models to investigate the causes of work roll failure and surface degradation

### 1.3. Thesis Outline

---

The process of building on the White model to fulfil the project objectives required a relatively diverse range of tasks to be performed. As such the layout for this thesis draws the different tasks into logical subject groupings rather than purely reflecting chronological order.

A review of the literature along with work performed at Cardiff University and Tata Steel will be discussed in Chapter 2 (Page 7) along with a description of the work carried out by this project's predecessor.

Chapters 3 (Page 66) will be concerned solely with the thermal models. This will cover such planned activities as the expansion of the mill centre plane symmetry condition and adaptations to allow temperature varying material properties. Also included are unplanned improvements necessitated by discoveries during the course of the project.

Chapter 4 (Page 156) will detail the use of ABAQUS for this project, a commercial finite element software package used to generate all the stress results. The structure of the stress models will be discussed along with the boundary conditions, meshing strategy, element choice, etc.

Chapter 5 (Page 213) will present the results of the testing and validation work carried out during the development of the thermal and stress models. Much of the thermal model development is discussed in Chapter 3, so Chapter 5 mainly focuses on the stress models.

Chapter 6 (Page 258) will describe the results of a short experiment, investigating the effects of oxide layer thickness on the work roll surface, the effect of introducing temperature dependency to the work roll's material properties and the influence of process history on the temperature and stress distributions in the roll.

Finally, the conclusions and recommendations for further work can be found in chapter 7 (Page 280).

## 2. Literature Review

### 2.1. General Literature Review

The modelling of the thermal response of hot rolling mills ostensibly first began in 1954<sup>5</sup> with a study by Peck et al<sup>6</sup>. However, the complexity of the boundary conditions for this problem meant that most models were limited by the simplifications required to render the problems practically solvable<sup>7</sup>. The first solutions developed were analytical in nature, involving laborious mathematical formulations but minimising the amount of computation required.

In 1989 Tseng et al<sup>8</sup> developed an uncoupled analytical thermomechanical solution to the steady-state (quasi-static) work rolling problem in two dimensions,  $(r, \theta)$ , as illustrated on the left of Figure 2-1.

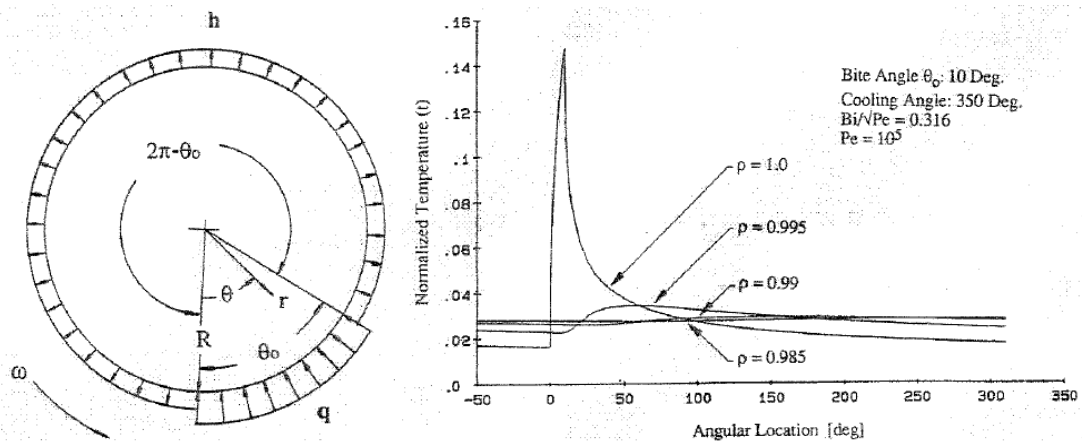


Figure 2-1: Left: Model geometry and Right: Angular temperature distributions at various radial locations ( $\rho = \frac{r}{R}$ , where  $r$  is the radial location and  $R$  is the roll radius) from Tseng et al<sup>8</sup>.

In the figure,  $\theta_0$  is the roll bite size in radians,  $R$  is the work roll radius,  $h$  is the heat transfer coefficient,  $q$  is the heat input from contact with the strip,  $\omega$  is the roll rotational velocity and  $\theta$  and  $r$  are the angular and radial coordinates respectively. The study assumed conduction in the circumferential direction was negligible, converted the governing equation to a dimensionless form and achieved a solution by converting the equation into an infinite series. The resulting temperature profile is given on the right of Figure 2-1.

Using the plane strain assumption, a formulation for each component of stress was developed using the thermoelastic displacement potential approach. The assumptions of the study stipulated a zero-traction condition at the roll surface, which the displacement potential approach fails to deliver. A general stress function was added to improve the approach by removing the surface stresses. The results indicated that, for high angular work roll velocities, thermal stresses vary rapidly within a small depth range (1.5% of the roll radius), the maximum

stress is found towards the end of the roll bite and some localised yielding might occur in the roll bite.

In 1997 Yiannopoulos et al<sup>9</sup> introduced an internal pressure distribution to the inside face of a hollow cylindrical model, in order to gauge the effect of the resulting prestress on the stress distribution experienced by the roll during rolling. The model geometry is shown in Figure 2-2, where  $T_e$  and  $h_e$  are the external temperature and heat transfer coefficient,  $T_i$  and  $h_i$  are the internal temperature and heat transfer coefficient,  $a$  and  $b$  are the internal and external radii,  $2\beta$  is the width of the roll bite,  $\phi$  is the angle of the roll bite to the horizontal axis,  $T_o$  is the work roll temperature field and  $r$  and  $\theta$  are the coordinates of any arbitrary point  $M$ .

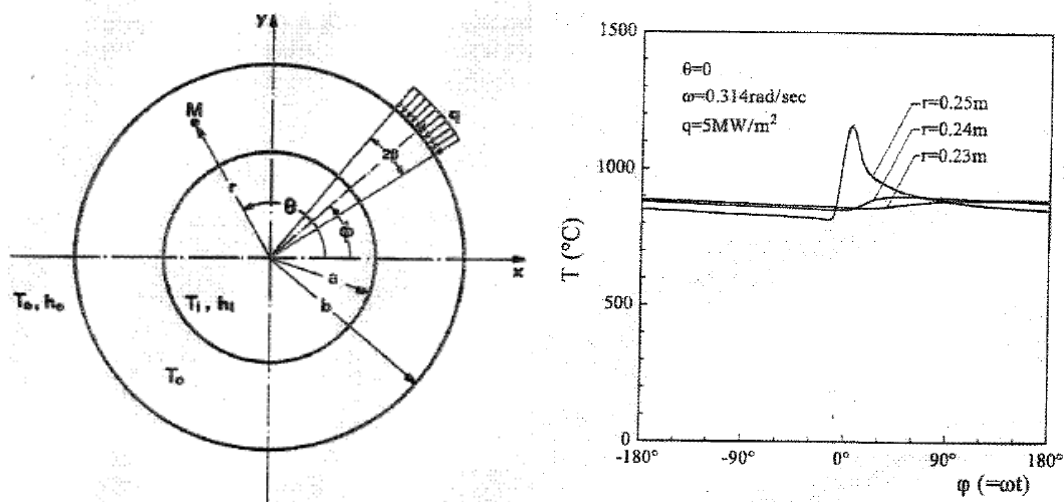


Figure 2-2: Left: Model geometry and Right: Steady state temperature distribution for a rotating work roll, as presented in Yiannopoulos et al<sup>9</sup>.

A two-dimensional time varying temperature formulation was developed and predicted the existence of a thin thermal boundary layer (2.5% of the radius). The boundary conditions specified temperature differences for cooling and a constant heat flux for heat input in the roll bite, hence the high temperatures in Figure 2-2, where heat input is not limited by reducing temperature difference between the roll and strip.

The formulation for the temperature profile was used to develop a set of transient thermomechanical stress equations using the plane-strain assumption. This was then used to calculate the principal stresses, to separate the constant and alternating parts of the stress distribution and identify the level of internal pressure that would give minimum equivalent stresses and therefore maximise fatigue life.



Even by the present day computational resources are a limiting factor in the development of more complicated numerical simulations, although the advancement of technology is allowing for increasingly accurate models to be produced. In the past 10 years a number of different research groups have created numerical models to predict temperature and stress effects in work rolls.

In 1999 Chang<sup>10</sup> produced a radial-circumferential model that used variable heat transfer in the roll bite by modelling a section of strip as well as the work roll. The model was two dimensional and semi-infinite, and used a finite difference solution in the circumferential direction and an analytical solution in the radial direction to save processing time. Only the finite difference formulation is given in this article, for sliding areas of the roll bite;

$$t_{rm,n} = t_{rm-1,n} + C_m \Delta t_{rm,n}^f + \Delta t_{rm,n}^c + \Delta t_{rm,n}^b$$

where  $t_{rm,n}$  is the temperature at position  $(rm,n)$ ,  $C_m$  is a coefficient which determines the amount of heat which goes into the work roll,  $\Delta t^f$  is the temperature rise associated with frictional heating,  $\Delta t^c$  is the temperature rise associated with heat diffusing into the depth of the roll and  $\Delta t^b$  is the influence of the contact patch on the surface temperature. For sticking areas of the roll bite;

$$t_{rm,n} = t_{rm-1,n} + C_m \Delta t_{rm,n}^s + \Delta t_{rm,n}^c + \Delta t_{rm,n}^b$$

where  $\Delta t^s$  is an arbitrarily chosen heating term to replace the frictional heat generation term. The temperature of the work roll and strip at the entrance to the roll bite are considered “known” and all other temperatures can be found by sweeping through the roll bite with the equations given above, for both work roll and strip.

This work found that within a thermal boundary layer thermal stresses were dominant, whereas outside the thermal boundary layer mechanical stresses predominated.

In 2000 a very compact model was developed by Campos et al<sup>11</sup> to calculate the thermal crown of a work roll. This model considers heat flow in the radial and axial directions while any variation in the circumferential direction is ignored by using a single average temperature. The basic conduction equation considered is stated as;

$$\frac{\partial^2 T}{\partial x^2} + \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial r^2} = \frac{1}{\alpha} \frac{\partial T}{\partial t}$$

Where  $T$  is the temperature of the material,  $x$  is the axial location,  $r$  is the radial location,  $t$  is time and  $\alpha$  is the thermal diffusivity. This is used to calculate the rate of heat transfer  $\dot{q}$  between control volumes for isolated moments in time to give;

$$\dot{q}_{x,r \rightarrow x+1,r} = -k_{x,r} \frac{A_{x,r}}{x_{x+1,r} - x_{x,r}} (T_{x+1,r} - T_{x,r})$$

$$\dot{q}_{x,r \rightarrow x,r+1} = -k_{x,r} \frac{2\pi w}{\ln\left(\frac{r_{x,r+1}}{r_{x,r}}\right)} (T_{x,r+1} - T_{x,r})$$

Where  $k$  is the thermal conductivity,  $A$  is the area between two control volumes and  $w$  is the axial width of the element. All the conduction components can be gathered into a single term;

$$\dot{q}_{x,r} = \dot{q}_{x-1,r \rightarrow x,r} - \dot{q}_{x,r \rightarrow x+1,r} + \dot{q}_{x,r-1 \rightarrow x,r} - \dot{q}_{x,r \rightarrow x,r+1}$$

Which can be used to calculate the change in the temperature of an element given;

$$\Delta T_{x,r} = \frac{1}{m_{x,r} c_{x,r}} \dot{q}_{x,r} t_s$$

Where  $t$  is the time-step length,  $m$  is the mass of material in the element and  $c$  is the specific heat capacity. Boundary conditions are applied using the general relationship;

$$\dot{q}_{x,4 \rightarrow \beta} = -h_{\beta} S_{x,4} (T_{\beta} - T_{x,4})$$

Where the  $\beta$  term can be replaced with  $S$ ,  $W$  or  $A$  for strip, water or ambient coefficient and temperature values as the situation merits. In this equation  $h$  is a heat transfer coefficient and  $S$  represents the surface area of the exposed element face. Index 4 refers to the radial location index, only four elements are considered in the radial direction to ensure a fast response time, with indices ranging from 1 at the centre to 4 at the edge. The structure of the model is illustrated in Figure 2-3 below.

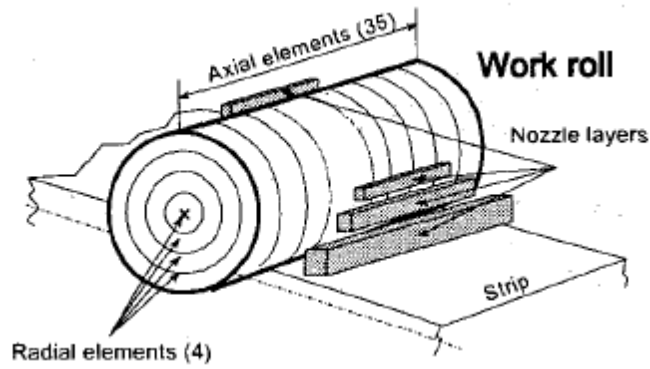


Figure 2-3: The work roll thermal model of Campos et al<sup>11</sup>.

The model uses the forward difference method, otherwise known as the Euler Method, to achieve a solution. This method is inherently unstable for longer time steps so the model calculates a safe length of time step based on the largest change in temperature in the previous time step. The larger the change the more susceptible to instability the model will become and the shorter the time step must be. To ensure that the model can run in real time the worst possible single iteration time is used to calculate the maximum number of iterations allowed. If a solution is achieved before this figure is reached then a new thermal profile is calculated, if not the previous thermal profile is used.

This system was purportedly validated against results from the literature, operation in a test mill and finally implemented on a full scale rolling operation, used to control the cooling systems based on work roll profile change due to thermal growth. It was well suited to its purpose but sacrifices detail for efficiency. Those sacrifices are not significant enough to affect the thermal growth, hence its fitness for purpose is valid, but it is far too coarse to provide data for a valid stress simulation and does not consider circumferential variation.

Also in 2000 Lee et al<sup>12</sup> introduced a three dimensional finite element solution to the transient thermal problem. The model bases its formulation around the three dimensional heat conduction term;

$$\frac{\partial T}{\partial t} = \alpha \nabla^2 T$$

which is equivalent to;

$$\frac{\partial T}{\partial t} = \alpha \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right)$$

in rectangular coordinates  $(x,y,z)$  or;

$$\frac{\partial T}{\partial t} = \alpha \left( \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial r^2} + \frac{1}{r^2} \frac{\partial^2 T}{\partial \theta^2} + \frac{\partial^2 T}{\partial z^2} \right)$$

in cylindrical polar coordinates  $(r,\theta,z)$ , where  $\alpha$  is the thermal diffusivity and  $z$  is the roll axis. The model only considers the work roll barrel, disregarding the journals, and utilises symmetry around the centre plane to only model half of the roll. The boundary condition on the end of the work roll opposite the symmetry condition can be varied, assuming either no-heat-transfer, a specified heat flux or the surface set to a specified ambient temperature. The whole mesh is assumed to rotate, eliminating the need to consider convection. The boundary conditions at the work roll surface are separated into zones as shown in Figure 2-4 below.

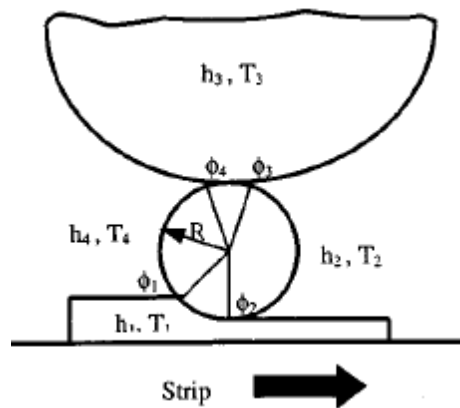
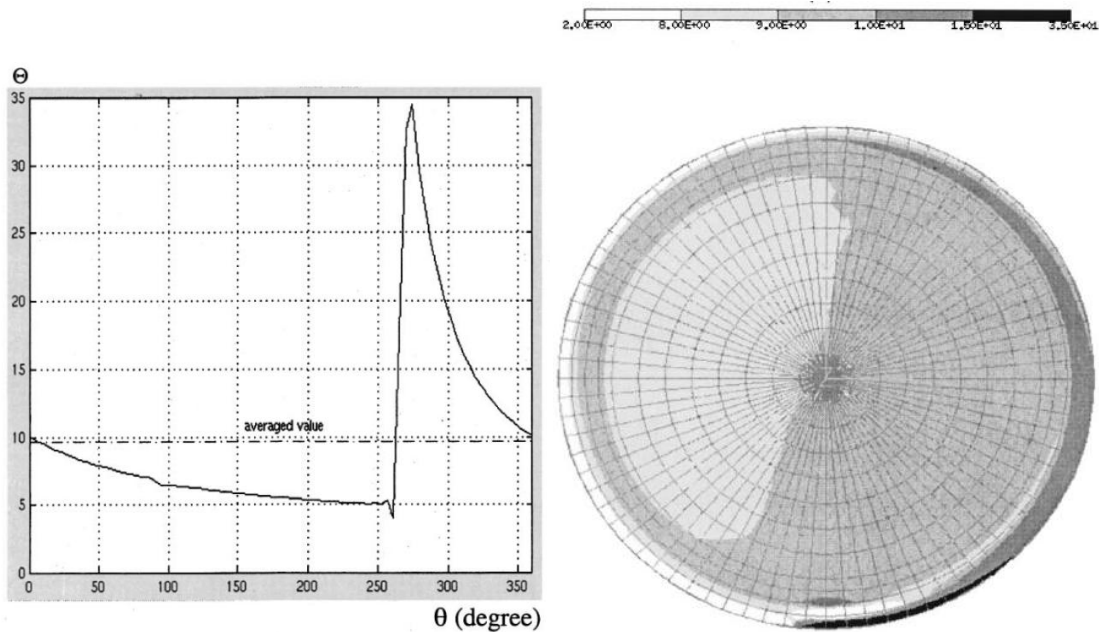


Figure 2-4: Boundary condition arrangement in Lee et al<sup>12</sup>.

$\phi_1 \rightarrow \phi_2$  covers the roll bite,  $\phi_2 \rightarrow \phi_3$  the first bank of water cooling sprays,  $\phi_3 \rightarrow \phi_4$  the contact with the backup roll and  $\phi_4 \rightarrow \phi_1$  the second bank of water cooling sprays. The model mesh rotates so that, with each time step, every surface element is ascribed a contact temperature and heat transfer coefficient related to which of the aforementioned zones it is passing through. A complicated Finite Element formulation is derived and converted into dimensionless units for easier comparison to other models, though due to lack of published details a direct comparison with any one solution was not possible. The temperature results are shown in Figure 2-5 below and displayed for the first time the significant circumferential fluctuations that occur at the work roll surface.



**Figure 2-5: Circumferential temperature variation at the roll surface (left) and temperature distribution at  $z = 0$  (right) from Lee et al<sup>12</sup>.**

The model of Lee et al<sup>12</sup> was a big step forward in considering 4 dimensions ( $r, z, \theta, t$ ) rather than the 2 or 3 that had previously been achieved. The significant contributions of this model were in highlighting the importance of the  $\theta$  direction to the thermal experience of a work roll.

A study in 2002 by Zhang et al<sup>13</sup> later used a different formulation of the concept used by Campos et al<sup>11</sup> to prove the viability of the approach for predicting thermal crown in CVC (continuously varying crown) work rolls. The equation for conduction within the work roll is given as;

$$\rho C_p \frac{\partial T}{\partial t} = k \left[ \frac{\partial^2 T}{\partial r^2} + \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial x^2} \right] + \dot{q}$$

This is equivalent to the equation used by Campos et al with the addition of a  $\dot{q}$  term to directly apply heat input from boundary conditions to the heat balance equation. This equation is solved internally using the Euler method. The model disregards the roll journals and makes use of symmetry along the roll centre plane, as shown in the layout diagram below.

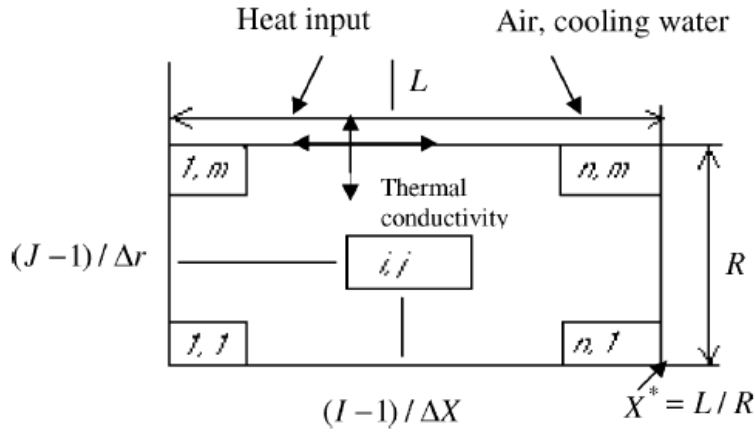


Figure 2-6: Structure of the model of Zhang et al<sup>13</sup>.

The temperatures at the surface are calculated using the equation;

$$T_{i,m} = \frac{(T_{i,m-1}/\Delta r^*) + (Rh/k)T_f}{(1/\Delta r^*) + (Rh/k)} + \dot{q}_i^* \Delta \tau$$

Where  $R$  is the work roll radius,  $\Delta r^*$  is the dimensionless radial mesh spacing,  $m$  is the radial index at the roll surface,  $h$  is the surface thermal conductivity coefficient,  $k$  is the material thermal conductivity,  $T_f$  is the temperature of the fluid in contact with the surface,  $\dot{q}_i^*$  is the dimensionless rate of heat flow and  $\Delta \tau$  is the dimensionless time step. There is no information on the derivation of this equation, although if it is assumed that the flow of heat into the surface is occurring at the same rate as conduction to material just below the surface the conduction equation could be written as;

$$k \frac{T_{i,m} - T_{i,m-1}}{\Delta r} = h(T_f - T_{i,m})$$

which can be rearranged to give;

$$\frac{T_{i,m} - T_{i,m-1}}{\Delta r^*} + \frac{Rh}{k}(T_{i,m} - T_f) = 0$$

given that  $\Delta r^* = \frac{\Delta r}{R}$ . Isolating  $T_{i,m}$ ;

$$T_{i,m} = \frac{(T_{i,m-1}/\Delta r^*) + (Rh/k)T_f}{(1/\Delta r^*) + (Rh/k)}$$

The initial surface temperature could then be obtained by adding a heat generation term to account for all the various boundary conditions to give;

$$T_{i,m} = \frac{(T_{i,m-1}/\Delta r^*) + (Rh/k)T_f}{(1/\Delta r^*) + (Rh/k)} + \dot{q}_i^* \Delta \tau$$

The boundary conditions are considered in some detail, the deformation heat flow,  $Q_p$ , given by;

$$Q_p = \beta_1 A W_p$$

where  $A$  is a constant,  $\beta_1$  is the proportion of deformation dissipated as heat and is set at 0.9 and  $W_p$  can be obtained from;

$$W_p = p_m \ln \left( \frac{h_1}{h_2} \right)$$

Where  $p_m$  is the rolling pressure,  $h_1$  is the strip thickness before rolling and  $h_2$  is the strip thickness after rolling.

The frictional heat flow,  $Q_r$ ;

$$Q_r = \beta_2 A W_r$$

Where  $A$  is a constant,  $\beta_2$  is a constant with the value 0.9 and  $W_r$  can be calculated from;

$$W_r = \mu p_m \bar{V}_r t_r$$

Where  $\mu$  is the friction coefficient,  $p_m$  is the rolling pressure,  $t_r$  is the contact time between the work roll and the strip and  $\bar{V}_r$  is average relative velocity between the work roll and the strip.

The contact heat flow,  $Q_R$ ;

$$Q_R = \frac{2k_s \sqrt{t} (T_{s0} - T_m)}{\sqrt{\alpha_s \pi}}$$

Where  $k_s$  is the strip thermal conductivity,  $T_{s0}$  is the initial strip temperature,  $t$  is the time and  $\alpha_s = \lambda_s/\rho_s C_s$ , where  $\rho_s$  and  $C_s$  are the density and specific heat capacity of the strip, respectively.  $T_m$  is given by;

$$T_m = \frac{\lambda_s T_{s0} \sqrt{\alpha_s} + \lambda_R T_{R0} \sqrt{\alpha_R}}{\lambda_s/\alpha_s + \lambda_R \sqrt{\alpha_R}}$$

Where  $T_{R0}$  is the initial roll temperature,  $\lambda_R$  is the roll thermal conductivity and  $\alpha_R = \lambda_R/\rho_R C_R$ , where  $\rho_R$  and  $C_R$  are the roll density and specific heat capacity respectively.

A comparison between the results calculated by this model and measurements taken from an actual work roll are shown in Figure 2-7 below. There are clearly some disparities but the general overall results were found to be in good agreement with experimental data.

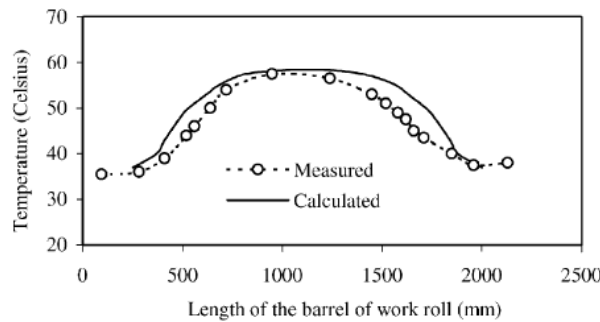


Figure 2-7: Comparison of calculated and measured surface temperature after rolling 18 coils, from Zhang et al<sup>13</sup>.

Another 2002 work by Serajzadeh et al<sup>14</sup> also included a model of the strip passing through the roll bite to model unsteady state behaviour. The general heat transfer equation is given as;

$$\frac{1}{r} \frac{\partial}{\partial r} \left( kr \frac{\partial T}{\partial r} \right) + \frac{1}{r^2} \frac{\partial}{\partial \theta} \left( k \frac{\partial T}{\partial \theta} \right) + \frac{\partial}{\partial z} \left( k \frac{\partial T}{\partial z} \right) = \rho C_p \frac{\partial T}{\partial t}$$

Where  $k$ ,  $\rho$  and  $C_p$  are the thermal conductivity, density and specific heat capacity of the material respectively. From the roll pressure and flow stress calculated from the interaction of these two bodies the following formula was used to calculate the contact heat transfer coefficient.

$$h_{con} = \frac{\bar{k}}{c_1} \left( \frac{p_r}{\bar{\sigma}_s} \right)^{1.7}, \bar{k} = \frac{k_r k}{k_r + k}$$



Where  $k_r$  and  $k$  are the roll and strip thermal conductivity respectively,  $p_r$  is the mean roll pressure,  $\bar{\sigma}$  is the mean flow stress and  $c_1$  is a constant empirically derived as 0.035mm for C-Mn steels. This relationship was given by Hadly et al<sup>15</sup>.

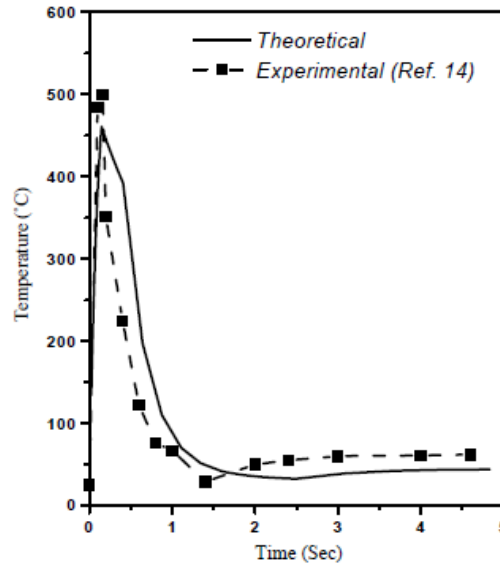


Figure 2-8: Experimental model result compared to theoretical expectation

The model results, shown in Figure 2-8, showed a large variation in temperature around the roll circumference. Figure 2-9 shows the change in temperature with time for points held stationary on the roll and at varying depths below the roll surface. From this information it was ascertained that very fast changes of temperature with time were limited to a shallow band of material near to the roll surface. The project also found that steady state conditions may or may not be attained depending on the process parameters. The latter conclusion was based on the findings for the reversing stand, which used far shorter bar lengths with considerable idle time between passes.

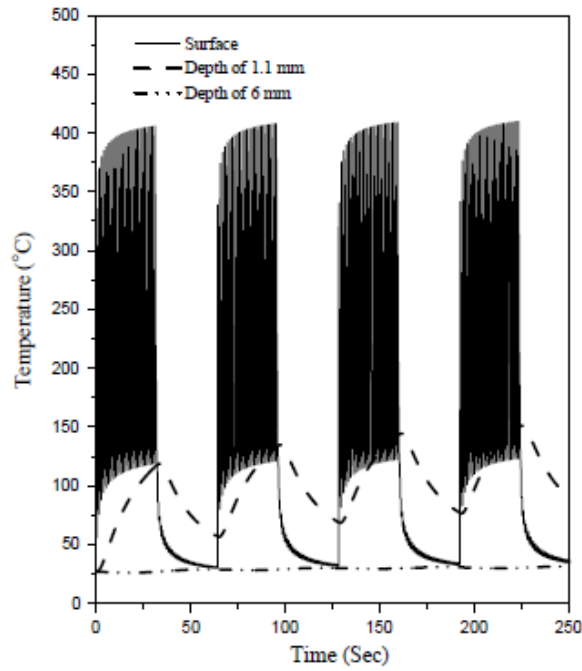


Figure 2-9: Temperature change with time at various depths

Later that year Hwang et al<sup>16</sup> published a Finite Element model that gave a thermo-mechanical analysis of both the work roll and strip. The interaction between the two media was defined by iterative contact calculations and gave results which compared very well with experimental data. In particular this article successfully implemented a frictional stress calculation based on a modified Coulomb friction model.

$$\sigma_t = -\mu\sigma_n g(|\Delta\mathbf{u}|)$$

where  $\sigma_t$  represents the tangential stress in the direction  $\Delta\mathbf{u}$ , and  $g(|\Delta\mathbf{u}|)$  can be attained from the following equation;

$$g(|\Delta\mathbf{u}|) = \frac{2}{\pi} \tan^{-1} \left( \frac{|\Delta\mathbf{u}|}{a} \right)$$

Where  $a$  is a very small constant. This allows the separate Coulomb equations for the sliding and sticking zones of the contact to be roughly represented in one equation.

In 2004 Corral et al<sup>17</sup> presented an analytic-numerical model based on a circular cross-section of a work roll. The roll surface was broken into multiple zones which represent the different areas of heating and cooling around the roll surface as shown below, and assumed steady-state conditions.

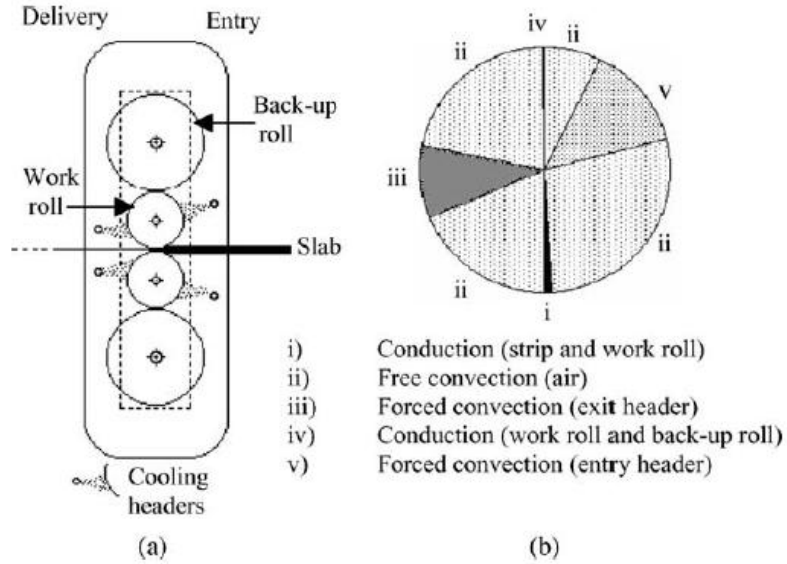


Figure 2-10: a) layout of a 4-high rolling stand, b) division of work roll based on boundary conditions in Corral et al<sup>17</sup>.

The analytical equation used to define the temperature at any position is given below.

$$T = T_a + T_s^* A_0 + T_s^* \sum_{n=1}^{50} \frac{\exp(-\alpha\sqrt{n})}{\sqrt{n}} \times [A_n \cos(n\theta_t - \alpha\sqrt{n}) + B_n \sin(n\theta_t - \alpha\sqrt{n})]$$

where;

$$\alpha = \sqrt{\frac{\omega_r}{2\kappa_r}} r, \omega_r = \frac{2\pi}{t_r}, \theta_t = \frac{2\pi t}{t_r}, T_s^* = \frac{T_s^0 - T_a + (\beta_S - 1)(T_{2\pi} - T_a)}{\beta_S}, \beta_S = 1 + \sqrt{\frac{(\rho c_p k)_r}{(\rho c_p k)_b}}$$

where;

- T – Temperature at a point in the model.
- $T_a$  – Ambient temperature.
- $T_s^0$  – Temperature in the strip surface before contact with the work roll.
- $\rho$  – Density of the work roll material.
- $C_p$  – Specific heat of the roll material subscript *r* denotes roll, **b** denotes backup roll.
- $k$  – Thermal conductivity.
- $r$  – Radial coordinate.
- $b$  – Outer radius.

- $T_{2\pi}$  – Work roll surface temperature before contact with the strip.
- $A_0$  – Coefficient of the Fourier series.
- $\omega_r$  – Work roll angular velocity.
- $\kappa_r$  – Thermal diffusivity of the work roll.
- $n$  – Iterator.
- $A_n$  – Coefficient of the Fourier series.
- $\theta_t$  – Angular position as a function of time.
- $B_n$  – Coefficient of the Fourier series.
- $t_r$  – Rotation period of work roll.

The resulting surface temperature profile is shown in Figure 2-11.

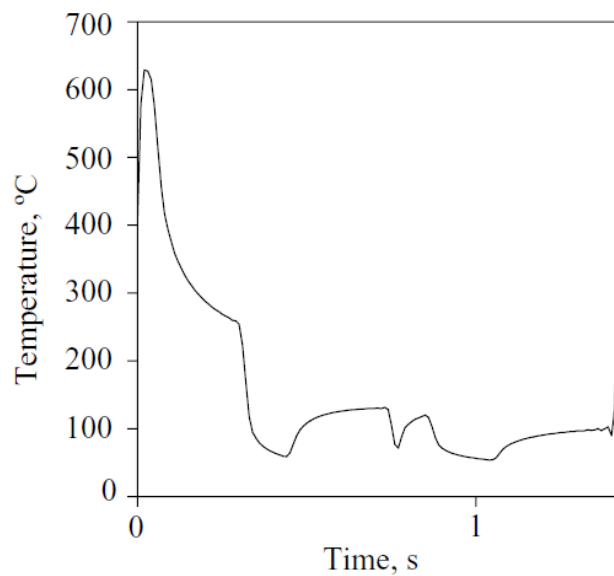


Figure 2-11: Surface temperature profile used by Corral et al<sup>17</sup>.

The model was used to predict the cycling of compressive and tensile stresses, showing that compressive effects existed around the roll bite and that tensile fluctuations could be found below the cooling headers. Based on this data and crack initiation tests (Figure 2-12) Corral et al found that work rolls can suffer from low-cycle fatigue.

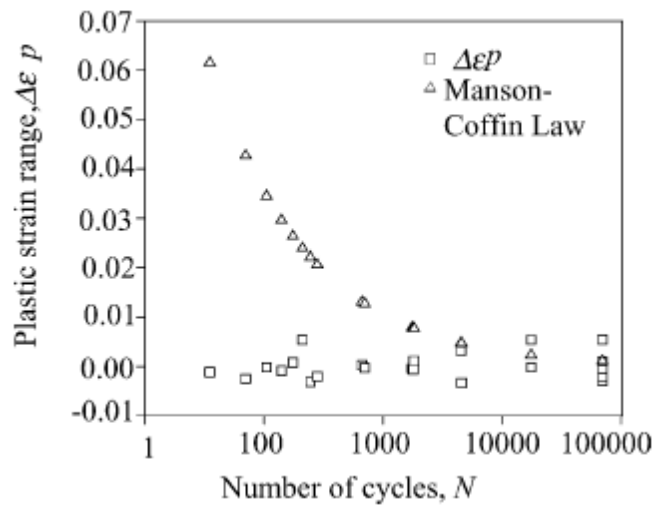


Figure 2-12: Crack initiation in the roll, from Corral et al<sup>17</sup>.

In 2007 Chang-Sheng et al<sup>18</sup> developed a fully three dimensional and transient representation of a work roll. This model was constructed in the finite element software package ANSYS and was simplified by assuming axial symmetry across the centre line and by removing the centre portion of the roll. The subdivision of the model into angular segments for boundary condition treatment and into elements for analysis is shown in Figure 2-13 below.

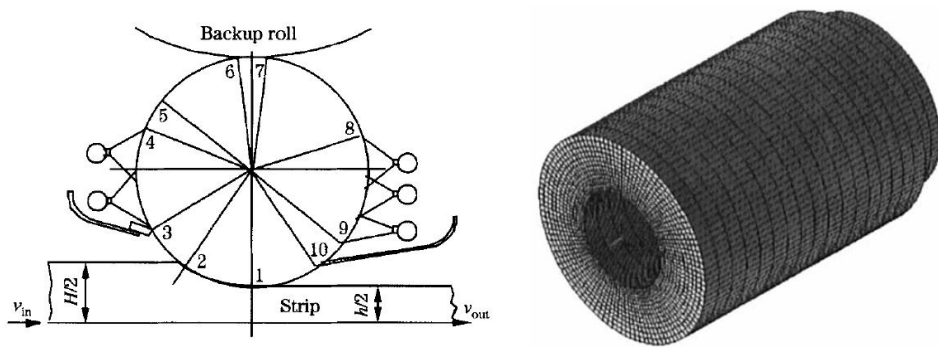


Figure 2-13: (left) location of boundary conditions, (right) finite element mesh structure, from Chang-Sheng et al<sup>18</sup>.

These boundary conditions do not have to remain static over time, so iterations of rolling and idle time could be modelled effectively. The thermal prediction was linked with an indirect coupling structural analysis to provide an accompanying stress field.

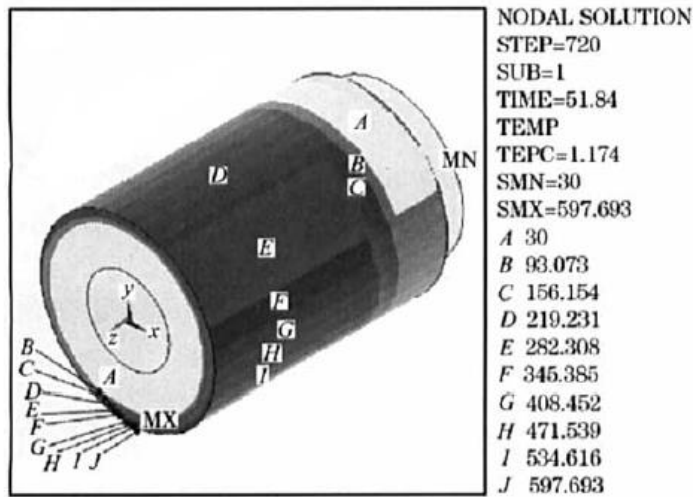


Figure 2-14: Temperature results from the model of Chang-Sheng et al<sup>18</sup>.

Figure 2-14 shows the temperatures calculated after 51.84 seconds of rolling time. The figure uses a false colour scheme with temperature thresholds for each colour as indicated in the legend, and is reproduced as published. The figure shows a very high variation in temperature around the roll surface reducing to very little variation a small distance below the surface. The article states that serious variation occurs only within the top 50mm of the roll, below which only very gradual changes occur.

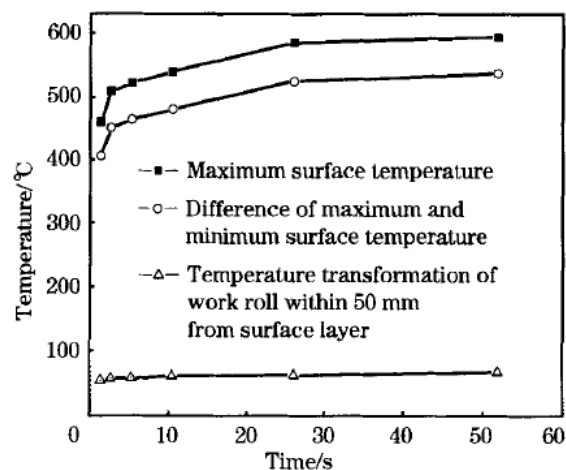


Figure 2-15: Change of average measures of temperature with time, from Chang-Sheng et al<sup>18</sup>.

Figure 2-15 shows the relative levels of change of maximum temperature at the work roll surface and 50mm below the surface. It can be seen that the rate of change at the surface is relatively fast towards the beginning of a rolling phase and diminishes with time. 50mm below the surface very little change has occurred even toward the end of the rolling phase.

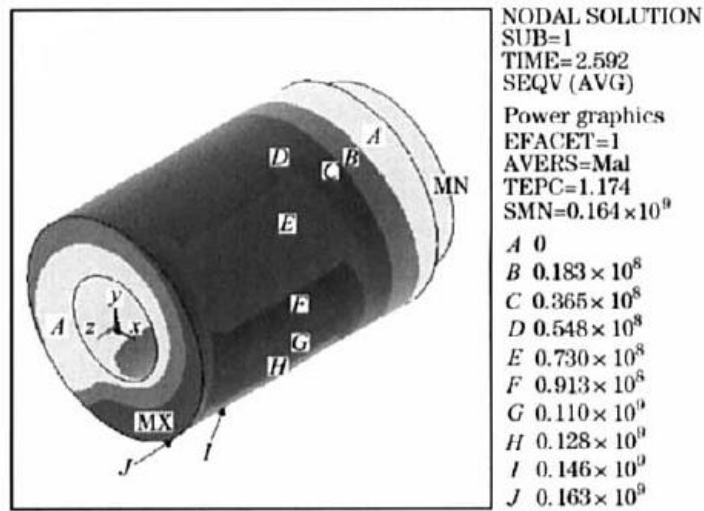


Figure 2-16: Stress results from the model of Chang-Sheng et al<sup>18</sup>.

Figure 2-16 shows the stress results a short amount of time after the beginning of a rolling phase. The false colour scheme is based on stress threshold levels which are indicated in Pascals in the legend at the right of the figure. The stress measure used is a von-Mises equivalent stress and close inspection shows that the main peaks and valleys in this plot correspond with the peaks and valleys in the thermal profile. Although there is very little interpretation it was noted that both positive and negative thermal stresses exist at the surface and that prolonged rolling leads to fatigue due to thermal stress cycling.

Also in 2007 Wang et al<sup>19</sup> produced a multi-faceted model of work roll profile. This model combines the effect of worn, ground and thermal profiles to give as close a representation to the actual roll profile as possible. The thermal model in this work is understandably simple as the temperature profile is a means to an end and need not be accurate in its fine details. The one dimensional model represents the work roll as a stack of discs, with each disc having its own boundary conditions. The equation used to solve the temperature problem is;

$$T_k^{n+1} = T_k^n + \frac{\alpha_s arc}{C_p \rho \pi r^2} (T_s - T_k^n) dt + \frac{2\alpha_w \xi_w}{C_p \rho r} (T_w - T_k^n) dt + \frac{2\alpha_A \xi_A}{C_p \rho r} (T_A - T_k^n) dt + \frac{\lambda}{C_p \rho \Delta l^2} (T_{k+1}^n + T_{k-1}^n - 2T_k^n) dt$$

$T$  refers to temperature, the  $k$  suffixes refer to axial location in the roll, the  $n$  suffixes to new and old time steps and  $S$ ,  $W$  and  $A$  refer to strip, cooling water and ambient temperatures respectively. The  $\alpha$  coefficients represent heat transfer coefficients, suffixes denoting different media types as previously described.  $\lambda$  is the conductivity of the work roll,  $arc$  is the arc length of the roll bite,  $r$  is the radius of the work roll and  $\xi_w$  and  $\xi_A$  are the contact ratios for the

cooling water and ambient environment respectively.  $\Delta l$  is the width of each slice of work roll,  $m$  is the mass of the slice,  $C_p$  is the specific heat and  $\rho$  is the material density.

To derive this equation, one can consider a one dimensional temperature change model and what would be important for that model. The crucial terms would be the change of temperature with time, which would have to be equal to the rate of conduction to the surface from the local environment and the diffusion below the surface. Using the same notation as above, and assuming small angles so that a rectangular coordinate frame can be used, this can be written mathematically as;

$$\frac{\partial T}{\partial t} = A(T_S - T_k^n) + B(T_W - T_k^n) + C(T_A - T_k^n) + D \frac{\partial^2 T}{\partial x^2}$$

Where  $A$ ,  $B$ ,  $C$  and  $D$  are the coefficients of thermal conductivity. Introducing finite difference approximations gives;

$$\frac{T_k^{n+1} - T_k^n}{dt} = A(T_S - T_k^n) + B(T_W - T_k^n) + C(T_A - T_k^n) + D \frac{T_{k+1}^n + T_{k-1}^n - 2T_k^n}{\Delta l^2}$$

Which rearranges to;

$$T_k^{n+1} = T_k^n + A(T_S - T_k^n)dt + B(T_W - T_k^n)dt + C(T_A - T_k^n)dt + \frac{D}{\Delta l^2} (T_{k+1}^n + T_{k-1}^n - 2T_k^n)dt$$

This equation would give the change in temperature at the surface and by setting  $A$ ,  $B$  and  $C$  to zero would also account for temperature change below the surface. The assumptions in the boundary conditions are that radiation, conduction to the backup roll and frictional heat generation in the roll bite are all negligible. The complete contour prediction package was found to be reliable when applied to online strip shape control in a rolling mill, presumably owned by the supporting Anshan Iron and Steel Corporation.

Another article from 2007 by Mercado-Solis et al<sup>20</sup> produced a simplified rolling test rig comprising a narrow disk to represent the work roll and another narrower disk complete with induction heater to represent the feedstock. The experimental rig was set up as shown in Figure 2-17 with a contact force of 1.5KN and the surface of the feedstock disk held at 1000°C. The points labelled 1 through 6 correspond to the points labelled on the temperature plots in Figure 2-18, which shows how the temperature profile responds to the varying boundary conditions at the labelled locations.



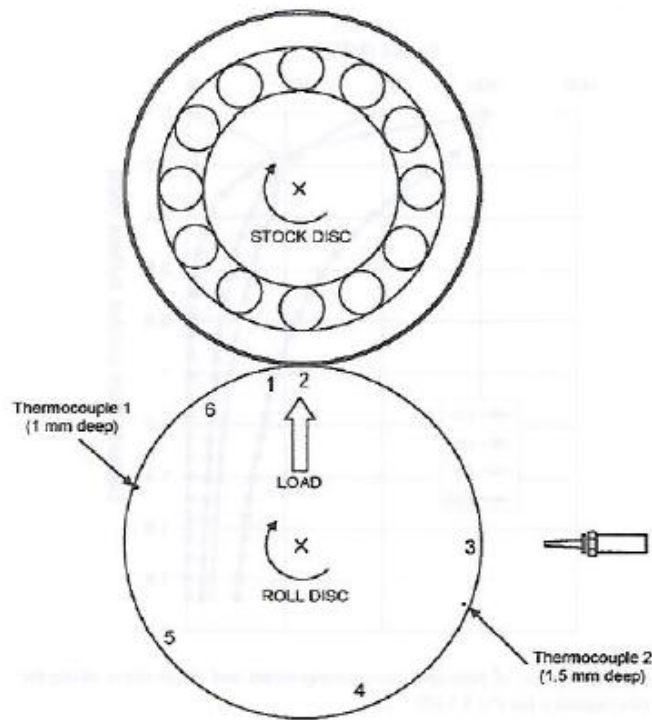


Figure 2-17: Layout of the test rig<sup>20</sup>

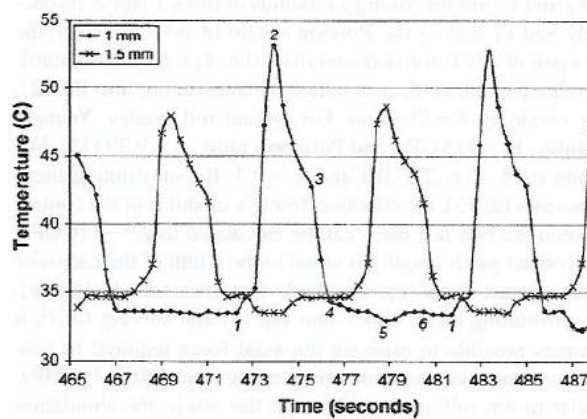


Figure 2-18: Temperature profiles at the two thermocouples with labelled positions<sup>20</sup>

A 2D finite element reproduction of this test rig accompanied the experimental data and showed good agreement with experimental results. The software used for this modelling, MSC.MARC/MENTAT, comes from a commercial software package and so its inner workings are not available for scrutiny.

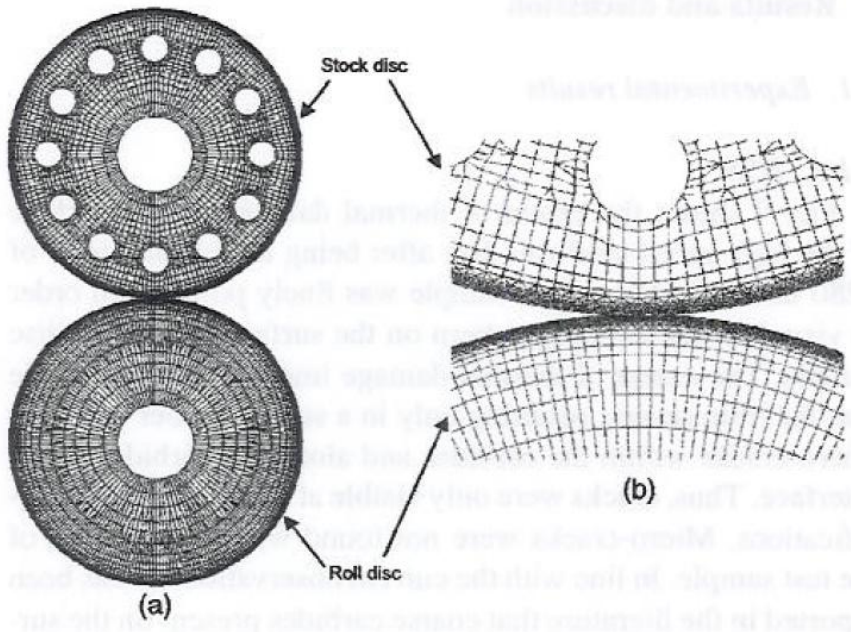


Figure 2-19: Finite element mesh structures

The comparison of this models predictions and the experimentally observed values (Figure 2-20) shows a good correspondence. The simulated results are slightly more angular and peak higher than the recorded temperatures but the margin of error is small.

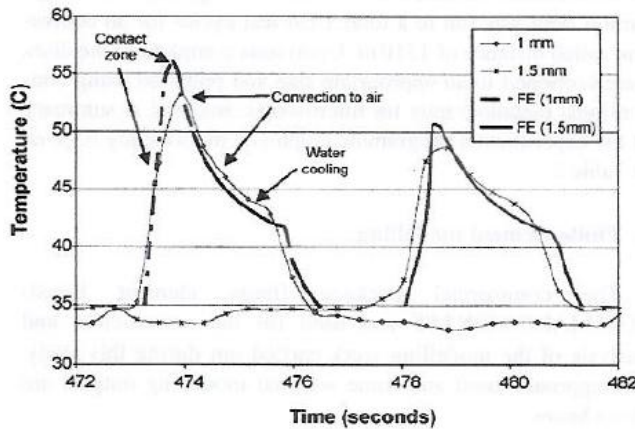


Figure 2-20: Comparison of measured and simulated results

Although the maximum temperatures for this study ( $\approx 144^{\circ}\text{C}$  at their highest) are recognised as being much lower than in an actual rolling campaign they were still high enough to initiate thermal fatigue cracks.

In 2008 Abbaspour and Saboonchi<sup>21</sup> released the latest iteration on a quasi-three dimensional model comprising a series of planes on the axial and radial axes. While Corral et al ([17]) assume that the roll is long enough compared to its radius that there will be little axial heat flow, this model assumes that the rotation of the work roll will account for enough

simulated heat flow that conductive heat flow in the circumferential direction will be negligible. The energy balance equation used is given by;

$$\nabla \cdot (k\nabla T) + \dot{q} = \rho C \frac{DT}{Dt}$$

Where  $\nabla$  is the Laplacian operator,  $T$  is the temperature at any point in the roll,  $\dot{q}$  is the heat transferred to the roll from external sources and  $\rho$  and  $C$  are the roll material density and specific heat capacity respectively. This three dimensional variation on the form used by Campos<sup>11</sup> is quickly simplified by assuming that heat transport in the circumferential direction is negligible compared to “convection” caused by the rotation of the roll, there is no internal energy production and movement of the work roll in any motion other than angular is not significant. To remove the heat transfer term required by the rotation of the roll the authors used the coordinate transformation  $\zeta = \vartheta - \omega t$  to give the final form of the equation as;

$$\frac{\partial T_W}{\partial t} = \alpha_W \left[ \frac{1}{r} \frac{\partial}{\partial r} \left( r \frac{\partial T_W}{\partial r} \right) + \frac{\partial^2 T}{\partial z^2} \right]$$

This article introduces the model and provides results (Figure 2-21, Figure 2-22) to prove that the model has been calibrated to actual process data from a hot rolling mill at the Mobarakeh Steel Complex. The calibration is taken from a surface temperature profile roughly 15 minutes after the roll has been removed from the hot mill.

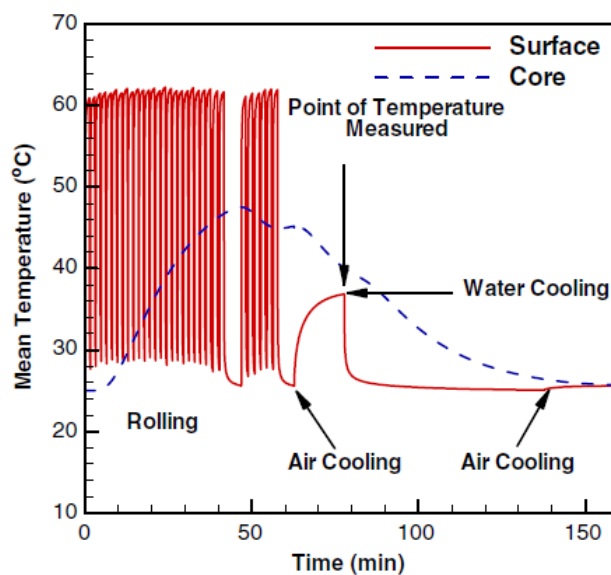


Figure 2-21: Testing regime

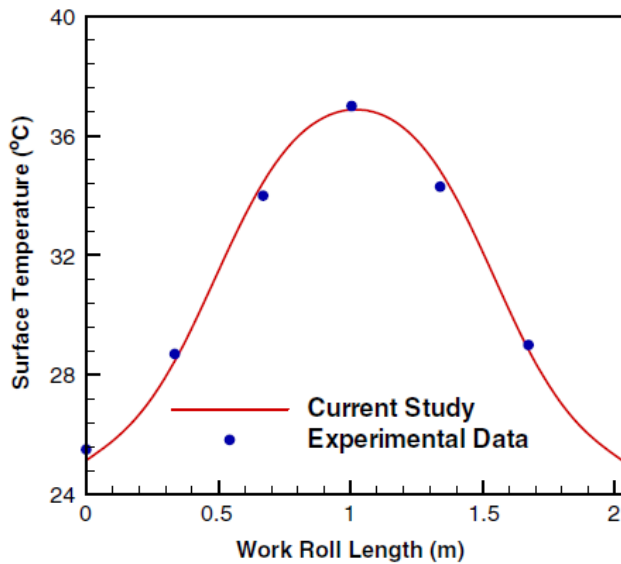


Figure 2-22: Comparison of experimental and simulated values across the roll length

The study also provided a predicted temperature profile for the work roll surface which shows a very large disparity between temperatures at different points of the roll surface; Figure 2-23.

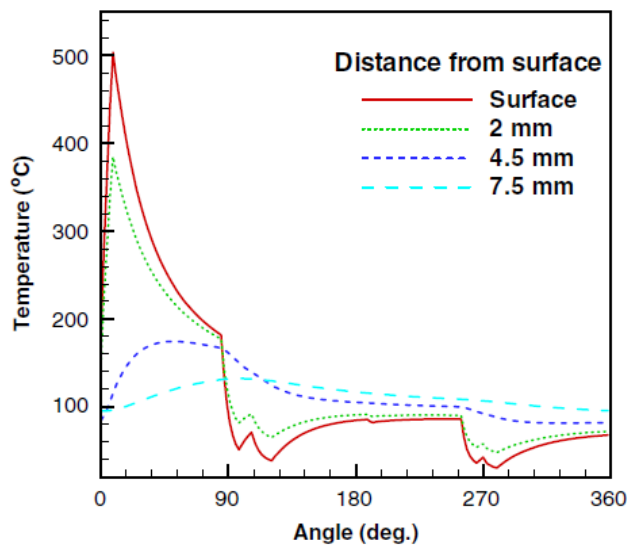


Figure 2-23: Circumferential temperature distribution

The clean abruptness of the lines betrays the simplified boundary conditions but it is reasonable to assume that the general shape of this temperature profile is correct. This lends further weight to the conclusion that circumferential temperature changes are significant and that damage due to cyclic thermal loading is a serious risk.

The most recent study was released in April 2010 by Benasciutti et al<sup>22</sup> and proposed a simplified finite element model that would provide characteristic if not accurate results at a fraction of the computational load.

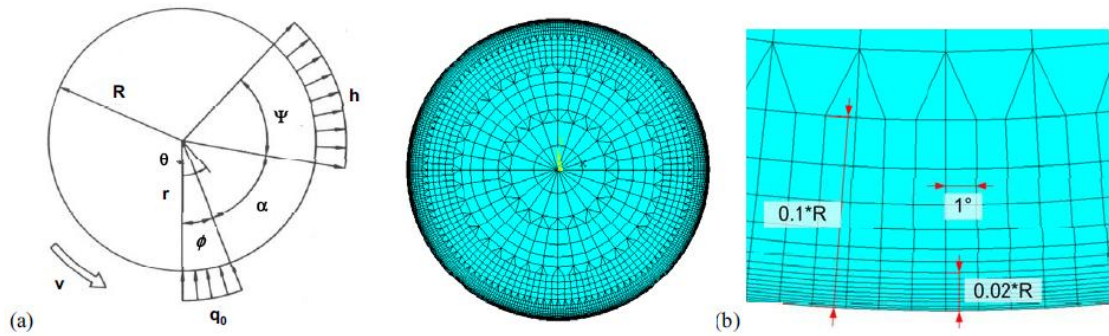


Figure 2-24: a) boundary conditions and overall mesh structure, b) mesh spacing at the roll surface

The model was non-steady-state and built using the ANSYS software package with the APDL (Ansys Parametric Design Language) scripting language. The simplified boundary conditions consist of a constant heating zone and a single convective cooling zone. The model was thermally validated against an analytical solution for the steady-state temperature given by a Fourier series from Patula<sup>23</sup>.

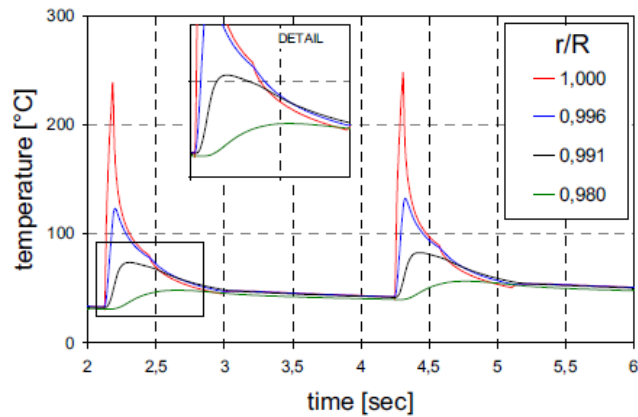


Figure 2-25: Evolution of the temperature of a point

Figure 2-25 shows how the temperature at a point on the work roll surface and points just below develop with time. The points are held static relative to the work roll surface so the sharp peaks correspond to the point's passage through the strip contact patch. The rate at which the temperature is rising is quite high, resulting in the overall roll temperature being higher than generally found through the literature. This is probably due to the 2D nature of the model; the area outside the strip width is not taken into account so the cooling effect of this extra material is missing.

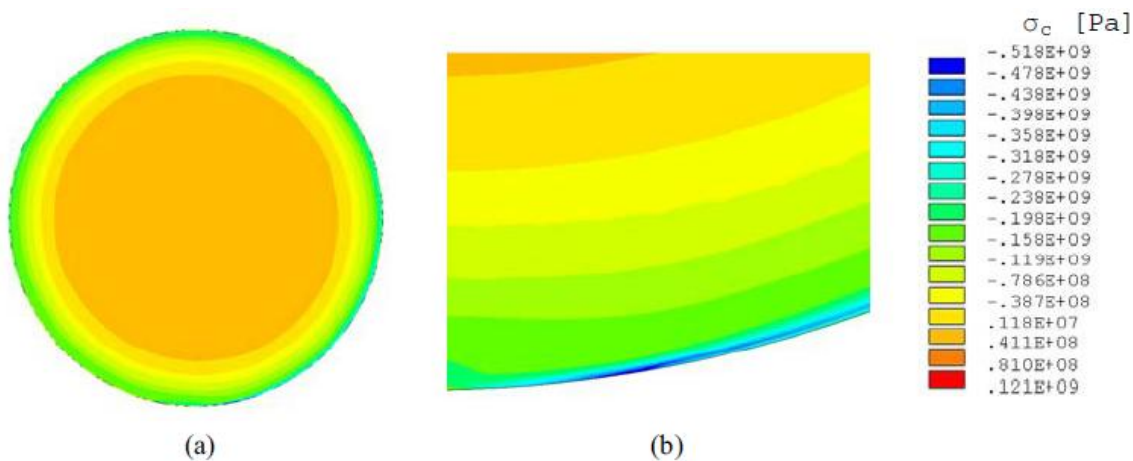


Figure 2-26: Circumferential normal thermal stress a) through the model and b) around the strip contact

Figure 2-26 shows the circumferential normal stress caused by the temperature distribution. There is no time included in the figure so it is not possible to determine what temperatures gave these stresses. While being the finest resolution so far the lack of a third dimension impacts on the accuracy of the model, which was proposed as a methodological example rather than an accurate portrayal.

## 2.2. Research in the Engineering Doctorate Scheme

---

The history of work roll modelling within the Engineering Doctorate scheme begins with Lee<sup>24</sup>. The project failed to provide a three dimensional temperature analysis but succeeded in producing a two dimensional axial-radial explicit finite difference model of the temperature and thermal deflection fields. The limiting factor in the three dimensional model was computational resources.

Pernas<sup>25</sup> followed Lee's project with a two dimensional radial-circumferential implicit finite difference model, sacrificing the capacity to model axial thermal behaviour for an insight into circumferential behaviour. This model included a convection term to simulate the rotation of the roll without needing to rotate the mesh, which along with the conversion to an implicit solver greatly improved its efficiency over the previous model. Unfortunately, due to an error discovered too late in the undergraduate project to remedy, the model only worked for very small angular velocities.

The previous models reached their current configuration through the work of Clarke<sup>26</sup> on rolling/sliding lubricated contact. Clarke's models combined a model with the axial-radial approach of Lee with a three dimensional model with Pernas's convection term. By using the results of the two dimensional model as boundary conditions for the three dimensional model, only a very shallow depth of material needed to be modelled in 3D. The model provided very

detailed and comprehensive results for a relatively small computational cost. Although not developed with hot rolling in mind, the problem of a cylindrical solid with a rotating heating / cooling boundary condition relates directly to the hot rolling process.

The next model in the chain of development in the Engineering Doctorate scheme is the White model<sup>3</sup>, which will be covered in section 2.4.

### 2.3. Research at Tata Steel

---

While the academic work mentioned previously was being carried out, researchers at Tata Steel were developing their own group of models. There are currently four work roll and rolling models available within Tata Steel, TEX, Strip\_Rollbite, TITAN and Roll Stress Model.

The TEX model, or Thermal EXpansion model, calculates the three dimensional temperature profile using two finite difference models to simulate first temperature flow in the axial direction, then in the circumferential direction. The model is efficient and fast to calculate changes in work roll thermal camber in-line with the process. A considerable portion of the development work was applied to calculating the pattern of heat extraction from the work roll surface due to the impingement of cooling water sprays. Due to the detail and validation available in these “cooling maps” these results were incorporated in the White model and into the current work.

The Strip\_Rollbite model combines a collection of one-dimensional radial paths that represent the work roll and a two dimensional representation of the strip. This model has been used to investigate the interaction between the work roll and the strip and includes a frictional heating model, strip temperature effects, oxide layer effects and contact pressure effects.

TITAN is a whole mill model which presents a user with a graphical user interface from which a virtual rolling mill can be assembled complete with multiple simulation models for different aspects of the rolling process. As a modelling effort it is incredibly broad in scope thanks to a modular design which enables different models to be “plugged in” with very little effort. If one element is missing from this model it is consideration of work roll temperature, though this may soon be remedied using one of the lightweight models such as TEX or Strip\_Rollbite.

The Roll Stress Model was developed in Swinden Technology Centre for long product rolling. This process differs significantly from strip rolling in that strip is far wider than it is thick while long products tend to have a relatively even cross-section. The work rolls in long product rolling therefore have a different cross-section, tending to be larger radially than axially. Furthermore, not all rolls are single piece constructions, some are formed by shrink-

fitting the working surface to a re-usable spindle. While thermal conduction plays the majority role in dictating work roll temperature in strip rolling, heat in a long product work roll comes predominantly from friction and deformation heating. The work roll stress in long product rolling is further complicated by sometimes complicated product cross-sections, causing uneven pressure distributions and introducing stress gradients<sup>27</sup>.

## 2.4. The White Model<sup>3</sup>

---

The function of the White model is to use a combination of process data and a minimum of data input from the user to calculate the thermal stress and expansion profile of a work roll. To achieve this task a software package was developed which performed the following functions.

- Reading a user-specified range of process data from the hot mill database.
- Accepting user input to supply necessary data not held on the database.
- Calculating the three dimensional heat distribution of the work roll varying in time over the period for which process data was acquired.
- Opening and running ABAQUS CAE to calculate the stress results and extracting them without requiring user intervention.
- Presenting the results in an easily digestible format.

As an introduction to the C# programming language in which the White model was constructed the user interface was re-written at the beginning of the current project. As such the interface aspects of the White model will not be discussed. The stress components of the White model will also be ignored as later changes to the thermal models and updates to the ABAQUS software package have rendered the previous models non-functional. What follows is a brief description of the mathematics behind the thermal models which were implemented in the new user interface. It should be noted that the lessons learned in the White model contribute heavily to the design of the GUI (Graphical User Interface) and the program structure surrounding the core and shell models. The data extraction via SQL (Structured Query Language, a popular language for interacting with databases) from the hot mill database and processing into the format used by the models could be carried across almost verbatim.



### 2.4.1. General Introduction

---

The approach taken towards solving the work roll thermal problem is an adaptation of that taken by Clarke<sup>26</sup> and exploits the two time-scales on which the temperature of the material is changing.

#### *Time scale 1*

---

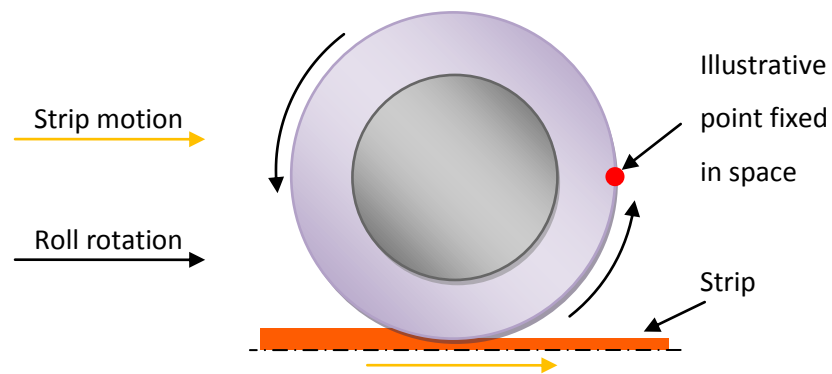
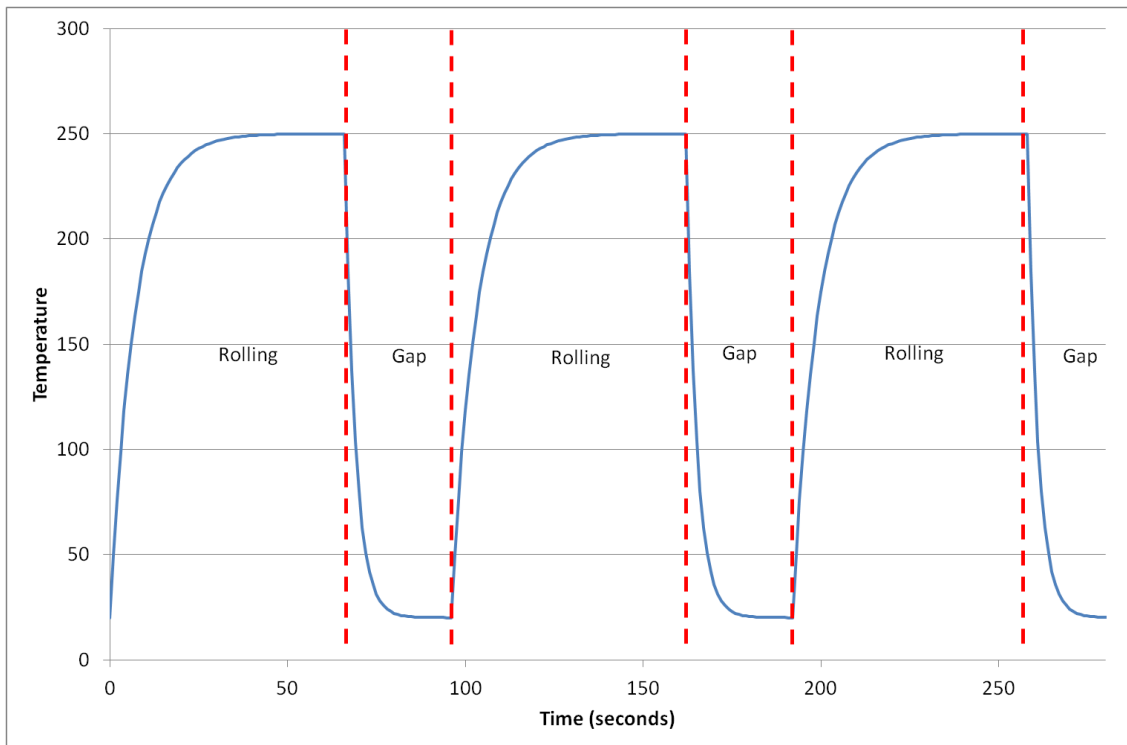


Figure 2-27: A stationary point in a moving work roll

Time scale 1 considers periods of time over which bulk changes to temperature are significant. Consider a point positioned at 90 degrees to the roll bite (the red circle in Figure 2-27, the roll bite is the area of the work roll in contact with the strip) and held stationary relative to the stand geometry so that as the work roll rotates the point stays 90 degrees behind the roll bite at all times. If the overall magnitude of heating and cooling were balanced the temperature at this point could be expected to remain constant. For each revolution the same amount of heat would be absorbed from the strip and the same amount removed by the cooling sprays.



**Figure 2-28: The time varying temperature of a stationary point**

Figure 2-28 shows the typical temperature variation that could actually be expected at the point in Figure 2-27 in a real rolling application. When a strip first makes contact with a work roll it is relatively cool due to its previous period of cooling with no significant heat sources. Heat conduction is driven by temperature difference so this sudden contact between a hot strip and a cold roll drives heat into the work roll causing it to warm up. As the overall work roll temperature is raised the temperature difference between the strip and roll – and hence the amount of heat transferred to the roll – is reduced. Conversely the amount of heat transferred from the work roll to the cooling spray increases as the roll temperature increases. If these conditions prevailed for long enough the work roll would reach a new equilibrium, though in practice in a non-continuous hot strip mill the end of the strip always arrives before this can happen.

The model based on time scale 1 describes this slowly time varying behaviour.

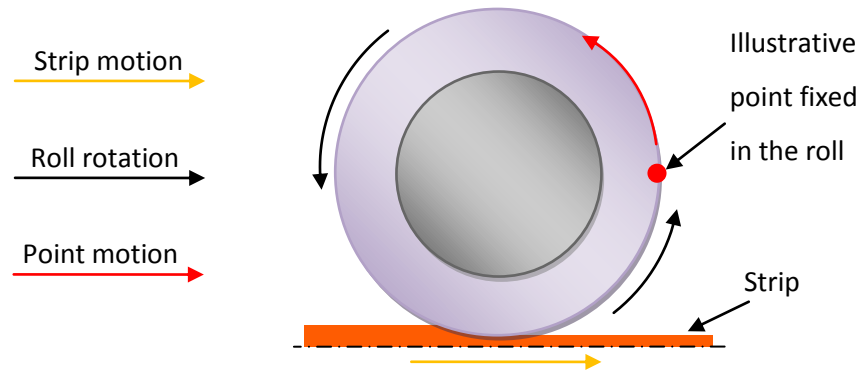


Figure 2-29: A point which moves with the work roll

Time scale 2 considers the shorter periods of time between work roll revolutions. Consider again a point on the work roll (Figure 2-29) which this time is allowed to rotate with the work roll, i.e. stationary relative to the work roll material. Figure 2-30 shows the temperature variation that can be expected by the material at that point.

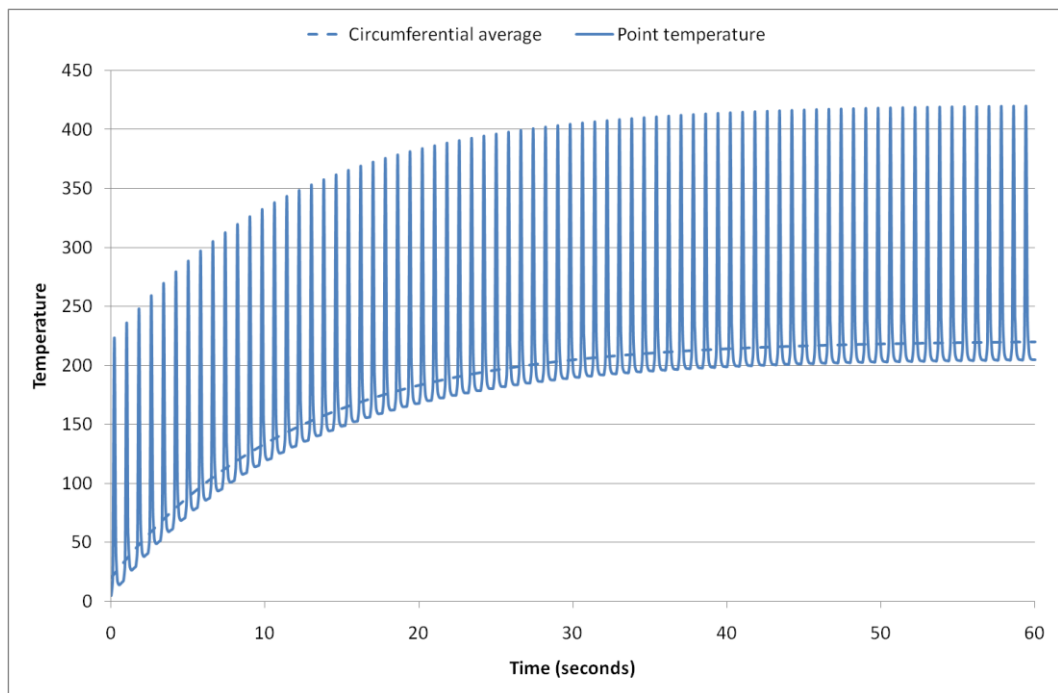
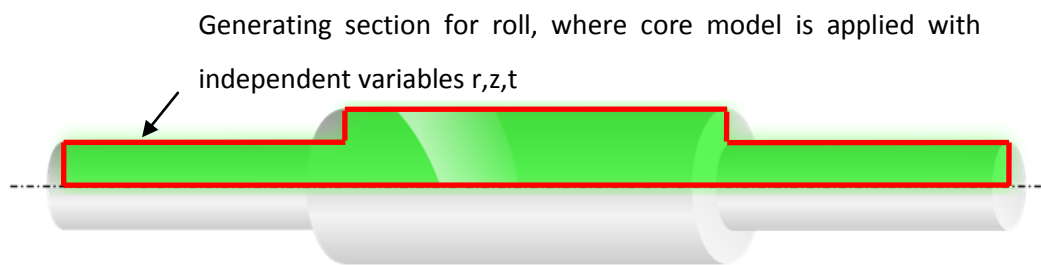


Figure 2-30: The time varying temperature of a point moving with the work roll

Compared to the slow and steady increases and decreases due to total heat input and heat output imbalance this variation is fast and abrupt. The material point enters the roll bite cold and is heated very quickly by contact with the strip, leaving the roll bite at a high temperature. Almost immediately the point enters the range of the cooling water jets and is brought back down to a low temperature. The material is held at a low temperature for the rest of its

rotation to draw as much heat out of the work roll as possible before it arrives at the roll bite entrance again. The time scale 2 model describes this fast paced variation.

The first point to observe in Figure 2-30 is that the average temperature of all the points around the circumference changes at the same rate as a point held stationary in relation to the mill geometry. This is because they are both responding to the same process, the transfer of heat left over from the heating-cooling imbalance to material below the surface. The time scale 1 model can therefore be simplified to model only the average circumferential temperatures at any radial-axial location and still be valid in areas displaying time scale 2 behaviour. This simplified model is called the “Core model” and is outlined in Figure 2-31 below (The White model took advantage of the symmetry of the work roll by only modelling the right half of the model and assuming the left side was identical).



**Figure 2-31: the Core model**

The task of modelling the time scale 2 behaviour can also be simplified due to its predictability and the relative stability of the oscillations after the first few seconds of the rolling period. There is only a small difference between the peak or trough of a feature later in the rolling period and the corresponding point on the next feature. This allows the model predicting time scale 2 behaviour to be steady-state, as the slow change in total work roll temperature has an insignificant impact on the temperature of a point during one revolution of the roll. The steady-state assumption (that temperature does not change with time) means only one feature can be calculated at a time but the repetitive nature of the features means that only one feature needs to be calculated for behaviour over a much longer span of time to be known.

A further simplification can be made by acknowledging that the temperature fluctuations in a cylinder with a moving heat source on its surface diminish in magnitude below the surface. At a given depth dictated by the intensity of the source, the speed at which it moves and the material properties of the cylinder there will be no perceivable variation in temperature around the circumference. In other words, below this depth the temperature of any point will be equal to the average temperature as predicted by the Core model. Therefore, rather than modelling the whole roll in three dimensions it is only necessary to model material

immediately beneath any moving heat sources. The time scale 2 model takes advantage of this, is called the “Shell model” and is illustrated in green in Figure 2-32 (as with the Core model, the White model only considered the right hand side of the model shown below).

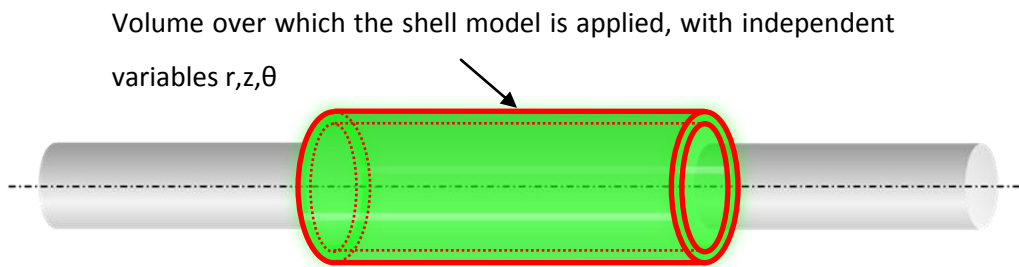


Figure 2-32: the Shell model

By using the Core and Shell models it is possible to obtain the complete temperature solution for the work roll at any point in time. The core model describes the change in the general averaged temperature profile with time, potentially taking account of the effect of multiple changes in external conditions in the work rolls thermal history. The shell model is solved once to fill in the circumferential variation in temperature experienced by material near the surface of the work roll barrel, using temperatures from the final core model solution as a boundary condition.

#### 2.4.2. Energy Balance in a Control Volume

---

Both models derive their heat transport equations from the same heat balance, which is itself derived with the assumption that all heat transfer between internal model elements will obey Fourier’s Law of Conduction.

$$\dot{Q} = -kA \left( \frac{dT}{dx} \right)$$

Equation 2-1

Where  $\dot{Q}$  is the heat flow (W),  $k$  is the thermal conductivity (W/mK),  $A$  is the area of the face through which the heat flow is occurring ( $m^2$ ) and  $\frac{dT}{dx}$  is the temperature gradient in the  $x$ -direction (K/m). Heat flow in one direction can be considered using the element in Figure 2-33 below.

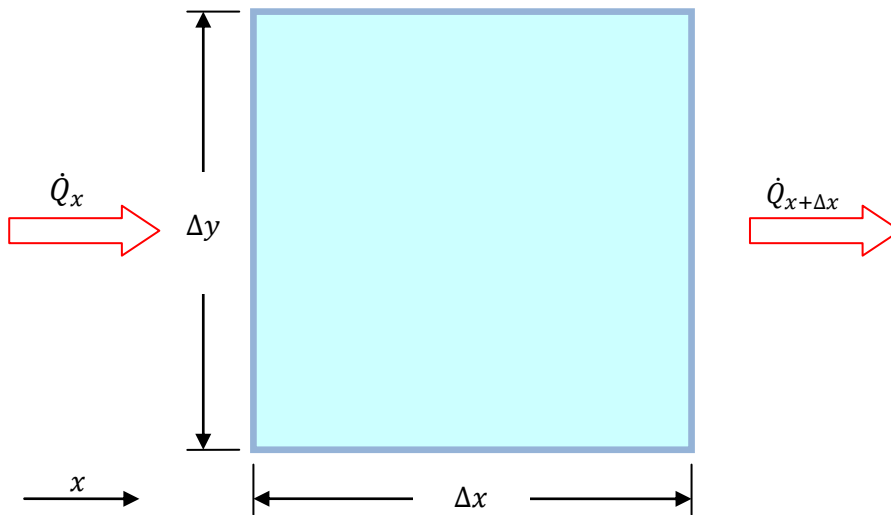


Figure 2-33: Elementary heat flow in the x-direction

The heat conduction into the left face is given by;

$$\dot{Q}_x = -k\Delta y\Delta z \left(\frac{dT}{dx}\right)_x \quad \text{Equation 2-2}$$

And heat conduction out of the right face is;

$$\dot{Q}_{x+\Delta x} = -k\Delta y\Delta z \left(\frac{dT}{dx}\right)_{x+\Delta x} \quad \text{Equation 2-3}$$

The change in thermal energy in this control volume will therefore be;

$$\begin{aligned} \dot{Q}_{net} &= \dot{Q}_x - \dot{Q}_{x+\Delta x} \\ &= k\Delta y\Delta z \left( -\left(\frac{dT}{dx}\right)_x + \left(\frac{dT}{dx}\right)_{x+\Delta x} \right) \\ &= \Delta x\Delta y\Delta z k \left( \frac{\left(\frac{dT}{dx}\right)_{x+\Delta x} - \left(\frac{dT}{dx}\right)_x}{\Delta x} \right) \end{aligned} \quad \text{Equation 2-4}$$

Which as  $\Delta x$  approaches zero becomes;

$$\dot{Q}_x = \Delta V k \frac{\partial^2 T}{\partial x^2} \quad \text{Equation 2-5}$$

This can be expanded to include all three dimensions to give;

$$\dot{Q}_c = \Delta V k \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right) \quad \text{Equation 2-6}$$

Equation 2-6 now represents the rate of change of internal energy in an element of volume  $\Delta V$ . The first law of thermodynamics is given by;

$$Q = \Delta U + W \quad \text{Equation 2-7}$$

Where  $Q$  is the energy supplied to the element,  $\Delta U$  is the change in internal energy and  $W$  is the work done by the element. The system being considered is a small element inside the work roll, so it is fair to assume that it is not doing any work. For a fixed period of time  $\Delta t$  Equation 2-7 becomes;

$$Q = \Delta t \Delta V k \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right) \quad \text{Equation 2-8}$$

Alternatively, using the volume and heat capacity of the material  $\Delta U$  can be represented as;

$$\Delta U = \Delta V \rho C_p \Delta T \quad \text{Equation 2-9}$$

Where  $C_p$  is the specific heat capacity (J/Kg K) and  $\rho$  is the density (Kg/m<sup>3</sup>). Equating the two equations gives;

$$\Delta V \Delta t k \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right) = \Delta V \rho C_p \Delta T \quad \text{Equation 2-10}$$

Simplifying, rearranging and taking  $\Delta t$  to the limit;

$$\frac{\partial T}{\partial t} = \frac{k}{\rho C_p} \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right) \quad \text{Equation 2-11}$$

Introducing the thermal diffusivity,  $\alpha = k/\rho C_p$ , and converting from rectangular to cylindrical polar coordinates  $(r, \theta, z)$  to make the application of this basic equation to work roll geometry more straightforward:

$$\frac{\partial T}{\partial t} = \alpha \left( \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial r^2} + \frac{1}{r^2} \frac{\partial^2 T}{\partial \theta^2} + \frac{\partial^2 T}{\partial z^2} \right) \quad \text{Equation 2-12}$$

This equation now stands as an accurate representation of the thermal behaviour of a small volume of material in a work roll with the axes fixed in the rotating roll, which means that the heating and cooling zones, which are fixed in space, are moving with  $\theta$  as far as Equation 2-12 is concerned. However, it is far easier to model the work roll with the axes fixed in space, so that the boundary conditions are fixed in the axis set. This can be achieved by introducing a convective term to account for the flow of heat in the circumferential direction of the stationary axis set brought about by the rigid body motion of the roll, and in this thesis "convection" is the term used to describe this process.

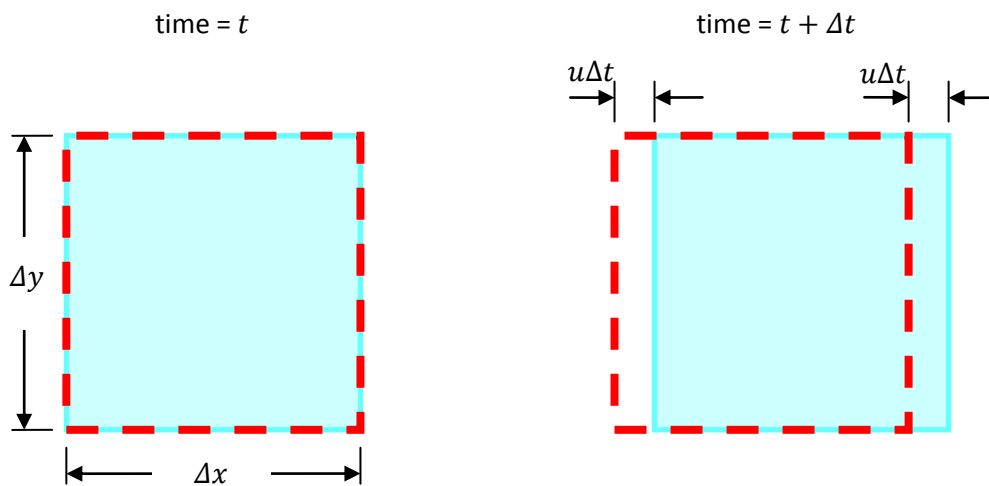


Figure 2-34: A static element in a moving solid

Figure 2-34 illustrates the situation of an element which is held static in a material body which is moving at a constant speed  $u$ . At time  $t + \Delta t$  the element has remained in the same location but the material it once contained has moved to the right. From an energy balance perspective the internal energy in the material that is now outside the right element boundary has been lost. Conversely material entering the left element boundary is carrying energy in with it. The energy entering and exiting the element can be expressed mathematically as...

$$Q_x = T_x \Delta V \rho C_p = T_x \times u \Delta t \Delta y \Delta z \times \rho C_p \quad \text{Equation 2-13}$$

and

$$Q_{x+\Delta x} = -T_{x+\Delta x} \Delta V \rho C_p = -T_{x+\Delta x} \times u \Delta t \Delta y \Delta z \times \rho C_p \quad \text{Equation 2-14}$$

respectively. So the total change of energy in the element is:



$$\begin{aligned}
Q &= u\Delta t\Delta y\Delta z\rho C_p(T_x - T_{x+\Delta x}) \\
&= u\Delta t\Delta V\rho C_p \frac{(T_x - T_{x+\Delta x})}{\Delta x}
\end{aligned}
\tag{Equation 2-15}$$

By the same token as before, equating this with Equation 2-9 gives;

$$\Delta V\rho C_p\Delta T = u\Delta t\Delta V\rho C_p \frac{(T_x - T_{x+\Delta x})}{\Delta x}
\tag{Equation 2-16}$$

Rearranging and simplifying;

$$\frac{\Delta T}{\Delta t} = -u \frac{T_{x+\Delta x} - T_x}{\Delta x}
\tag{Equation 2-17}$$

Which as  $\Delta t$  and  $\Delta x \rightarrow 0$  becomes;

$$\frac{\partial T}{\partial t} = -u \frac{\partial T}{\partial x}
\tag{Equation 2-18}$$

This gives a mathematical representation of the time varying temperature at a fixed point due only to material moving through that point in one direction. In the work roll the only significant velocity component is rotation, so converting to cylindrical polar coordinates and substituting  $\omega r$  for  $u$  gives the convection term;

$$\frac{\partial T}{\partial t} = -\omega r \left( \frac{1}{r} \frac{\partial T}{\partial \theta} \right) = -\omega \frac{\partial T}{\partial \theta}
\tag{Equation 2-19}$$

Adding Equation 2-12 and Equation 2-19 provides the three dimensional heat transfer equation with the effects of both conduction and convection.

$$\frac{\partial T}{\partial t} = \alpha \left( \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial r^2} + \frac{1}{r^2} \frac{\partial^2 T}{\partial \theta^2} + \frac{\partial^2 T}{\partial z^2} \right) - \omega \frac{\partial T}{\partial \theta}
\tag{Equation 2-20}$$

For the two dimensional model the heat transfer equation can be generated by removing the circumferential terms. The resulting equation is essentially identical to that used by Campos et al<sup>11</sup>.

$$\frac{\partial T}{\partial t} = \alpha \left( \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial r^2} + \frac{\partial^2 T}{\partial z^2} \right) \quad \text{Equation 2-21}$$

The heat transfer equation for the steady state three dimensional shell model is created by setting the time-varying terms of Equation 2-20 to zero as required by the steady state assumption.

$$0 = \alpha \left( \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial r^2} + \frac{1}{r^2} \frac{\partial^2 T}{\partial \theta^2} + \frac{\partial^2 T}{\partial z^2} \right) - \omega \frac{\partial T}{\partial \theta} \quad \text{Equation 2-22}$$

### 2.4.3. Finite Difference Solution

To attempt to solve Equation 2-21 or Equation 2-22 analytically and exactly would be prohibitively complicated, so a finite difference approach is taken to provide a numerical solution. The finite difference terms can be derived by considering an arbitrary function  $w(x)$ .

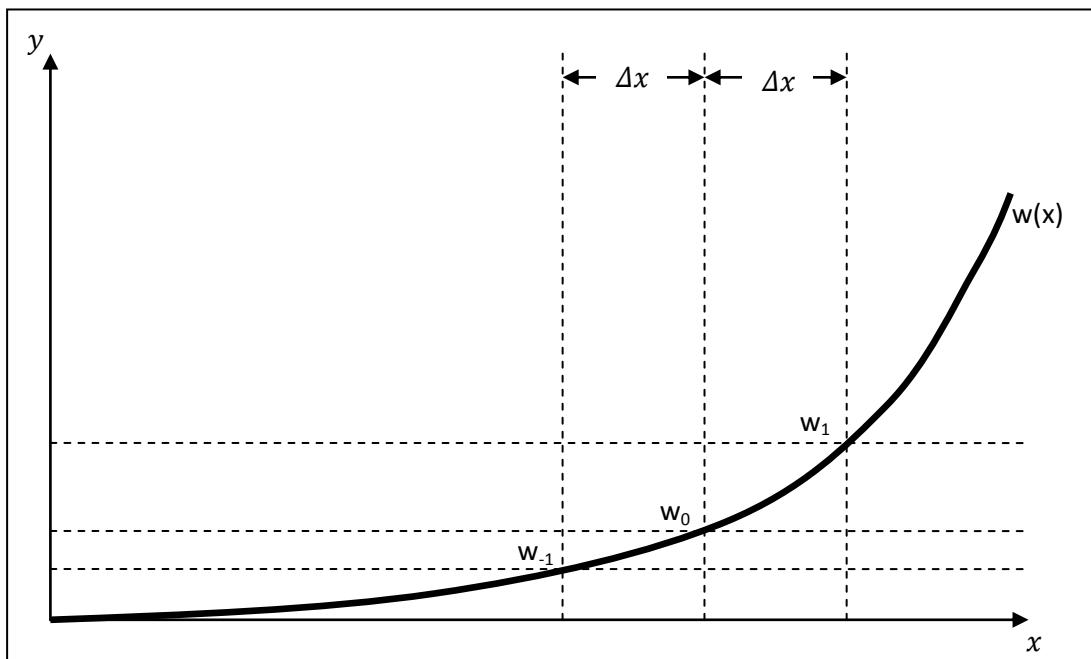


Figure 2-35: Finite difference approximation of an arbitrary function

Taylor's theorem gives;

$$w(x + \Delta x) = w(x) + \Delta x \frac{\partial w(x)}{\partial x} + \frac{\Delta x^2}{2!} \frac{\partial^2 w(x)}{\partial x^2} + \frac{\Delta x^3}{3!} \frac{\partial^3 w(x)}{\partial x^3} + \dots \quad \text{Equation 2-23}$$

With this it is possible to predict the value of  $w_1$  and  $w_{-1}$  from  $w_0$  and its derivatives.

$$w_1 = w_0 + \Delta x \frac{\partial w_0}{\partial x} + \frac{\Delta x^2}{2!} \frac{\partial^2 w_0}{\partial x^2} + \frac{\Delta x^3}{3!} \frac{\partial^3 w_0}{\partial x^3} + \dots \quad \text{Equation 2-24}$$

$$w_{-1} = w_0 - \Delta x \frac{\partial w_0}{\partial x} + \frac{\Delta x^2}{2!} \frac{\partial^2 w_0}{\partial x^2} - \frac{\Delta x^3}{3!} \frac{\partial^3 w_0}{\partial x^3} + \dots \quad \text{Equation 2-25}$$

Subtracting Equation 2-25 from Equation 2-24;

$$w_1 - w_{-1} = 2\Delta x \frac{\partial w_0}{\partial x} + \frac{2\Delta x^3}{3!} \frac{\partial^3 w_0}{\partial x^3} + \frac{2\Delta x^5}{5!} \frac{\partial^5 w_0}{\partial x^5} + \dots \quad \text{Equation 2-26}$$

And adding Equation 2-24 and Equation 2-25;

$$w_1 + w_{-1} = 2w_0 + \Delta x^2 \frac{\partial^2 w_0}{\partial x^2} + \frac{2\Delta x^4}{4!} \frac{\partial^4 w_0}{\partial x^4} + \dots \quad \text{Equation 2-27}$$

Ignoring higher order terms and rearranging Equation 2-26 and Equation 2-27 gives approximations for the differential expressions.

$$\frac{\partial w_0}{\partial x} = \frac{w_1 - w_{-1}}{2\Delta x} + 0(\Delta x^2) \quad \text{and} \quad \frac{\partial^2 w_0}{\partial x^2} = \frac{w_1 + w_{-1} - 2w_0}{\Delta x^2} + 0(\Delta x^3) \quad \text{Equation 2-28}$$

These approximations are known as central difference formulae. Forward and backward difference formulae can be derived directly from Equation 2-24 and Equation 2-25 as follows, though their accuracy suffers from disregarding lower order terms.

$$\frac{\partial w_0}{\partial x} = \frac{w_1 - w_0}{\Delta x} + 0(\Delta x) \quad \text{and} \quad \frac{\partial w_0}{\partial x} = \frac{w_0 - w_{-1}}{\Delta x} + 0(\Delta x) \quad \text{Equation 2-29}$$

For a simple one-dimensional system (recalling Equation 2-11 and ignoring the  $y$  and  $z$  spatial components) the heat transfer process can be represented using Equation 2-28 and Equation 2-29. If the temperature is  $T_i^{t_s}$  at the  $i$ 'th mesh point of a uniform mesh then the spatial term can be approximated by the central difference formulae for a grid with regular spacing equal to  $\Delta x$ . At time  $t_s$  the temporal term can be approximated by the forward difference formulae for a time step  $\Delta t$ , giving the resulting equation;

$$\frac{T_i^{t_s+1} - T_i^{t_s}}{\Delta t} = \alpha \frac{T_{i+1}^{t_s} + T_{i-1}^{t_s} - 2T_i^{t_s}}{\Delta x^2} \quad \text{Equation 2-30}$$

This is known as the Euler method and is useful in computational applications, or even hand calculations in simple situations. The temporal term only being first order accurate makes this form unstable<sup>28</sup> for time steps larger than the limit  $\Delta t = \frac{\Delta x^2}{2\alpha}$ . A complex model will require a fine mesh to accurately resolve details and this condition of stability dictates the use of a small time-step for meshes with a small spacing. This can make the Euler method an unwieldy solution for simulating long time periods.

The Crank-Nicolson method solves this stability problem by approximating the equation at mesh point  $i$  midway between the old and new time steps. If the change in each value with time is assumed to be linear then the midway time step values can be found by taking the average of the current values and those predicted by the Euler method. This is expressed mathematically as;

$$\frac{T_i^{t_s+1} - T_i^{t_s}}{\Delta t} = \frac{\alpha}{2} \left[ \frac{T_{i+1}^{t_s} + T_{i-1}^{t_s} - 2T_i^{t_s}}{\Delta x^2} + \frac{T_{i+1}^{t_s+1} + T_{i-1}^{t_s+1} - 2T_i^{t_s+1}}{\Delta x^2} \right] \quad \text{Equation 2-31}$$

This equation involves the already predicted previous time step values ( $t_s$ ) and the unknown time step values ( $t_s + 1$ ). When the full range of  $i$  is considered this equation forms a series of simultaneous equations that can be translated into a tri-diagonal matrix. These kinds of matrix problems are well understood and can be easily and efficiently solved using highly refined algorithms.

Unfortunately a problem arises when more than one dimension is considered using the Crank-Nicolson method. Additional terms enter into the solution matrix outside of the tri-diagonal structure making the matrix considerably more difficult to solve. At this point it becomes more efficient to obtain the solution by the Gauss-Seidel iterative method than by matrix elimination.

Separating the terms of Equation 2-31 and substituting for  $\lambda = \frac{\alpha \Delta t}{2\Delta x^2}$  gives;

$$T_i^{t_s+1} = \frac{\lambda}{1 + 2\lambda} T_{i+1}^{t_s+1} + \frac{\lambda}{1 + 2\lambda} T_{i-1}^{t_s+1} + \frac{\lambda}{1 + 2\lambda} T_{i+1}^{t_s} + \frac{1 - 2\lambda}{1 + 2\lambda} T_i^{t_s} + \frac{\lambda}{1 + 2\lambda} T_{i-1}^{t_s} \quad \text{Equation 2-32}$$

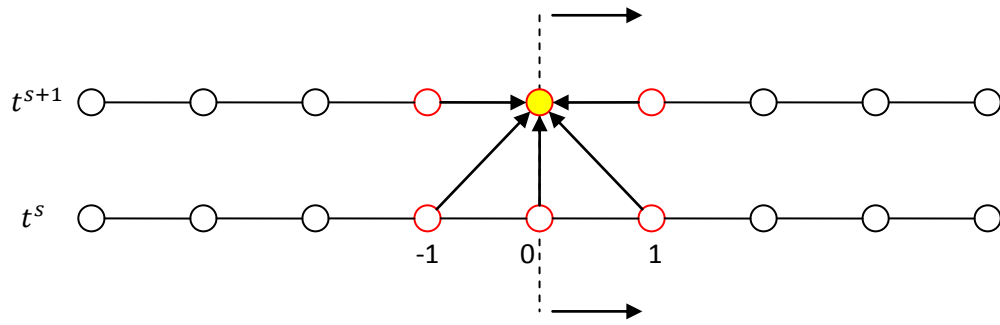


Figure 2-36: Illustration of Gauss-Seidel iterative approach

The new time step is created as a direct copy of the current time step and Equation 2-32 is used to update the new values for the full range of  $i$ . If values are updated from left to right as shown above in Figure 2-36 then the left  $T^{t_s+1}$  value will have been updated more recently than the right  $T^{t_s+1}$  value, implying an error in the calculation. Therefore calculations need to be repeated until the error becomes vanishingly small. This can take a large number of iterative cycles so it is wise to impose a level of error deemed acceptable ( $1 \times 10^{-10}$  degrees was used throughout this project). The error is calculated by taking the difference in the values of  $t^{s+1}$  between iterations. When further repetition yields very little difference in the predicted values the solutions (real and predicted) can be considered to have ‘converged’.

#### 2.4.4. Application of Finite Difference to the Core Model

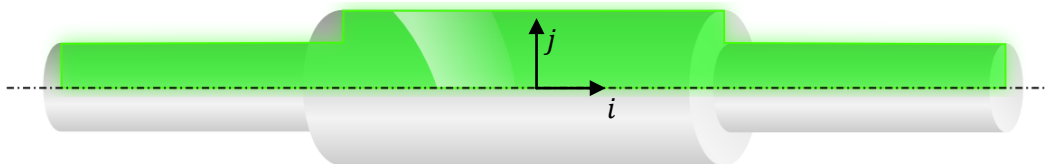


Figure 2-37:  $i$  and  $j$  directions in the Core model

Equation 2-21 gives the equation for heat flow in a small control volume in the Core model. Using the difference formulae derived in the previous section, applying the Crank-Nicolson method and using  $K$  as the time-step index and  $i$  and  $j$  as the coordinate indices as defined in Figure 2-37 this can be transformed into a discretised version.

$$\frac{T_{i,j}^{K+1} - T_{i,j}^K}{\Delta t} = \frac{\alpha}{2} \left( \frac{T_{i,j+1} + T_{i,j-1} - 2T_{i,j}}{\Delta r^2} + \frac{T_{i,j+1} - T_{i,j-1}}{2j\Delta r^2} + \frac{T_{i+1,j} + T_{i-1,j} - 2T_{i,j}}{\Delta z^2} \right)^{K+1} + \frac{\alpha}{2} \left( \frac{T_{i,j+1} + T_{i,j-1} - 2T_{i,j}}{\Delta r^2} + \frac{T_{i,j+1} - T_{i,j-1}}{2j\Delta r^2} + \frac{T_{i+1,j} + T_{i-1,j} - 2T_{i,j}}{\Delta z^2} \right)^K$$

Equation 2-33

Where  $r$  (the radial distance of the point from the roll centre line) can be expressed as the radial location index multiplied by the mesh spacing in the radial direction, or  $j\Delta r$ . This equation can be converted into a suitable form for Gauss-Seidel iteration by rearranging to combine the  $T$  terms and using  $\gamma_0 = \frac{\Delta r^2}{\Delta z^2}$ ,  $\lambda_0 = \frac{\Delta r^2}{\alpha \Delta t}$ ,  $j_{0a} = \frac{2j+1}{4j}$  and  $j_{0b} = \frac{2j-1}{4j}$ .

$$T_{i,j}^{K+1} = AT_{i,j+1}^{K+1} + BT_{i,j-1}^{K+1} + CT_{i+1,j}^{K+1} + DT_{i-1,j}^{K+1} + AT_{i,j+1}^K + BT_{i,j-1}^K + CT_{i+1,j}^K + DT_{i-1,j}^K + ET_{i,j}^K + F$$

Equation 2-34

Where;

$$A = \frac{j_{0a}}{1 + \gamma_0 + \lambda_0} \quad B = \frac{j_{0b}}{1 + \gamma_0 + \lambda_0} \quad C = \frac{\gamma_0}{2(1 + \gamma_0 + \lambda_0)}$$

$$D = \frac{\gamma_0}{2(1 + \gamma_0 + \lambda_0)} \quad E = \frac{\lambda_0 - 1 - \gamma_0}{1 + \gamma_0 + \lambda_0} \quad F = 0$$

Equation 2-35

These substitutions not only make a very long equation more legible but also make composing the software solution more straightforward. For each of the internal model points Equation 2-34 can be used to update the value until a stable result is achieved. For points around the edges of the model boundary conditions need to be applied.

### Reduced point boundary conditions

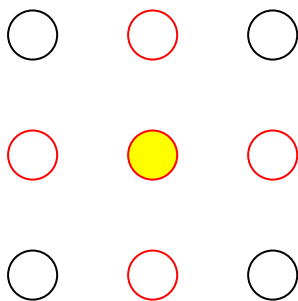


Figure 2-38: 5-point Crank-Nicolson setup

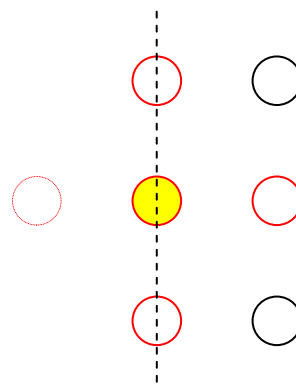


Figure 2-39: Reduced point boundary condition

Figure 2-38 represents any area of the core model not near a boundary. The yellow point is the point whose value is being calculated and the red ringed points are all those whose value is used to update the yellow point. Figure 2-39 represents an area of the core model on a left-facing edge. The situation is the same as in Figure 2-38 except that the left-most red ringed point could be described as "virtual", since it is not inside the model geometry. No values exist outside of the core model boundaries so Equation 2-34 cannot be used. The exact solution to this problem will vary depending on the boundary in question but the following approach is always taken;

1. Mathematically define the boundary condition
2. Rearrange the relationship to make the value at the virtual point the subject
3. Substitute the relationship into Equation 2-33 to replace the value of the virtual point
4. Rearrange as necessary to give an equation in the same format as Equation 2-34

Rearranging every equation into the same format as Equation 2-34 is very useful from a programming perspective because it means only one equation needs to be written into the solution algorithm. Otherwise multiple equations would need to be defined along with a method of choosing which equation to use at any point. The process defined above can be applied to each boundary in turn until every boundary has been considered and the mathematical treatment of the Core model is complete. The differences between boundary and non boundary points are given by differences in the coefficients of Equation 2-34.

### *Centre-plane symmetry face*

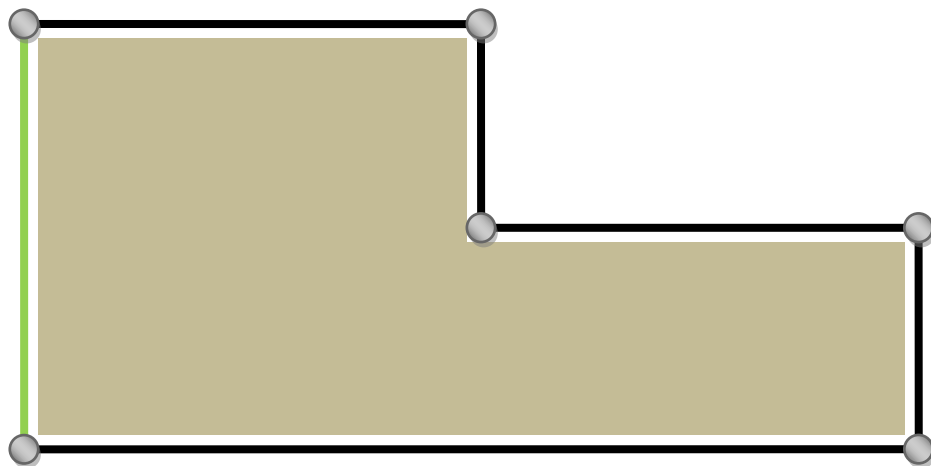


Figure 2-40: The centre-plane symmetry face

The assumption that allows the collapse to half a work roll relies on the fact that the thermal load either side of the centre plane will be identical. If this is the case then material either side of the centre plane should be the same temperature. Since heat flow is driven by temperature difference this means there should be no heat flow across the centre plane. Mathematically this can be expressed as;

$$\frac{\partial T}{\partial z} = \frac{T_{i+1,j} - T_{i-1,j}}{2\Delta z} = 0 \quad \text{Equation 2-36}$$

Therefore;

$$T_{i-1,j} = T_{i+1,j} \quad \text{Equation 2-37}$$

Equation 2-37 can be substituted into Equation 2-33 to give;

$$\begin{aligned} & \frac{T_{i,j}^{K+1} - T_{i,j}^K}{\Delta t} \\ &= \frac{\alpha}{2} \left( \frac{T_{i,j+1} + T_{i,j-1} - 2T_{i,j}}{\Delta r^2} + \frac{T_{i,j+1} - T_{i,j-1}}{2j\Delta r^2} + \frac{2T_{i+1,j} - 2T_{i,j}}{\Delta z^2} \right)^{K+1} \\ &+ \frac{\alpha}{2} \left( \frac{T_{i,j+1} + T_{i,j-1} - 2T_{i,j}}{\Delta r^2} + \frac{T_{i,j+1} - T_{i,j-1}}{2j\Delta r^2} + \frac{2T_{i+1,j} - 2T_{i,j}}{\Delta z^2} \right)^K \end{aligned} \quad \text{Equation 2-38}$$

Which gives the following coefficients.

$$\begin{aligned} A &= \frac{j_{0a}}{1 + \gamma_0 + \lambda_0} & B &= \frac{j_{0b}}{1 + \gamma_0 + \lambda_0} & C &= \frac{\gamma_0}{1 + \gamma_0 + \lambda_0} \\ D &= 0 & E &= \frac{\lambda_0 - 1 - \gamma_0}{1 + \gamma_0 + \lambda_0} & F &= 0 \end{aligned}$$

Equation 2-39



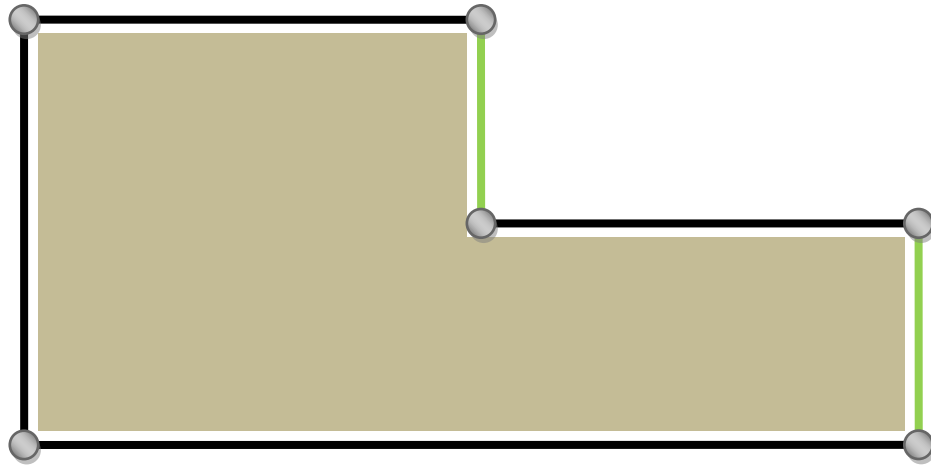


Figure 2-41: Convection boundary conditions

The side faces of the work roll are cooled by exposure to the atmosphere. This heat transfer is driven by the temperature difference between the work roll and the environment;

$$-k \frac{\partial T}{\partial z} = h_{conv}(T_{i,j} - T_f) \quad \text{Equation 2-40}$$

Where  $T_f$  is the atmospheric temperature and  $h_{conv}$  is a heat transfer coefficient which can be used to control the heat transfer. Recalling the difference formulae defined in Equation 2-28 this can be rewritten as;

$$-k \frac{T_{i+1,j} - T_{i-1,j}}{2\Delta z} = h_{conv}(T_{i,j} - T_f) \quad \text{Equation 2-41}$$

The unknown value in this instance is  $T_{i+1,j}$  so rearranging to make this the subject of the equation gives;

$$T_{i+1,j} = T_{i-1,j} - \frac{2h_{conv}\Delta z}{k}(T_{i,j} - T_f) \quad \text{Equation 2-42}$$

Substituting Equation 2-42 into Equation 2-33 gives the modified form of the discretised equation;

$$\begin{aligned}
& \frac{T_{i,j}^{K+1} - T_{i,j}^K}{\Delta t} \\
&= \frac{\alpha}{2} \left( \frac{T_{i,j+1} + T_{i,j-1} - 2T_{i,j}}{\Delta r^2} + \frac{T_{i,j+1} - T_{i,j-1}}{2j\Delta r^2} \right. \\
&\quad \left. + \frac{T_{i-1,j} - \frac{2h_{conv}\Delta Z}{k}(T_{i,j} - T_f) + T_{i-1,j} - 2T_{i,j}}{\Delta Z^2} \right)^{K+1} \\
&\quad + \frac{\alpha}{2} \left( \frac{T_{i,j+1} + T_{i,j-1} - 2T_{i,j}}{\Delta r^2} + \frac{T_{i,j+1} - T_{i,j-1}}{2j\Delta r^2} \right. \\
&\quad \left. + \frac{T_{i-1,j} - \frac{2h_{conv}\Delta Z}{k}(T_{i,j} - T_f) + T_{i-1,j} - 2T_{i,j}}{\Delta Z^2} \right)^K
\end{aligned}$$

**Equation 2-43**

Which can be rearranged into the same form as Equation 2-34 if;

$$\begin{aligned}
A &= \frac{j_0 a}{1 + \gamma_0 + \lambda_0 + \alpha_{bc}} & B &= \frac{j_0 b}{1 + \gamma_0 + \lambda_0 + \alpha_{bc}} & C &= 0 \\
D &= \frac{\gamma_0}{1 + \gamma_0 + \lambda_0 + \alpha_{bc}} & E &= \frac{\lambda_0 - 1 - \gamma_0 - \alpha_{bc}}{1 + \gamma_0 + \lambda_0 + \alpha_{bc}} & F &= \frac{2\alpha_{bc}T_f}{1 + \gamma_0 + \lambda_0 + \alpha_{bc}}
\end{aligned}$$

**Equation 2-44**

where  $\alpha_{bc} = \frac{h_{conv}\Delta Z\gamma_0}{k}$ . It is worth noting that the value of the  $C$  coefficient is zero. Recalling Equation 2-34 this means that the value of  $T_{i,j}$  at this point will not be affected by the value at  $T_{i+1,j}$ . This is a useful outcome as it means no programming logic is required to dictate that the value at  $T_{i+1,j}$  should not be used, the coefficient value ensures this without the need for any additional processing overhead.

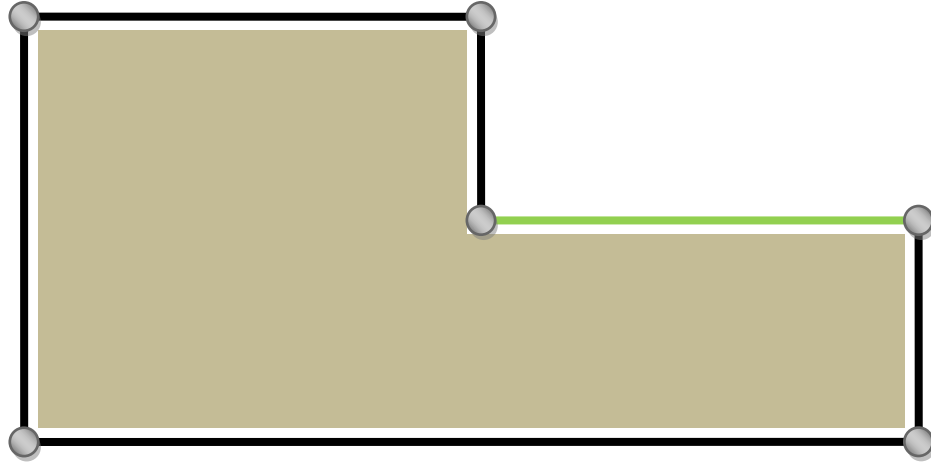


Figure 2-42: Journal surfaces

Following the same process as with the previous boundary condition the value of the virtual point above the journal surface can be shown to be;

$$T_{i,j+1} = T_{i,j-1} - \frac{2h_{ConvB}\Delta r}{k}(T_{i,j} - T_B) \quad \text{Equation 2-45}$$

where  $h_{ConvB}$  is the heat transfer coefficient for the bearing contact area and  $T_B$  is the temperature of the bearing chock. As the temperature magnitudes are not likely to be very high the bearing is assumed to be in contact with the work roll journal across the full length. This is inaccurate but much more straightforward than a more realistic condition would be and will have little impact as the bearings supply only marginal amounts of heat compared to the strip contact. This assumption is based on the fact that the chock temperature is 40 degrees compared to strip temperatures of 900-1100 degrees, and is corroborated by later thermal model results (see the results in Chapter 6, page 258).

Equation 2-45 can be substituted into Equation 2-34 to eliminate  $T_{i,j+1}$  and rearranged to give coefficient values of;

$$\begin{aligned} A &= 0 & B &= \frac{j_{0a} + j_{0b}}{1 + \gamma_0 + \lambda_0 + \alpha_{bc}} & C &= \frac{\gamma_0}{2(1 + \gamma_0 + \lambda_0 + \alpha_{bc})} \\ D &= \frac{\gamma_0}{2(1 + \gamma_0 + \lambda_0 + \alpha_{bc})} & E &= \frac{\lambda_0 - 1 - \gamma_0 - \alpha_{bc}}{1 + \gamma_0 + \lambda_0 + \alpha_{bc}} & F &= \frac{2\alpha_{bc}T_B}{1 + \gamma_0 + \lambda_0 + \alpha_{bc}} \end{aligned} \quad \text{Equation 2-46}$$

$$\text{Where } \alpha_{bc} = \frac{2h_{ConvB}\Delta r j_{0a}}{k}$$

## Barrel surface

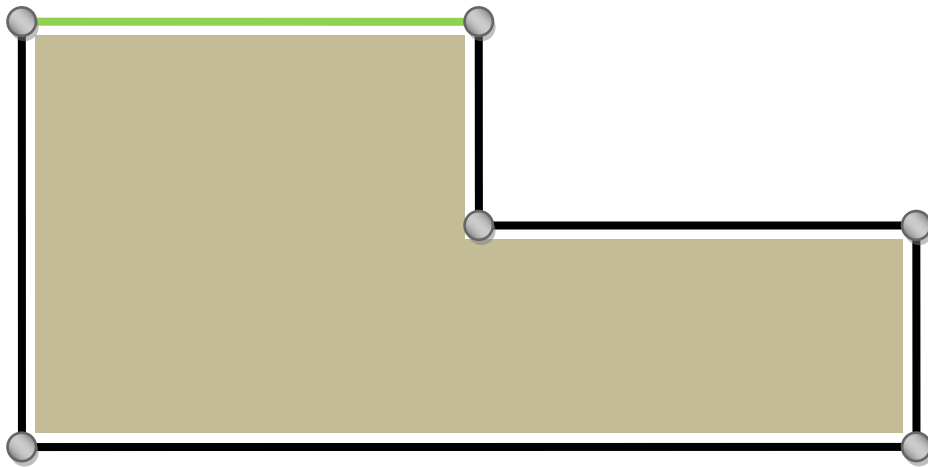


Figure 2-43: The barrel surface

The work roll surface is the most complicated boundary condition, as there are many different effects taking place. In the roll bite heat is being produced by friction and deformation of the strip and this heat input is added to by conduction due to the strip-roll temperature difference. Outside the roll bite the roll surface is kept cool by convection to the ambient environment and liberal use of cooling water.

Of the sources of heating and cooling mentioned in Section 1.1, heat radiation from the strip to the roll and conduction to the backup roll were ignored. Radiation was ignored due to the moving layer of water which coats the surface of the work roll, shielding it from the potential heat gain. Conduction to the backup roll is ignored due to the vast difference in the potential heat flow into the work roll from the strip and the likely heat flow out at the backup roll contact. The results given in Chapter 6 show the surface temperature of the work roll reduced to approximately 120 degrees Centigrade at the backup roll position, giving a temperature difference between the work roll and the backup roll of around 100 degrees, assuming room temperature starting conditions. By contrast, the temperature difference between the work roll surface and the strip is around 900 degrees and the contact length at this position is longer due to the deformation of the strip. A significant frictional heat generation component is also present at this location.

The heat generation components, friction and deformation heat, can be approximated together by use of the mechanical power required to rotate the work rolls. This value is available on the hot mill database and represents the amount of energy being absorbed by the process. There are two rolls so this value must be halved, giving;

$$\dot{q}_g = \beta \frac{P}{2A_c} \quad \text{Equation 2-47}$$

Where  $\dot{q}_g$  represents the rate of heat flow into the work roll per unit area from heat generation components,  $P$  is the mechanical power provided by the rolling stand,  $A_c$  is the contact area and  $\beta$  is a constant representing the proportion of heat flowing into the work roll compared to the strip.

The heat conduction term can be provided by use of an arbitrary heat transfer coefficient in the same manner as previous boundary conditions have been solved;

$$\dot{q}_{cond} = \frac{\theta_b}{2\pi} h_{str}(T_{str} - T_{i,j}) \quad \text{Equation 2-48}$$

Where  $\dot{q}_{cond}$  is the rate of heat flow into the work roll per unit area conducted from the strip,  $\theta_b$  is the angle of the contact arc,  $h_{str}$  is the heat transfer coefficient and  $T_{str}$  is the strip temperature. The contact arc angle is required because only a small portion of the work roll will be subject to this conduction. The necessary value of the heat transfer coefficient was found by comparison of predicted temperatures to measured values.

The effect of the cooling sprays can be calculated in the same way;

$$\dot{q}_{conv} = \frac{2\pi - \theta_b}{2\pi} h_{spr}(T_{i,j} - T_{spr}) \quad \text{Equation 2-49}$$

In this case  $\dot{q}_{conv}$  is the rate of heat flow out of the work roll per unit area via convection,  $h_{spr}$  is the heat transfer coefficient for the cooling sprays and  $T_{spr}$  is the temperature of the cooling water.

The complete boundary condition can be represented as the sum of all the influences described above.

$$k \frac{dT}{dr} = \dot{q}_g + \dot{q}_{cond} + \dot{q}_{conv} \quad \text{Equation 2-50}$$

Equation 2-50 expands to give;

$$k \frac{dT}{dr} = \beta \frac{P}{2A_c} + \frac{\theta_b}{2\pi} h_{str}(T_{str} - T_{i,j}) - \frac{2\pi - \theta_b}{2\pi} h_{spr}(T_{i,j} - T_{spr}) \quad \text{Equation 2-51}$$

Which can be simplified to;

$$k \frac{dT}{dr} = H_0 + H_1 T_{i,j} + H_2 T_{str} + H_3 T_{spr} \quad \text{Equation 2-52}$$

Given;

$$H_0 = \beta \frac{P}{2A_c} \quad \text{Equation 2-53}$$

$$H_1 = \frac{-\theta_b}{2\pi} h_{str} - \left( \frac{2\pi - \theta_b}{2\pi} \right) h_{spr} \quad \text{Equation 2-54}$$

$$H_2 = \frac{\theta_b}{2\pi} h_{str} \quad \text{Equation 2-55}$$

$$H_3 = \left( \frac{2\pi - \theta_b}{2\pi} \right) h_{spr} \quad \text{Equation 2-56}$$

Substituting in the finite difference approximation for the derivative term;

$$k \left( \frac{T_{i,j+1} - T_{i,j-1}}{2\Delta r} \right) = H_0 + H_1 T_{i,j} + H_2 T_{str} + H_3 T_{spr} \quad \text{Equation 2-57}$$

Rearranging and collecting all constants gives;

$$T_{i,j+1} = G_0 + G_1 T_{i,j} + T_{i,j-1} \quad \text{Equation 2-58}$$

Where;

$$G_0 = \frac{2\Delta r}{k} (H_0 + H_2 T_{str} + H_3 T_{spr}) \quad \text{Equation 2-59}$$

$$G_1 = \frac{2\Delta r}{k} H_1 \quad \text{Equation 2-60}$$

Substituting this into Equation 2-34 results in a new set of coefficients.

$$\begin{aligned} A &= 0 & B &= \frac{j_{0a} + j_{0b}}{1 + \gamma_0 + \lambda_0 + \alpha_{bc}} & C &= \frac{\gamma_0}{2(1 + \gamma_0 + \lambda_0 + \alpha_{bc})} \\ D &= \frac{\gamma_0}{2(1 + \gamma_0 + \lambda_0 + \alpha_{bc})} & E &= \frac{\lambda_0 - 1 - \gamma_0 - \alpha_{bc}}{1 + \gamma_0 + \lambda_0 + \alpha_{bc}} & F &= \frac{2j_{0a}G_0}{1 + \gamma_0 + \lambda_0 + \alpha_{bc}} \end{aligned}$$

Where  $\alpha_{bc} = j_{0a}G_1$

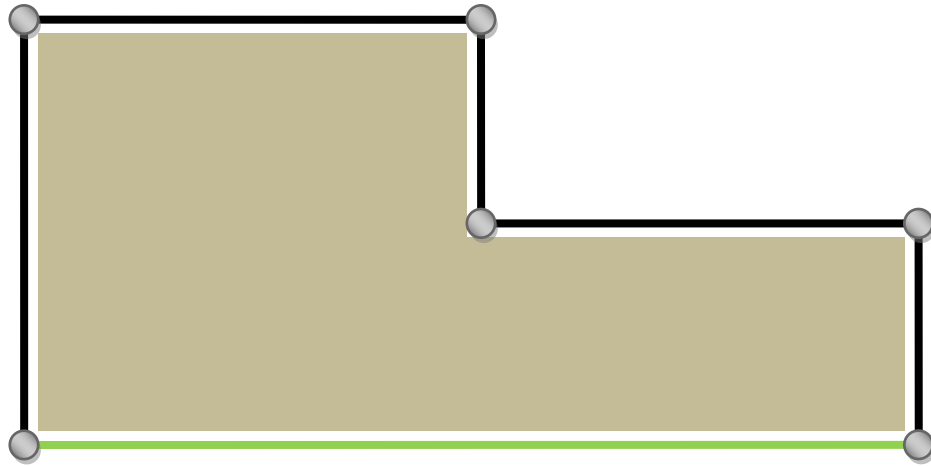


Figure 2-44: The axisymmetric symmetry face

The symmetry condition for the rotational axis is complicated by the fact that as  $r$  tends to zero some components of Equation 2-34 become singular. For this reason the equation for the centre line must be constructed from a separate energy balance.

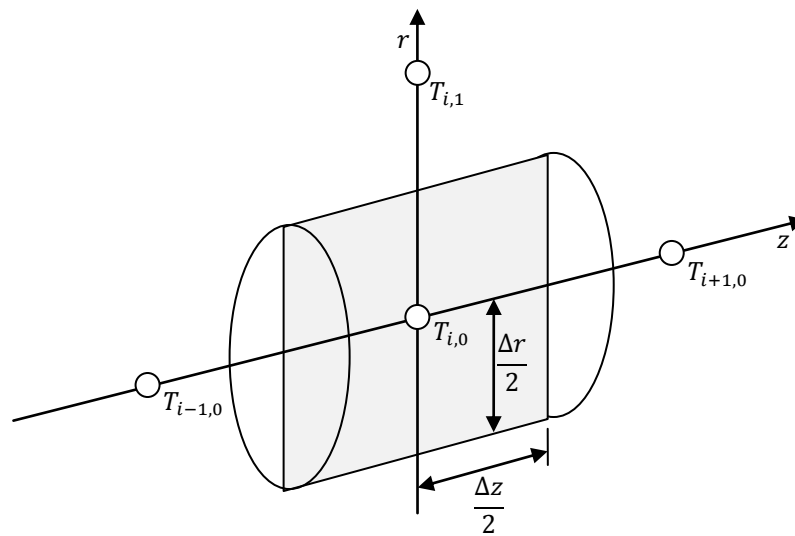


Figure 2-45: Example diagram of cylindrical centre-line element

Figure 2-45 illustrates the element used to provide the new equation. The outer surface area of this element is  $2\pi\Delta z \Delta r/2 = \pi\Delta z\Delta r$ , so the heat flux into the volume through the outer surface is:

$$\dot{Q}_r = \pi\Delta z\Delta r k \frac{\partial T}{\partial r} \Big|_{j=\frac{1}{2}} \quad \text{Equation 2-61}$$

The heat flux in the z-direction is;

$$\dot{Q}_z = \pi \frac{\Delta r^2}{4} k \left( \frac{\partial T}{\partial z_{i+\frac{1}{2}}} - \frac{\partial T}{\partial z_{i-\frac{1}{2}}} \right) \quad \text{Equation 2-62}$$

Therefore the net heat flow will be given by;

$$d\dot{Q}_{Net} = \pi \frac{\Delta r^2}{4} k \left( \frac{\partial T}{\partial z_{i+\frac{1}{2}}} - \frac{\partial T}{\partial z_{i-\frac{1}{2}}} \right) + \pi \Delta z \Delta r k \frac{\partial T}{\partial r}_{j=\frac{1}{2}} \quad \text{Equation 2-63}$$

Recalling Equation 2-7 and Equation 2-9 (Page 39), where  $V = \pi \frac{\Delta r^2}{4} \Delta z$ , Equation 2-63 becomes;

$$\pi \frac{\Delta r^2}{4} \Delta z \rho C_p \frac{\partial T}{\partial t} = \pi \frac{\Delta r^2}{4} k \left( \frac{\partial T}{\partial z_{i+\frac{1}{2}}} - \frac{\partial T}{\partial z_{i-\frac{1}{2}}} \right) + \pi \Delta z \Delta r k \frac{\partial T}{\partial r}_{j=\frac{1}{2}} \quad \text{Equation 2-64}$$

Substituting in finite difference terms and applying the Crank-Nicholson method;

$$\begin{aligned} & \pi \frac{\Delta r^2}{4} \Delta z \rho C_p \frac{T_{i,0}^{K+1} - T_{i,0}^K}{\Delta t} \\ &= \frac{1}{2} \left[ \pi \frac{\Delta r^2}{4} k \left( \frac{T_{i+1,0}^{K+1} - T_{i,0}^{K+1}}{\Delta z} - \frac{T_{i,0}^{K+1} - T_{i-1,0}^{K+1}}{\Delta z} \right) + \pi \Delta z \Delta r k \frac{T_{i,1}^{K+1} - T_{i,0}^{K+1}}{\Delta r} \right] \\ &+ \frac{1}{2} \left[ \pi \frac{\Delta r^2}{4} k \left( \frac{T_{i+1,0}^K - T_{i,0}^K}{\Delta z} - \frac{T_{i,0}^K - T_{i-1,0}^K}{\Delta z} \right) + \pi \Delta z \Delta r k \frac{T_{i,1}^K - T_{i,0}^K}{\Delta r} \right] \end{aligned} \quad \text{Equation 2-65}$$

Expanding, rearranging and simplifying gives;

$$\begin{aligned} (1 + \gamma_1 + \lambda_1) T_{i,0}^{K+1} &= \frac{\gamma_1}{2} T_{i+1,0}^{K+1} + \frac{\gamma_1}{2} T_{i-1,0}^{K+1} + T_{i,1}^{K+1} \\ &+ \frac{\gamma_1}{2} T_{i+1,0}^K + \frac{\gamma_1}{2} T_{i-1,0}^K + T_{i,1}^K - (1 + \gamma_1 - \lambda_1) T_{i,0}^K \end{aligned} \quad \text{Equation 2-66}$$

With the coefficients;

$$\gamma_1 = \frac{\Delta r^2}{2\Delta z^2} \quad \lambda_1 = \frac{\Delta r^2 \rho C_p}{2k\Delta t} \quad \text{Equation 2-67}$$

Minor rearrangements put Equation 2-66 in the standard form of Equation 2-34, giving;



$$\begin{aligned}
 A &= \frac{1}{1 + \gamma_1 + \lambda_1} & B &= 0 & C &= \frac{\gamma_1}{1 + \gamma_1 + \lambda_1} \\
 D &= \frac{\gamma_1}{2(1 + \gamma_1 + \lambda_1)} & E &= \frac{\lambda_1 - 1 - \gamma_1}{1 + \gamma_1 + \lambda_1} & F &= 0
 \end{aligned}$$

Equation 2-68

### Compound boundary conditions

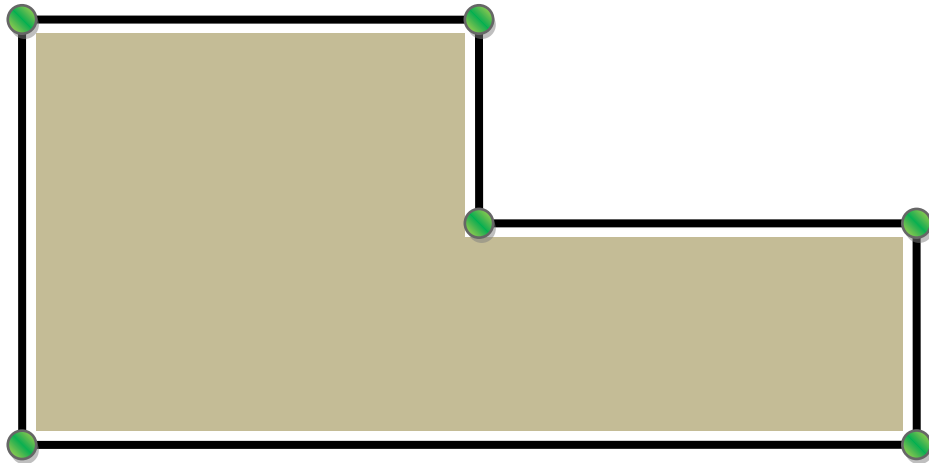


Figure 2-46: The compound boundary conditions

The only areas of the model not fully described by the previously derived equations are those points which lie on more than one boundary condition. These will not be repeated as they are formed simply by applying the substitutions described earlier in concert and therefore add little to the discussion. Further detail can be found in White's thesis<sup>3</sup>.

### 2.4.5. Application of Finite Difference to the Shell Model

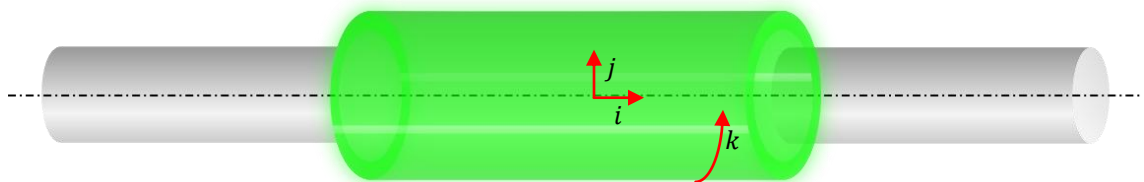


Figure 2-47: i, j and k directions in the shell model

Equation 2-69 gives the heat flow equation for a small control volume in the shell model.

$$0 = \alpha \left( \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial r^2} + \frac{1}{r^2} \frac{\partial^2 T}{\partial \theta^2} + \frac{\partial^2 T}{\partial z^2} \right) - \omega \frac{\partial T}{\partial \theta}$$

Equation 2-69

Using the finite difference approximations developed in earlier parts of this chapter and the dimensional sense shown in Figure 2-47, Equation 2-69 can be expanded to its point-wise form.

$$0 = \alpha \left( \frac{1}{r} \frac{T_{i,j+1,k} - T_{i,j-1,k}}{2\Delta r} + \frac{T_{i,j+1,k} + T_{i,j-1,k} - 2T_{i,j,k}}{\Delta r^2} + \frac{1}{r^2} \frac{T_{i,j,k+1} + T_{i,j,k-1} - 2T_{i,j,k}}{\Delta \theta^2} + \frac{T_{i+1,j,k} + T_{i-1,j,k} - 2T_{i,j,k}}{\Delta z^2} \right) - \omega \frac{T_{i,j,k} - T_{i,j,k-1}}{\Delta \theta}$$

Equation 2-70

A backward difference is used for the convection term because the approximation had to replace a first order differential. The central difference of a first order differential would have been  $\frac{T_{i,j,k+1} - T_{i,j,k-1}}{2\Delta \theta}$ , which does not contain a  $T_{i,j,k}$  term, making it numerically unstable. Equation 2-70 can be arranged into a standard form similar to the standard form of the core model, with the same benefit of allowing the coefficients to be calculated outside of the iterative process.

$$T_{i,j,k} = AT_{i+1,j,k} + BT_{i-1,j,k} + CT_{i,j+1,k} + DT_{i,j-1,k} + ET_{i,j,k+1} + FT_{i,j,k-1} + G$$

Equation 2-71

Where

$$\begin{aligned} A &= \frac{\gamma}{2 + 2\gamma + \lambda + 2j_c} & B &= \frac{\gamma}{2 + 2\gamma + \lambda + 2j_c} & C &= \frac{j_a}{2 + 2\gamma + \lambda + 2j_c} \\ D &= \frac{j_b}{2 + 2\gamma + \lambda + 2j_c} & E &= \frac{j_c}{2 + 2\gamma + \lambda + 2j_c} & F &= \frac{\lambda + j_c}{2 + 2\gamma + \lambda + 2j_c} \\ G &= 0 \end{aligned}$$

Equation 2-72

And

$$\begin{aligned} \gamma &= \frac{\Delta r^2}{\Delta z^2} & \lambda &= \frac{\omega \Delta r^2}{\alpha \Delta \theta} & j_a &= 1 + \frac{1}{2j_{offset} + 2j} \\ j_b &= 1 - \frac{1}{2j_{offset} + 2j} & j_c &= \frac{1}{(j_{offset} + j)^2 \Delta \theta^2} \end{aligned}$$

Equation 2-73

Equation 2-71 describes the heat flow within the volume of the shell model. As with the core model, special consideration is needed to emulate the thermal conditions at the volume boundaries.

## Surface

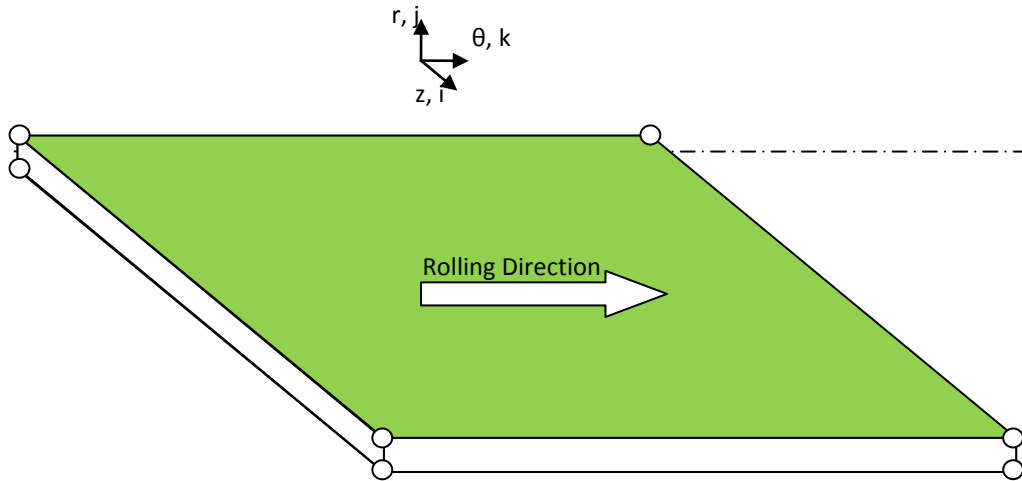


Figure 2-48: Shell model top surface

Figure 2-48 highlights the top surface of the shell model, which is displayed "unwrapped", since this rectangular form reflects the three component coordinate scheme  $(i, j, k)$  used by the software. Each dimension refers to one component of a cylindrical polar coordinate system  $(r, z, \theta)$ , so the cylindrical behaviour of the shape is preserved.

The substitution used by the core model to mimic the thermal conditions at the work roll surface (Equation 2-51) can be expanded to allow angular variability.

$$T_{i,j+1,k} = T_{i,j-1,k} + \frac{2\Delta r}{k} \dot{q}_{friction}(\theta) - \frac{2\Delta r h_{str}}{k} (T_{i,j,k} - T_{str[i,k]}) - \frac{2\Delta r h_{spr[i,k]}}{k} (T_{i,j,k} - T_{spr})$$

Equation 2-74

In Equation 2-74,  $T_{str[i,k]}$  and  $h_{spr[i,k]}$  are two dimensional arrays containing the temperature and heat transfer coefficient maps, respectively, that control heat flow to and from the roll surface. Substituting Equation 2-74 into Equation 2-70 and rearranging gives an equation for the surface boundary.

$$\begin{aligned}
& \left( 2 + 2\gamma + \lambda + 2j_c + \frac{2\Delta r j_a}{k} (h_{str} + h_{spr[i,k]}) \right) T_{i,j,k} \\
& = \gamma T_{i+1,j,k} + \gamma T_{i-1,j,k} + j_a T_{i,j+1,k} + j_b T_{i,j-1,k} + j_c T_{i,j,k+1} \\
& + (\lambda + j_c) T_{i,j,k-1} - \frac{\Delta r^2}{\alpha} T' \\
& + \frac{2\Delta r j_a}{k} (\dot{q}_{friction}(\theta) + h_{str} T_{str[i,k]} + h_{spr[i,k]} T_{spr})
\end{aligned} \tag{Equation 2-75}$$

Equation 2-75 can then be used to define the new coefficient definitions for Equation 2-71.

$$\begin{aligned}
A &= \frac{\gamma}{2 + 2\gamma + \lambda + 2j_c + j_a \frac{2\Delta r}{k} (h_{str} + h_{spr[i,k]})} \\
B &= \frac{\gamma}{2 + 2\gamma + \lambda + 2j_c + j_a \frac{2\Delta r}{k} (h_{str} + h_{spr[i,k]})} \\
C &= 0 \\
D &= \frac{j_a + j_b}{2 + 2\gamma + \lambda + 2j_c + j_a \frac{2\Delta r}{k} (h_{str} + h_{spr[i,k]})} \\
E &= \frac{j_c}{2 + 2\gamma + \lambda + 2j_c + j_a \frac{2\Delta r}{k} (h_{str} + h_{spr[i,k]})} \\
F &= \frac{\lambda + j_c}{2 + 2\gamma + \lambda + 2j_c + j_a \frac{2\Delta r}{k} (h_{str} + h_{spr[i,k]})} \\
G &= \frac{-\frac{\Delta r^2}{\alpha} T' + j_a \frac{2\Delta r}{k} (\dot{q}_{friction}(\theta) + h_{str} T_{str[i,k]} + h_{spr[i,k]} T_{spr})}{2 + 2\gamma + \lambda + 2j_c + j_a \frac{2\Delta r}{k} (h_{str} + h_{spr[i,k]})}
\end{aligned} \tag{Equation 2-76}$$

## Contact faces

---

Figure 2-49 gives an illustration of the two faces created by "unwrapping" the shell model. In its original cylindrical form these two faces would be in perfect contact, with heat flowing between the two as it would between any other two internal points. To imitate this behaviour the temperature arrays are extended by one position at each end of the  $\theta$  axis. Before each iteration the values at the  $\theta_{max}$  position are copied to the  $0 - \Delta\theta$  position and the values at the 0 position are copied to the  $\theta_{max} + \Delta\theta$  position. This approach removes the need for a boundary condition at these locations, the general equation references valid temperatures within the model volume and the values at the new positions do not need to be calculated by the solver.

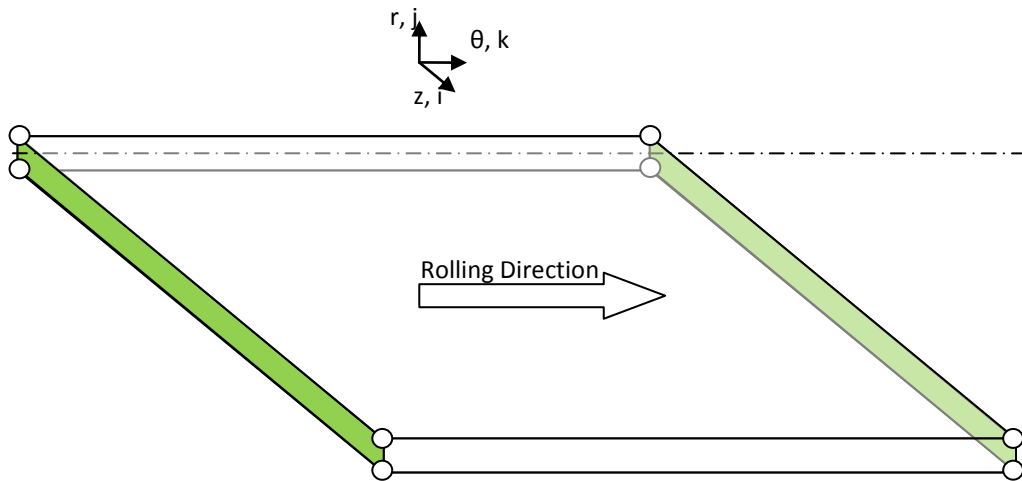


Figure 2-49: Shell model with the contact faces highlighted

### Symmetry plane

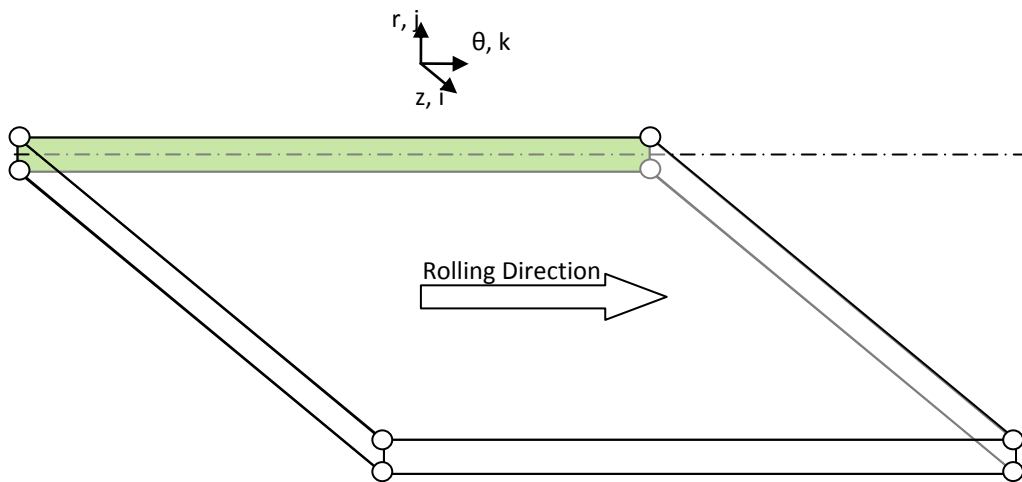


Figure 2-50: Shell model symmetry plane

The symmetry plane assumption for the shell model is also the symmetry assumption for the core model, that the temperatures either side of the boundary will be identical so heat flow will be zero.

$$\frac{\partial T}{\partial z} = \frac{T_{i+1,j,k} - T_{i-1,j,k}}{2\Delta z} = 0$$

Rearranging this equation gives  $T_{i-1,j,k} = T_{i+1,j,k}$ . Substituting into Equation 2-71 gives;

$$A = \frac{2\gamma}{2 + 2\gamma + \lambda + 2j_c} \quad B = 0 \quad \text{Equation 2-77}$$

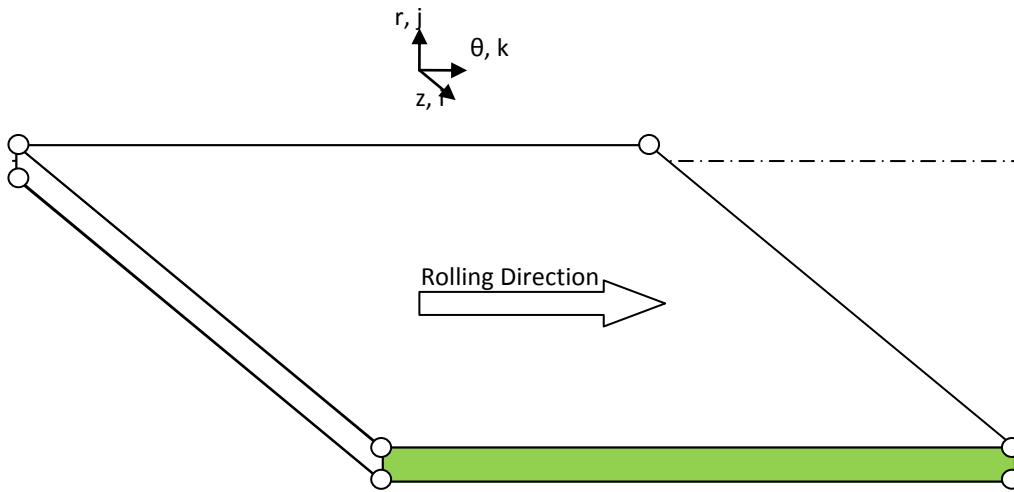


Figure 2-51: Shell model convection face

The face highlighted in Figure 2-51 is open to the ambient environment, experiencing the same temperature difference dependent heat flow as given by Equation 2-42. The equation needs to be adapted to account for the extra dimension.

$$T_{i+1,j,k} = T_{i-1,j,k} - \frac{2h_{conv}\Delta z}{k} (T_{i,j,k} - T_{ambient}) \quad \text{Equation 2-78}$$

Substituting Equation 2-78 into Equation 2-71 gives;

$$A = 0$$

$$B = \frac{2\gamma}{2 + 2\gamma + \lambda + 2j_c + \frac{2h_{conva}\Delta z\gamma}{k}}$$

$$D = \frac{j_b}{2 + 2\gamma + \lambda + 2j_c + \frac{2h_{conva}\Delta z\gamma}{k}}$$

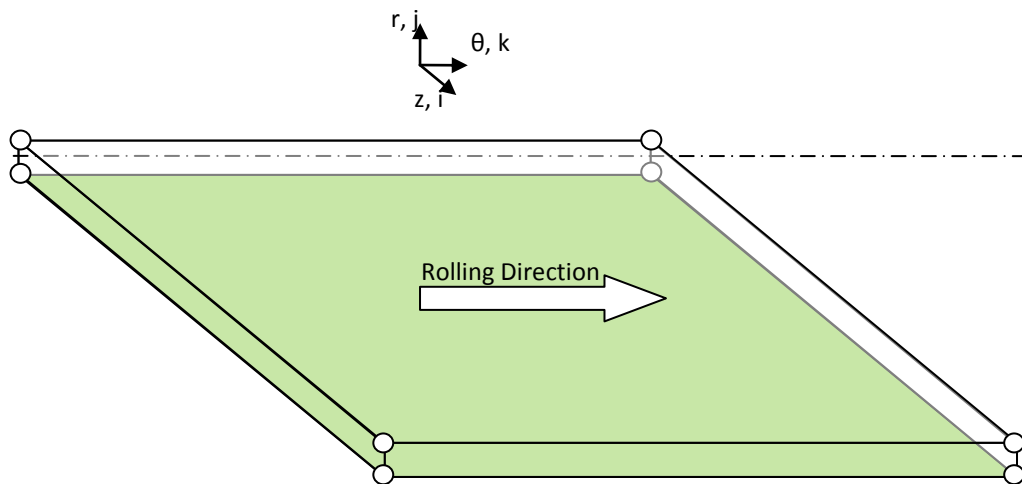
$$F = \frac{\lambda + j_c}{2 + 2\gamma + \lambda + 2j_c + \frac{2h_{conva}\Delta z\gamma}{k}}$$

$$C = \frac{j_a}{2 + 2\gamma + \lambda + 2j_c + \frac{2h_{conva}\Delta z\gamma}{k}}$$

$$E = \frac{j_c}{2 + 2\gamma + \lambda + 2j_c + \frac{2h_{conva}\Delta z\gamma}{k}}$$

$$G = \frac{-\frac{\Delta r^2}{\alpha} T' + \frac{2h_{conva}\Delta z\gamma}{k} T_{ambient}}{2 + 2\gamma + \lambda + 2j_c + \frac{2h_{conva}\Delta z\gamma}{k}}$$

Equation 2-79



The core model temperatures are used to fill the entire bottom plane of the model, since at this depth there should be no significant angular variation. These values do not need to be updated and act as a direct temperature substitution similar to the symmetry condition. As such no adjustments need to be made to the general equation, the core temperature values are accessed as if they were any other value in the model volume.

## 2.5. Summary of the Literature

---

The first consideration in reviewing the literature must be the place of this project in the current development of this area of science. As the project is part funded by Tata Steel this consideration must include the project's place in the expansion of the industrial sponsor's technology base. While many work roll temperature prediction models have been introduced [10-14, 16- 22], very few go on to consider the stresses implied. Those that do consider the thermal stresses do so in isolation and with simplified boundary conditions [18, 20] or in two dimensions only [10, 14, 16, 17, 19, 21, 22]. Within Tata Steel there are different models of temperature prediction but only one model capable of stress prediction and this is not applicable to strip rolling. What is lacking both in the literature and in the Tata Steel modelling suite is a full description of stresses in work rolls.

The second consideration is the method of providing the deliverables required by the project objectives. From the literature it is clear that there are numerous ways of approaching the problem of simulating work roll thermal behaviour. Each approach has its strengths and weaknesses and the best approach for any application will depend on how the priorities ascribed to any prospective model compare to those strengths and weaknesses.

- Analytical solutions are prohibitively complicated in terms of providing a solution to the exact problem but can produce results very quickly if adequately simplified. This makes them very useful for bulk-behaviour analysis and in-line implementation, but less useful for detailed study.
- Finite Element solutions require complicated software development or the use of commercial software, but can simulate the thermal and mechanical responses in a single model. The convenience of containing all the modelling tasks in one package and making use of professionally optimised and validated ready-made software is often countered by the cost and prescriptive nature of the software.
- The finite difference technique can provide solutions to multi-dimensional problems without an enormous rise in complexity associated with complex boundary conditions. Solvers are generally relatively easy to implement which means that in-house software can be developed, giving the researcher the flexibility to change any part of the solution, data acquisition, data manipulation or boundary condition implementation processes as the need arises. However, the finite difference method is unsuited to stress analysis, so can only be used as a part of the total solution.

The only method of solution capable of solving both the thermal and stress problems simultaneously, or at least within one package, is a finite element solution. However, a finite difference thermal solution already exists in the shape of the White model and there is no reason why a finite element thermal model should be superior in terms of speed or quality. An analytical solution would patently not be superior, if only due to the number of assumptions required to render such a solution viable. Given the previous success of the White model in imitating the Port Talbot hot mill, it makes more sense to continue development of the thermal solution with finite difference methods, rather than wipe the slate clean for little or no added benefit.

Finite difference methods are not suited to solving three dimensional stress problems and an analytical solution does not yet exist, either within Tata or the academic community. With the state of current measurement technology and financial considerations an analytical solution could only be developed through the insights garnered from a finite element model. Given these factors finite element modelling is the only salient technique. An added benefit to this approach is the previous project's initial investigations which act as an introduction to Abaqus, a widely used and respected commercial finite element software package. The only potential alternative would be the development of specialised finite element software. Time constraints make this alternative untenable.



To conclude, a review of the available literature indicates that the best path towards achieving the project goals would comprise further development of the finite difference thermal model as inherited from the White model and the development of a finite element stress model using the Abaqus software package.

## 3. The Thermal Model

---

### 3.1. Introduction

---

At the conclusion of the literature review it had been established that the White Model presented a valuable, ready-made and tested thermal solution. That element of the model was introduced in the literature review. Section 3.2 expands on this introduction to outline the process flows for the core and shell models, along with the other program elements required for those two models to function. The author of the model and the project supervisors also made some recommendations for improvements that would increase the utility of the model. Section 3.3 discusses these improvements. During early work with the finite element software discussed in Chapter 4, several difficulties were discovered in the thermal models. Section 3.4 describes the difficulties discovered and the measures required to resolve them.

In September 2010 a visit to the IJmuiden RD&T department yielded a set of recommendations for improvements to the thermal models. While it was not practical to implement them all, they are all introduced in section 3.5, in detail where appropriate. Finally, in October of 2011, a second visit to IJmuiden resulted in a second set of recommendations, primarily involving a change to the boundary conditions of the thermal models. The implementation of the new boundary conditions, and associated updates to the cooling boundary conditions, uncovered a lack of energy conservation when temperature dependent material properties were used. The updates to the boundary conditions and the body of work which followed is presented in section 3.6.

NB – All of the results in this chapter were generated using the data sets given in Appendix D.1 or Appendix D.2, two data sets that are identical in all but length and reduced to a very small size to allow quick and easy debugging.

### 3.2. Process Outline

---

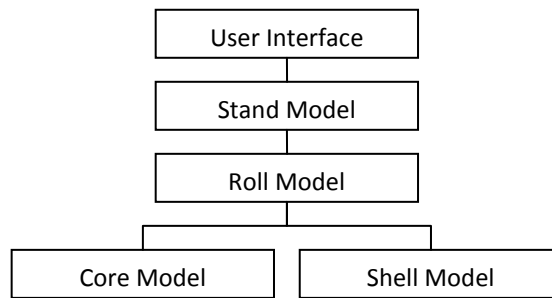


Figure 3-1: Model hierarchy

While discussion of the C# coding of the thermal models and surrounding architecture is consigned to Appendix B, understanding the structure of the software is important for the following chapter.

Figure 3-1 shows the hierarchy of the components used in the thermal analysis. The top component is the user interface, which controls the configuration of and data used by the thermal models through direct user interaction. Below the user interface, the stand model is responsible for dividing the input data into stages of analysis. The roll model works with a single stage of data, coordinating the core and shell models, which perform the actual analysis.

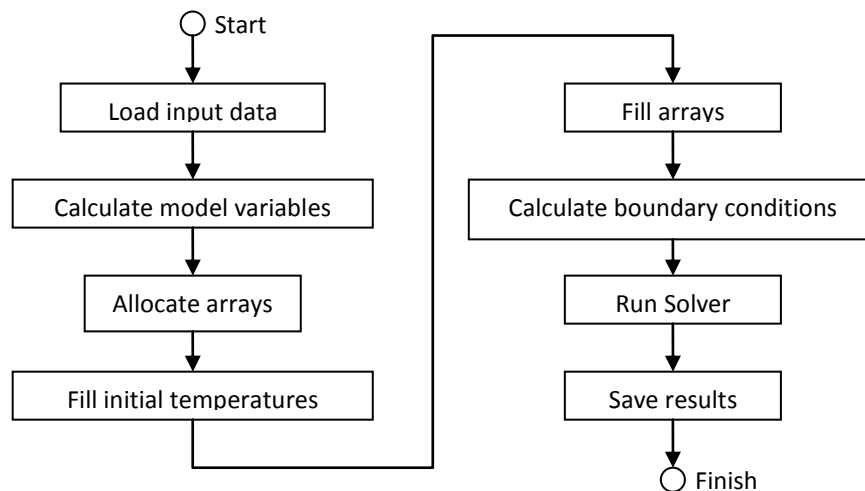
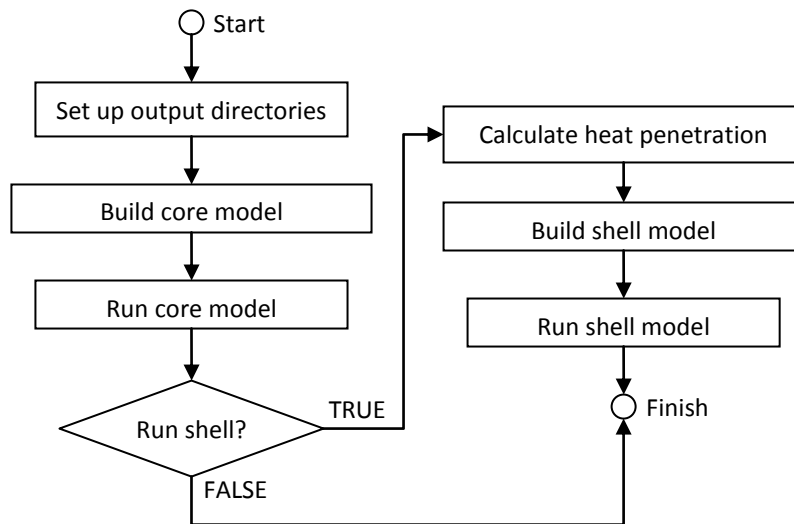


Figure 3-2: Generalised thermal model process flow

The lowest elements of the structure, the thermal models, both share the same basic process flow as illustrated in Figure 3-2. The “load input data” step stores the data passed to the thermal models in the model’s own internal variables. “Calculate model variables” then performs the basic calculations which define the mesh spacing and the array sizes which are subsequently used in the “Allocate arrays” step to allocate the memory space needed to store the model data. “Fill initial temperatures” either sets the entire model to an initial

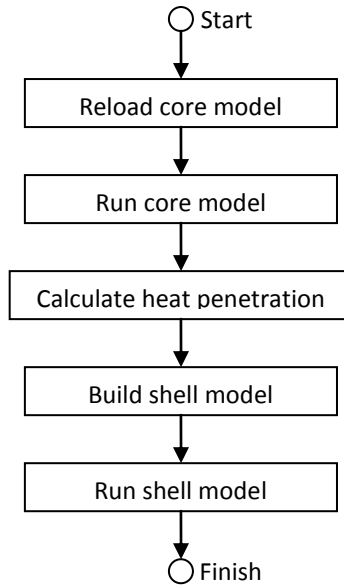
temperature or imports the final temperature of a previous analysis as a starting condition. “Fill arrays” loads the supplied cooling map and other heat transfer coefficients into the appropriate arrays for use in the next step. “Calculate boundary conditions” evaluates all the non-temperature terms of the general and boundary condition equations, storing the resultant values in coefficient arrays. Finally “Run Solver” iteratively calculates new temperatures for new time-steps until the pre-defined run time has been achieved.



**Figure 3-3: Roll model general process flow**

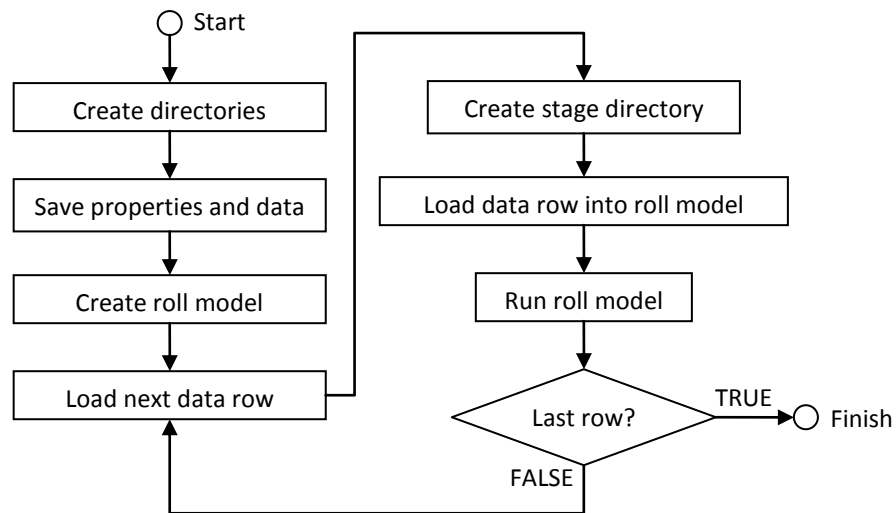
Figure 3-3 gives the process flow for the roll model, which is the next level up in the model hierarchy. The first task of this model is to create the set of folders which will be used by the core and shell models to store the output files separately. Using data provided by a model further up the hierarchy the core model is created and in the following step set running. Once the core model analysis is complete a Boolean (true or false) value, also provided at the beginning of the process, is consulted as to whether a shell model analysis is required. If the value is true the heat penetration depth is calculated, giving the depth at which circumferential temperature variation becomes vanishingly small. With a combination of the initially supplied data, the core model results and the heat penetration depth the shell model is constructed. Finally the shell model solution is calculated and the process is concluded.

Figure 3-4 gives a variation on the previous figure, a roll model which is constructed from saved files rather than as a link in a chain of analyses. In this case the core model will already have been solved, so the results must be loaded from file. One small additional time-step is performed so that the temperature variation with time can be calculated. The use of this is discussed in section 3.4.2 (Page 85). The final three steps are identical to those in Figure 3-3. The option of not running the shell model is removed in this case, since there is no other reason for running this model than to run the shell model.



**Figure 3-4: Roll model run shell from saved files process flow**

In Figure 3-5 the process flow for the stand model is illustrated.



**Figure 3-5: Stand model process flow**

As with the roll model the first task is to create new folders, this time to hold the whole solution. In the next step the properties and process data table are saved, for the record. Next the roll model is created outside of the following loop, to allow the same instance of that model to be used for each successive stage of analysis. The “load next data row” step marks the beginning of the loop, where a single row is extracted from the process data provided by the user interface. A new folder is created to house the results of this stage of the analysis, and then the data row is loaded into the roll model. The first time this action is performed an initial temperature is applied to the models, for subsequent repeats only the boundary

conditions are overwritten. Running the roll model then begins the processes described in Figure 3-3 and Figure 3-2. If there are no more rows left to load from the data table the process finishes, otherwise the next row is loaded and the process repeats until the end of the data table is reached.

### 3.3. Recommended Improvements

---

Most of the processes described above were carried over from the White model intact. This section describes the first improvements made to the software on the advice of the author of the previous project.

#### 3.3.1. Solver Efficiency Improvement

A solver is a programming element whose sole function is to solve an equation or series of equations. In the thermal models it fills a position towards the end of the generalised flow diagram illustrated in Figure 3-2. As described in the previous chapter, the equation which needs to be solved to acquire the temperature of any point in the model can be represented by a general equation.

$$T_{i,j}^{K+1} = \theta T_{i,j}^K + (1 - \theta)(AT_{i,j+1}^{K+1} + BT_{i,j-1}^{K+1} + CT_{i+1,j}^{K+1} + DT_{i-1,j}^{K+1}) + (1 - \theta)(AT_{i,j+1}^K + BT_{i,j-1}^K + ET_{i,j}^K + CT_{i+1,j}^K + DT_{i-1,j}^K + F)$$

**Equation 3-1**

Equation 3-1 states the general equation for the core model, with the addition of a relaxation term.  $T$  gives the temperature at a point,  $i$  and  $j$  are the coordinate indices of the point,  $K$  is the current time-step,  $\theta$  is the relaxation and  $A$  through  $F$  are coefficients. This general form is true for any position in the core model, with the values of  $A$  through  $F$  changing dependent on the boundary conditions near the edges. The relaxation term allows instability to be quelled if it arises by incorporating a proportion of the original value with the predicted value, though is set to zero as standard. The iteration process involves repeatedly calculating the result of this equation for every point in the model. When new values have been calculated for every point they must be re-calculated, since updating the new values changes the right hand side of the equation. The equation can therefore be further simplified to combine those terms which will remain identical between iterations.

$$T_{i,j}^{K+1} = \theta \times T_{i,j}^K + (1 - \theta)(AT_{i,j+1}^{K+1} + BT_{i,j-1}^{K+1} + CT_{i+1,j}^{K+1} + DT_{i-1,j}^{K+1} + RHS) \quad \text{Equation 3-2}$$

Where

$$RHS = AT_{i,j+1}^K + BT_{i,j-1}^K + ET_{i,j}^K + CT_{i+1,j}^K + DT_{i-1,j}^K \quad \text{Equation 3-3}$$

All the temperature terms in Equation 3-3 refer to the old time step and will therefore be constant, allowing them to be gathered into a single term. This reduces the length of the equation, simplifying the code required to implement them and reducing the number of calculations per step. It is in this condition that the solver of the White Model operates, with four unknowns, six arrays of constants and a relaxation value. It is the storage of these constants that is the object of this improvement.

Data storage in a computer is essentially one dimensional, i.e. all data can be represented by a single column of binary values. This holds true for all data types and data structures, with the more complex types being represented differently for ease of interpretation (e.g. the core model uses a two dimensional array to hold temperature data, as in this context it is a two dimensional field). However, the only difference between a two dimensional array and a one dimensional array is the way the indexing system is portrayed. The benefit of converting the data storage to one dimensional arrays is the freedom to arrange the data to best suit the solution process.

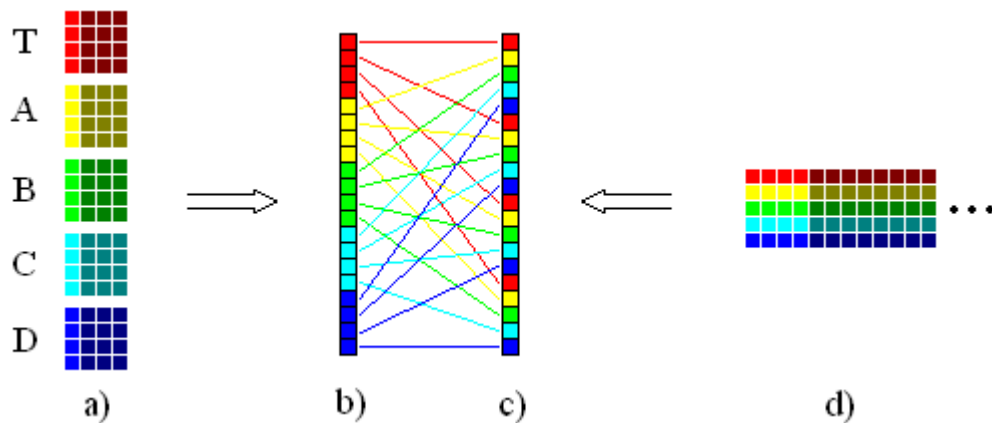


Figure 3-6: Separated and combined coefficient storage schemes

Figure 3-6 a) shows the initial storage scheme, with each term being stored in its own array. Figure 3-6 b) shows the data which can be loaded at one time, taking an equal number of entries from each coefficient array. Figure 3-6 c) then shows the order in which the data will be used as the calculations are performed. Starting from the initial storage scheme this adds an extra process which must be performed before the data can be used.

Figure 3-6 d) shows the proposed storage scheme where a two dimensional array is used as a collection of one dimensional arrays, each of which refers to a different coefficient. With the data in this format it can be used directly without any extra sorting. In reality the solution requires the temperature values of a few surrounding points at each location. To reduce the searching required in returning these values the temperatures are stored independently of the coefficients.

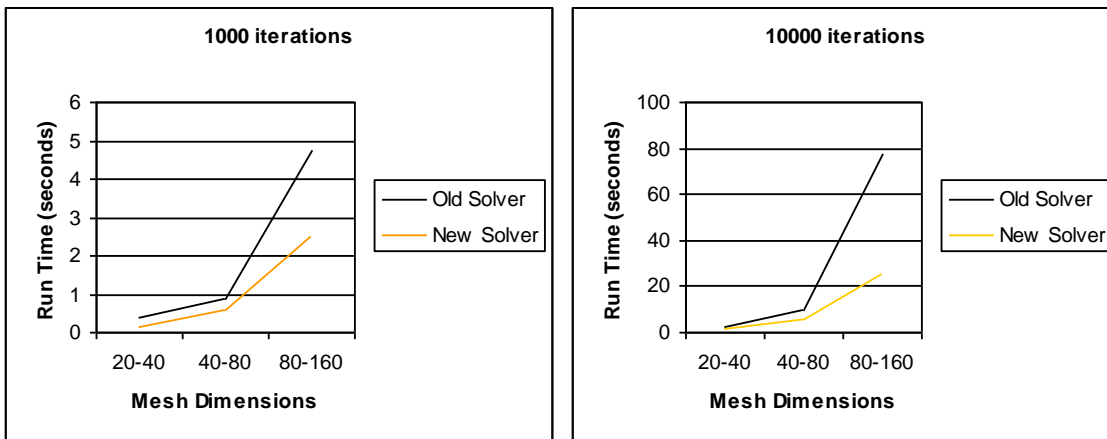


Figure 3-7: Comparison of old and new solver run-times

Figure 3-7 shows the effect of the efficiency improvements on the run time of the solver loop, using different mesh resolutions and performing different numbers of iterations. The iteration count is set arbitrarily high, to give a total run time of great enough length to not be significantly affected by any errors in specifying starting and finishing times. For reference, the computer used to generate the data in the previous figure was a Toshiba Tecra M11-11M, running Windows 7 Professional 32-bit, with 4GB ram and a 2.67GHz Intel Core i7 Processor. Both sides of the figure show a dramatic reduction in run time when using the new solver, with the greater benefit coming at higher resolutions and longer run times. While the core model gets some benefit from this improvement (normally running with dimensions of 50-100) the bulk of the improvement will be in the shell model, which is very high resolution (256-100-120) and often takes tens of thousands of iterations to converge.

### 3.3.2. Expansion of the Symmetry Condition

The symmetry condition which divides the work roll into two equal halves either side of the strip centre line is very useful. It reduces the number of calculations needed per iteration, the total run time and the amount of disk space required to store the results. However, if the strip is running off-centre or asymmetrical cooling is applied, intentionally or otherwise, the



model will be unable to take account of any asymmetrical response. It was for this reason that the expansion of the symmetry condition was recommended, while retaining the symmetrically reduced model so that advantage can still be made of symmetry when it occurs.

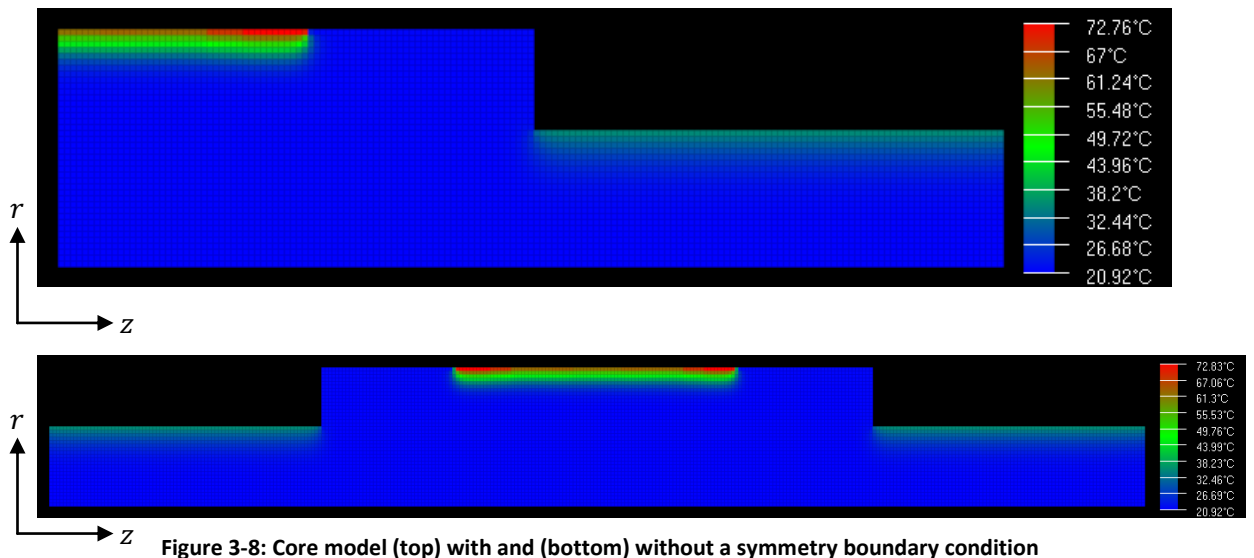


Figure 3-8: Core model (top) with and (bottom) without a symmetry boundary condition

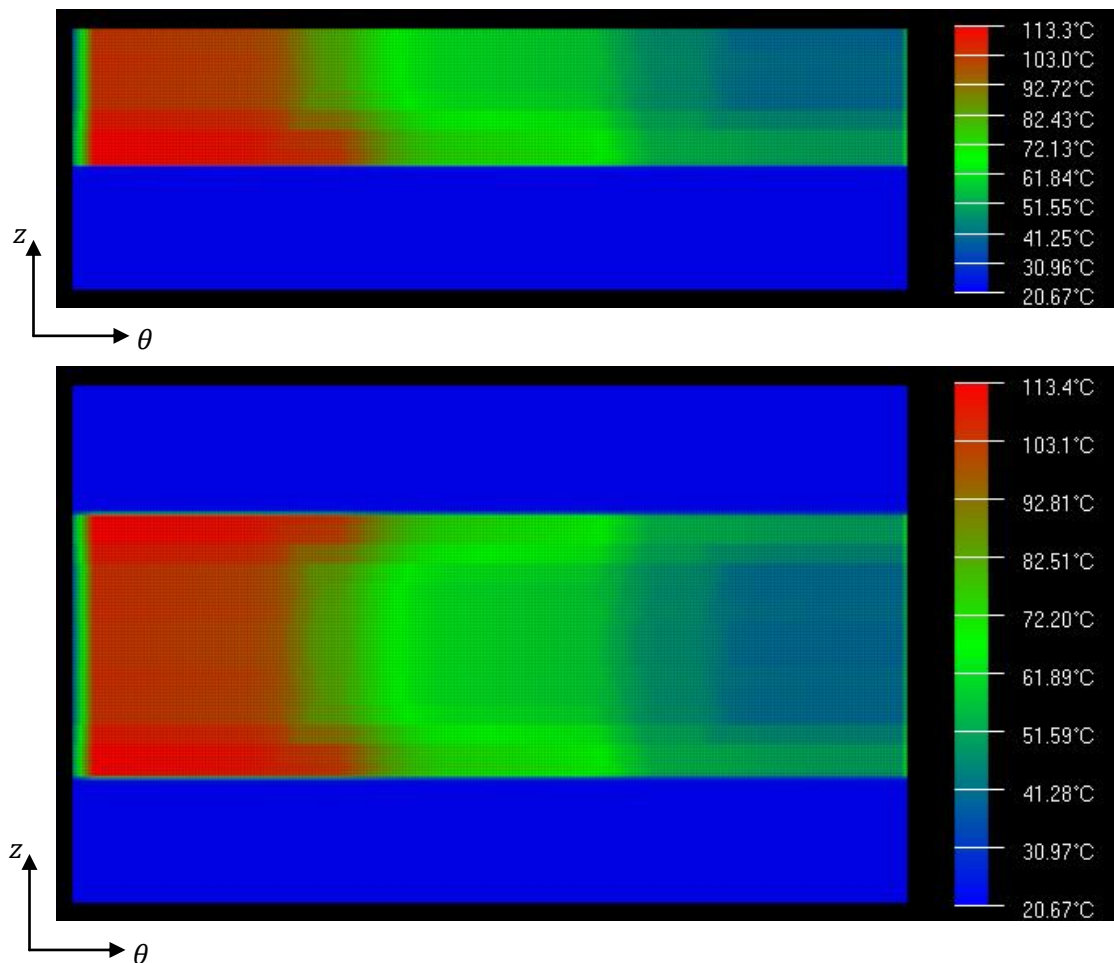


Figure 3-9: Shell model (top) with and (bottom) without a symmetry boundary condition

Figure 3-8 shows the core model with and without the symmetry boundary condition.

Figure 3-9 shows a temperature contour map from the surface of the work roll, as though the surface layer of the roll has been peeled off and unrolled. In this figure,  $z$  is the axial dimension i.e. running down the centreline of the model, while  $\theta$  increases with angular position around the roll.

To affect these changes a new model was created for both the core and shell models, which makes use of the principle of “Inheritance” to incorporate a vast majority of the code from the original core and shell models. This not only reduces the programming workload by not requiring the replication of code but also means that if further changes are required they can be made once. By changing the original models, these new models will “inherit” those changes too. Departure from the original code is made by “overriding” the original methods (functions).

`LoadSettingsFromProperties` – this method loads all the data required to run the model. The only change needed was to double the *isize* parameter, which corresponds to the width of each model in node points.

`FillModelArrays` – this method contains a number of arrays that needed adapting:

- The *dimensions* array stores the size of each “element” in the model. Boundary conditions along the edges of the model replace one half of an element with an equation based on the other half of the element and values relating to heat input or cooling. Hence, all the elements on the boundary are effectively half-sized, requiring different treatment in various places in the program.
- The bite condition arrays need to be expanded to allow for both halves of the strip contact area
- The strip temperature and thermal conductivity (of the interface between the strip and the roll) arrays need to be expanded. The origin for these remains the same, i.e. at its original position on the symmetry line with the loops used to fill the arrays running from negative to positive. This is easier to apply than moving the origin as the majority of the same procedures will be applicable with only minor adaptations, rather than a complete rewrite.
- The cooling water heat transfer arrays need to be expanded. In this instance a complete rewrite is required, moving the origin from the centre line to the edge of the barrel. This allows the use of asymmetrical cooling maps.

`CalculateModelVariables` – this method breaks the large scale dimensions and specifications into the smaller units and dimensions required by the program. To implement this method the strip width needs to be doubled to represent both halves of the strip.

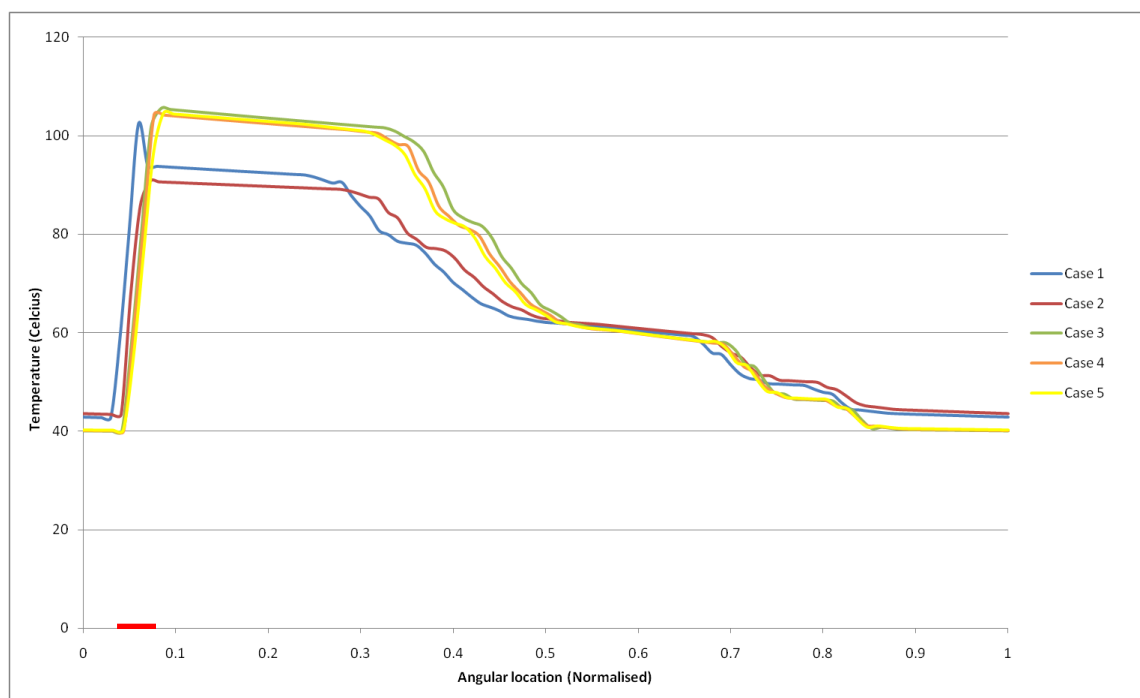
With the modifications described above, only the new boundary conditions need to be implemented. These will not be discussed in detail as no new terms are required, only minor rearrangement of the original equations to create the mirror image of the boundary conditions introduced in the previous chapter.

The expanded thermal models were compared to the original thermal models to check for coherency when identical boundary conditions were applied to both. The highest difference was of the order  $10^{-13}$  degrees, allowing the validation work of the previous project to be carried across.

### 3.4. Repairs

---

With the recommended improvements implemented, work began on using the thermal solution output with a finite element model to obtain a stress solution. During this process it became clear that the thermal model outputs were unstable. Figure 3-10 shows some example temperature data taken from five different finite element models, each of which used the same input data but different resolutions.



**Figure 3-10: Surface temperature variation around the circumference in the rolling direction (normalised to 1 revolution). Heavy line on abscissa corresponds to the roll bite.**

The variations present in this data arose due to a number of issues, one of which was a simple error in the algorithm responsible for importing the temperature data. The three

remaining issues pertained to the thermal models themselves. An initial investigation showed that the core model was giving identical results for each of the five test cases, it was only the shell model data that showed any variation.

Figure 3-11 below shows a comparison between the core model data and the shell model data. Since the core model temperature data corresponds to the circumferentially averaged value at that radial-axial position, the shell model data can be converted into the same format by averaging around the circumference. There are two main discrepancies between the two data sets.

1. The shell model temperatures are consistently higher than the core model temperatures.
2. The shell model temperature varies almost linearly with depth, whereas the core model temperature follows a more pronounced curve.

These discrepancies did not necessarily highlight a potential cause of the dissimilarity in Figure 3-10 but represented errors that needed to be addressed.

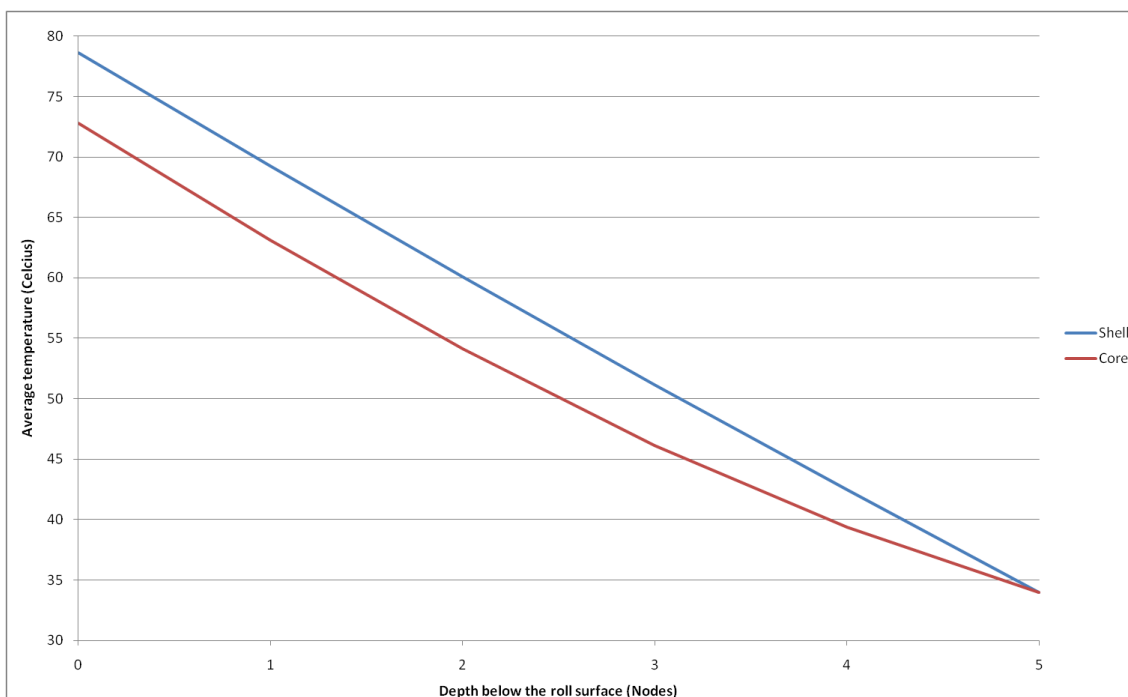


Figure 3-11: Circumferentially averaged temperature at the barrel centre, varying with distance below the surface

### 3.4.1. Boundary Condition Adjustments

The first of the inconsistencies was discovered by comparing the heat flow into the model from the strip and out of the model through the cooling spray in the two models.

### Heat flow from the strip

---

Given that the heat transfer coefficients used represent the heat flow per degree Kelvin temperature difference per unit area, the heat flow can be expressed as:

$$\dot{Q} = hA(T - T_s) \quad \text{Equation 3-4}$$

where  $\dot{Q}$  is the heat flow,  $h$  is the heat transfer coefficient,  $A$  is the surface area of material that the node represents,  $T$  is the temperature at the node being considered and  $T_s$  is the strip temperature. The dimensions of each element are described in the generic terms of mesh spacing as  $\Delta z$ ,  $\Delta r$  and  $r\Delta\theta$  (NB: the  $\theta$  direction spacing is measured in radians and so the distance between nodes will be different for different radial positions). Substituting the area of the top surface into Equation 3-4 gives the following relationship;

$$\dot{Q} = hr\Delta\theta\Delta z(T - T_s) \quad \text{Equation 3-5}$$

By summing  $\dot{Q}$  for all points on the surface of the shell model, and with  $\Delta\theta$  replaced with  $2\pi$  in the core model, the total heat flow can be acquired.

### Heat flow to the cooling sprays

---

The heat flow to the cooling sprays can be treated in very much the same manner as the heat flow from the strip.

$$\dot{Q} = hr\Delta\theta\Delta z(T - T_f) \quad \text{Equation 3-6}$$

In this case  $T_f$  represents the temperature of the cooling water. The effect of variations in cooling water temperature due to varying mass flow of water at different positions on the work roll surface are all taken care of by the Van Steden cooling map (by providing  $h$ ), so the supply temperature can be used.

It was immediately apparent upon implementing the heat flow calculations that the two models were behaving differently. The heat going into the shell model was higher, at 1.52MW compared to 1.44MW for the core model. The heat leaving the core model was far higher – 2.05MW to the shell model’s 1.45MW – explaining the shell model’s higher average temperature.

The model was run again and the boundary heat flow equations performed before the solver equation was allowed to run. This simplifies the problem and reduces the number of factors that could be concealing the real source of the error. The result of this was 1.5MW of heat flowing to the core and 1.46MW of heating in the shell model, and 2.17MW leaving from the core and 2.14MW leaving from the shell to the cooling sprays.

Figure 3-12 explains the differences in the heating and cooling of the core and shell models. The blue line shows the cooling heat transfer coefficient multiplied by the temperature difference between the work roll surface and the coolant temperature at different points around the circumference. The heat transfer coefficient varies significantly, based on the layout of the cooling sprays and the amount of water being supplied to them. The temperature will also vary significantly. This is equivalent to the way the cooling is experienced by the shell model.

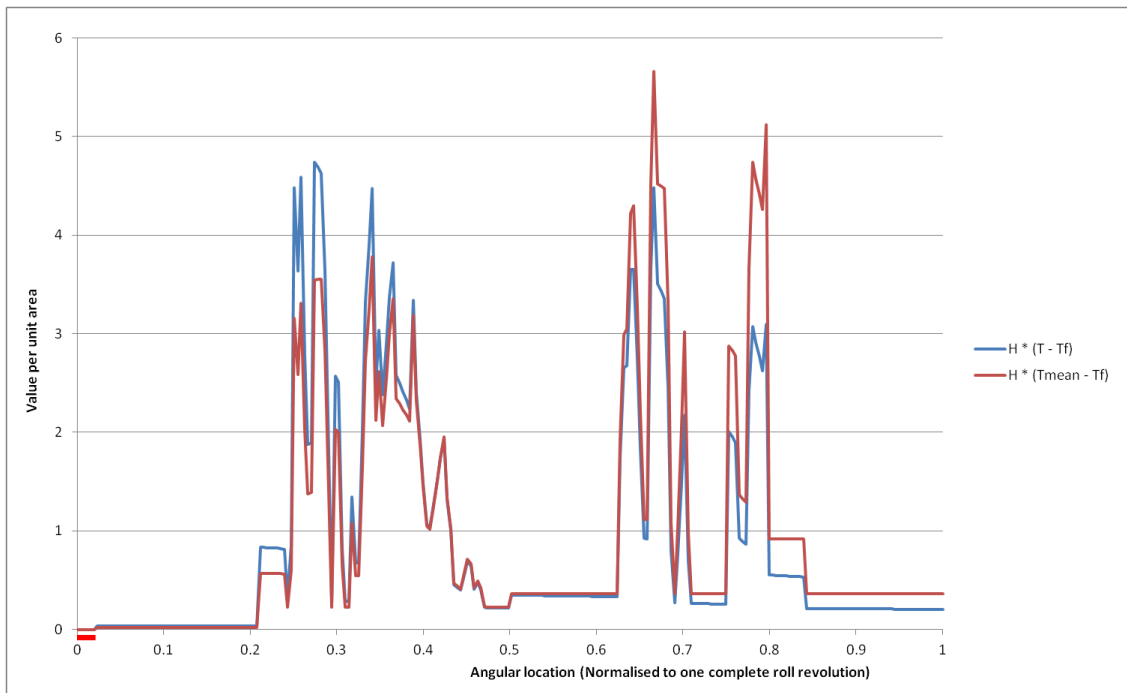


Figure 3-12: Difference between heat transfer rates using averaged and non-averaged temperatures. Roll bite location is indicated by the thick red line just below the abscissa.

The red line in Figure 3-12 shows the cooling heat transfer coefficient multiplied by the average temperature. While the heat transfer coefficients will still vary significantly, the temperature remains constant around the circumference. This is equivalent to the cooling experienced by the core model. The product in this case is not weighted by the higher temperatures immediately after the roll bite (the left side of the figure) and lower temperatures further round the roll. This is not a zero-sum difference; the area under the red curve comes to 247 units and the area under the blue curve is 229 units. This difference in cooling intensity for the same cooling map is what leads to the final temperature difference between the shell and core models.

### Solution

---

To account for the discrepancy it was necessary to adjust one of the cooling boundary conditions to match the other. The shell model was chosen as the model to adjust since the core model had been validated against experimental data, so the cooling effect witnessed by this model is likely to be the more accurate (this approach was later reversed, as described in section 3.6.2, page 119, following adjustments to the terms related to conduction between the strip and the work roll).

$$\dot{Q}_{Spr} = h_{sprAdjust_i} \times h_{spr_{i,k}} (T_{i,j,k} - T_{spr}) \quad \text{Equation 3-7}$$

An array of adjustment factors (varying with axial position) was created which multiplies into the  $\dot{Q}_{Spr}$  term for the shell model surface boundary condition, as seen in Equation 3-7. This factor can be calculated from the temperature and heat transfer coefficient profiles to give the same net effect as the equivalent boundary conditions in the core model.

$$h_{sprAdjust_i} = \frac{\sum(h_{spr_{i,k}} \times T_{i,k})}{\sum h_{spr_{i,k}} \times \sum T_{i,k}} \quad \text{Equation 3-8}$$

Equation 3-8 uses the difference between the effective (top) and actual (bottom) proportional heat flow to provide a new adjustment factor.  $h_{spr_{i,k}}$  is the heat transfer coefficient and  $T_{i,k}$  is the temperature at the given surface coordinates. The new adjustment factor will change the effective heat transfer coefficients used by the boundary conditions, so the boundary conditions will have to be recalculated to give the new solver coefficients. The new solver process is given below.

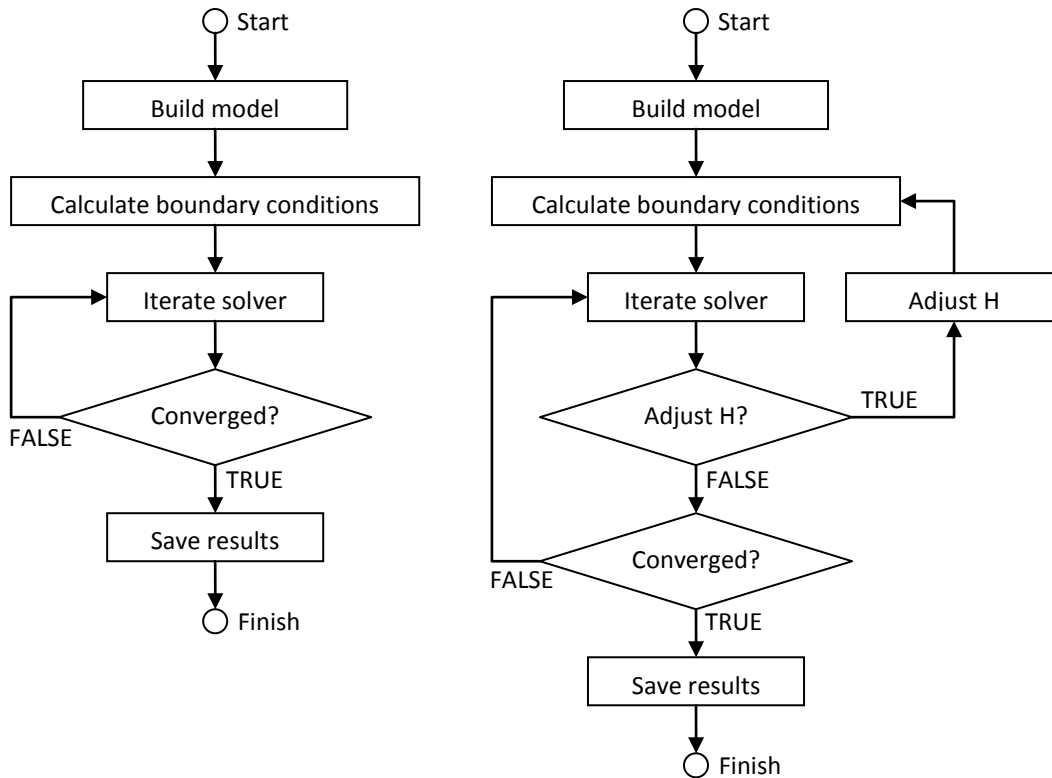


Figure 3-13: old (left) and new (right) shell model flow diagrams

The left of Figure 3-13 gives the original process flow, which is a slightly rearranged version of Figure 3-2. The new process on the right of Figure 3-13 is identical save for a new conditional switch and an extra process step. The extra conditional switch controls the application of the adjustment factor recalculation, only recalculating the boundary conditions after the temperatures have been allowed to approach convergence. This iterative approach is required as the imbalance between the shell and core models will change as the shell model temperature profile develops. The “Adjust H” step performs the calculation in Equation 3-8 for the full range of axial positions in the model to provide updated values for the adjustment factor.

The criteria of the “Adjust H?” switch is that a certain number of iterations have been performed, with 100 found to give a good balance of convergence rate against efficiency. For larger resolutions the process of converging the adjustment factor in-line with the solution still results in long run times. To minimise the run time an “Optimise” step is also introduced which makes an estimate at the settling value of the adjustment factor.

As a safeguard against the convergence of the thermal model being declared prematurely, the main solver loop is forced to repeat until the adjustment factor updates meet their own convergence criteria, though in practice it is normally the first to converge. This is achieved by making the “Converged?” switch dependent on an “H Converged?” switch being “TRUE”.



Conversely if the adjustment factor reaches convergence first, further updates are still performed as a check but are not implemented. This saves processing time, since the boundary condition calculation step is processor-intensive.

The process diagram in Figure 3-13 can be expanded to include all of the conditions mentioned above, as presented in Figure 3-14 below.

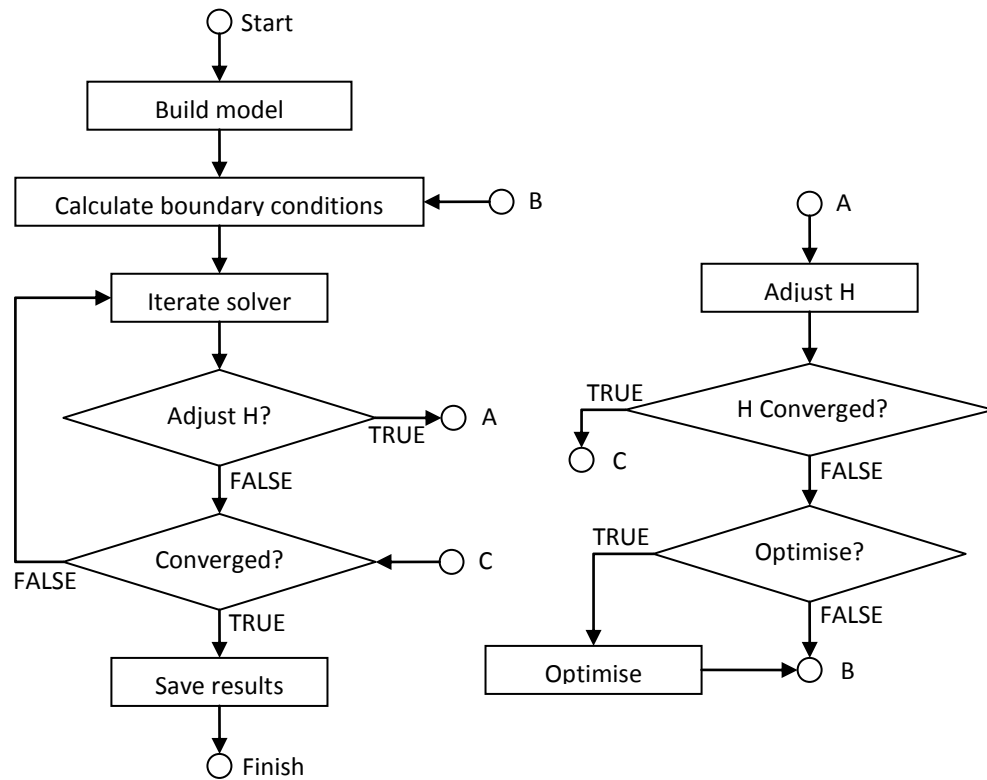


Figure 3-14: Optimised shell model process flow diagram

In Figure 3-14 the effect of the convergence criteria on the adjustment factor can be seen in the way the “TRUE” branch of the “H Converged?” switch leads to the same element as the “FALSE” branch of the “Adjust H?” switch. In this scenario the “Calculate boundary conditions” step is not performed, so the new adjustment values are not implemented, saving time.

### Optimisation

To ensure that the thermal solution convergence was not held up by the convergence rate of the adjustment factor, an optimisation algorithm was developed. This algorithm uses three data points as  $(x, y)$  coordinates – with  $x$  given by the number of iterations and  $y$  given by the adjustment factor value – to predict the settling value, i.e. the  $(x, y)$  point at which the maximum value of  $y$  is attained.

Very few points will be available to the optimisation routine, since it takes a significant amount of time (100 iterations worth of time) to generate each data point. A simple second order polynomial relationship was chosen due to the low requirement for input data and the added benefit of having only one maxima or minima, making the prediction element of the routine more straight forward. The relationship is given below. The following equations use an abstract set of symbols A, B, C and M for simplicity, not to be confused with the definitions stated in the nomenclature.

$$y = Ax^2 + Bx + C \quad \text{Equation 3-9}$$

The three data points will be in the format  $(x_1, y_1)$ ,  $(x_2, y_2)$  and  $(x_3, y_3)$ . These values can be used to convert Equation 3-9 to a set of linear equations, giving;

$$M = \left[ \begin{array}{ccc|c} x_1^2 & x_1 & 1 & y_1 \\ x_2^2 & x_2 & 1 & y_2 \\ x_3^2 & x_3 & 1 & y_3 \end{array} \right] \quad \text{Equation 3-10}$$

Where  $M$  represents the matrix form of the three simultaneous equations. From  $M$  the values of  $A$ ,  $B$  and  $C$  can be found using the Gauss-Jordan elimination process, which reduces the left side of  $M$  to an identity matrix. The process will also affect the right hand side of  $M$ , converting it from an input vector into an output vector containing the values of  $A$ ,  $B$  and  $C$ .

In Equation 3-11 the first row is divided by  $x_1^2$  to reduce the first entry to one. The other entries values are represented by an asterisk (\*) for brevity.

$$M = \left[ \begin{array}{ccc|c} 1 & * & * & * \\ x_2^2 & x_2 & 1 & y_2 \\ x_3^2 & x_3 & 1 & y_3 \end{array} \right] \quad \text{Equation 3-11}$$

Next a multiple of the top row is added to or subtracted from the other rows of the matrix to reduce every other value in the first column to zero.

$$M = \left[ \begin{array}{ccc|c} 1 & * & * & * \\ 0 & * & * & * \\ 0 & * & * & * \end{array} \right] \quad \text{Equation 3-12}$$

By dividing the second row by the value in the second column of the second row the next member of the diagonal is reduced to one.

$$M = \left[ \begin{array}{ccc|c} 1 & * & * & * \\ 0 & 1 & * & * \\ 0 & * & * & * \end{array} \right]$$

Equation 3-13

The second row can now be used to eliminate the remaining value from the lower half of the matrix.

$$M = \left[ \begin{array}{ccc|c} 1 & * & * & * \\ 0 & 1 & * & * \\ 0 & 0 & * & * \end{array} \right]$$

Equation 3-14

Dividing the bottom row by the value in the third column of the bottom row reduces that value to one.

$$M = \left[ \begin{array}{ccc|c} 1 & * & * & * \\ 0 & 1 & * & * \\ 0 & 0 & 1 & Y_3 \end{array} \right]$$

Equation 3-15

The bottom row of the left matrix is now in the form of the identity matrix and so will not need to change any more. In turn, this means that the bottom row of the right hand matrix will not be changing, so it can formally be named  $Y_3$  as the final output value. Substituting the values from the bottom row into Equation 3-9 gives;

$$Y_3 = A(0) + B(0) + C(1)$$

In other words,  $Y_3 = C$ . It will therefore follow that, once they are calculated,  $Y_2 = B$  and  $Y_1 = A$ . These could be acquired by back-substitution from Equation 3-15, but since the matrix elimination process is easy to automate, carrying on with the elimination route provides a more flexible system for later expansion.

$$M = \left[ \begin{array}{ccc|c} 1 & * & 0 & * \\ 0 & 1 & 0 & Y_2 \\ 0 & 0 & 1 & Y_3 \end{array} \right]$$

Equation 3-16

The upward elimination is no different to downward elimination, in this case multiples of the third row are added to or subtracted from the second and first rows to reduce the value in the third column to zero. This first upward sweep fully defines the second row.

$$M = \begin{bmatrix} 1 & 0 & 0 & | & Y_1 \\ 0 & 1 & 0 & | & Y_2 \\ 0 & 0 & 1 & | & Y_3 \end{bmatrix}$$

**Equation 3-17**

The second sweep fully defines the first row. The fourth column can now be extracted to define Equation 3-9, and the settling point estimated based on the turning point of the curve. The gradient of Equation 3-9 at the turning point can be found from the following equation.

$$\frac{dy}{dx} = 2Ax + B = 0$$

Which gives  $x$  at the turning point as;

$$x = -\frac{B}{2A}$$

**Equation 3-18**

The value of  $y$  calculated from Equation 3-9 with the  $x$  value from Equation 3-18 will give an estimate of the settling value of the adjustment factor, shortcutting the iteration process to reduce the number of iterations required to reach convergence.

### *Comparison*

---

Figure 3-15 shows a comparison of the core model and circumferentially averaged temperatures at the surface, varying in the axial direction. The two sets of values lie almost perfectly on top of one another, showing that the new routines introduced at the top surface boundary have successfully corrected the mismatch in heating and cooling.

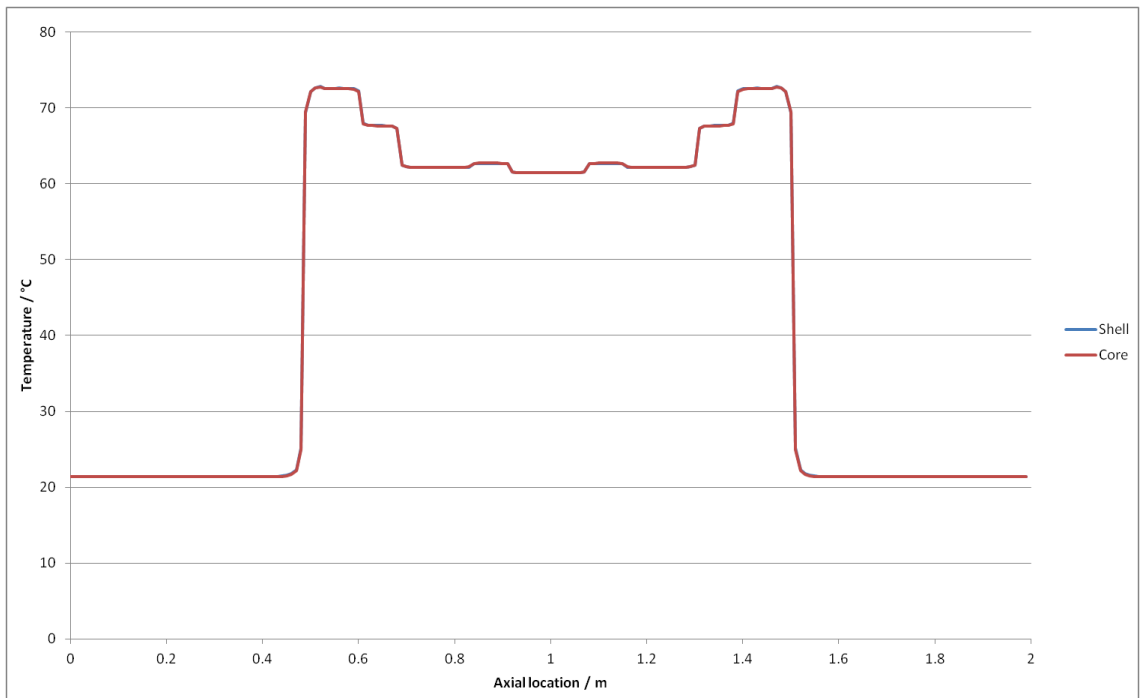


Figure 3-15: core model versus averaged shell model axial temperature profile

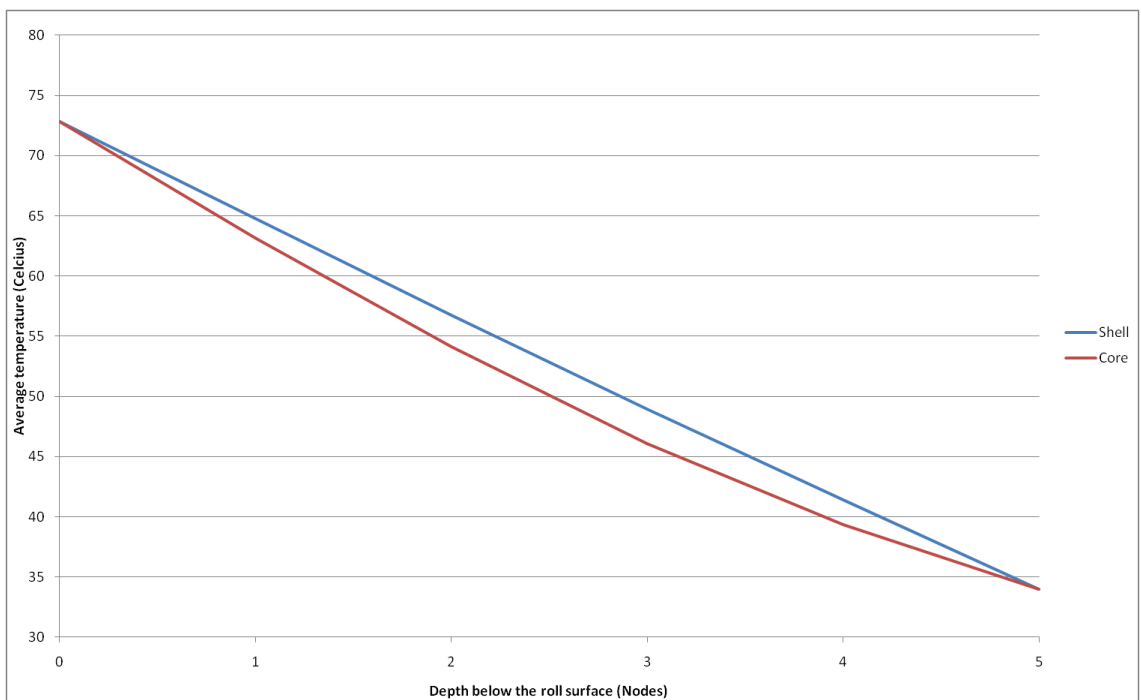


Figure 3-16: Circumferentially averaged temperature at the barrel centre, varying with distance below the surface

### 3.4.2. Transiency

Figure 3-16 shows the same comparison between the core and averaged shell model results as Figure 3-11, but with the boundary condition adjustments in effect. The difference

in cooling intensity at the surface has been nullified very effectively but there is still a significant difference between the results of the two models.

In this instance the difference stems from the treatment of time by each model. The core model is transient, so the temperatures predicted by the core model will be highly dependent on time. In Figure 3-16 this manifests as a dip in the temperature profile, where the temperature at the roll surface is high but the temperature below the roll surface has not yet had time to heat up. The shell model, by contrast, is static. The results of the shell model describe a temperature profile where an infinite amount of time has passed and thermal equilibrium has been achieved. With no insulation present the temperature profile between two points at equilibrium will always be a straight line, as seen in Figure 3-16.

### Solution

---

It would be impractical to create a transient shell model, the computational cost of running any reasonable length of simulation would be prohibitively high. However, transient behaviour can be affected without the increased processing overhead if a measure of the required transient behaviour is obtained.

$$\frac{\partial T}{\partial t} = \frac{T_{i,j}^{K+1} - T_{i,j}^K}{\Delta t} = T'_{i,j} \quad \text{Equation 3-19}$$

The core model can provide this by running a second of simulated time. For every possible value of  $(i, j)$ ,  $T_{i,j}^K$  can be taken from the old results file and  $T_{i,j}^{K+1}$  can be taken from the new results file. The difference of these two values, divided by the length of the time step used to generate the new results, will give a value for  $T'_{i,j}$ . The gradient of the temperature with time will not be directly dependent on element sizes, so there will be no need to treat the  $T'_{i,j,k}$  term needed for the shell model any differently than the  $T'_{i,j}$ . This means that the  $T'$  term can be substituted directly into the original shell model equation;

$$\frac{\partial T}{\partial t} = \alpha \left( \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial r^2} + \frac{1}{r^2} \frac{\partial^2 T}{\partial \theta^2} + \frac{\partial^2 T}{\partial z^2} \right) - \omega \frac{\partial T}{\partial \theta} \quad \text{Equation 3-20}$$

to give;

$$T' = \alpha \left( \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial r^2} + \frac{1}{r^2} \frac{\partial^2 T}{\partial \theta^2} + \frac{\partial^2 T}{\partial z^2} \right) - \omega \frac{\partial T}{\partial \theta} \quad \text{Equation 3-21}$$

Thanks to the presence of a deprecated  $T'$  term in the original code the process of applying this change was a process of minor adaptations rather than wholly new code generation. The single value of  $T'$  was expanded into an array to allow  $T'$  to vary in the radial and axial directions and the shell model construction process was adapted to allow the shell model to request  $T'$  values from the core model.

### *Heat flow checking*

---

In order to ensure that the solution to this problem is correct it is necessary to quantify the amount of energy that temporal effects are absorbing. This is the amount of energy that is being used to heat up the material and can be seen as the difference between the heat energy flowing into the shell area of the core model and the heat energy flowing out. This is contained in 5 quantities;

1. The heat conducted from the strip
2. The heat absorbed by the cooling sprays
3. The heat generated in the roll bite due to friction
4. The heat leaving via convection from the side of the roll
5. The heat conducted deeper into the core material

The heat gained from / lost to the strip / cooling sprays was calculated in the previous section, so only three new methods are required in the Core and Shell models.

### *Friction*

---

Most of the hard work has already been done for this value, as there is a quantity called *spow* which represents the watts per square metre induced by mechanical loads. The surface heat flow from friction can therefore be calculated by multiplying this value by the area for each surface node.

$$\dot{Q} = spow \times \Delta z \times r \Delta \theta \quad \text{Equation 3-22}$$

Replacing  $\Delta\theta$  with  $2\pi$  puts the equation in the correct form for use in the core model.

### Convection from the roll sides

---

For the convection from the sides of the roll it is necessary to turn again to Fourier's Law of Conduction. This time the area to calculate is a segment of a circle as illustrated below.

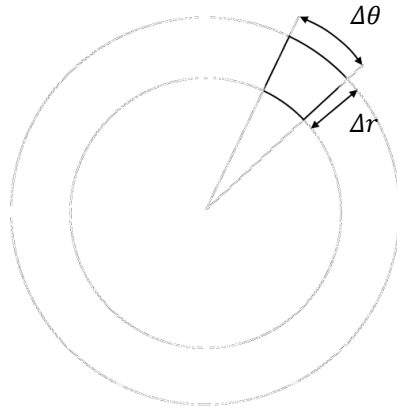


Figure 3-17: Area of an element on the side of the work roll

The area for this shape is  $r\Delta\theta\Delta r$ , where  $r$  is the distance from the centre of the circle to the middle of the element (i.e. the radius of the smallest circle plus  $\Delta r/2$ ). Once again, expanding  $\Delta\theta$  to  $2\pi$  will give the equivalent area for the core model. Substituting this for the area in Fourier's Law of Conduction gives;

$$\dot{Q} = hr\Delta\theta\Delta r(T - T_a) \qquad \text{Equation 3-23}$$

Where  $T$  is the temperature of the node on the roll side,  $h$  is the heat transfer coefficient and  $T_a$  is the ambient air temperature.

### Conduction to the core

---

The amount of energy conducted to the core model is the sum of all heat transfer across the boundary between the nodes above the shell-core interface and the nodes below the shell-core interface. This is relatively simple in the case of the shell model as the model includes in its temperature array the temperatures for the row of nodes below the interface.



The core model calculates averages for the whole of the roll so can be approached in the same manner.

To calculate the amount of heat energy being transferred between two nodes in either model requires the heat difference between the two nodes, the area of the face that separates the two volumes of material the nodes represent and the thermal conductivity of the material. The equation takes the form;

$$\dot{Q} = kA \frac{\partial T}{\partial r} \quad \text{Equation 3-24}$$

Considering a node with index  $j$  and another node with index  $j + 1$ , the figure below shows that the plane separating the two volumes of material is at  $r + \Delta r/2$ .

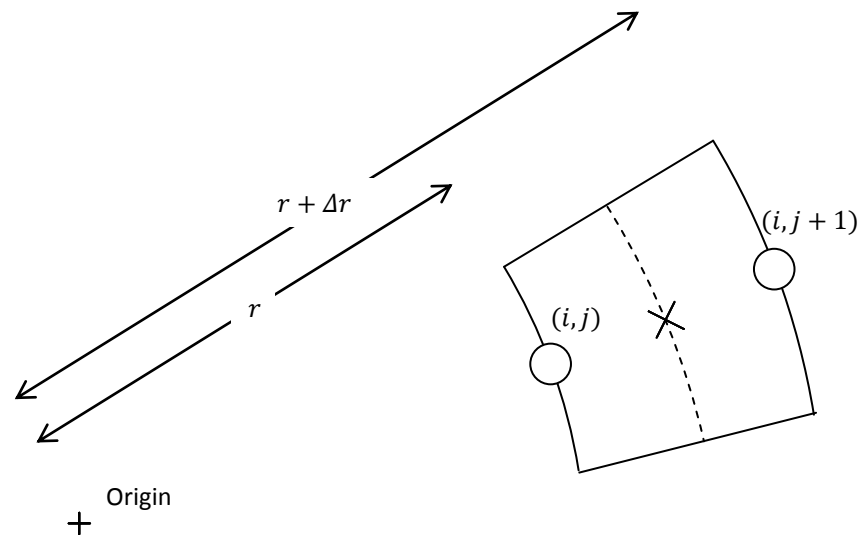


Figure 3-18: The interface plane between two nodes

This being the case Equation 3-24 becomes;

$$\dot{Q} = k \left( r + \frac{\Delta r}{2} \right) \Delta \theta \Delta z \quad \text{Equation 3-25}$$

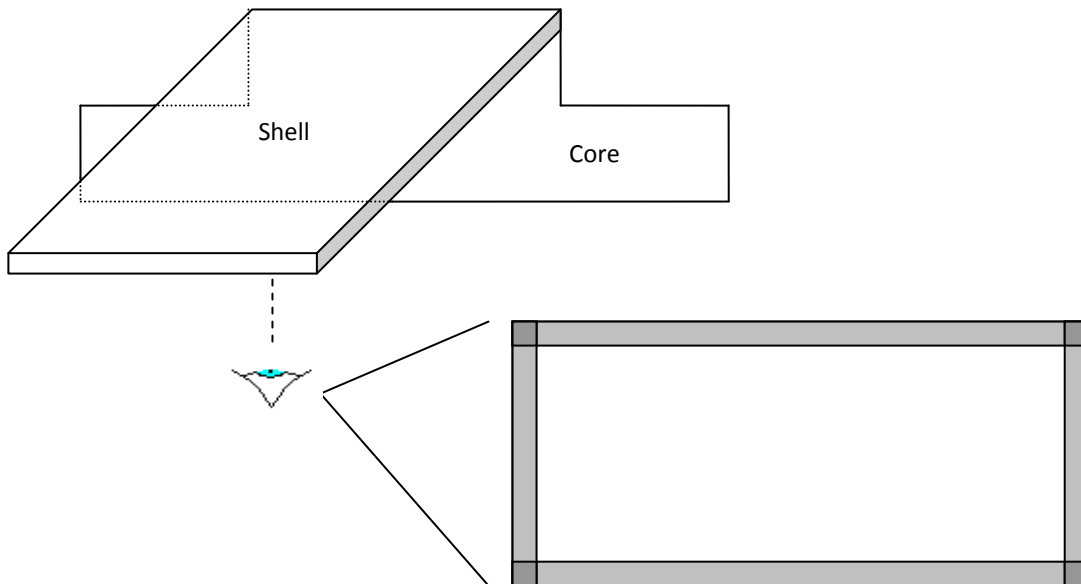


Figure 3-19: View of the interface with boundary regions

Figure 3-19 shows a view of the interface surface of the shell model. The highlighted areas around the edge of this surface show the area represented by nodes on this surface that lie on a boundary condition. The light gray areas are in contact with one boundary condition and so one of their physical dimensions is half size (as shown in Figure 3-20 below, either  $\Delta\theta$  or  $\Delta z$ ). The dark gray areas are in contact with two boundary conditions so both of their physical dimensions are half size.

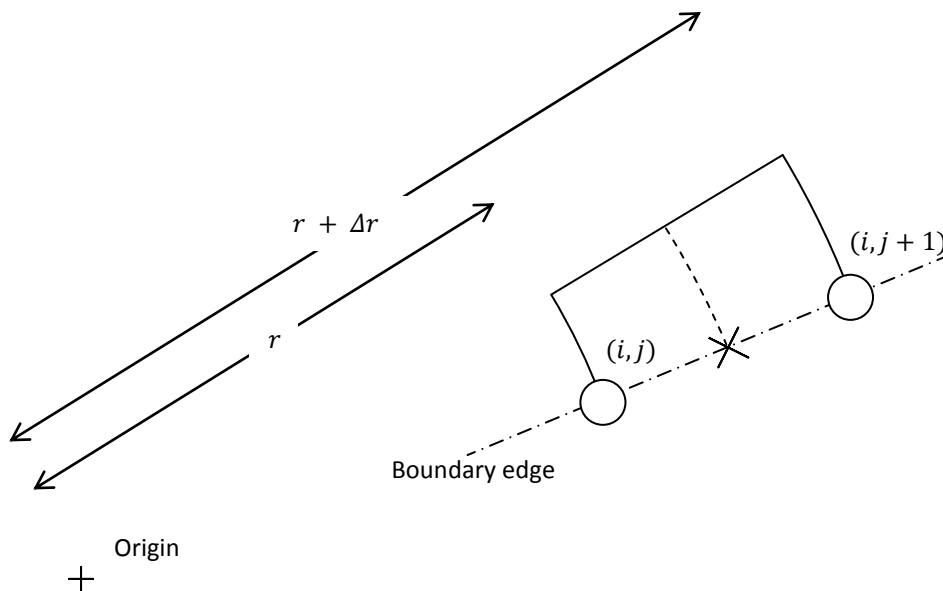


Figure 3-20: The interface plane between two nodes on a boundary edge

	Shell	Core
Spray	-2,034,325	-2,028,325
Strip	1,443,180	1,449,204
Friction	640,286	640,286
Hconv	-4.52	-5.02
Interface	-49,308	-34,722
<b>Total</b>	<b>4.52</b>	<b>26,438</b>

Table 3-1: Comparison of heat transfer in the shell and core models

Table 3-1 shows the heat transfer values obtained from the two models via the methods developed in this section. The two sets of values are noticeably different as would be expected of the two models when one is solving a time-dependent equation and the other is not. From the totals it can be seen that the shell model is balancing the heat equation to 0.0002% of the total heat input (the strip heating plus the frictional heating). The core model balances to 1%. This difference gives a direct comparison of how much energy is being absorbed by the material and a method of quantifying how well the solution is working once this section’s adaptations are applied.

	Shell	Core
Spray	-2,022,645	-2,028,325
Strip	1,443,464	1,449,204
Friction	640,286	640,286
Hconv	-5.02	-5.02
Interface	-34,405	-34,722
<b>Total</b>	<b>26,876</b>	<b>26,438</b>

Table 3-2: Comparison of heat transfer in the adapted shell and core models

Table 3-2 gives the new heat transfer values after the transient adaptation is implemented in the shell model. Although a small error remains in the totals the two models are far more coherent, the effect of which can be seen in Figure 3-21 below.

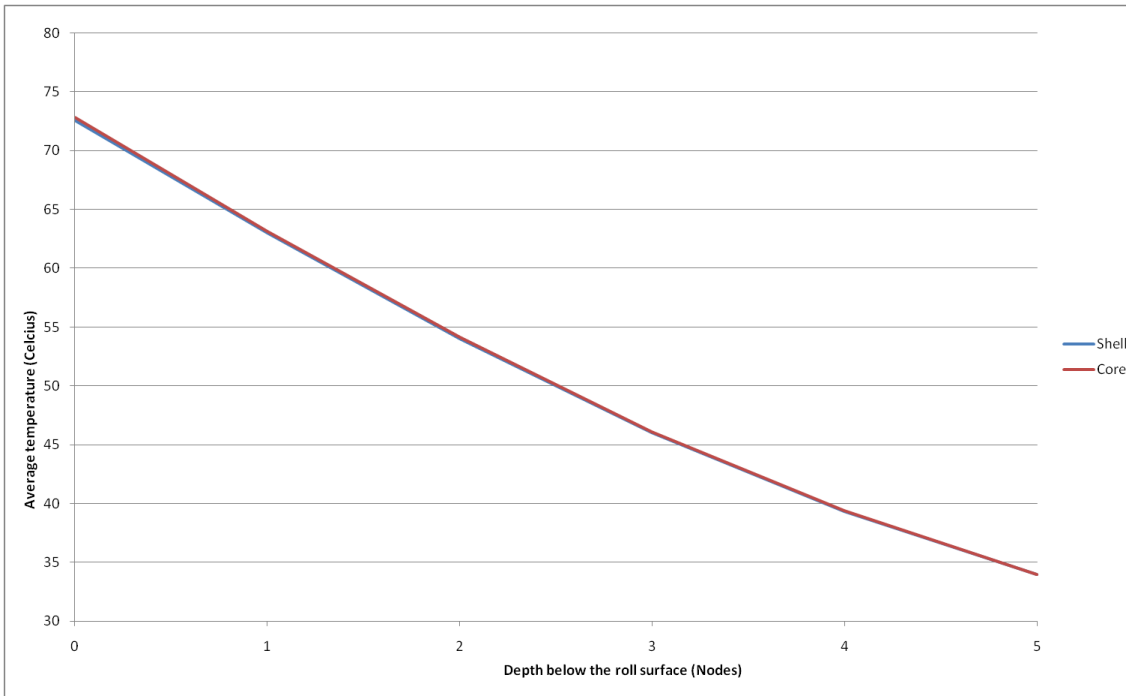


Figure 3-21: Circumferentially averaged temperature at the barrel centre, varying with distance below the surface

### 3.4.3. Shell Mesh Resolution

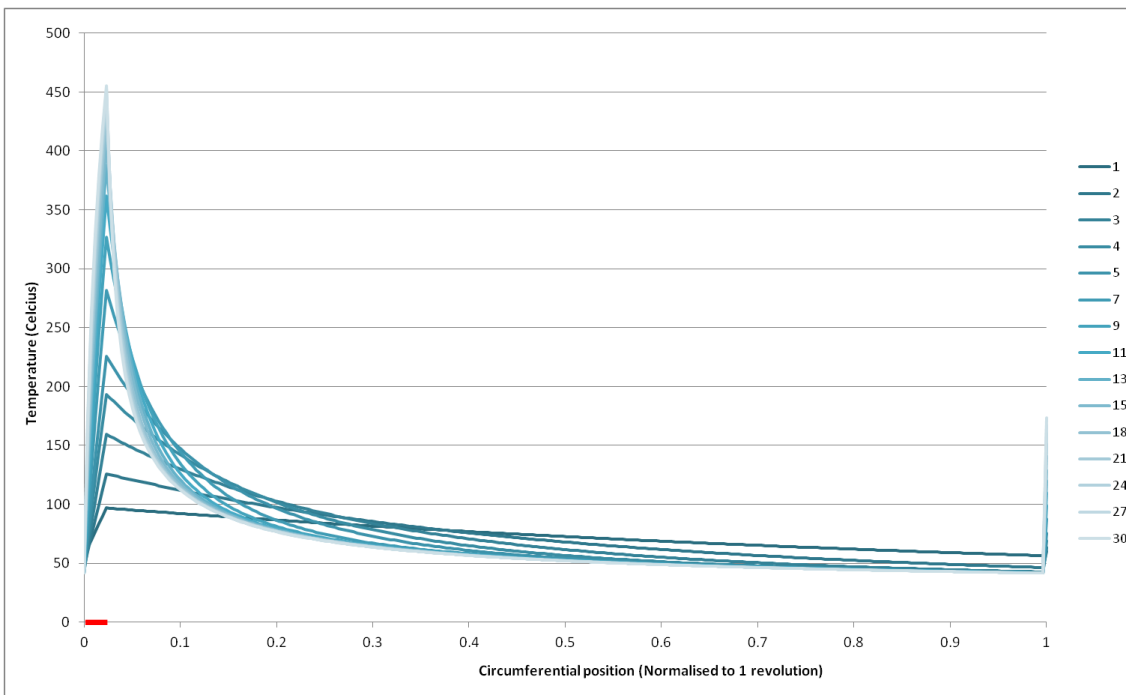
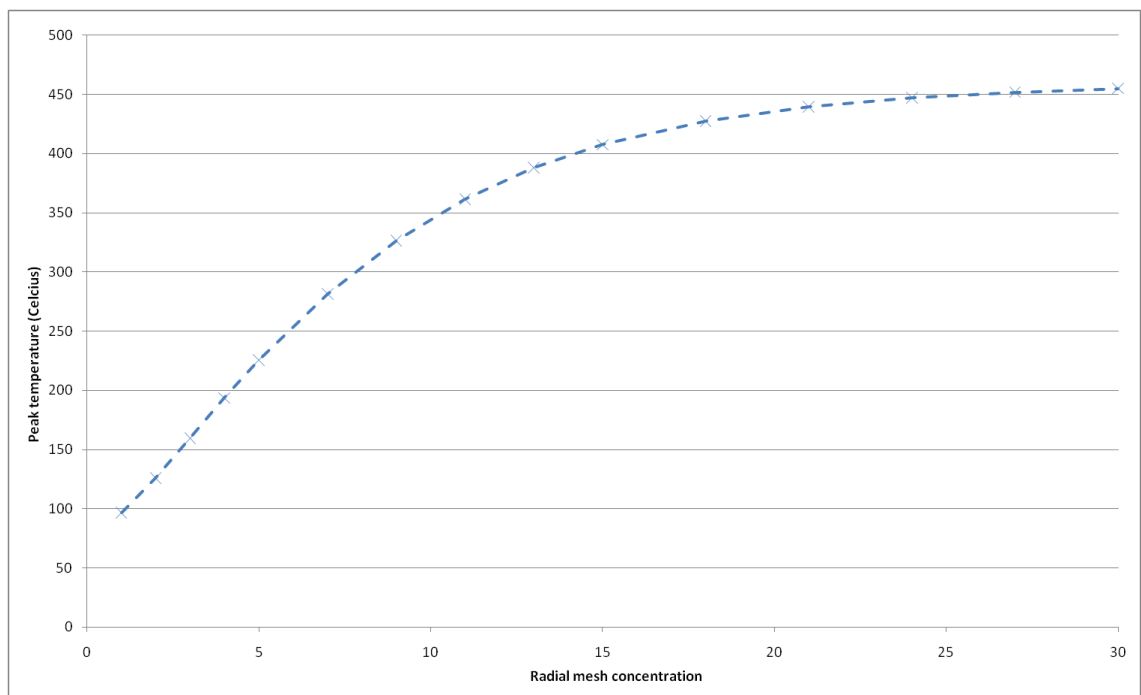


Figure 3-22: Surface temperature variation around the circumference, varying with resolution. Location of the roll bite is indicated by the thick red line on the abscissa.

The previous two sections resolved the issues presented by Figure 3-11, but did not address the troubling variations observed in the temperature profiles of Figure 3-10. The problem in this case is illustrated in Figure 3-22, in which each data set represents a shell model whose radial resolution is finer than the core model by a magnitude, as indicated in the

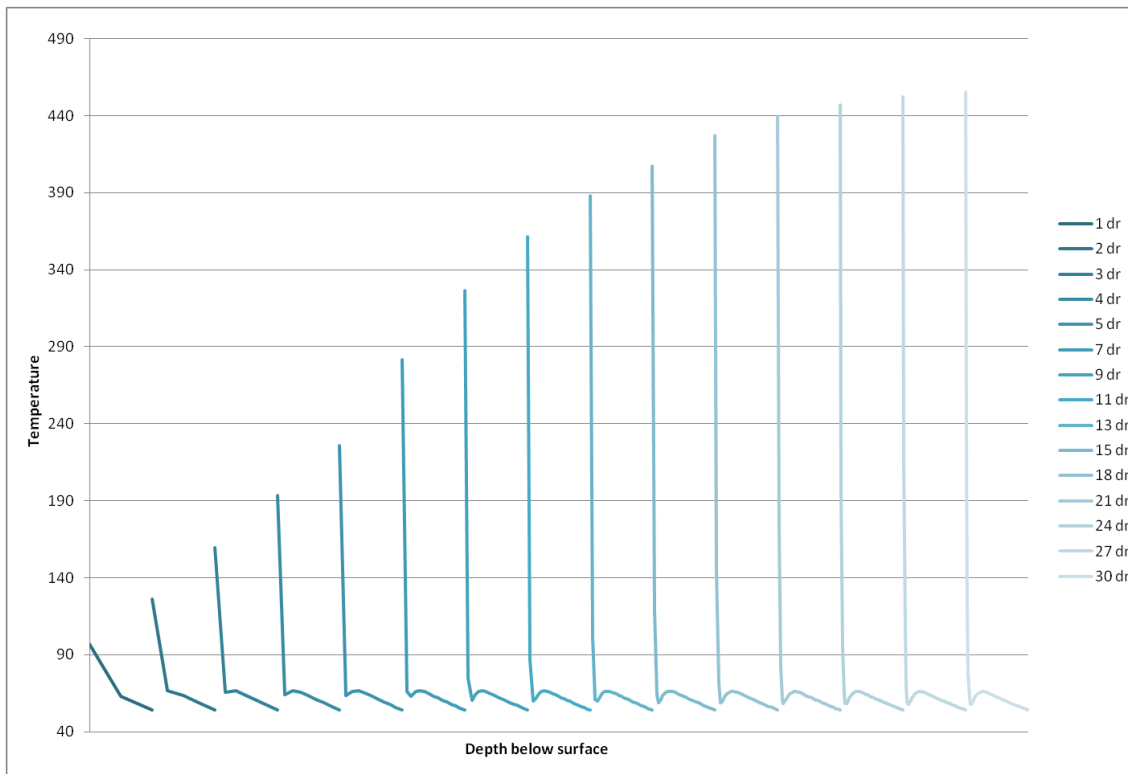
legend. The darkest line is 1:1 with the core model, whereas the lightest line has thirty times the radial resolution, i.e. a thirtieth of the mesh spacing. Average heating and cooling heat transfer coefficients were applied to keep the boundary conditions simple and so that the adjustment factors could be disabled. This reduces the chance of any factor other than radial resolution influencing the results.

The response of the model to changes in resolution, as shown in Figure 3-22, is indicative of one of two possible issues. The first potential issue is that the resolution of the shell model was not sufficient to accurately define the temperature distribution. The second potential issue is growing instability.



**Figure 3-23: Change in maximum temperature with radial mesh concentration**

Figure 3-23 was produced to check whether the change in peak temperature was due to a feedback mechanism or runaway instability. If feedback or instability were the cause it would be reasonable to assume that the effects would become more pronounced as the resolution continued to rise. Instead, Figure 3-23 shows that beyond five times the density of the core model, greater increases are required to achieve the same increase in peak temperature. The figure would therefore seem to suggest that higher radial resolutions are required to adequately capture the circumferential variation.



**Figure 3-24: Offset radial temperature change at the roll bite exit for various resolutions**

Figure 3-24 shows radial change in temperature below the roll bite, with the data sets separated for easier interpretation. For each individual data set the left hand side gives the surface temperature and the right hand side gives the interface between the shell and core models. Since the core model temperatures are fixed, the right hand temperature is always the same. By contrast, the left hand temperature changes drastically, following the trend shown in Figure 3-23.

What Figure 3-24 reveals is a feature of the temperature profile which only comes into being once the resolution of the shell model is fine enough to outline it. The model on the far left is too coarse to capture any detail below the surface temperature, but by the fourth model a second peak is clearly developing. Further increases in resolution reveal no further details but act to better define both the surface temperature peak and the sub-surface peak. However, to have confidence that the sub-surface peak is a valid feature which has previously been hidden by low resolution, it should be explainable by thermal considerations.

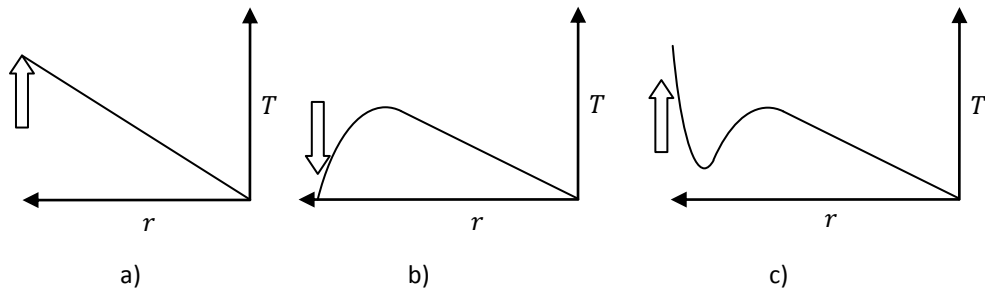


Figure 3-25: Development of a sub-surface temperature peak

Figure 3-25 shows how such a sub-surface peak could be formed. In part a) a heat source is applied to the surface (left of the graph) for an extended period of time, transferring a lot of heat to the material below the surface. Part b) takes the temperature distribution from part a) and cools the surface for a short period of time, quickly reducing the surface temperature and conducting away heat from a short distance below the surface. Since the time period is short, a considerable amount of the heat energy remains behind. This is equivalent to a slice of the work roll (which has been working for some time) passing beneath the cooling water jets. In part c) a strong heat source is then applied to the surface for a very small amount of time. Although the strong heat source makes a big difference very near to the surface, the time period is so short that the heat does not have time to diffuse into the material and fill the dip created by the short cooling period which preceded it.

The Fourier number for the short time period scenario can be calculated in order to provide some theoretical backing to the proposal that the time period required for one rotation is too short for significant heat penetration. If the Fourier number is large (unity or higher) then the material is likely to experience significant temperature change according to its Biot number. If the Fourier number for the whole roll is determined, the characteristic length will be the radius of the roll, so that the fourier number becomes;

$$F_o = \frac{\alpha t}{R^2} \quad \text{Equation 3-26}$$

Using the data in Appendix D.1;

$$\alpha = \frac{k}{\rho C_p} = \frac{25}{7800 \times 410} = 7.82 \times 10^{-6} \quad \text{Equation 3-27}$$

$$t = \frac{1}{\omega} = \frac{1}{12.5} = 0.08 \quad \text{Equation 3-28}$$

$$F_o = \frac{\alpha t}{R^2} = \frac{7.82 \times 10^{-6} \times 0.08}{0.351425^2} = 5.06 \times 10^{-6}$$

Equation 3-29

A Fourier number of  $5.06 \times 10^{-6}$  is very small, supporting the idea that the radial temperature profile does not have time to significantly change during one roll revolution. At this rotational speed, a Fourier number of unity is obtained with a length scale of 0.79mm, which suggests that the layer experiencing transient changes on the scale of the time for one revolution is of the order of a millimetre deep. This is in line with the conclusions drawn by White<sup>3</sup> who determined heat penetration depths based on the classic solution for one-dimensional heat conduction into a semi-infinite body.

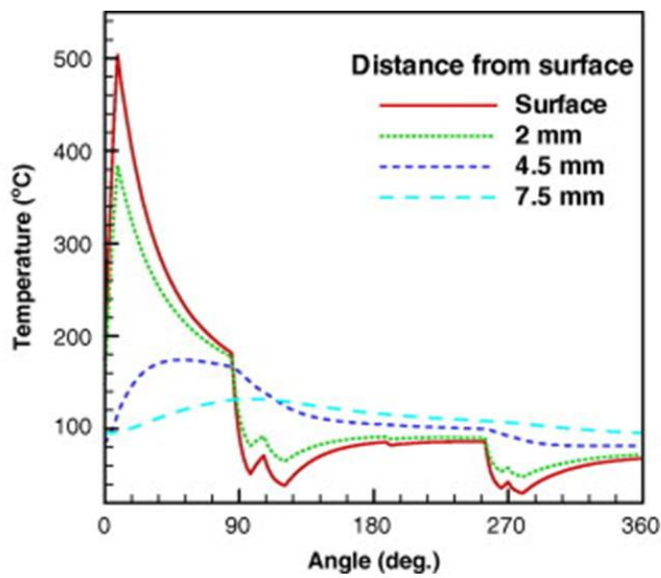
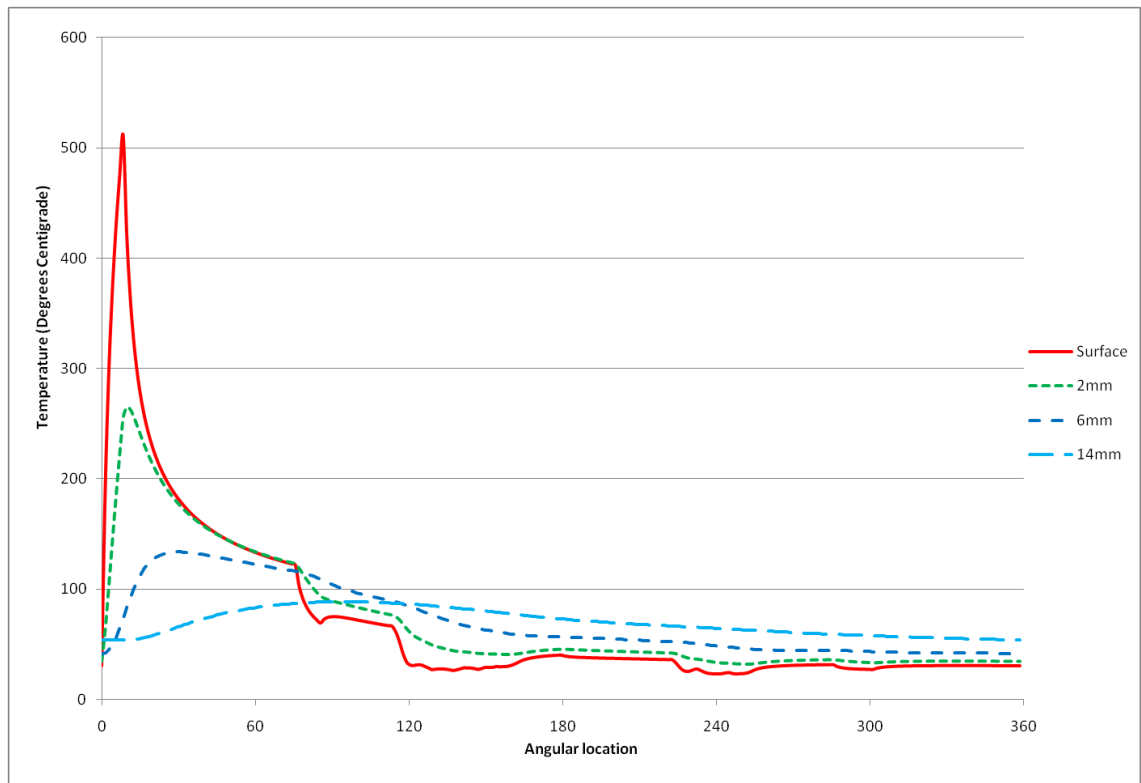


Figure 3-26: Circumferential variation in temperatures at various depths below the roll surface<sup>21</sup>





**Figure 3-27: Circumferential variation in temperatures at various depths below the roll surface**

Further supporting evidence can be found by comparing the high resolution results to the temperature profiles found in the literature. Figure 3-26 shows a thermal result from the work of Abbaspour and Saboonchi [21], using parameters derived from experimental data from the Mobarakeh Steel Complex. Figure 3-27 shows the equivalent output from the shell model. There are some differences between the two profiles, but the character and magnitude of both results are very similar. The main differences are likely due to differences in boundary conditions, since both the shell model and [21] use empirically determined heat transfer coefficients. The similarity between these two results provides further reassurance that the sharp surface temperature peak is not merely the product of runaway instability.

Table 3-3 below shows the material and geometric properties used by both models. In this table,  $\alpha$  is the thermal diffusivity,  $\rho$  is the material density,  $C_p$  is the heat capacity,  $\lambda$  is the thermal expansion coefficient,  $L$  is the barrel length,  $R_b$  is the backup roll radius and  $R_w$  is the work roll radius. The shallower heat penetration in [21] would suggest a higher work roll angular velocity and would normally result in lower surface temperatures. The difference in thermal diffusivity would allow more heat into the work roll in [21], explaining why this is not the case.

	[21]	this study
$\alpha$ ( $m^2/s$ )	$17 \times 10^{-6}$	$7.82 \times 10^{-6}$
$\rho$ ( $kg/m^3$ )	7800	7800
$C_p$ ( $J/kg K$ )	444	410
$\lambda$ ( $1/^\circ C$ )	12.06	N/A
$L$ ( $mm$ )	2050	2000
$R_b$ ( $mm$ )	718	N/A
$R_w$ ( $mm$ )	320 – 370	343

Table 3-3: Comparison of material properties

### 3.5. Recommendations from IJmuiden – September 2010

---

#### 3.5.1. Temperature Dependent Material Properties

The primary recommendation from the IJmuiden RD&T department was that temperature variability of the material properties should be taken into account. This change needed to be implemented with care, since the values of the material properties are involved in calculating the coefficient values used by the solver loop (See Chapter 2). Recalculating the coefficient values with every iteration would drastically reduce the efficiency of the model, resulting in potentially impractical run times. As such the solver equation needed to be reformulated to isolate those terms which need to be changed with each iteration.

NB – The following work describes the formulation of the models in the "CoreModelTDP.cs" and "ShellModelTDP.cs" code files. Both files have been superseded by the "ConservativeCoreModel.cs" and "ConservativeShellModel.cs" files, whose formulation is described in section 3.6 (Page 115). The following description is included because the original models were used extensively in testing the stress solution.

#### Core Model – General equation

---

The differential form of the basic equation for the 2D model is shown below.

$$\frac{\partial T}{\partial t} = \alpha \left( \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial r^2} + \frac{\partial^2 T}{\partial z^2} \right)$$

Making use of finite difference approximations for the differential terms and using the Crank-Nicolson method transforms the partial differential equation into the form shown below.

$$\begin{aligned} \frac{T_{i,j}^{K+1} - T_{i,j}^K}{\Delta t} = & \frac{\alpha}{2} \left( \frac{T_{i,j+1} + T_{i,j-1} - 2T_{i,j}}{\Delta r^2} + \frac{T_{i,j+1} - T_{i,j-1}}{2j\Delta r^2} + \frac{T_{i+1,j} + T_{i-1,j} - 2T_{i,j}}{\Delta z^2} \right)^{(K+1)} \\ & + \frac{\alpha}{2} \left( \frac{T_{i,j+1} + T_{i,j-1} - 2T_{i,j}}{\Delta r^2} + \frac{T_{i,j+1} - T_{i,j-1}}{2j\Delta r^2} \right. \\ & \left. + \frac{T_{i+1,j} + T_{i-1,j} - 2T_{i,j}}{\Delta z^2} \right)^{(K)} \end{aligned}$$

Equation 3-30

This can be rearranged to collect the temperature terms and isolate  $T_{i,j}^{K+1}$ .

$$\begin{aligned} T_{i,j}^{K+1} = & T_{i,j+1}^{K+1} \left( \frac{\frac{1}{2\Delta r^2} + \frac{1}{4j\Delta r^2}}{\frac{1}{\alpha\Delta t} + \frac{1}{\Delta r^2} + \frac{1}{\Delta z^2}} \right) + T_{i,j-1}^{K+1} \left( \frac{\frac{1}{2\Delta r^2} - \frac{1}{4j\Delta r^2}}{\frac{1}{\alpha\Delta t} + \frac{1}{\Delta r^2} + \frac{1}{\Delta z^2}} \right) \\ & + T_{i+1,j}^{K+1} \left( \frac{\frac{1}{2\Delta z^2}}{\frac{1}{\alpha\Delta t} + \frac{1}{\Delta r^2} + \frac{1}{\Delta z^2}} \right) + T_{i-1,j}^{K+1} \left( \frac{\frac{1}{2\Delta z^2}}{\frac{1}{\alpha\Delta t} + \frac{1}{\Delta r^2} + \frac{1}{\Delta z^2}} \right) \\ & + T_{i,j+1}^K \left( \frac{\frac{1}{2\Delta r^2} + \frac{1}{4j\Delta r^2}}{\frac{1}{\alpha\Delta t} + \frac{1}{\Delta r^2} + \frac{1}{\Delta z^2}} \right) + T_{i,j-1}^K \left( \frac{\frac{1}{2\Delta r^2} - \frac{1}{4j\Delta r^2}}{\frac{1}{\alpha\Delta t} + \frac{1}{\Delta r^2} + \frac{1}{\Delta z^2}} \right) \\ & + T_{i+1,j}^K \left( \frac{\frac{1}{2\Delta z^2}}{\frac{1}{\alpha\Delta t} + \frac{1}{\Delta r^2} + \frac{1}{\Delta z^2}} \right) + T_{i-1,j}^K \left( \frac{\frac{1}{2\Delta z^2}}{\frac{1}{\alpha\Delta t} + \frac{1}{\Delta r^2} + \frac{1}{\Delta z^2}} \right) \\ & + T_{i,j}^K \left( \frac{\frac{1}{\alpha\Delta t} - \frac{1}{\Delta r^2} - \frac{1}{\Delta z^2}}{\frac{1}{\alpha\Delta t} + \frac{1}{\Delta r^2} + \frac{1}{\Delta z^2}} \right) \end{aligned}$$

Equation 3-31

This equation gives a basic starting point for calculating the temperature of any point in the model. Points away from the boundaries can be solved using this equation as it is, while those not completely surrounded by material will need to implement approximations to represent thermal boundary behaviour.

As with the original solver equation, a general form is required so that a single equation can be used for any point in the model. In the non-temperature-dependent case, the values

multiplying into each temperature term were collected into single coefficients, as reiterated in Equation 3-32.

$$T_{i,j}^{K+1} = AT_{i,j+1}^{K+1} + BT_{i,j-1}^{K+1} + CT_{i+1,j}^{K+1} + DT_{i-1,j}^{K+1} + AT_{i,j+1}^K + BT_{i,j-1}^K + CT_{i+1,j}^K + DT_{i-1,j}^K + ET_{i,j}^K + F$$

Equation 3-32

However, this format cannot be used with temperature dependent material properties. To take an example, the value of the A coefficient is given by the following equation.

$$A = \frac{\frac{1}{2\Delta r^2} + \frac{1}{4j\Delta r^2}}{\frac{1}{\alpha\Delta t} + \frac{1}{\Delta r^2} + \frac{1}{\Delta z^2}}$$

Equation 3-33

Equation 3-33 contains an  $\alpha$  term, which is based on material properties and means that  $A$  would need to be recalculated for every iteration. To avoid this issue, the following definitions provide a new set of coefficients whose values will not change with temperature.

$$A = \frac{1}{2\Delta r^2} + \frac{1}{4j\Delta r^2} = \frac{2j + 1}{4j\Delta r^2}$$

$$B = \frac{1}{2\Delta r^2} - \frac{1}{4j\Delta r^2} = \frac{2j - 1}{4j\Delta r^2}$$

$$C = D = \frac{1}{2\Delta z^2}$$

$$E = \frac{1}{\Delta t}$$

$$F = -\frac{1}{\Delta r^2} - \frac{1}{\Delta z^2}$$

With these new definitions Equation 2-34 becomes;

$$T_{i,j}^{K+1} = \frac{AT_{i,j+1}^{K+1} + BT_{i,j-1}^{K+1} + CT_{i+1,j}^{K+1} + DT_{i-1,j}^{K+1}}{\frac{E}{\alpha} - F} + \frac{AT_{i,j+1}^K + BT_{i,j-1}^K + CT_{i+1,j}^K + DT_{i-1,j}^K + \left(\frac{E}{\alpha} + F\right)T_{i,j}^K}{\frac{E}{\alpha} - F}$$

Equation 3-34

Equation 3-34 is not quite in the right form to act as a general equation for the whole model since it does not take account of variations at the boundaries. To develop a more generic equation consider the substitution for position  $j + 1$  at the work roll surface.

$$T_{i,j+1} = G_0 + G_1 T_{i,j} + T_{i,j-1} \quad \text{Equation 3-35}$$

Where the  $G_0$  and  $G_1$  coefficients are given by;

$$G_0 = \frac{2\Delta r}{k} (H_0 + H_2 T_{str} + H_3 T_{fs}) \quad G_1 = \frac{2\Delta r}{k} H_1$$

A more comprehensive explanation of the surface substitution can be found in Chapter 2. These terms cannot be incorporated as they stand, since the  $k$  term represents the thermal conductivity of the material, which will change with temperature. Consider defining  $G_0$  and  $G_1$  as;

$$G_0 = 2\Delta r (H_0 + H_2 T_{str} + H_3 T_{fs}) \quad G_1 = 2\Delta r H_1$$

The new  $G_0$  and  $G_1$  definitions have removed the material property term, which must now be implemented in a new version of Equation 3-35.

$$T_{i,j+1} = \frac{G_0}{k} + \frac{G_1}{k} T_{i,j} + T_{i,j-1} \quad \text{Equation 3-36}$$

When Equation 3-36 is substituted into Equation 3-34 nearly all the necessary terms are present.

$$T_{i,j}^{K+1} = \frac{(A + B)T_{i,j-1}^{K+1} + CT_{i+1,j}^{K+1} + DT_{i-1,j}^{K+1} + (A + B)T_{i,j-1}^K + CT_{i+1,j}^K + DT_{i-1,j}^K}{\frac{E}{\alpha} - F - \frac{AG_1}{k}} + \frac{\left(\frac{E}{\alpha} + F + \frac{AG_1}{k}\right)T_{i,j}^K + \frac{2AG_0}{k}}{\frac{E}{\alpha} - F - \frac{AG_1}{k}}$$

$$\text{Equation 3-37}$$

There are some components of Equation 3-37 which are not completely generic. The  $A$  terms which premultiply  $G_0$  and  $G_1$  come from the elimination of the  $j + 1$  index. If the  $i + 1$  index were eliminated then  $C$  terms would replace the  $A$  terms (the  $A + B$  coefficient pairing

would also become a  $C + D$  pairing, following the elimination of the  $T_{i+1,j}$  term. However, this is already dealt with by altering the values of  $A$ ,  $B$ ,  $C$  and  $D$  at the boundaries to reflect these changes, allowing the form of Equation 2-34 to be maintained). To account for this, the  $A$  term is taken inside the definition of  $G_0$  and  $G_1$ , making the choice of term boundary-specific.

$$G_0 = 2A\Delta r(H_0 + H_2T_{str} + H_3T_{fs}) \qquad G_1 = 2A\Delta rH_1$$

To reduce the length of the solver equation a new set of coefficients are introduced.

$$E^* = \frac{E}{\alpha} + F + \frac{G_1}{k} \qquad F^* = \frac{E}{\alpha} - F - \frac{G_1}{k} \qquad G^* = \frac{2G_0}{k}$$

This gives a final arrangement of Equation 2-34 as;

$$T_{i,j}^{K+1} = \frac{AT_{i,j+1}^{K+1} + BT_{i,j-1}^{K+1} + CT_{i+1,j}^{K+1} + DT_{i-1,j}^{K+1}}{F^*} + \frac{AT_{i,j+1}^K + BT_{i,j-1}^K + CT_{i+1,j}^K + DT_{i-1,j}^K + E^*T_{i,j}^K + G^*}{F^*} \qquad \text{Equation 3-38}$$

In Equation 3-38 the  $A$ ,  $B$ ,  $C$  and  $D$  terms contain no material property values and as such can remain constant throughout. The  $E^*$ ,  $F^*$  and  $G^*$  terms do contain material property values, but the  $E$ ,  $F$ ,  $G_0$  and  $G_1$  values do not. Only recalculating the material property values and the starred terms restricts the number of calculations required between each iteration to a bare minimum. Equation 3-38 therefore functions as a superior general equation for the temperature-dependent core model.

### *Core Model – Axisymmetric symmetry*

---

Equation 3-38 can be used to calculate the temperature of any point within the material and along most boundary conditions with the use of a substitution for any virtual temperature value, as described above and in Chapter 2. However, along the rotational axis the  $j$  coordinate, which represents the radial distance from the rotational axis, will be zero. Since this would result in singularity, a different formulation of the general equation is required for every point at this  $j$  coordinate.

$$\begin{aligned}
& \pi \frac{\Delta r^2}{4} \Delta z \rho C_p \frac{T_{i,0}^{K+1} - T_{i,0}^K}{\Delta t} \\
&= \frac{1}{2} \left[ \pi \frac{\Delta r^2}{4} k \left( \frac{T_{i+1,0}^{K+1} - T_{i,0}^{K+1}}{\Delta z} - \frac{T_{i,0}^{K+1} - T_{i-1,0}^{K+1}}{\Delta z} \right) + \pi \Delta z \Delta r k \frac{T_{i,1}^{K+1} - T_{i,0}^{K+1}}{\Delta r} \right] \\
&+ \frac{1}{2} \left[ \pi \frac{\Delta r^2}{4} k \left( \frac{T_{i+1,0}^K - T_{i,0}^K}{\Delta z} - \frac{T_{i,0}^K - T_{i-1,0}^K}{\Delta z} \right) + \pi \Delta z \Delta r k \frac{T_{i,1}^K - T_{i,0}^K}{\Delta r} \right]
\end{aligned} \tag{Equation 3-39}$$

Equation 3-39 presents the heat flow equation as developed in Chapter 2, before it was rearranged to collect static terms.

$$\begin{aligned}
& T_{i,0}^{K+1} \\
&= \frac{\frac{\Delta r^2}{4\Delta z^2} T_{i+1,0}^{K+1} + \frac{\Delta r^2}{4\Delta z^2} T_{i-1,0}^{K+1} + T_{i,1}^{K+1}}{\frac{\Delta r^2}{2\Delta t\alpha} + \frac{\Delta r^2}{2\Delta z^2} + 1} \\
&+ \frac{\frac{\Delta r^2}{4\Delta z^2} T_{i+1,0}^K + \frac{\Delta r^2}{4\Delta z^2} T_{i-1,0}^K + T_{i,1}^K + \left( \frac{\Delta r^2}{2\Delta t\alpha} - \frac{\Delta r^2}{2\Delta z^2} - 1 \right) T_{i,0}^K}{\frac{\Delta r^2}{2\Delta t\alpha} + \frac{\Delta r^2}{2\Delta z^2} + 1}
\end{aligned} \tag{Equation 3-40}$$

Equation 3-40 shows how Equation 3-39 can be rearranged in such a way as to reduce the number of occurrences of the material property terms.  $\alpha$  can be substituted for  $k/\rho C_p$  since these terms all appear together.

$$\begin{aligned}
T_{i,j}^{K+1} &= \frac{A_2 T_{i,j+1}^{K+1} + B_2 T_{i,j-1}^{K+1} + C_2 T_{i+1,j}^{K+1} + D_2 T_{i-1,j}^{K+1}}{F_2^*} \\
&+ \frac{A_2 T_{i,j+1}^K + B_2 T_{i,j-1}^K + C_2 T_{i+1,j}^K + D_2 T_{i-1,j}^K + E_2^* T_{i,j}^K}{F_2^*}
\end{aligned} \tag{Equation 3-41}$$

The following substitutions provide Equation 3-41, which simplifies the equation by collecting static values into single coefficients.

$$\begin{aligned}
A_2 &= 1 & B_2 &= 0 & C_2 &= \frac{\Delta r^2}{4\Delta z^2} \\
D_2 &= C_2 & E_2^* &= \frac{\Delta r^2}{2\Delta t\alpha} - 2C_2 - A_2 & F_2^* &= \frac{\Delta r^2}{2\Delta t\alpha} + 2C_2 + A_2
\end{aligned}$$

$$\tag{Equation 3-42}$$

As with the previous general equation, further modifications are needed to allow for boundary conditions which feature the substitution of an approximate heat flow at a boundary condition. Consider the substitution required to emulate heat loss from the journal edge.

$$T_{i+1,j} = T_{i-1,j} - \frac{2h_{convB}\Delta Z}{k}(T_{i,j} - T_{fB}) \quad \text{Equation 3-43}$$

When Equation 3-43 is substituted into Equation 3-40 the following equation is obtained.

$$\begin{aligned} T_{i,0}^{K+1} &= \frac{\frac{\Delta r^2}{2\Delta Z^2} T_{i-1,0}^{K+1} + T_{i,1}^{K+1} + \frac{\Delta r^2}{2\Delta Z^2} T_{i-1,0}^K + T_{i,1}^K}{\frac{\Delta r^2}{2\Delta t\alpha} + \frac{\Delta r^2}{2\Delta Z^2} + 1 + \frac{2h_{convB}\Delta Z}{k} \left( \frac{\Delta r^2}{4\Delta Z^2} \right)} \\ &+ \frac{\left( \frac{\Delta r^2}{2\Delta t\alpha} - \frac{\Delta r^2}{2\Delta Z^2} - 1 - \frac{2h_{convB}\Delta Z}{k} \left( \frac{\Delta r^2}{4\Delta Z^2} \right) \right) T_{i,0}^K + \frac{T_{fB}h_{convB}\Delta r^2}{k\Delta Z}}{\frac{\Delta r^2}{2\Delta t\alpha} + \frac{\Delta r^2}{2\Delta Z^2} + 1 + \frac{2h_{convB}\Delta Z}{k} \left( \frac{\Delta r^2}{4\Delta Z^2} \right)} \end{aligned} \quad \text{Equation 3-44}$$

Equation 3-44 can be simplified down to a general form similar to Equation 3-41 with one additional coefficient definition and some minor alterations to the starred terms.

$$\begin{aligned} T_{i,j}^{K+1} &= \frac{A_2 T_{i,j+1}^{K+1} + B_2 T_{i,j-1}^{K+1} + C_3 T_{i+1,j}^{K+1} + D_3 T_{i-1,j}^{K+1}}{F_2^*} \\ &+ \frac{A_2 T_{i,j+1}^K + B_2 T_{i,j-1}^K + C_3 T_{i+1,j}^K + D_3 T_{i-1,j}^K + E_2^* T_{i,j}^K + G_2^*}{F_2^*} \end{aligned} \quad \text{Equation 3-45}$$

Where

$$\begin{aligned} C_3 &= 0 & D_3 &= 2D_2 & E_2^* &= \frac{\Delta r^2}{2\Delta t\alpha} - 2C_2 - 1 - \frac{G_1}{k} \\ F_2^* &= \frac{\Delta r^2}{2\Delta t\alpha} + 2C_2 + 1 + \frac{G_1}{k} & G_2^* &= \frac{G_0}{k} & G_0 &= \frac{T_{fB}h_{convB}\Delta r^2}{\Delta Z} \\ G_1 &= 2Ch_{convB}\Delta Z \end{aligned} \quad \text{Equation 3-46}$$

By changing the values of  $G_0$  and  $G_1$  various boundary conditions can be affected, meeting the requirement for universal applicability. Equation 3-45 also matches Equation 3-38 in structure, meaning that the same arrays can be used in both cases and the same equation can



perform both calculations, with the only change being the input data. This completes the mathematical reworking of the core model to allow temperature dependency.

### Shell Model – General equation

The partial differential equation describing heat flow in the shell model is given in Equation 3-47 below.

$$T' = \alpha \left( \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial r^2} + \frac{1}{r^2} \frac{\partial^2 T}{\partial \theta^2} + \frac{\partial^2 T}{\partial z^2} \right) - \omega \frac{\partial T}{\partial \theta} \quad \text{Equation 3-47}$$

The equation is significantly different to the equation describing heat flow in the core model. The number of terms is increased by the extra dimension being considered and decreased by the lack of time, eliminating the  $K + 1$  timestep. In Chapter 2, Equation 3-47 was expanded with finite different approximations and simplified to a level that was easily digestible by a solver equation.

$$\begin{aligned} & \left( 2 + \frac{2\Delta r^2}{\Delta z^2} + \frac{\omega \Delta r^2}{\alpha \Delta \theta} + \frac{2}{(j_{offset} + j)^2 \Delta \theta^2} \right) T_{i,j,k} \\ &= \left( \frac{1}{2j_{offset} + 2j} + 1 \right) T_{i,j+1,k} + \left( 1 - \frac{1}{2j_{offset} + 2j} \right) T_{i,j-1,k} \\ &+ \left( \frac{1}{(j_{offset} + j)^2 \Delta \theta^2} \right) T_{i,j,k+1} + \left( \frac{1}{(j_{offset} + j)^2 \Delta \theta^2} + \frac{\omega \Delta r^2}{\alpha \Delta \theta} \right) T_{i,j,k-1} \\ &+ \left( \frac{\Delta r^2}{\Delta z^2} \right) T_{i+1,j,k} + \left( \frac{\Delta r^2}{\Delta z^2} \right) T_{i-1,j,k} - \frac{\Delta r^2}{\alpha} T' \end{aligned} \quad \text{Equation 3-48}$$

Equation 3-48 presents the equation at its most reduced without trapping material property terms inside complex coefficients. The coefficients can now be reformulated to isolate the material values.

$$\begin{aligned} A &= \frac{\Delta r^2}{\Delta z^2} & B &= A & C &= 1 + \frac{1}{2j_{offset} + 2j} \\ D &= 1 - \frac{1}{2j_{offset} + 2j} & E &= \frac{1}{(j_{offset} + j)^2 \Delta \theta^2} & F_a &= \frac{\omega \Delta r^2}{\Delta \theta} \\ G &= -\Delta r^2 T' & H &= 2 + 2A + 2E \end{aligned} \quad \text{Equation 3-49}$$

Substituting the coefficients from Equation 3-49 into Equation 3-38 gives;

$$T_{i,j,k} = \frac{AT_{i+1,j,k} + BT_{i-1,j,k} + CT_{i,j+1,k} + DT_{i,j-1,k} + ET_{i,j,k+1}}{H + \frac{F_a}{\alpha}} + \frac{\left(E + \frac{F_a}{\alpha}\right)T_{i,j,k-1} + \frac{G}{\alpha}}{H + \frac{F_a}{\alpha}}$$

Equation 3-50

For the sake of keeping the solver equation simple, temperature varying coefficients can also be introduced.

$$F^* = E + \frac{F_a}{\alpha} \qquad G^* = \frac{G}{\alpha} \qquad H^* = H + \frac{F_a}{\alpha}$$

Equation 3-51

Giving

$$T_{i,j,k} = \frac{AT_{i+1,j,k} + BT_{i-1,j,k} + CT_{i,j+1,k} + DT_{i,j-1,k} + ET_{i,j,k+1} + F^*T_{i,j,k-1} + G^*}{H^*}$$

Equation 3-52

As with the previous general equations, these coefficient definitions need to be expanded to allow for variations at the model boundaries. The surface temperature substitution that will provide the template for these variations is given below.

$$T_{i,j+1,k} = T_{i,j-1,k} + \frac{2\Delta r}{k} \dot{q}_{friction}(\theta) - \frac{2\Delta r h_{str}}{k} (T_{i,j,k} - T_{str[i,k]}) - \frac{2\Delta r h_{spr[i,k]}}{k} (T_{i,j,k} - T_{spr})$$

Equation 3-53

Substituting Equation 3-53 into Equation 3-50 gives;

$$\begin{aligned}
& \left( H + \frac{F_a}{\alpha} + \frac{2C\Delta r}{k} (h_{str} + h_{spr[i,k]}) \right) T_{i,j,k} \\
& = (A)T_{i+1,j,k} + (B)T_{i-1,j,k} + (D + C)T_{i,j-1,k} + (E)T_{i,j,k+1} \\
& + \left( E + \frac{F_a}{\alpha} \right) T_{i,j,k-1} + \frac{G}{\alpha} \\
& + \frac{2C\Delta r}{k} (\dot{q}_{friction}(\theta) + T_{str[i,k]}h_{str} + T_{spr}h_{spr[i,k]})
\end{aligned}$$

Equation 3-54

Equation 3-54 can be made to yield Equation 3-52 given that;

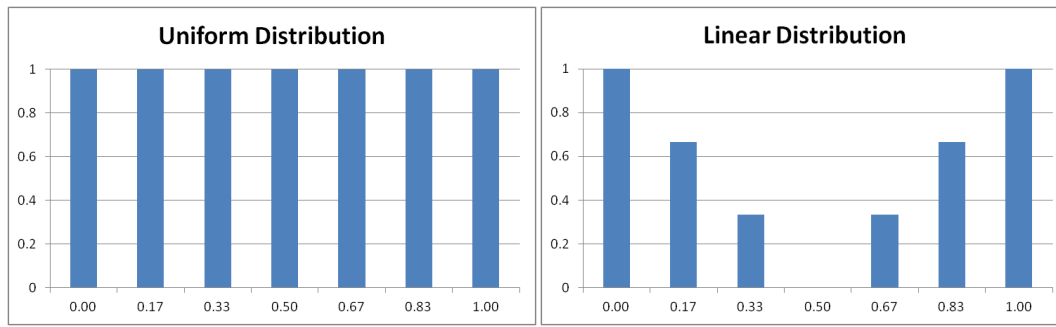
$$\begin{aligned}
C &= 0 & D &= C + D & H^* &= H + \frac{F_a}{\alpha} + \frac{G_B}{k} \\
G_T &= 2C\Delta r (\dot{q}_{friction}(\theta) + T_{str[i,k]}h_{str} + T_{spr}h_{spr[i,k]}) & G^* &= \frac{G}{\alpha} + \frac{G_T}{k} \\
G_B &= 2C\Delta r (h_{str} + h_{spr[i,k]})
\end{aligned}$$

Equation 3-55

Equation 3-55 gives the coefficient definitions required to give the temperature dependent shell model's general equation. The definitions of  $G_T$  and  $G_B$  can be altered to reflect a boundary condition or set to zero for points within the volume, while  $A$ ,  $B$ ,  $C$  and  $D$  can change accordingly. Only the starred terms need to be recalculated with each iteration, and within those terms only the  $\alpha$  and  $k$  values will change. Due to the design of the shell model, with  $k = 0$  and  $k = k_{max}$  referring to the same location (the shell model represents a tube of material, so  $k = 0$  and  $k_{max}$  represent the location where the "cut" was made to allow the tube to be "unrolled") and the core model results supplying the values for the inner radius, there are no more complicated boundary conditions to consider. Equation 3-52 will now account for heat transfer in the shell model in its entirety.

### 3.5.2. Roll Bite Conditions

Another concern of the IJmuiden RD&T department was that the variability of thermal effects through the roll bite was not being adequately simulated by the current configuration. Specifically, the heat input due to friction was being distributed evenly over the bite area.



**Figure 3-28: Flat and linear heat transfer distributions, normalised to 1 in magnitude (vertical scale) and across the roll bite (horizontal scale)**

Figure 3-28 shows the two distributions that were used to assess the impact of the heating distribution. The design and results of the test are considered in Chapter 4. The figure only gives normalised distributions, not giving any indication of the magnitude of the heating effect. This is intentional, since the heat generation is estimated by reference to the power consumed by the work roll drive motors. The assumption is that this power is converted into heat in the roll bite, accounting for both friction and deformation heat, and a proportion is transferred to the work roll. This heat is shared among all the locations in the roll bite. Since only the distribution of the heating must be changed, the total effect of the original heating should be the same as the total effect of the new distribution. This is expressed in equation form below.

$$\sum_{n=0}^{bitewidth} power_n = magnitude \times \sum_{n=0}^{bitewidth} f(k_n) \quad \text{Equation 3-56}$$

Equation 3-56 sums *power* (i.e. the power dissipated at each location in the roll bite) over the roll bite and equates that to the sum of  $f(k_n)$  (the normalised heating intensity at index  $k = n$ ) multiplied by a reference magnitude value. This can be rearranged to give the value of the reference magnitude.

$$magnitude = \frac{\sum_{n=0}^{bitewidth} power_n}{\sum_{n=0}^{bitewidth} f(k_n)} \quad \text{Equation 3-57}$$

A relationship that gives the linear distribution in Figure 3-28 is;

$$y = \left| 1 - \frac{2k}{bitewidth} \right| \quad \text{Equation 3-58}$$

Equation 3-58 can be used with Equation 3-57 to give the heating due to friction for  $k = 0 \rightarrow k = bitewidth$ .

$$Friction_k = \frac{\sum_{n=0}^{bitwidth} power_n}{\sum_{n=0}^{bitwidth} \left|1 - \frac{2n}{bitwidth}\right|} \times \left|1 - \frac{2k}{bitwidth}\right| \quad \text{Equation 3-59}$$

Equation 3-59 was used to test whether variable frictional heating and uniformly distributed frictional heating makes a significant difference to the thermal profile. The results of this analysis can be found in Chapter 5.

### 3.5.3. Efficiency Improvements Considered but not Implemented

During the visit to the Ilmuiden RD&T department the runtime of the model was discussed and highlighted as an area for improvement. Unfortunately the only methods available for improving the run-time of the model required extensive reworking of the code, a task which gave too little benefit for the requisite expenditure of development time. While efforts were focused on improving functionality rather than efficiency, some research was performed on potential efficiency savings. The results of that research are presented here.

#### *Sparse Matrices*

---

Although the finite difference method has been approached from a point-by-point perspective in this project, matrices are often used as a short-hand for the processes involved. The underlying theory is identical in both cases, the two approaches only vary after the general equation is formulated. For simplicity, consider an explicit one dimensional heat diffusion problem.

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} \quad \text{Equation 3-60}$$

Recalling Chapter 2, finite difference approximations can be substituted into Equation 3-60 and the resulting equation can be rearranged to isolate the temperature terms.

$$T_i^{K+1} = AT_{i+1}^K + BT_i^K + CT_{i-1}^K \quad \text{Equation 3-61}$$

In this project, this format is as far as the mathematical manipulation needs to go. The solver equation would iterate through all the values of  $i$ , calculating the result of each

simultaneous equation in turn. To use a matrix approach, the simultaneous equations must be converted into a matrix form.

$$Y = MT$$

Where  $Y$  and  $T$  are column vectors and  $M$  is a square matrix. Consider the simultaneous equations themselves.

$$\begin{aligned} T_0^1 &= \text{constant} \\ T_i^1 &= AT_{i+1}^0 + BT_i^0 + CT_{i-1}^0 \\ T_n^1 &= \text{constant} \end{aligned}$$

Considering each possible value of  $i$ , these equations can be combined into a matrix formulation.

$$\begin{bmatrix} T_0^1 \\ T_1^1 \\ T_2^1 \\ T_3^1 \\ \vdots \\ T_{n-2}^1 \\ T_{n-1}^1 \\ T_n^1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A & B & C & 0 & 0 & 0 & 0 & 0 \\ 0 & A & B & C & 0 & 0 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 & 0 & 0 \\ 0 & 0 & 0 & \ddots & \ddots & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & A & B & C \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \text{constant} \\ T_1^0 \\ T_2^0 \\ T_3^0 \\ \vdots \\ T_{n-2}^0 \\ T_{n-1}^0 \\ \text{constant} \end{bmatrix}$$

The problem above can be solved by readily available and high efficient software applications. The addition of extra dimensions complicate the matter significantly by introducing extra terms offset from the "tri-diagonal" structure shown above. There are software solutions to these "sparse" matrices, so-named for the large number of "zero" values, which become very efficient when the matrices are large. However, the solution processes are not straight forward and would require a great deal of work and extra research. As such, the choice was made to focus on those tasks which add functionality, rather than improvements to efficiency.

## Language Conversion

---

The C# language is not the most efficient language available due to its high "level". The lowest level of language is machine code, the binary commands which order hardware

components to perform certain actions. Machine code is very powerful as it has direct access to the hardware, but it would take a very long time to write a function to do a complicated task in machine code, since practically every step must be coded manually. A higher level language abstracts away hardware-specific concerns, providing built-in functions to automate the most common tasks and providing simplified interfaces for those components that the user may want more control over. The level of the language defines how far the language designers prioritise convenience and ease of use over the ability to manage small details of general processes.

A case in point would be memory allocation in C# and C++. An array in C++ or C# is a storage variable capable of holding multiple values. The array will need somewhere to store the values it is given and in both cases this means allocating a space in the memory of the computer. In C++ this allocation must be initiated by the programmer, who must also remember to de-allocate the memory if the array is no longer needed. In C# both the allocation and de-allocation are taken care of automatically, making C# the higher level language. The benefits of the C# approach are that the programmer does not need to waste time writing code to allocate and de-allocate memory for every complex data type, and the threat of leaving memory tied to a non-existent array is eliminated. The benefits of the C++ approach are that the programmer can allocate and de-allocate memory at will. When combined with the numerous possibilities offered by being able to work with memory addresses directly, the C++ approach can result in a solution which is much more optimised to an individual application.

If efficiency is the top priority and cost is no object then a lower level language would be preferable to C#. However, there is a major time consideration involved in translating a program from one language to another, especially when descending in language level. Many tasks would need to be performed which had previously been automated. For this project, the requisite time investment made language conversion an impractical option.

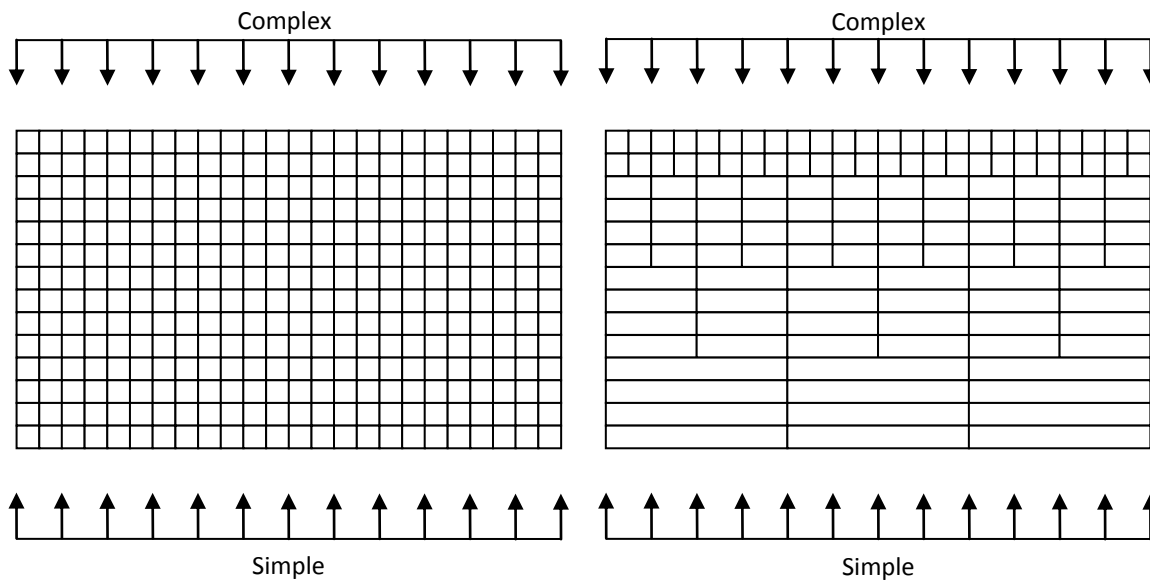


Figure 3-29: A square mesh with (right) and without (left) linking elements

An effective way of reducing the run time of the thermal models would be to reduce the number of nodes (points of information) the model is evaluating. One potential way of achieving this aim is to relax the assumption that the mesh is completely uniform in all directions. Although this will result in a drop in accuracy, the effects can be reduced by only expanding the mesh density in areas where there is only a very small thermal gradient.

By deriving a formulation of the control volume that allows for two bordering volumes along the upper surface, the axial mesh spacing could be increased for greater depths below the surface. This would greatly reduce the number of elements near the centre of the work roll, where temperatures are quite steady and thermal gradients are shallow. Figure 3-29 presents an example. Unfortunately, in the case of this project, time constraints preclude the possibility of implementing this approach.

### Graded Mesh

---

Another option for reducing the size of the mesh involves introducing a variable mesh spacing in one or more directions. By grading the size of the elements in one direction, high detail can be achieved near the surface with greatly reduced density below the surface. Errors can be minimised by ensuring that the difference in mesh spacing is constant, as shown in Figure 3-30, where each element is twice the depth of the element above it.



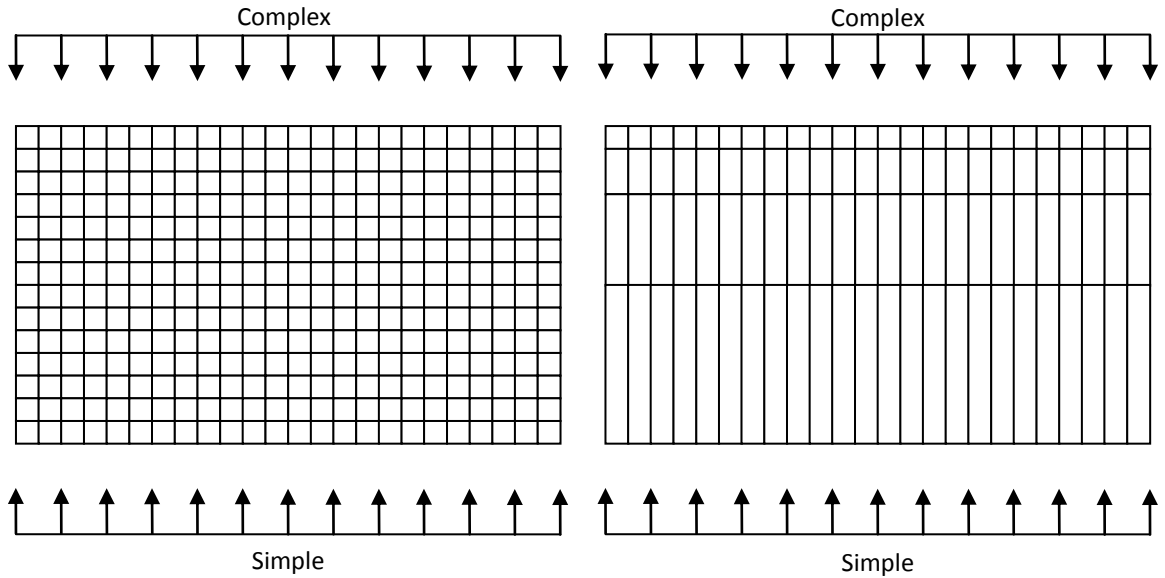


Figure 3-30: A square mesh with (right) and without (left) variable mesh spacing

As with the previous technique, the amount of existing code that would need to be altered or rewritten to implement this approach would be extensive.

### Multi-Grid

The last technique suggested to improve the efficiency of the thermal models was the multi-grid technique. This essentially involves calculating the temperature distribution for a very low resolution model, then using the results to set the initial temperature of a higher resolution model. Successive models are used until the required resolution is attained, as illustrated in the example in Figure 3-31.

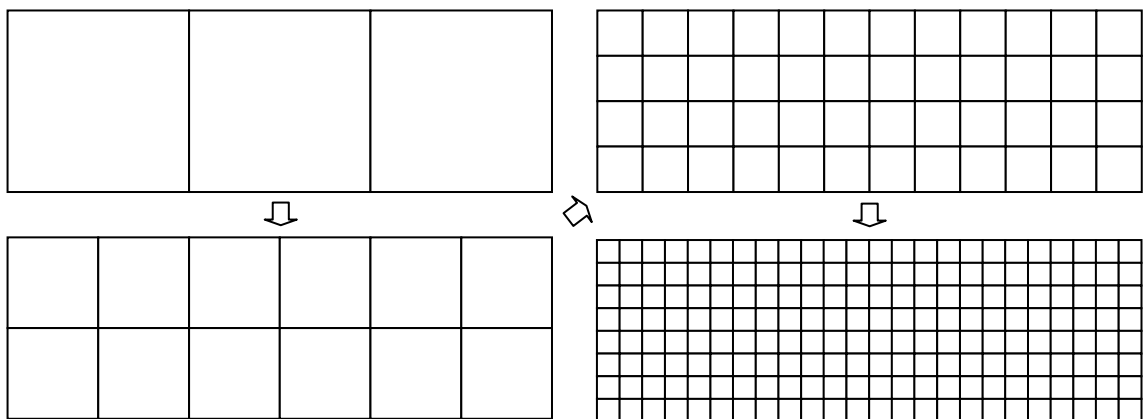


Figure 3-31: A multi-grid model sequence for a rectangular volume

By using interpolation algorithms to estimate smoothed profiles, this technique can effectively reduce run-times by providing a starting condition to each model which is much closer to the solution than a constant initial condition.

The only model this would apply to is the shell model, the core model reaches a solution too quickly for this technique to be worthwhile. While this technique would be much easier to apply it is doubtful that it would provide any quicker resolution of the final result. The core temperature distribution is already used as an initial condition, the features which take time to resolve are those formed by the fast-moving thermal sources and sinks at the surface. From previous sections it has been shown that the effect of these sources only properly resolves at higher resolutions. Therefore, it seems highly likely that earlier multi-grid models would only be able to resolve a distribution which would not actually reflect the final models temperature distribution. Improved interpolation routines or other methods of predicting the likely final temperature profile might provide a faster solution, but the time involved in developing these solutions could have been excessive.

While this technique offers the potential of speeding up the convergence rate of problems reliant on large meshes, the potential pitfalls detract too much from the potential improvements. As such, development was restricted to functionality improving measures rather than efficiency improving measures.

#### 3.5.4. Surface cooling map

It was stated in the project objectives that incorporation the surface cooling heat transfer coefficient generation facility of the TEX model from the IJmuiden RD&T department was an important objective. Following a meeting with Harry van-Steden, it was decided that the amount of work involved in combining the two software packages was prohibitive. The cooling map data output by the TEX model could already be automatically imported directly into the current software, so the benefits of the cooling maps can be gained without the time expenditure by means of cooperation.

## 3.6. Second Set of Recommendations from IJmuiden

---

### 3.6.1. Roll bite conditions

The original core and shell models used a constant HTC (heat transfer coefficient) of  $24000 \text{ W/m}^2\text{K}$ . This value was chosen empirically, to give the right balance of heating and cooling to match temperature measurements taken from a roll which had just been taken out of service. However, during a second visit to IJmuiden in October 2011, Jan de-Roo of the IJmuiden RD&T department provided a more detailed relationship<sup>29</sup>.

$$h_{str} = \frac{1}{\left( \frac{1}{50000} + \frac{d_{ox}}{\lambda_{ox}} + \sqrt{\frac{\pi t_{contact}}{\lambda_{steel}\rho_{steel}C_{p_{steel}}}} \right)} \quad \text{Equation 3-62}$$

This relationship can be split into three sections, considering that Equation 3-62 is of the form;  $\frac{1}{h_{str}} = \frac{1}{A} + \frac{1}{B} + \frac{1}{C}$ ;

A – The contact HTC is  $50000 \text{ W/m}^2\text{K}$ . This seems much higher than the original, but is reduced by the contributions of the following two terms.

B – The effect of the oxide layer which coats the surface of the work roll can be given by  $\frac{\lambda_{ox}}{d_{ox}}$ . The oxide layer thickness varies through the roll bite, but is believed to be between  $3\mu\text{m}$  and  $10\mu\text{m}$  thick<sup>4</sup>. Both thicknesses were tested and the results are shown in Chapter 6 (Page 258).

C – The term  $\sqrt{\frac{\lambda_{steel}\rho_{steel}C_{p_{steel}}}{\pi t_{contact}}}$  is the most complicated, being derived from a one-dimensional transient heat conduction problem<sup>30</sup> which used the following equation to describe the change in temperature.

$$\frac{\partial^2 T}{\partial x^2} = \frac{1}{\alpha} \frac{\partial T}{\partial t} \quad \text{Equation 3-63}$$

The problem described a semi-infinite body starting at a uniform temperature  $T_i$  and with the surface held at a constant temperature  $T_s$  from time  $t = 0$ . Using  $\eta = \frac{x}{\sqrt{4\alpha t}}$  as a substitute coordinate transformation, to combine  $x$  and  $t$  into one "similarity variable", gives (after simplification);

$$\frac{d^2T}{d\eta^2} = -2\eta \frac{dT}{d\eta} \quad \text{Equation 3-64}$$

This can be integrated by introducing  $w = dT/d\eta$  to give;

$$\frac{dw}{d\eta} = -2\eta w \rightarrow \frac{dw}{w} = -2\eta d\eta \rightarrow \ln w = -\eta^2 + C_0 \rightarrow w = C_1 e^{-\eta^2} \quad \text{Equation 3-65}$$

where  $C_1 = \ln C_0$ . Substituting back for  $w = dT/d\eta$  and using the boundary conditions of  $T = T_i$  at  $t = 0$  and  $T = T_s$  at  $t = \infty$ , given that  $\eta = 0$  when  $t = 0$  and  $\eta = \infty$  when  $t = \infty$ , gives;

$$T = C_1 \int_0^\eta e^{-u^2} du + C_2 \quad \text{Equation 3-66}$$

where  $u$  is a dummy integration variable. The boundary condition at  $t = 0$  gives  $C_2 = T_s$ , and the condition at  $t = \infty$  gives;

$$C_1 = \frac{2(T_i - T_s)}{\sqrt{\pi}} \quad \text{Equation 3-67}$$

The heat transfer coefficient component emerges from the boundary condition at the surface, by substituting relationships for  $\frac{dT}{d\eta}$  and  $\frac{\partial \eta}{\partial x}$ .

$$\dot{Q}_{str} = -k \frac{\partial T}{\partial x} \Big|_{x=0} = -k \frac{dT}{d\eta} \frac{\partial \eta}{\partial x} \Big|_{\eta=0} = -k C_1 e^{-\eta^2} \frac{1}{\sqrt{4\alpha t}} \Big|_{\eta=0} \quad \text{Equation 3-68}$$

Substituting for  $C_1$ ;

$$\dot{Q}_{str} = \frac{k(T_s - T_i)}{\sqrt{\pi\alpha t}} \quad \text{Equation 3-69}$$

[30] produces a set of results using this formulation, but if  $\alpha = \frac{k}{\rho C_p}$  then it is possible to rearrange Equation 3-69.

$$\dot{Q}_{str} = \frac{k(T_s - T_i)}{\sqrt{\frac{\pi t k}{\rho C_p}}} = \frac{(T_s - T_i)}{\sqrt{\frac{\pi t}{k \rho C_p}}} \quad \text{Equation 3-70}$$

Given that the boundary heat flow from the strip in the thermal models is given by;

$$\dot{Q}_{str} = h_{str}(T_{str} - T_{i,j}) \quad \text{Equation 3-71}$$

where  $h_{str}$  is the heat transfer coefficient,  $T_{str}$  is the strip temperature and  $T_{i,j}$  is the temperature of the work roll at the surface, Equation 3-70 would give a heat transfer coefficient of;

$$h_{str} = \frac{1}{\sqrt{\frac{\pi t}{k \rho C_p}}} = \sqrt{\frac{k \rho C_p}{\pi t}} \quad \text{Equation 3-72}$$

So the derivation of term  $C$  can be understood as emerging from Fourier's Law, given the coordinate transform required to turn a one-dimensional transient problem into an ordinary differential equation.

In the core model the temperature difference used at the surface is the difference between the strip and work roll circumferentially averaged temperature. Equation 3-72 therefore provides a good method for allowing the core model surface heat flow to reflect changing conditions through the roll bite. However, the temperature difference used at the surface of the shell model is the difference between the strip temperature and the work roll temperature at that point. The shell model includes circumferential variation, so the temperature difference will change through the roll bite. Therefore, there is no need to include Equation 3-72 in the heat transfer coefficient of the shell model.

$$h_{str} = \frac{1}{\left(\frac{1}{50000} + \frac{d_{ox}}{\lambda_{ox}}\right)} \quad \text{Equation 3-73}$$

To ensure that both models are presenting the same behaviour, the shell model strip conduction term is forced to give the same overall effect as the core model strip conduction term. This is achieved using a modified form of Equation 3-71.

$$\dot{Q}_{str} = h_{strAdjust} \times h_{str}(T_{str} - T_{i,j}) \quad \text{Equation 3-74}$$

where  $h_{strAdjust}$  is an adjustment factor given by;

$$h_{strAdjust} = \frac{\bar{Q}_{core_i}}{\bar{Q}_{shell_i}} \quad \text{Equation 3-75}$$

where  $\bar{Q}_{core_i}$  emulates the total heat flow into the core model at axial index position  $i$ , and  $\bar{Q}_{shell_i}$  is the total heat flow into the shell model at axial index position  $i$ . The two total heat flow terms can be found using the following equations;

$$\bar{Q}_{core_i} = h_{str} \left( T_{str} - \frac{\sum_{k=1}^{kSize} \bar{T}_{i,j,k}}{kSize} \right) \quad \text{Equation 3-76}$$

and

$$\bar{Q}_{shell_i} = \frac{\sum_{k=1}^{kSize} h_{str}(T_{str} - T_{i,j,k})}{kSize} \quad \text{Equation 3-77}$$

where  $T_{i,j,k}$  is the temperature of the shell model at the surface and  $kSize$  is the shell model circumferential length in node points. The surface area of the elements are not considered, since the total area of the shell model surface would equal the total area of the core model surface and cancel out in Equation 3-75 anyway. The definition of  $h_{str}$  used by Equation 3-76 is given by Equation 3-62, whereas the definition of  $h_{str}$  used by Equation 3-77 is given by Equation 3-73. The difference between using a circumferentially averaged temperature and a complete circumferential temperature profile should be accounted for by the different definitions of  $h_{str}$ . Therefore, the further the value of  $h_{strAdjust}$  is from 1, the greater the difference between the effects of term C of Equation 3-62 and the effects of the circumferential temperature variation in the shell model.

With a total heat flow in the core model of 0.898MW,  $h_{strAdjust}$  was found to have a value of between 1.1 and 1.15 along the roll barrel, corresponding to a 10-15% difference between the heating experienced by the core and shell models. The total heat flow into the shell model was 0.899MW, a difference of .07%, showing that the adjustment factor is successfully ensuring that both models experience very nearly the same heat flow.

### 3.6.2. Cooling conditions

The cooling of the work roll surface is very uneven, with large amounts of water flow focused in some areas while lacking in others. This presents a problem when using two thermal models in concert, where the assumptions about circumferential behaviour differ. A model whose surface temperature difference is calculated from circumferentially averaged temperatures (roughly 90°C in general) will predict a very different amount of cooling than a model whose surface temperature can reach 500°C in one area while dropping to the temperature of the coolant in another.

Earlier in the chapter the core model was assumed to be the more accurate, since the heat fluxes had been balanced to give the right temperature distribution at the surface after rolling, based on empirical data. The approach taken by the new strip conduction boundary condition is to provide accurate heat transfer coefficients and then adjust the average values to account for temperature variability, i.e. to assume that the shell model is the more accurate and ensure that the core model matches it in the circumferential average. The new approach is more rigorous and therefore more satisfying, but requires that the cooling boundary conditions are adjusted in a similar way.

The problem with adjusting the core model cooling to reflect the angular variations of the shell model cooling, is that the core model is the first model in the sequence to be run. A crude solution to this problem was found by first turning off those segments of code which forced the shell model cooling to have the same effect as the core model cooling. The resulting shell model would be based on an unrealistic boundary temperature but would have a realistic cooling applied at the surface. The temperature results were saved to file as temperature differences, i.e.  $T = T_{i,j,k} - \bar{T}_{i,j}$ . The model was then rerun and the core model could make use of the saved file to predict what the temperature deviations from the mean would be in the shell model.

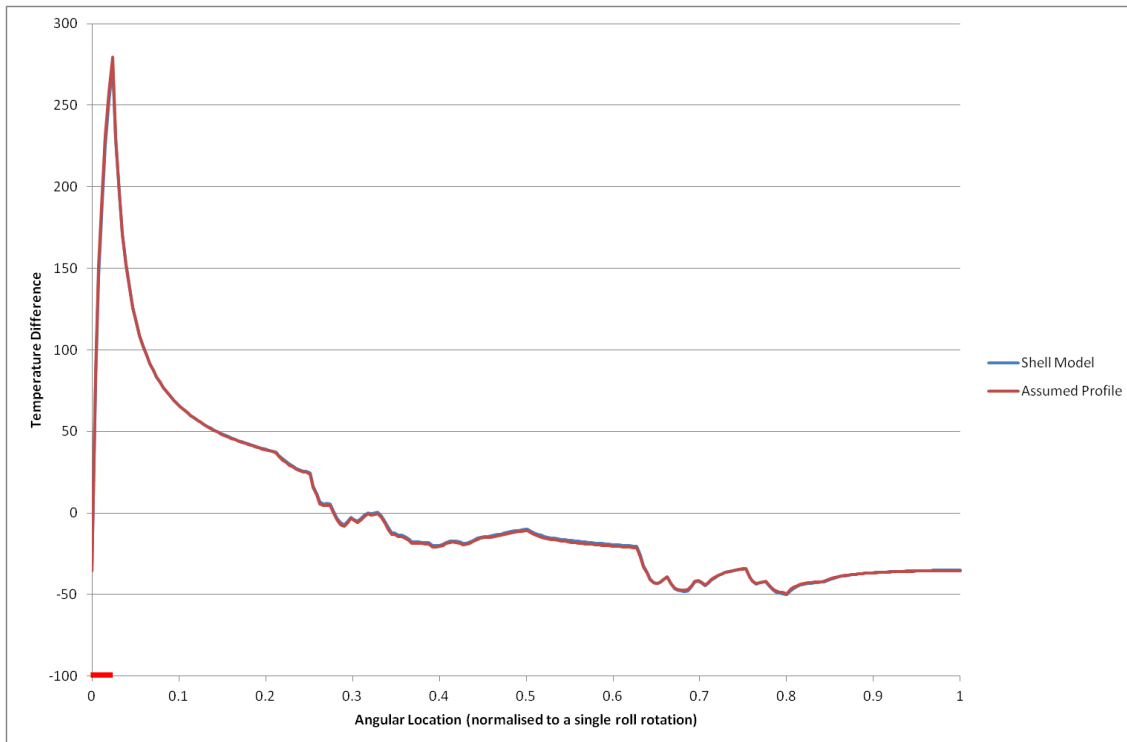
The original average cooling heat transfer coefficient in the core model,  $\bar{h} = \frac{\sum h(\theta)}{n_\theta}$ , was adjusted to take account of the predicted temperature variability.

$$\bar{h} = \frac{\sum T(\theta, t)h(\theta)}{T(t)n_\theta} = \frac{\sum (T(\theta) + T(t))h(\theta)}{T(t)n_\theta} = \frac{\sum T(\theta)h(\theta) + T(t)\sum h(\theta)}{T(t)n_\theta} \quad \text{Equation 3-78}$$

In Equation 3-78  $h$  is the cooling heat transfer coefficient, which is a function of  $\theta$ , the circumferential location,  $n_\theta$  is the number of nodes in the circumferential direction, and  $T$  is a function of time  $t$ , and  $\theta$ .  $T(\theta, t)$ , a function of location and time, is the same as  $T(\theta) + T(t)$  if the circumferential temperature difference remains the same with changing time. This is the

assumption which is made by using the results of a shell model to inform the boundary conditions of the core model.

The effect of these changes made the core model much warmer, raising the average surface temperature of the core model from 46°C to roughly 70°C. The shell model was rerun to generate a new angular temperature profile, based on more accurate interface temperatures. The comparison between the old and new angular temperature profiles at  $z = 0$  (the centre symmetry plane) is shown in Figure 3-32.



**Figure 3-32: Comparison of shell model circumferential temperature profile and the circumferential profile assumed for the core model. Location of the roll bite is indicated by the thick red line on the abscissa.**

The difference between the two profiles in Figure 3-32 is minimal, lending credence to the idea that only the average temperature of the work roll changes, while the circumferential variations stay more or less identical. Furthermore, different rolling stages with different process parameters also generate identical circumferential profiles, showing that these temperature profiles are very resilient to small variations in process parameters during a single rolling campaign, in which the material properties of the strip are largely identical and the reduction amounts vary only slightly. This is a useful finding, since it allows one estimated profile to be used for a whole rolling campaign without fear of significant error.

A correction term is applied to the shell model cooling, comparing the cooling experienced by the shell model and the cooling that would be experienced by a core model with the same circumferentially averaged temperatures. A correction of around 1% was observed within the



width of the strip, with a correction of 80% required outside of the strip width. Given that the temperature difference outside of the strip width is mostly negligible, the 80% difference is no cause for concern, while the 1% is a very good agreement.

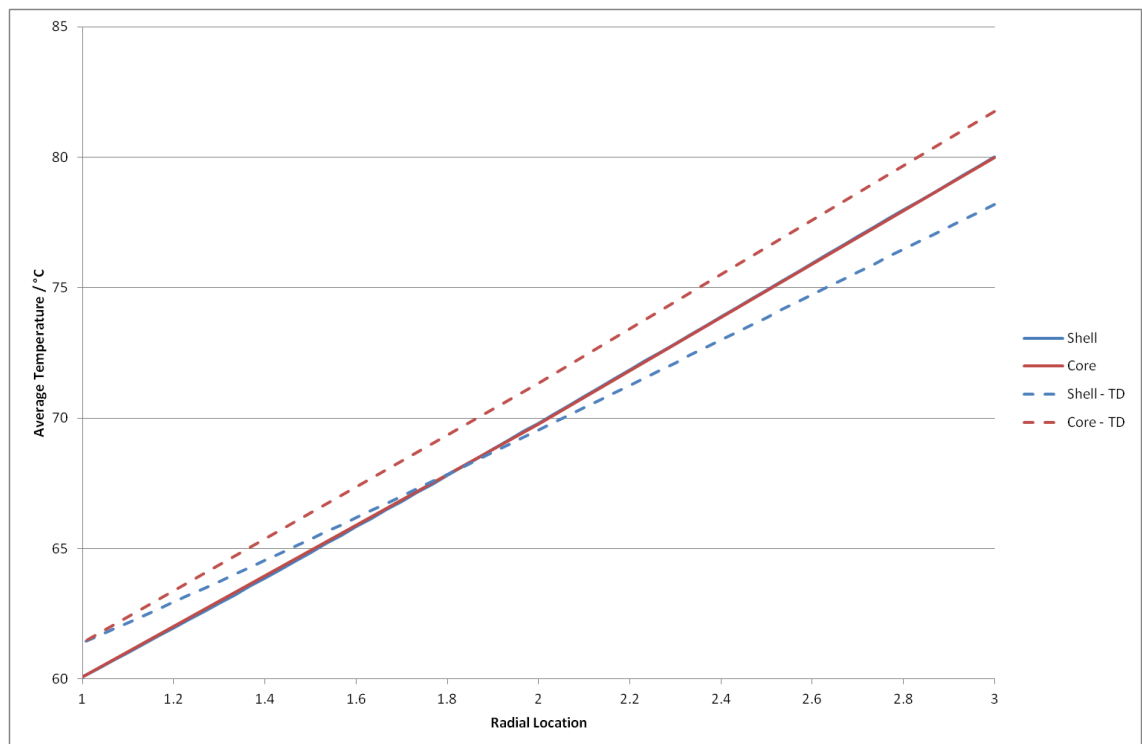


Figure 3-33

Figure 3-33 shows the difference between the core model temperatures and the circumferential average of the shell model temperatures varying radially and at the centre symmetry plane, for temperature dependent (Shell – TD and Core – TD) and non temperature dependent material properties. The two sets of non temperature dependent results are nearly indistinguishable, showing a very good agreement. The two sets of temperature dependent results, however, disagree strongly. A total balance of the heat flow into and out of the shell model showed that 1.125MW of energy was being supplied to the roll, but only 0.965MW of that was accounted for by cooling and changing temperatures. The 0.16MW of unaccounted energy is due to differences in conduction between internal model elements, i.e. the model was not conserving energy.

### 3.6.3. Conservative thermal models

The lack of energy conservation of the previous thermal models was due to the change in the mechanical properties with temperature. The heat flow out of or in to each element was based on the conductivity of that element. If two neighbouring elements were at different

temperatures, the heat flow out of one would not equal the heat flow into the other, resulting in a loss or gain of energy. Correcting the problem required the reformulation of the heat flow equations from scratch, taking the variable thermal properties into account and using a control volume formulation rather than the simple discretisation of the differential equation presented above.

The simple approach leads to a model that does not conserve energy in internal heat conduction, as the variation of conduction with position is not accounted for. One option would have been to formulate terms such as  $\frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right)$  as  $\frac{\partial k}{\partial x} \frac{\partial T}{\partial x} + k \frac{\partial^2 T}{\partial x^2}$ , etc, but this approach does not impose exact energy conservation. A more robust approach is to use a control volume formulation which explicitly ensures that heat conduction across the control volume boundaries is viewed identically for the two control volumes.

### Energy balances

---

The first step in redesigning the thermal model equations is to rework the elementary heat balance with thermally variable properties for each component of heat flow.

#### Axial component

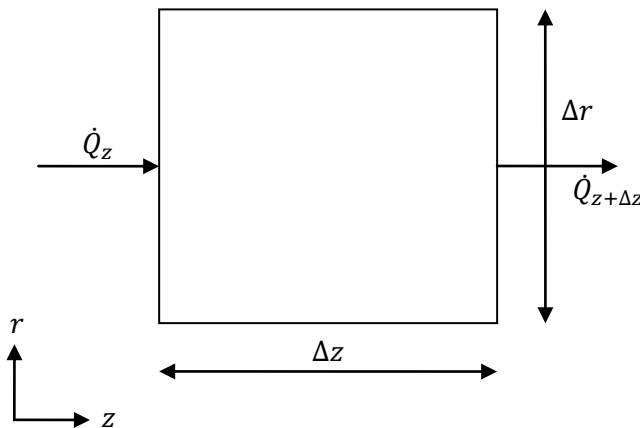


Figure 3-34

Figure 3-34 shows a typical element from a perspective perpendicular to the axial direction. Following Fourier's law of conduction, the heat flow in to the left side of the element will be given by Equation 3-79.

$$\dot{Q}_z = - \left( k \frac{dT}{dz} \right)_{i-\frac{1}{2}} r \Delta \theta \Delta r$$

Equation 3-79

The heat flow out of the right hand side of the element will be given by Equation 3-80.

$$\dot{Q}_{z+\Delta z} = -\left(k \frac{dT}{dz}\right)_{i+\frac{1}{2}} r\Delta\theta\Delta r \quad \text{Equation 3-80}$$

In both Equation 3-79 and Equation 3-80 the thermal conductivity term is now location-specific. The net change in heat in the element will be given by the difference between the two heat flows.

$$\dot{Q}_{net} = \dot{Q}_z - \dot{Q}_{z+\Delta z} = -\left(k \frac{dT}{dz}\right)_{i-\frac{1}{2}} r\Delta\theta\Delta r + \left(k \frac{dT}{dz}\right)_{i+\frac{1}{2}} r\Delta\theta\Delta r \quad \text{Equation 3-81}$$

Equation 3-81 can be rearranged to isolate similar terms.

$$\dot{Q}_{net} = r\Delta\theta\Delta r\Delta z \left( \frac{\left(k \frac{dT}{dz}\right)_{i+\frac{1}{2}} - \left(k \frac{dT}{dz}\right)_{i-\frac{1}{2}}}{\Delta z} \right) \quad \text{Equation 3-82}$$

Previously Equation 3-82 was simplified by substituting in a higher order differential term to replace the two lower order terms. However, in this case the conductivity term is also location specific and cannot be separated from the temperature terms without taking  $\Delta z$  to the limit. To preserve the direct correlation between heat flow out of one element and the heat flow into the next, substitutions will be made directly into Equation 3-82. Substituting  $\Delta V = r\Delta\theta\Delta r\Delta z$ ,  $\frac{dT}{dz}_{i-\frac{1}{2}} = \frac{T_i - T_{i-1}}{\Delta z}$  and  $\frac{dT}{dz}_{i+\frac{1}{2}} = \frac{T_{i+1} - T_i}{\Delta z}$ , and if  $k_+$  and  $k_-$  are the values of  $k$  at positions  $i + \frac{1}{2}$  and  $i - \frac{1}{2}$  respectively;

$$\dot{Q}_{net} = \Delta V \left( k_+ \frac{T_{i+1} - T_i}{\Delta z^2} - k_- \frac{T_i - T_{i-1}}{\Delta z^2} \right) \quad \text{Equation 3-83}$$

Finally, complete the net heat flow equation by collecting the temperature terms.

$$\dot{Q}_{net} = \Delta V \left( \left(\frac{k_+}{\Delta z^2}\right) T_{i+1} + \left(-\frac{k_+}{\Delta z^2} - \frac{k_-}{\Delta z^2}\right) T_i + \left(\frac{k_-}{\Delta z^2}\right) T_{i-1} \right) \quad \text{Equation 3-84}$$

The values of  $k_+$  and  $k_-$  are specified by linear interpolation, which ensures energy conservation at the interface.  $k_+ = \frac{k_i+k_{i+1}}{2}$  and  $k_- = \frac{k_{i-1}+k_i}{2}$ . The change in the internal energy of the element will be  $\Delta U = \Delta V(\rho C_p)_i \Delta T$ , and the first law of thermodynamics states that  $Q = \Delta U + W$ . Given that the element under consideration is arbitrarily placed inside the work roll, it is safe to assume that it is not doing any work. Therefore;

$$\Delta V(\rho C_p)_i \Delta T = \Delta t \Delta V \left( \left( \frac{k_+}{\Delta Z^2} \right) T_{i+1} + \left( -\frac{k_+}{\Delta Z^2} - \frac{k_-}{\Delta Z^2} \right) T_i + \left( \frac{k_-}{\Delta Z^2} \right) T_{i-1} \right) \quad \text{Equation 3-85}$$

Equation 3-85 can be rearranged to create the first order temperature difference with time term.

$$\frac{\Delta T}{\Delta t} = \frac{1}{(\rho C_p)_i} \left( \left( \frac{k_+}{\Delta Z^2} \right) T_{i+1} + \left( -\frac{k_+}{\Delta Z^2} - \frac{k_-}{\Delta Z^2} \right) T_i + \left( \frac{k_-}{\Delta Z^2} \right) T_{i-1} \right) \quad \text{Equation 3-86}$$

Finally, dividing the inside and multiplying the outside of the main brackets by  $k_i$  and substituting  $\alpha_i = \frac{k_i}{\rho C_p}$ ,  $\tau_{i+} = \frac{k_+}{k_i}$  and  $\tau_{i-} = \frac{k_-}{k_i}$ ;

$$\frac{\Delta T}{\Delta t} = \alpha_i \left( \left( \frac{\tau_{i+}}{\Delta Z^2} \right) T_{i+1} + \left( -\frac{\tau_{i+}}{\Delta Z^2} - \frac{\tau_{i-}}{\Delta Z^2} \right) T_i + \left( \frac{\tau_{i-}}{\Delta Z^2} \right) T_{i-1} \right) \quad \text{Equation 3-87}$$

Equation 3-87 gives the contribution of axial heat flow to the temperature change with time of a point within the work roll.

## Radial component

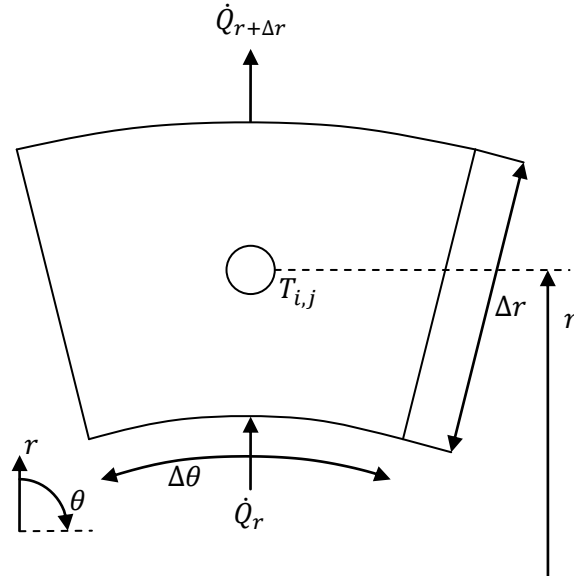


Figure 3-35

Figure 3-35 shows a small element subject to radial heat flow. The heat flow into the bottom face of the element is given by;

$$\dot{Q}_r = -\left(k \frac{dT}{dr}\right)_{j-\frac{1}{2}} \left(r - \frac{\Delta r}{2}\right) \Delta\theta \Delta z \quad \text{Equation 3-88}$$

And the heat flow out of the top surface is given by;

$$\dot{Q}_{r+\Delta r} = -\left(k \frac{dT}{dr}\right)_{j+\frac{1}{2}} \left(r + \frac{\Delta r}{2}\right) \Delta\theta \Delta z \quad \text{Equation 3-89}$$

So the total net heat flow can be given by subtracting Equation 3-88 from Equation 3-89.

$$\begin{aligned} \dot{Q}_{net} &= \dot{Q}_r - \dot{Q}_{r+\Delta r} \\ &= -\left(k \frac{dT}{dr}\right)_{j-\frac{1}{2}} \left(r - \frac{\Delta r}{2}\right) \Delta\theta \Delta z \\ &\quad + \left(k \frac{dT}{dr}\right)_{j+\frac{1}{2}} \left(r + \frac{\Delta r}{2}\right) \Delta\theta \Delta z \end{aligned} \quad \text{Equation 3-90}$$

Expanding;

$$\dot{Q}_{net} = -\left(k \frac{dT}{dr}\right)_{j-\frac{1}{2}} r \Delta \theta \Delta z + \left(k \frac{dT}{dr}\right)_{j-\frac{1}{2}} \frac{\Delta r \Delta \theta \Delta z}{2} + \left(k \frac{dT}{dr}\right)_{j+\frac{1}{2}} r \Delta \theta \Delta z + \left(k \frac{dT}{dr}\right)_{j+\frac{1}{2}} \frac{\Delta r \Delta \theta \Delta z}{2}$$

Equation 3-91

Substitute the finite difference approximations  $\frac{dT}{dr}_{j-\frac{1}{2}} = \frac{T_j - T_{j-1}}{\Delta r}$  and  $\frac{dT}{dr}_{j+\frac{1}{2}} = \frac{T_{j+1} - T_j}{\Delta r}$  and denote the boundary  $k$  values by  $k_+$  and  $k_-$  for positions  $j + \frac{1}{2}$  and  $j - \frac{1}{2}$  respectively.

$$\dot{Q}_{net} = -k_- \frac{T_j - T_{j-1}}{\Delta r} r \Delta \theta \Delta z + k_- \frac{T_j - T_{j-1}}{\Delta r} \frac{\Delta r \Delta \theta \Delta z}{2} + k_+ \frac{T_{j+1} - T_j}{\Delta r} r \Delta \theta \Delta z + k_+ \frac{T_{j+1} - T_j}{\Delta r} \frac{\Delta r \Delta \theta \Delta z}{2}$$

Equation 3-92

Again, linear interpolation is used for  $k_+$  and  $k_-$ ,  $k_+ = \frac{k_j + k_{j+1}}{2}$  and  $k_- = \frac{k_{j-1} + k_j}{2}$ . The geometric quantities can be isolated by multiplying certain terms by  $\frac{r}{r}$  or  $\frac{\Delta r}{\Delta r}$ .

$$\dot{Q}_{net} = -k_- \frac{T_j - T_{j-1}}{\Delta r^2} r \Delta r \Delta \theta \Delta z + k_- \frac{T_j - T_{j-1}}{r \Delta r} \frac{r \Delta r \Delta \theta \Delta z}{2} + k_+ \frac{T_{j+1} - T_j}{\Delta r^2} r \Delta r \Delta \theta \Delta z + k_+ \frac{T_{j+1} - T_j}{r \Delta r} \frac{r \Delta r \Delta \theta \Delta z}{2}$$

Equation 3-93

It is now possible to substitute  $\Delta V = r \Delta r \Delta \theta \Delta z$ .

$$\dot{Q}_{net} = \Delta V \left( -k_- \frac{T_j - T_{j-1}}{\Delta r^2} + k_- \frac{T_j - T_{j-1}}{2r \Delta r} + k_+ \frac{T_{j+1} - T_j}{\Delta r^2} + k_+ \frac{T_{j+1} - T_j}{2r \Delta r} \right)$$

Equation 3-94

Collect temperature terms.

$$\dot{Q}_{net} = \Delta V \left( \left( \frac{k_+}{\Delta r^2} + \frac{k_+}{2r \Delta r} \right) T_{j+1} + \left( -\frac{k_-}{\Delta r^2} + \frac{k_-}{2r \Delta r} - \frac{k_+}{\Delta r^2} - \frac{k_+}{2r \Delta r} \right) T_j + \left( \frac{k_-}{\Delta r^2} - \frac{k_-}{2r \Delta r} \right) T_{j-1} \right)$$

Equation 3-95

Introduce the change of energy in the volume,  $\Delta U = \Delta V (\rho C_p)_j \Delta T$ , which requires a time step length to be introduced to Equation 3-95 to convert the energy flow rate into a specific energy quantity.

$$\begin{aligned}
\Delta U &= \Delta V(\rho C_p)_j \Delta T \\
&= \Delta t \Delta V \left( \left( \frac{k_+}{\Delta r^2} + \frac{k_+}{2r\Delta r} \right) T_{j+1} + \left( -\frac{k_-}{\Delta r^2} + \frac{k_-}{2r\Delta r} - \frac{k_+}{\Delta r^2} - \frac{k_+}{2r\Delta r} \right) T_j \right. \\
&\quad \left. + \left( \frac{k_-}{\Delta r^2} - \frac{k_-}{2r\Delta r} \right) T_{j-1} \right)
\end{aligned}$$

Equation 3-96

Rearrange to create and isolate  $\frac{\Delta T}{\Delta t}$ .

$$\begin{aligned}
\frac{\Delta T}{\Delta t} &= \frac{1}{(\rho C_p)_j} \left( \left( \frac{k_+}{\Delta r^2} + \frac{k_+}{2r\Delta r} \right) T_{j+1} + \left( -\frac{k_-}{\Delta r^2} + \frac{k_-}{2r\Delta r} - \frac{k_+}{\Delta r^2} - \frac{k_+}{2r\Delta r} \right) T_j \right. \\
&\quad \left. + \left( \frac{k_-}{\Delta r^2} - \frac{k_-}{2r\Delta r} \right) T_{j-1} \right)
\end{aligned}$$

Equation 3-97

Introducing  $k_j$  allows the substitution of  $\alpha_j = \frac{k_j}{\rho C_p}$ ,  $\tau_{j+} = \frac{k_+}{k_j}$  and  $\tau_{j-} = \frac{k_-}{k_j}$ .

$$\begin{aligned}
\frac{\Delta T}{\Delta t} &= \alpha_j \left( \left( \frac{\tau_{j+}}{\Delta r^2} + \frac{\tau_{j+}}{2j\Delta r^2} \right) T_{j+1} + \left( -\frac{\tau_{j-}}{\Delta r^2} + \frac{\tau_{j-}}{2j\Delta r^2} - \frac{\tau_{j+}}{\Delta r^2} - \frac{\tau_{j+}}{2j\Delta r^2} \right) T_j \right. \\
&\quad \left. + \left( \frac{\tau_{j-}}{\Delta r^2} - \frac{\tau_{j-}}{2j\Delta r^2} \right) T_{j-1} \right)
\end{aligned}$$

Equation 3-98

Equation 3-98 gives the contribution of radial heat flow to the temperature change with time of a point within the work roll.

## Angular component

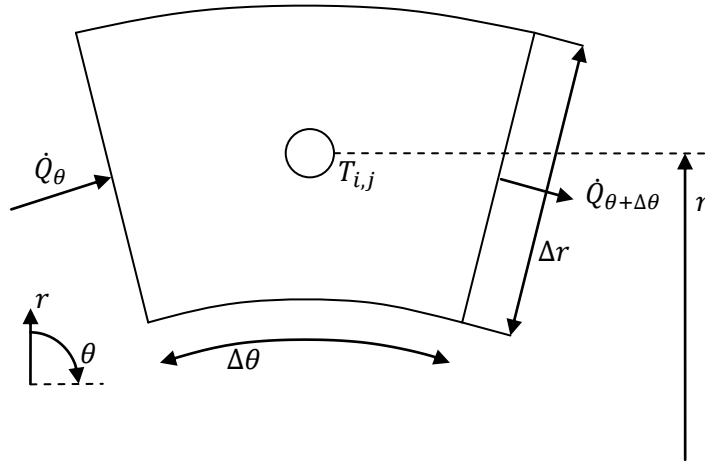


Figure 3-36

Figure 3-36 shows a small element subject to circumferential heat flow. The heat flow into the left face and out of the right face are given by the following equations.

$$\dot{Q}_\theta = -\left(\frac{k}{r} \frac{dT}{d\theta}\right)_{k-\frac{1}{2}} \Delta r \Delta z \quad \text{Equation 3-99}$$

$$\dot{Q}_{\theta+\Delta\theta} = -\left(\frac{k}{r} \frac{dT}{d\theta}\right)_{k+\frac{1}{2}} \Delta r \Delta z \quad \text{Equation 3-100}$$

Thus the net heat flow is given by;

$$\dot{Q}_{net} = \dot{Q}_\theta - \dot{Q}_{\theta+\Delta\theta} = -\left(\frac{k}{r} \frac{dT}{d\theta}\right)_{k-\frac{1}{2}} \Delta r \Delta z + \left(\frac{k}{r} \frac{dT}{d\theta}\right)_{k+\frac{1}{2}} \Delta r \Delta z \quad \text{Equation 3-101}$$

Equation 3-101 can be rearranged to give;

$$\dot{Q}_{net} = \frac{r \Delta r \Delta \theta \Delta z}{r^2} \left( \frac{\left(\frac{k}{r} \frac{dT}{d\theta}\right)_{k+\frac{1}{2}} - \left(\frac{k}{r} \frac{dT}{d\theta}\right)_{k-\frac{1}{2}}}{\Delta \theta} \right) \quad \text{Equation 3-102}$$

Which can be simplified by substituting for  $\Delta V = r \Delta r \Delta \theta \Delta z$ ,  $\frac{dT}{d\theta}_{k+\frac{1}{2}} = \frac{T_{k+1} - T_k}{\Delta \theta}$  and

$\frac{dT}{d\theta}_{k-\frac{1}{2}} = \frac{T_k - T_{k-1}}{\Delta \theta}$ , and denoting the boundary  $k$  values by  $k_+$  and  $k_-$  for positions  $k + \frac{1}{2}$  and

$k - \frac{1}{2}$  respectively.



$$\dot{Q}_{net} = \frac{\Delta V}{r^2} \left( k_+ \frac{T_{k+1} - T_k}{\Delta\theta^2} - k_- \frac{T_k - T_{k-1}}{\Delta\theta^2} \right) \quad \text{Equation 3-103}$$

where  $k_+ = \frac{k_k + k_{k+1}}{2}$  and  $k_- = \frac{k_{k-1} + k_k}{2}$  from linear interpolation of  $k$ . Collect the temperature terms.

$$\dot{Q}_{net} = \Delta V \left( \left( \frac{k_+}{r^2 \Delta\theta^2} \right) T_{k+1} + \left( -\frac{k_+}{r^2 \Delta\theta^2} - \frac{k_-}{r^2 \Delta\theta^2} \right) T_k + \left( \frac{k_-}{r^2 \Delta\theta^2} \right) T_{k-1} \right) \quad \text{Equation 3-104}$$

With the addition of the time step length, Equation 3-104 can be equated to the change in energy of the element, i.e.  $\Delta U = \Delta V (\rho C_p)_k \Delta T = \Delta t \times \dot{q}_{net}$ .

$$\Delta V (\rho C_p)_k \Delta T = \Delta t \Delta V \left( \left( \frac{k_+}{r^2 \Delta\theta^2} \right) T_{k+1} + \left( -\frac{k_+}{r^2 \Delta\theta^2} - \frac{k_-}{r^2 \Delta\theta^2} \right) T_k + \left( \frac{k_-}{r^2 \Delta\theta^2} \right) T_{k-1} \right) \quad \text{Equation 3-105}$$

Rearranging to isolate  $\frac{\Delta T}{\Delta t}$ ;

$$\frac{\Delta T}{\Delta t} = \frac{1}{(\rho C_p)_k} \left( \left( \frac{k_+}{r^2 \Delta\theta^2} \right) T_{k+1} + \left( -\frac{k_+}{r^2 \Delta\theta^2} - \frac{k_-}{r^2 \Delta\theta^2} \right) T_k + \left( \frac{k_-}{r^2 \Delta\theta^2} \right) T_{k-1} \right) \quad \text{Equation 3-106}$$

And finally, introduce  $k_j$  to allow the substitution of  $\alpha_k = \frac{k_k}{\rho C_p}$ ,  $\tau_{k+} = \frac{k_+}{k_k}$  and  $\tau_{k-} = \frac{k_-}{k_k}$ .

$$\frac{\Delta T}{\Delta t} = \alpha_k \left( \left( \frac{\tau_{k+}}{r^2 \Delta\theta^2} \right) T_{k+1} + \left( -\frac{\tau_{k+}}{r^2 \Delta\theta^2} - \frac{\tau_{k-}}{r^2 \Delta\theta^2} \right) T_k + \left( \frac{\tau_{k-}}{r^2 \Delta\theta^2} \right) T_{k-1} \right) \quad \text{Equation 3-107}$$

Equation 3-107 gives the contribution of circumferential heat flow to the temperature change with time of a point within the work roll.

### Convective component

The convective term was initially reformulated as a central difference, to make use of the same interpolated values at the boundaries as the conduction terms. However, difficulties

were encountered during the application of the resulting equations, when rapid circumferential temperature variations provoked an unstable response. The equations were reformulated using a backward difference representation, to ensure numerical stability, and are presented in this form.

Figure 3-37 shows a small element whose boundaries remain stationary while the body of material it represents is moving at a constant velocity of  $u$  meters per second. After  $\Delta t$  seconds have passed, the material will have moved  $u\Delta t$  meters. As material moves out of the bounds of the element, it can be considered to carry away a portion of the internal energy of the element. Similarly, some material will enter the bounds of the element, carrying with it the internal energy of the element from which it has come. This can be expressed mathematically as follows.

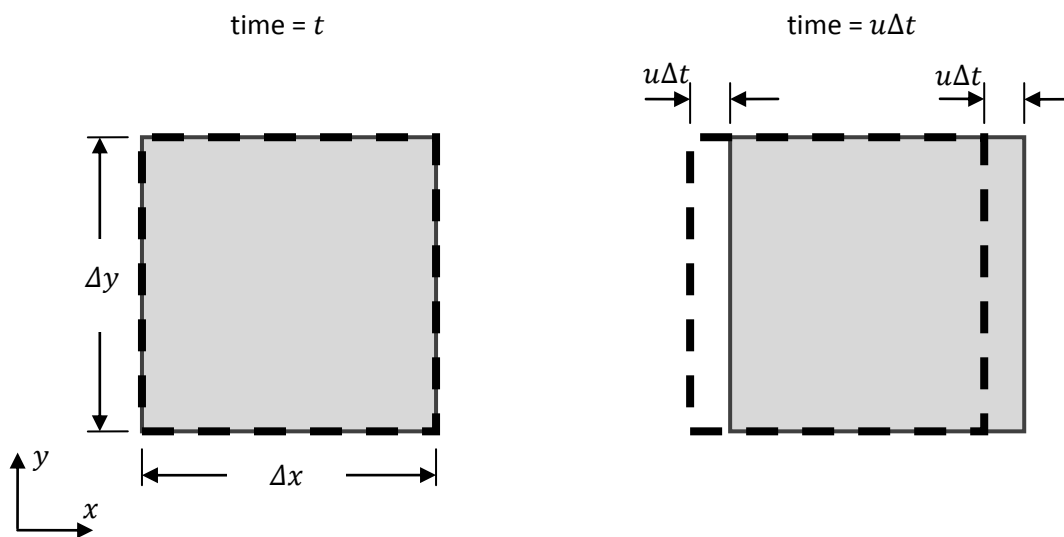


Figure 3-37

$$\dot{Q}_x = u\Delta t\Delta y\Delta z(\rho C_p T)_{i-1}$$

Equation 3-108

$$\dot{Q}_{x+\Delta x} = u\Delta t\Delta y\Delta z(\rho C_p T)_i$$

Equation 3-109

The net heat flow will therefore be given by;

$$\dot{Q}_{net} = \dot{Q}_x - \dot{Q}_{x+\Delta x} = u\Delta t\Delta y\Delta z \left( (\rho C_p T)_{i-1} - (\rho C_p T)_i \right)$$

Equation 3-110

which can be rearranged to isolate the volume of the element.

$$\begin{aligned}\dot{Q}_{net} &= u\Delta t\Delta x\Delta y\Delta z \left( \frac{(\rho C_p T)_{i-1} - (\rho C_p T)_i}{\Delta x} \right) \\ &= u\Delta V \left( \frac{(\rho C_p T)_{i-1} - (\rho C_p T)_i}{\Delta x} \right)\end{aligned}$$

**Equation 3-111**

As with the previous sections, this heat flow can be equated to the change in the internal energy of the element,  $\Delta U = \Delta V(\rho C_p)_i \Delta T$ .

$$\Delta V(\rho C_p T)_i \Delta T = u\Delta t\Delta V \left( \frac{(\rho C_p T)_{i-1} - (\rho C_p T)_i}{\Delta x} \right)$$

**Equation 3-112**

Rearranging to isolate  $\frac{\Delta T}{\Delta t}$ ,

$$\frac{\Delta T}{\Delta t} = \frac{u}{(\rho C_p)_i} \left( \frac{(\rho C_p)_{i-1} T_{i-1} - (\rho C_p)_i T_i}{\Delta x} \right)$$

**Equation 3-113**

Substituting for  $\psi = \frac{(\rho C_p)_{i-1}}{(\rho C_p)_i}$ ,

$$\frac{\Delta T}{\Delta t} = \left( \frac{u\psi}{\Delta x} \right) T_{i-1} + \left( -\frac{u}{\Delta x} \right) T_i$$

**Equation 3-114**

Equation 3-114 must be converted into cylindrical coordinates for use in the work roll models, i.e.  $u = \omega r$  and  $\Delta x = r\Delta\theta$ .

$$\frac{\Delta T}{\Delta t} = \left( \frac{\omega\psi}{\Delta\theta} \right) T_{i-1} + \left( -\frac{\omega}{\Delta\theta} \right) T_i$$

**Equation 3-115**

Equation 3-115 gives the contribution of convective heat flow to the temperature change with time of a point within the work roll.

General equation

The core model equation ignores effects acting in the circumferential direction, each point in the axial-radial plane representing the circumferential average temperature. As such the heat flow equation will be made up of the axial and radial heat flow components. This is expressed in mathematical terms below.

$$\frac{\Delta T}{\Delta t} = \frac{\Delta T}{\Delta t}_{axial} + \frac{\Delta T}{\Delta t}_{radial} \quad \text{Equation 3-116}$$

Substituting Equation 3-87, Equation 3-98 and  $\frac{\Delta T}{\Delta t} = \frac{T_{i,j}^{K+1} - T_{i,j}^K}{\Delta t}$  into Equation 3-116 gives;

$$\begin{aligned} \frac{T_{i,j}^{K+1} - T_{i,j}^K}{\Delta t} = & \alpha_{i,j} \left( \left( \frac{\tau_{i+}}{\Delta z^2} \right) T_{i+1,j} + \left( -\frac{\tau_{i+}}{\Delta z^2} - \frac{\tau_{i-}}{\Delta z^2} \right) T_{i,j} + \left( \frac{\tau_{i-}}{\Delta z^2} \right) T_{i-1,j} \right. \\ & + \left( \frac{\tau_{j+}}{\Delta r^2} + \frac{\tau_{j+}}{2j\Delta r^2} \right) T_{i,j+1} \\ & + \left( -\frac{\tau_{j-}}{\Delta r^2} + \frac{\tau_{j-}}{2j\Delta r^2} - \frac{\tau_{j+}}{\Delta r^2} - \frac{\tau_{j+}}{2j\Delta r^2} \right) T_{i,j} \\ & \left. + \left( \frac{\tau_{j-}}{\Delta r^2} - \frac{\tau_{j-}}{2j\Delta r^2} \right) T_{i,j-1} \right) \end{aligned} \quad \text{Equation 3-117}$$

Isolating the temperature terms;

$$\begin{aligned} & \left( \frac{\Delta r^2}{\alpha_{i,j}\Delta t} \right) T_{i,j}^{K+1} - \left( \frac{\Delta r^2}{\alpha_{i,j}\Delta t} \right) T_{i,j}^K \\ & = \left( \frac{\Delta r^2 \tau_{i+}}{\Delta z^2} \right) T_{i+1,j} + \left( \frac{\Delta r^2 \tau_{i-}}{\Delta z^2} \right) T_{i-1,j} + \left( \tau_{j+} + \frac{\tau_{j+}}{2j} \right) T_{i,j+1} \\ & + \left( -\frac{\Delta r^2 \tau_{i+}}{\Delta z^2} - \frac{\Delta r^2 \tau_{i-}}{\Delta z^2} - \tau_{j-} + \frac{\tau_{j-}}{2j} - \tau_{j+} - \frac{\tau_{j+}}{2j} \right) T_{i,j} \\ & + \left( \tau_{j-} - \frac{\tau_{j-}}{2j} \right) T_{i,j-1} \end{aligned}$$

Equation 3-118

The equation can be simplified by using the substitutions  $\gamma_0 = \frac{\Delta r^2}{\Delta z^2}$  and  $\lambda_0 = \frac{\Delta r^2}{\alpha \Delta t}$ .

$$\begin{aligned}
(\lambda_0)T_{i,j}^{K+1} - (\lambda_0)T_{i,j}^K &= (\gamma_0\tau_{i+})T_{i+1,j} + (\gamma_0\tau_{i-})T_{i-1,j} + \left(\tau_{j+} + \frac{\tau_{j+}}{2j}\right)T_{i,j+1} \\
&+ \left(-\gamma_0\tau_{i+} - \gamma_0\tau_{i-} - \tau_{j-} + \frac{\tau_{j-}}{2j} - \tau_{j+} - \frac{\tau_{j+}}{2j}\right)T_{i,j} \\
&+ \left(\tau_{j-} - \frac{\tau_{j-}}{2j}\right)T_{i,j-1}
\end{aligned}$$

**Equation 3-119**

Some minor rearrangement permits two more simplifying substitutions,  $j_{0a} = \frac{2j+1}{4j}$  and  $j_{0b} = \frac{2j-1}{4j}$ .

$$\begin{aligned}
(\lambda_0)T_{i,j}^{K+1} - (\lambda_0)T_{i,j}^K &= (\gamma_0\tau_{i+})T_{i+1,j} + (\gamma_0\tau_{i-})T_{i-1,j} + (2j_{0a}\tau_{j+})T_{i,j+1} \\
&+ (-\gamma_0\tau_{i+} - \gamma_0\tau_{i-} - 2j_{0b}\tau_{j-} - 2j_{0a}\tau_{j+})T_{i,j} \\
&+ (2j_{0b}\tau_{j-})T_{i,j-1}
\end{aligned}$$

**Equation 3-120**

The right hand side of the equation must now be split into old and new time step values, as required by the Crank-Nicholson technique.

$$\begin{aligned}
(\lambda_0)T_{i,j}^{K+1} - (\lambda_0)T_{i,j}^K &= \frac{1}{2} [(\gamma_0\tau_{i+})T_{i+1,j}^{K+1} + (\gamma_0\tau_{i-})T_{i-1,j}^{K+1} + (2j_{0a}\tau_{j+})T_{i,j+1}^{K+1} \\
&+ (2j_{0b}\tau_{j-})T_{i,j-1}^{K+1} \\
&+ (-\gamma_0\tau_{i+} - \gamma_0\tau_{i-} - 2j_{0b}\tau_{j-} - 2j_{0a}\tau_{j+})T_{i,j}^{K+1}] \\
&+ \frac{1}{2} [(\gamma_0\tau_{i+})T_{i+1,j}^K + (\gamma_0\tau_{i-})T_{i-1,j}^K + (2j_{0a}\tau_{j+})T_{i,j+1}^K \\
&+ (2j_{0b}\tau_{j-})T_{i,j-1}^K \\
&+ (-\gamma_0\tau_{i+} - \gamma_0\tau_{i-} - 2j_{0b}\tau_{j-} - 2j_{0a}\tau_{j+})T_{i,j}^K]
\end{aligned}$$

**Equation 3-121**

Collecting terms;

$$\begin{aligned}
0 = & \left(\frac{\gamma_0 \tau_{i+}}{2}\right) T_{i+1,j}^{K+1} + \left(\frac{\gamma_0 \tau_{i-}}{2}\right) T_{i-1,j}^{K+1} + (j_{0a} \tau_{j+}) T_{i,j+1}^{K+1} + (j_{0b} \tau_{j-}) T_{i,j-1}^{K+1} \\
& + \left(-\frac{\gamma_0 \tau_{i+}}{2} - \frac{\gamma_0 \tau_{i-}}{2} - j_{0b} \tau_{j-} - j_{0a} \tau_{j+} - \lambda_0\right) T_{i,j}^{K+1} \\
& + \left(\frac{\gamma_0 \tau_{i+}}{2}\right) T_{i+1,j}^K + \left(\frac{\gamma_0 \tau_{i-}}{2}\right) T_{i-1,j}^K + (j_{0a} \tau_{j+}) T_{i,j+1}^K \\
& + (j_{0b} \tau_{j-}) T_{i,j-1}^K \\
& + \left(-\frac{\gamma_0 \tau_{i+}}{2} - \frac{\gamma_0 \tau_{i-}}{2} - j_{0b} \tau_{j-} - j_{0a} \tau_{j+} + \lambda_0\right) T_{i,j}^K
\end{aligned} \tag{Equation 3-122}$$

Isolating  $T_{i,j}^{K+1}$ , which will become the object of the equation;

$$\begin{aligned}
& \left(\lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+}\right) T_{i,j}^{K+1} \\
& = \left(\frac{\gamma_0 \tau_{i+}}{2}\right) T_{i+1,j}^{K+1} + \left(\frac{\gamma_0 \tau_{i-}}{2}\right) T_{i-1,j}^{K+1} + (j_{0a} \tau_{j+}) T_{i,j+1}^{K+1} \\
& + (j_{0b} \tau_{j-}) T_{i,j-1}^{K+1} + \left(\frac{\gamma_0 \tau_{i+}}{2}\right) T_{i+1,j}^K + \left(\frac{\gamma_0 \tau_{i-}}{2}\right) T_{i-1,j}^K \\
& + (j_{0a} \tau_{j+}) T_{i,j+1}^K + (j_{0b} \tau_{j-}) T_{i,j-1}^K \\
& + \left(\lambda_0 - \frac{\gamma_0 \tau_{i+}}{2} - \frac{\gamma_0 \tau_{i-}}{2} - j_{0b} \tau_{j-} - j_{0a} \tau_{j+}\right) T_{i,j}^K
\end{aligned} \tag{Equation 3-123}$$

So to attain the form that will be used by the solver, i.e.

$$\begin{aligned}
T_{i,j}^{K+1} = & AT_{i+1,j}^{K+1} + BT_{i,j-1}^{K+1} + CT_{i+1,j}^{K+1} + DT_{i-1,j}^{K+1} + AT_{i,j+1}^K + BT_{i,j-1}^K \\
& + ET_{i,j}^K + CT_{i+1,j}^K + DT_{i-1,j}^K + F
\end{aligned} \tag{Equation 3-124}$$

The coefficient definitions will be;

$$A = \frac{j_{0a} \tau_{j+}}{\lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+}} \tag{Equation 3-125}$$

$$B = \frac{j_{0b} \tau_{j-}}{\lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+}} \tag{Equation 3-126}$$

$$C = \frac{\gamma_0 \tau_{i+}}{2 \left(\lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+}\right)} \tag{Equation 3-127}$$

$$D = \frac{\gamma_0 \tau_{i-}}{2 \left(\lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+}\right)} \tag{Equation 3-128}$$

$$E = \frac{\lambda_0 - \frac{\gamma_0 \tau_{i+}}{2} - \frac{\gamma_0 \tau_{i-}}{2} - j_{0b} \tau_{j-} - j_{0a} \tau_{j+}}{\lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+}} \tag{Equation 3-129}$$

$$F = 0 \tag{Equation 3-130}$$

The set of coefficient definitions above provide the basis for the core model solution with thermally variable material properties. If the material properties were not thermally variable then these equations should reduce to the coefficient definitions of the original core model, providing an easy check of their validity. In the non-thermally varying scenario, all of the  $\tau$  terms will be equal to 1 (i.e.  $\frac{k_{i+}}{k_i} = \frac{k_{i-}}{k_i} = \frac{k}{k} = 1$ ), giving;

$$A = \frac{j_{0a}}{\lambda_0 + \frac{\gamma_0}{2} + \frac{\gamma_0}{2} + j_{0b} + j_{0a}} = \frac{j_{0a}}{\lambda_0 + \gamma_0 + 1} \quad \text{Equation 3-131}$$

$$B = \frac{j_{0b}}{\lambda_0 + \frac{\gamma_0}{2} + \frac{\gamma_0}{2} + j_{0b} + j_{0a}} = \frac{j_{0b}}{\lambda_0 + \gamma_0 + 1} \quad \text{Equation 3-132}$$

$$C = \frac{\gamma_0}{2(\lambda_0 + \frac{\gamma_0}{2} + \frac{\gamma_0}{2} + j_{0b} + j_{0a})} = \frac{\gamma_0}{2(\lambda_0 + \gamma_0 + 1)} \quad \text{Equation 3-133}$$

$$D = \frac{\gamma_0}{2(\lambda_0 + \frac{\gamma_0}{2} + \frac{\gamma_0}{2} + j_{0b} + j_{0a})} = \frac{\gamma_0}{2(\lambda_0 + \gamma_0 + 1)} \quad \text{Equation 3-134}$$

$$E = \frac{\lambda_0 - \frac{\gamma_0}{2} - \frac{\gamma_0}{2} - j_{0b} - j_{0a}}{\lambda_0 + \frac{\gamma_0}{2} + \frac{\gamma_0}{2} + j_{0b} + j_{0a}} = \frac{\lambda_0 - \gamma_0 - 1}{\lambda_0 + \gamma_0 + 1} \quad \text{Equation 3-135}$$

$$F = 0 \quad \text{Equation 3-136}$$

$$\text{NB: } j_{0a} + j_{0b} = \frac{2j+1}{4j} + \frac{2j-1}{4j} = \frac{2j}{4j} + \frac{1}{4j} + \frac{2j}{4j} - \frac{1}{4j} = 1$$

Comparing Equation 3-131 through Equation 3-136 to the original core model definition given in Equation 2-35 shows that the models will agree exactly on the solution when using constant material properties.

### Surface boundary condition

From the White model<sup>3</sup> the definitions for the heat flows in to and out of the surface are given as;

$$\dot{q}_g = \beta \frac{P_{stand}}{4A_{contact}} \quad \text{Equation 3-137}$$

$$\dot{q}_{cond} = \frac{\theta_b}{2\pi} h_{str}(T_{str} - T_{i,j}) \quad \text{Equation 3-138}$$

$$\dot{q}_{cool} = \frac{2\pi - \theta_b}{2\pi} h_{spr}(T_{i,j} - T_{spr}) \quad \text{Equation 3-139}$$

where  $\dot{q}_g$  is the heating due to friction,  $\dot{q}_{cond}$  is the heating due to conduction from the strip and  $\dot{q}_{cool}$  is the heating from the cooling sprays. These were combined to give a total heat flow term of;

$$k \frac{dT}{dr} = \dot{q}_g + \dot{q}_{cond} + \dot{q}_{cool} \quad \text{Equation 3-140}$$

which, through some rearrangement and substitution gave;

$$T_{i,j+1} = G_0 + G_1 T_{i,j} + T_{i,j-1} \quad \text{Equation 3-141}$$

where  $G_0 = \frac{2\Delta r}{k_{i,j}} (H_0 + H_2 T_{str} + H_3 T_{spr})$ ,  $G_1 = \frac{2\Delta r}{k_{i,j}} H_1$ ,  $H_0 = \beta \frac{P_{stand}}{4A_{contact}}$ ,  $H_1 = -\frac{\theta_b}{2\pi} h_{str} - \frac{2\pi-\theta_b}{2\pi} h_{spr}$ ,  $H_2 = \frac{\theta_b}{2\pi} h_{str}$  and  $H_3 = \frac{2\pi-\theta_b}{2\pi} h_{spr}$ .

Substituting Equation 3-141 into Equation 3-123 gives;

$$\begin{aligned} & \left( \lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} \right) T_{i,j}^{K+1} \\ & = \left( \frac{\gamma_0 \tau_{i+}}{2} \right) T_{i+1,j}^{K+1} + \left( \frac{\gamma_0 \tau_{i-}}{2} \right) T_{i-1,j}^{K+1} \\ & + (j_{0a} \tau_{j+}) (G_0 + G_1 T_{i,j}^{K+1} + T_{i,j-1}^{K+1}) + (j_{0b} \tau_{j-}) T_{i,j-1}^{K+1} \\ & + \left( \frac{\gamma_0 \tau_{i+}}{2} \right) T_{i+1,j}^K + \left( \frac{\gamma_0 \tau_{i-}}{2} \right) T_{i-1,j}^K \\ & + (j_{0a} \tau_{j+}) (G_0 + G_1 T_{i,j}^K + T_{i,j-1}^K) + (j_{0b} \tau_{j-}) T_{i,j-1}^K \\ & + \left( \lambda_0 - \frac{\gamma_0 \tau_{i+}}{2} - \frac{\gamma_0 \tau_{i-}}{2} - j_{0b} \tau_{j-} - j_{0a} \tau_{j+} \right) T_{i,j}^K \end{aligned} \quad \text{Equation 3-142}$$

Rearranging to gather terms gives;



$$\begin{aligned}
& \left( \lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - j_{0a} \tau_{j+} G_1 \right) T_{ij}^{K+1} \\
&= \left( \frac{\gamma_0 \tau_{i+}}{2} \right) T_{i+1,j}^{K+1} + \left( \frac{\gamma_0 \tau_{i-}}{2} \right) T_{i-1,j}^{K+1} + (j_{0b} \tau_{j-} + j_{0a} \tau_{j+}) T_{i,j-1}^{K+1} \\
&+ \left( \frac{\gamma_0 \tau_{i+}}{2} \right) T_{i+1,j}^K + \left( \frac{\gamma_0 \tau_{i-}}{2} \right) T_{i-1,j}^K + (j_{0b} \tau_{j-} + j_{0a} \tau_{j+}) T_{i,j-1}^K \\
&+ \left( \lambda_0 - \frac{\gamma_0 \tau_{i+}}{2} - \frac{\gamma_0 \tau_{i-}}{2} - j_{0b} \tau_{j-} - j_{0a} \tau_{j+} + j_{0a} \tau_{j+} G_1 \right) T_{ij}^K \\
&+ (2j_{0a} \tau_{j+} G_0)
\end{aligned}$$

Equation 3-143

A final substitution of  $\alpha_{rsb} = j_{0a} \tau_{j+} G_1$  and the coefficients can be given as;

$$A = 0 \quad \text{Equation 3-144}$$

$$B = \frac{j_{0b} \tau_{j-} + j_{0a} \tau_{j+}}{\lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \alpha_{rsb}} \quad \text{Equation 3-145}$$

$$C = \frac{\gamma_0 \tau_{i+}}{2 \left( \lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \alpha_{rsb} \right)} \quad \text{Equation 3-146}$$

$$D = \frac{\gamma_0 \tau_{i-}}{2 \left( \lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \alpha_{rsb} \right)} \quad \text{Equation 3-147}$$

$$E = \frac{\lambda_0 - \frac{\gamma_0 \tau_{i+}}{2} - \frac{\gamma_0 \tau_{i-}}{2} - j_{0b} \tau_{j-} - j_{0a} \tau_{j+} + \alpha_{rsb}}{\lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \alpha_{rsb}} \quad \text{Equation 3-148}$$

$$F = \frac{2j_{0a} \tau_{j+} G_0}{\lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \alpha_{rsb}} \quad \text{Equation 3-149}$$

### Symmetry plane boundary condition

The only modification needed to emulate symmetry between the two halves of the work roll is that, at the boundary,  $T_{i-1,j} = T_{i+1,j}$ . The adjustments to the coefficient definitions needed to achieve this are;

$$C = \frac{\gamma_0 (\tau_{i+} + \tau_{i-})}{2 \left( \lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} \right)} \quad \text{Equation 3-150}$$

$$D = 0 \quad \text{Equation 3-151}$$

### Barrel convection edge boundary condition

The substitution to allow convection at the edge of the barrel is;

$$T_{i+1,j} = T_{i-1,j} - \frac{2h_{conv}\Delta z}{k_{i,j}}(T_{i,j} - T_f) \quad \text{Equation 3-152}$$

Substituting Equation 3-152 and  $\alpha_{bce} = \frac{h_{conv}\Delta z\gamma_0\tau_{i+}}{k_{i,j}}$  into Equation 3-123 leads to the following set of coefficients.

$$A = \frac{j_{0a}\tau_{j+}}{\lambda_0 + \frac{\gamma_0\tau_{i+}}{2} + \frac{\gamma_0\tau_{i-}}{2} + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} + \alpha_{bce}} \quad \text{Equation 3-153}$$

$$B = \frac{j_{0b}\tau_{j-}}{\lambda_0 + \frac{\gamma_0\tau_{i+}}{2} + \frac{\gamma_0\tau_{i-}}{2} + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} + \alpha_{bce}} \quad \text{Equation 3-154}$$

$$C = 0 \quad \text{Equation 3-155}$$

$$D = \frac{\gamma_0\tau_{i-} + \gamma_0\tau_{i+}}{2\left(\lambda_0 + \frac{\gamma_0\tau_{i+}}{2} + \frac{\gamma_0\tau_{i-}}{2} + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} + \alpha_{bce}\right)} \quad \text{Equation 3-156}$$

$$E = \frac{\lambda_0 - \frac{\gamma_0\tau_{i+}}{2} - \frac{\gamma_0\tau_{i-}}{2} - j_{0b}\tau_{j-} - j_{0a}\tau_{j+} - \alpha_{bce}}{\lambda_0 + \frac{\gamma_0\tau_{i+}}{2} + \frac{\gamma_0\tau_{i-}}{2} + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} + \alpha_{bce}} \quad \text{Equation 3-157}$$

$$F = \frac{2\alpha_{bce}T_f}{\lambda_0 + \frac{\gamma_0\tau_{i+}}{2} + \frac{\gamma_0\tau_{i-}}{2} + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} + \alpha_{bce}} \quad \text{Equation 3-158}$$

### Journal surface boundary condition

The substitution to allow conduction from the chock to the work roll journal is;

$$T_{i,j+1} = T_{i,j-1} - \frac{2h_{ConvBs}\Delta r}{k_{i,j}}(T_{i,j} - T_B) \quad \text{Equation 3-159}$$

Substituting Equation 3-159 and  $\alpha_{js} = \frac{2j_{0a}\tau_{j+}h_{ConvBs}\Delta r}{k_{i,j}}$  into Equation 3-123 gives;

$$A = 0 \quad \text{Equation 3-160}$$

$$B = \frac{j_{0b}\tau_{j-} + j_{0a}\tau_{j+}}{\lambda_0 + \frac{\gamma_0\tau_{i+}}{2} + \frac{\gamma_0\tau_{i-}}{2} + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} + \alpha_{js}} \quad \text{Equation 3-161}$$

$$C = \frac{\gamma_0 \tau_{i+}}{2 \left( \lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} + \alpha_{js} \right)} \quad \text{Equation 3-162}$$

$$D = \frac{\gamma_0 \tau_{i-}}{2 \left( \lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} + \alpha_{js} \right)} \quad \text{Equation 3-163}$$

$$E = \frac{\lambda_0 - \frac{\gamma_0 \tau_{i+}}{2} - \frac{\gamma_0 \tau_{i-}}{2} - j_{0b} \tau_{j-} - j_{0a} \tau_{j+} - \alpha_{js}}{\lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} + \alpha_{js}} \quad \text{Equation 3-164}$$

$$F = \frac{2\alpha_{js} T_B}{\lambda_0 + \frac{\gamma_0 \tau_{i+}}{2} + \frac{\gamma_0 \tau_{i-}}{2} + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} + \alpha_{js}} \quad \text{Equation 3-165}$$

### Compound point boundary conditions

At every corner in the model outline there is a node where two boundary conditions meet. The solution of these points is achieved by substituting in new relationships for two values rather than one. Since the process is so similar to that detailed above, and the substituted relationships are exactly the same as those described above, they will not be repeated here.

### Centre symmetry axis

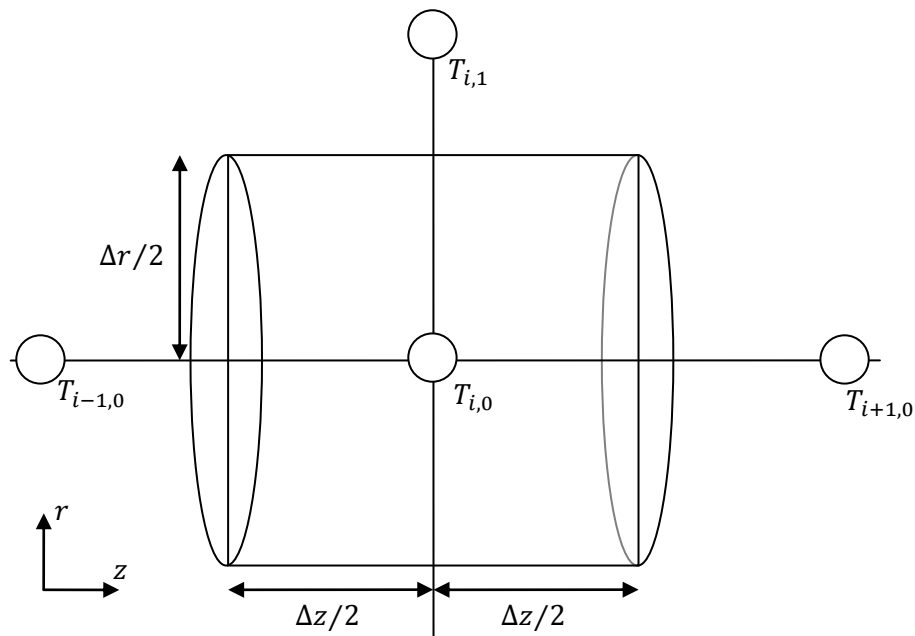


Figure 3-38

Figure 3-38 illustrates the size and shape of an element at the centre plane of the model. Compared to other elements in the work roll model, this model has a considerably different

shape, having a circular rather than an annular section. A new energy balance over this particular shape of element will give the required Axisymmetric boundary condition.

The outer surface area is  $2\pi \frac{\Delta r}{2} \Delta z = \pi \Delta r \Delta z$ . There will be no heat out, since the middle of the element is the centre point, giving;

$$\dot{Q}_r = \pi \Delta z \Delta r \left( k \frac{\partial T}{\partial r} \right)_{\frac{1}{2}} \quad \text{Equation 3-166}$$

and for the ends;

$$\dot{Q}_z = \pi \frac{\Delta r^2}{4} \left( \left( k \frac{\partial T}{\partial z} \right)_{i+\frac{1}{2}} - \left( k \frac{\partial T}{\partial z} \right)_{i-\frac{1}{2}} \right) \quad \text{Equation 3-167}$$

Equation 3-166 and Equation 3-167 can be combined to give the total heat flow.

$$\dot{Q}_{net} = \pi \frac{\Delta r^2}{4} \left( \left( k \frac{\partial T}{\partial z} \right)_{i+\frac{1}{2}} - \left( k \frac{\partial T}{\partial z} \right)_{i-\frac{1}{2}} \right) + \pi \Delta z \Delta r \left( k \frac{\partial T}{\partial r} \right)_{j+\frac{1}{2}} \quad \text{Equation 3-168}$$

Given that the conductivity term is location dependent, Equation 3-168 cannot be collapsed to a higher order differential term as the White model<sup>3</sup> did. Instead, the following substitutions will be applied directly;

$$\frac{\partial T}{\partial z_{i+\frac{1}{2}}} = \frac{T_{i+1,0} - T_{i,0}}{\Delta z}$$

$$\frac{\partial T}{\partial z_{i-\frac{1}{2}}} = \frac{T_{i,0} - T_{i-1,0}}{\Delta z}$$

$$\frac{\partial T}{\partial z_{j+\frac{1}{2}}} = \frac{T_{i,1} - T_{i,0}}{\Delta r}$$

giving;

$$\dot{Q}_{net} = \pi \frac{\Delta r^2}{4} \left( k_{i+} \frac{T_{i+1,0} - T_{i,0}}{\Delta z} - k_{i-} \frac{T_{i,0} - T_{i-1,0}}{\Delta z} \right) + \pi \Delta z \Delta r k_{j+} \frac{T_{i,1} - T_{i,0}}{\Delta r} \quad \text{Equation 3-169}$$

Collecting the temperature terms;

$$\begin{aligned}\dot{Q}_{net} = & \left(\frac{\pi\Delta r^2 k_{i+}}{4\Delta Z}\right)T_{i+1,0} + \left(\frac{\pi\Delta r^2 k_{i-}}{4\Delta Z}\right)T_{i-1,0} + (\pi\Delta Z k_{j+})T_{i,1} \\ & + \left(-\frac{\pi\Delta r^2 k_{i+}}{4\Delta Z} - \frac{\pi\Delta r^2 k_{i-}}{4\Delta Z} - \pi\Delta Z k_{j+}\right)T_{i,0}\end{aligned}$$

Equation 3-170

Substituting for  $V = \pi \frac{\Delta r^2}{4} \Delta Z$ ;

$$\begin{aligned}\dot{Q}_{net} = & V \left( \left(\frac{k_{i+}}{\Delta Z^2}\right)T_{i+1,0} + \left(\frac{k_{i-}}{\Delta Z^2}\right)T_{i-1,0} + \left(\frac{4}{\Delta r^2} k_{j+}\right)T_{i,1} \right. \\ & \left. + \left(-\frac{k_{i+}}{\Delta Z^2} - \frac{k_{i-}}{\Delta Z^2} - \frac{4}{\Delta r^2} k_{j+}\right)T_{i,0} \right)\end{aligned}$$

Equation 3-171

Since  $Q = \Delta U + W$ , where  $W = 0$  and  $\Delta U = V(\rho C_p)_{i,j} \Delta T$ , the substitution can be made for  $\Delta U = \dot{Q}_{net}$ ;

$$\begin{aligned}V(\rho C_p)_{i,j} \Delta T = & V \Delta t \left( \left(\frac{k_{i+}}{\Delta Z^2}\right)T_{i+1,0} + \left(\frac{k_{i-}}{\Delta Z^2}\right)T_{i-1,0} + \left(\frac{4}{\Delta r^2} k_{j+}\right)T_{i,1} \right. \\ & \left. + \left(-\frac{k_{i+}}{\Delta Z^2} - \frac{k_{i-}}{\Delta Z^2} - \frac{4}{\Delta r^2} k_{j+}\right)T_{i,0} \right)\end{aligned}$$

Equation 3-172

Equation 3-172 can be rearranged to isolate the  $\frac{\Delta T}{\Delta t}$  term.

$$\begin{aligned}\frac{\Delta T}{\Delta t} = & \frac{1}{(\rho C_p)_{i,j}} \left( \left(\frac{k_{i+}}{\Delta Z^2}\right)T_{i+1,0} + \left(\frac{k_{i-}}{\Delta Z^2}\right)T_{i-1,0} + \left(\frac{4}{\Delta r^2} k_{j+}\right)T_{i,1} \right. \\ & \left. + \left(-\frac{k_{i+}}{\Delta Z^2} - \frac{k_{i-}}{\Delta Z^2} - \frac{4}{\Delta r^2} k_{j+}\right)T_{i,0} \right)\end{aligned}$$

Equation 3-173

Introduce  $k_{i,j}$ , with  $\alpha = \frac{k}{\rho C_p}$  and  $\frac{k_{i-}}{k_{i,j}} = \tau_{i-}$ ,  $\frac{k_{i+}}{k_{i,j}} = \tau_{i+}$  and  $\frac{k_{j+}}{k_{i,j}} = \tau_{j+}$

$$\begin{aligned}\frac{\Delta T}{\Delta t} = & \alpha_{i,j} \left( \left(\frac{\tau_{i+}}{\Delta Z^2}\right)T_{i+1,0} + \left(\frac{\tau_{i-}}{\Delta Z^2}\right)T_{i-1,0} + \left(\frac{4\tau_{j+}}{\Delta r^2}\right)T_{i,1} \right. \\ & \left. + \left(-\frac{\tau_{i+}}{\Delta Z^2} - \frac{\tau_{i-}}{\Delta Z^2} - \frac{4\tau_{j+}}{\Delta r^2}\right)T_{i,0} \right)\end{aligned}$$

Equation 3-174

Introduce the Crank-Nicholson method...

$$\begin{aligned}
\frac{T_{i,0}^{K+1} - T_{i,0}^K}{\Delta t} &= \frac{\alpha_{i,j}}{2} \left( \left( \frac{\tau_{i+}}{\Delta Z^2} \right) T_{i+1,0}^{K+1} + \left( \frac{\tau_{i-}}{\Delta Z^2} \right) T_{i-1,0}^{K+1} + \left( \frac{4\tau_{j+}}{\Delta r^2} \right) T_{i,1}^{K+1} \right. \\
&\quad \left. + \left( -\frac{\tau_{i+}}{\Delta Z^2} - \frac{\tau_{i-}}{\Delta Z^2} - \frac{4\tau_{j+}}{\Delta r^2} \right) T_{i,0}^{K+1} \right) \\
&\quad + \frac{\alpha_{i,j}}{2} \left( \left( \frac{\tau_{i+}}{\Delta Z^2} \right) T_{i+1,0}^K + \left( \frac{\tau_{i-}}{\Delta Z^2} \right) T_{i-1,0}^K + \left( \frac{4\tau_{j+}}{\Delta r^2} \right) T_{i,1}^K \right. \\
&\quad \left. + \left( -\frac{\tau_{i+}}{\Delta Z^2} - \frac{\tau_{i-}}{\Delta Z^2} - \frac{4\tau_{j+}}{\Delta r^2} \right) T_{i,0}^K \right)
\end{aligned}$$

Equation 3-175

Collecting the temperature terms;

$$\begin{aligned}
0 &= \left( \frac{\tau_{i+}}{2\Delta Z^2} \right) T_{i+1,0}^{K+1} + \left( \frac{\tau_{i-}}{2\Delta Z^2} \right) T_{i-1,0}^{K+1} + \left( \frac{2\tau_{j+}}{\Delta r^2} \right) T_{i,1}^{K+1} \\
&\quad + \left( -\frac{\tau_{i+}}{2\Delta Z^2} - \frac{\tau_{i-}}{2\Delta Z^2} - \frac{2\tau_{j+}}{\Delta r^2} - \frac{1}{\alpha_{i,j}\Delta t} \right) T_{i,0}^{K+1} \\
&\quad + \left( \frac{\tau_{i+}}{2\Delta Z^2} \right) T_{i+1,0}^K + \left( \frac{\tau_{i-}}{2\Delta Z^2} \right) T_{i-1,0}^K + \left( \frac{2\tau_{j+}}{\Delta r^2} \right) T_{i,1}^K \\
&\quad + \left( -\frac{\tau_{i+}}{2\Delta Z^2} - \frac{\tau_{i-}}{2\Delta Z^2} - \frac{2\tau_{j+}}{\Delta r^2} + \frac{1}{\alpha_{i,j}\Delta t} \right) T_{i,0}^K
\end{aligned}$$

Equation 3-176

Multiplying through by  $\frac{\Delta r^2}{2}$  collects terms in a way which makes following substitutions easier.

$$\begin{aligned}
0 &= \left( \frac{\Delta r^2 \tau_{i+}}{4\Delta Z^2} \right) T_{i+1,0}^{K+1} + \left( \frac{\Delta r^2 \tau_{i-}}{4\Delta Z^2} \right) T_{i-1,0}^{K+1} + (\tau_{j+}) T_{i,1}^{K+1} \\
&\quad + \left( -\frac{\Delta r^2 \tau_{i+}}{4\Delta Z^2} - \frac{\Delta r^2 \tau_{i-}}{4\Delta Z^2} - \tau_{j+} - \frac{\Delta r^2}{2\alpha_{i,j}\Delta t} \right) T_{i,0}^{K+1} \\
&\quad + \left( \frac{\Delta r^2 \tau_{i+}}{4\Delta Z^2} \right) T_{i+1,0}^K + \left( \frac{\Delta r^2 \tau_{i-}}{4\Delta Z^2} \right) T_{i-1,0}^K + (\tau_{j+}) T_{i,1}^K \\
&\quad + \left( -\frac{\Delta r^2 \tau_{i+}}{4\Delta Z^2} - \frac{\Delta r^2 \tau_{i-}}{4\Delta Z^2} - \tau_{j+} + \frac{\Delta r^2}{2\alpha_{i,j}\Delta t} \right) T_{i,0}^K
\end{aligned}$$

Equation 3-177

Using  $\gamma_1 = \frac{\Delta r^2}{4\Delta Z^2}$ , which is more convenient in this formulation than the original  $\gamma_1 = \frac{\Delta r^2}{2\Delta Z^2}$

and  $\lambda_1 = \frac{\Delta r^2}{2\alpha_{i,j}\Delta t}$ ;

$$\begin{aligned}
0 = & (\gamma_1 \tau_{i+}) T_{i+1,0}^{K+1} + (\gamma_1 \tau_{i-}) T_{i-1,0}^{K+1} + (\tau_{j+}) T_{i,1}^{K+1} \\
& + (-\gamma_1 \tau_{i+} - \gamma_1 \tau_{i-} - \tau_{j+} - \lambda_1) T_{i,0}^{K+1} + (\gamma_1 \tau_{i+}) T_{i+1,0}^K \\
& + (\gamma_1 \tau_{i-}) T_{i-1,0}^K + (\tau_{j+}) T_{i,1}^K \\
& + (-\gamma_1 \tau_{i+} - \gamma_1 \tau_{i-} - \tau_{j+} + \lambda_1) T_{i,0}^K
\end{aligned}$$

**Equation 3-178**

Rearranging to provide the coefficient definitions;

$$\begin{aligned}
& (\lambda_1 + \gamma_1 \tau_{i+} + \gamma_1 \tau_{i-} + \tau_{j+}) T_{i,0}^{K+1} \\
& = (\gamma_1 \tau_{i+}) T_{i+1,0}^{K+1} + (\gamma_1 \tau_{i-}) T_{i-1,0}^{K+1} + (\tau_{j+}) T_{i,1}^{K+1} \\
& + (\gamma_1 \tau_{i+}) T_{i+1,0}^K + (\gamma_1 \tau_{i-}) T_{i-1,0}^K + (\tau_{j+}) T_{i,1}^K \\
& + (\lambda_1 - \gamma_1 \tau_{i+} - \gamma_1 \tau_{i-} - \tau_{j+}) T_{i,0}^K
\end{aligned}$$

**Equation 3-179**

gives;

$$A = \frac{\tau_{j+}}{\lambda_1 + \gamma_1 \tau_{i+} + \gamma_1 \tau_{i-} + \tau_{j+}}$$

**Equation 3-180**

$$B = 0$$

**Equation 3-181**

$$C = \frac{\gamma_1 \tau_{i+}}{\lambda_1 + \gamma_1 \tau_{i+} + \gamma_1 \tau_{i-} + \tau_{j+}}$$

**Equation 3-182**

$$D = \frac{\gamma_1 \tau_{i-}}{\lambda_1 + \gamma_1 \tau_{i+} + \gamma_1 \tau_{i-} + \tau_{j+}}$$

**Equation 3-183**

$$E = \frac{\lambda_1 - \gamma_1 \tau_{i+} - \gamma_1 \tau_{i-} - \tau_{j+}}{\lambda_1 + \gamma_1 \tau_{i+} + \gamma_1 \tau_{i-} + \tau_{j+}}$$

**Equation 3-184**

$$F = 0$$

**Equation 3-185**

which, if the  $\tau$  terms are all = 1 to give homogeneous material properties, and using

$\gamma_1 = \frac{\Delta r^2}{2\Delta z^2}$  instead of  $\gamma_1 = \frac{\Delta r^2}{4\Delta z^2}$  to match the original definition, gives;

$$A = \frac{1}{\lambda_1 + \gamma_1 + 1}$$

**Equation 3-186**

$$B = 0$$

**Equation 3-187**

$$C = \frac{\gamma_1}{2(\lambda_1 + \gamma_1 + 1)}$$

**Equation 3-188**

$$D = \frac{\gamma_1}{2(\lambda_1 + \gamma_1 + 1)}$$

**Equation 3-189**

$$E = \frac{\lambda_1 - \gamma_1 - 1}{\lambda_1 + \gamma_1 + 1}$$

**Equation 3-190**

$$F = 0$$

**Equation 3-191**

Equation 3-186 to Equation 3-191 matches with the core model coefficient definitions given in Equation 2-68 and onwards.

## Shell model

---

### General Equation

The shell model considers heat flow in all three spatial dimensions, and imports the change of temperature with time,  $\frac{\partial T}{\partial t}$ , from the core model.  $\frac{\partial T}{\partial t}$  is calculated for every axial-radial coordinate in the core model, so will be represented by the two-dimensional array,  $T'_{i,j}$ .

$$\begin{aligned}
 T'_{i,j} = \alpha_{i,j,k} & \left( \left( \frac{\tau_{i+}}{\Delta Z^2} \right) T_{i+1,j,k} + \left( -\frac{\tau_{i+}}{\Delta Z^2} - \frac{\tau_{i-}}{\Delta Z^2} \right) T_{i,j,k} + \left( \frac{\tau_{i-}}{\Delta Z^2} \right) T_{i-1,j,k} \right. \\
 & + \left( \frac{\tau_{j+}}{\Delta r^2} + \frac{\tau_{j+}}{2j\Delta r^2} \right) T_{i,j+1,k} + \left( -\frac{\tau_{j-}}{\Delta r^2} + \frac{\tau_{j-}}{2j\Delta r^2} - \frac{\tau_{j+}}{\Delta r^2} - \frac{\tau_{j+}}{2j\Delta r^2} \right) T_{i,j,k} \\
 & + \left( \frac{\tau_{j-}}{\Delta r^2} - \frac{\tau_{j-}}{2j\Delta r^2} \right) T_{i,j-1,k} + \left( \frac{\tau_{k+}}{r^2\Delta\theta^2} \right) T_{i,j,k+1} \\
 & + \left( -\frac{\tau_{k+}}{r^2\Delta\theta^2} - \frac{\tau_{k-}}{r^2\Delta\theta^2} \right) T_{i,j,k} + \left( \frac{\tau_{k-}}{r^2\Delta\theta^2} \right) T_{i,j,k-1} \left. \right) + \left( \frac{\omega\psi}{\Delta\theta} \right) T_{i,j,k-1} \\
 & + \left( -\frac{\omega}{\Delta\theta} \right) T_{i,j,k}
 \end{aligned}$$

**Equation 3-192**

Rearrange and substitute for  $r = j\Delta r$  and  $j = j_{offset} + j$  to allow for the radius of the bottom edge of the shell model not being zero.



$$\begin{aligned}
\frac{T'_{i,j}}{\alpha_{i,j,k}} = & \left(\frac{\tau_{i+}}{\Delta Z^2}\right) T_{i+1,j,k} + \left(-\frac{\tau_{i+}}{\Delta Z^2} - \frac{\tau_{i-}}{\Delta Z^2}\right) T_{i,j,k} + \left(\frac{\tau_{i-}}{\Delta Z^2}\right) T_{i-1,j,k} \\
& + \left(\frac{\tau_{j+}}{\Delta r^2} + \frac{\tau_{j+}}{2(j_{offset} + j)\Delta r^2}\right) T_{i,j+1,k} \\
& + \left(-\frac{\tau_{j-}}{\Delta r^2} + \frac{\tau_{j-}}{2(j_{offset} + j)\Delta r^2} - \frac{\tau_{j+}}{\Delta r^2} - \frac{\tau_{j+}}{2(j_{offset} + j)\Delta r^2}\right) T_{i,j,k} \\
& + \left(\frac{\tau_{j-}}{\Delta r^2} - \frac{\tau_{j-}}{2(j_{offset} + j)\Delta r^2}\right) T_{i,j-1,k} \\
& + \left(\frac{\tau_{k+}}{(j_{offset} + j)^2 \Delta r^2 \Delta \theta^2}\right) T_{i,j,k+1} \\
& + \left(-\frac{\tau_{k+}}{(j_{offset} + j)^2 \Delta r^2 \Delta \theta^2} - \frac{\tau_{k-}}{(j_{offset} + j)^2 \Delta r^2 \Delta \theta^2}\right) T_{i,j,k} \\
& + \left(\frac{\tau_{k-}}{(j_{offset} + j)^2 \Delta r^2 \Delta \theta^2}\right) T_{i,j,k-1} + \left(-\frac{\omega}{\alpha_{i,j,k} \Delta \theta}\right) T_{i,j,k} \\
& + \left(\frac{\omega \psi}{\alpha_{i,j,k} \Delta \theta}\right) T_{i,j,k-1}
\end{aligned}$$

Equation 3-193

Multiply by  $\Delta r^2$  to allow for simplifying substitutions.

$$\begin{aligned}
\frac{\Delta r^2 T'_{i,j}}{\alpha_{i,j,k}} = & \left(\frac{\Delta r^2 \tau_{i+}}{\Delta Z^2}\right) T_{i+1,j,k} + \left(-\frac{\Delta r^2 \tau_{i+}}{\Delta Z^2} - \frac{\Delta r^2 \tau_{i-}}{\Delta Z^2}\right) T_{i,j,k} + \left(\frac{\Delta r^2 \tau_{i-}}{\Delta Z^2}\right) T_{i-1,j,k} \\
& + \left(\tau_{j+} + \frac{\tau_{j+}}{2(j_{offset} + j)}\right) T_{i,j+1,k} \\
& + \left(-\tau_{j-} + \frac{\tau_{j-}}{2(j_{offset} + j)} - \tau_{j+} - \frac{\tau_{j+}}{2(j_{offset} + j)}\right) T_{i,j,k} \\
& + \left(\tau_{j-} - \frac{\tau_{j-}}{2(j_{offset} + j)}\right) T_{i,j-1,k} + \left(\frac{\tau_{k+}}{(j_{offset} + j)^2 \Delta \theta^2}\right) T_{i,j,k+1} \\
& + \left(-\frac{\tau_{k+}}{(j_{offset} + j)^2 \Delta \theta^2} - \frac{\tau_{k-}}{(j_{offset} + j)^2 \Delta \theta^2}\right) T_{i,j,k} \\
& + \left(\frac{\tau_{k-}}{(j_{offset} + j)^2 \Delta \theta^2}\right) T_{i,j,k-1} + \left(-\frac{\omega \Delta r^2}{\alpha_{i,j,k} \Delta \theta}\right) T_{i,j,k} \\
& + \left(\frac{\omega \psi \Delta r^2}{\alpha_{i,j,k} \Delta \theta}\right) T_{i,j,k-1}
\end{aligned}$$

Equation 3-194

Substitute for  $\gamma_0 = \frac{\Delta r^2}{\Delta z^2}$   $\lambda_0 = \frac{\omega \Delta r^2}{\alpha_{i,j,k} \Delta \theta}$  ,  $j_{0a} = \frac{2(j_{offset}+j)+1}{2(j_{offset}+j)}$  and  $j_{0b} = \frac{2(j_{offset}+j)-1}{2(j_{offset}+j)}$  and introduce  $j_c = \frac{1}{(j_{offset}+j)^2 \Delta \theta^2}$ .

$$\begin{aligned} \frac{\Delta r^2 T'_{i,j}}{\alpha_{i,j,k}} &= (\gamma_0 \tau_{i+}) T_{i+1,j,k} + (-\gamma_0 \tau_{i+} - \gamma_0 \tau_{i-}) T_{i,j,k} + (\gamma_0 \tau_{i-}) T_{i-1,j,k} + (j_{0a} \tau_{j+}) T_{i,j+1,k} \\ &+ (-j_{0b} \tau_{j-} - j_{0a} \tau_{j+}) T_{i,j,k} + (j_{0b} \tau_{j-}) T_{i,j-1,k} + (j_c \tau_{k+}) T_{i,j,k+1} \\ &+ (-j_c \tau_{k+} - j_c \tau_{k-}) T_{i,j,k} + (j_c \tau_{k-}) T_{i,j,k-1} + (-\lambda_0) T_{i,j,k} + (\lambda_0 \psi) T_{i,j,k-1} \end{aligned}$$

**Equation 3-195**

Gather the temperature terms;

$$\begin{aligned} &(j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} + \lambda_0) T_{i,j,k} \\ &= (\gamma_0 \tau_{i+}) T_{i+1,j,k} + (\gamma_0 \tau_{i-}) T_{i-1,j,k} + (j_{0a} \tau_{j+}) T_{i,j+1,k} + (j_{0b} \tau_{j-}) T_{i,j-1,k} \\ &+ (j_c \tau_{k+}) T_{i,j,k+1} + (j_c \tau_{k-} + \lambda_0 \psi) T_{i,j,k-1} - \frac{\Delta r^2 T'_{i,j}}{\alpha_{i,j,k}} \end{aligned}$$

**Equation 3-196**

The form of equation used by the solver is given by;

$$\begin{aligned} T_{i,j,k} &= AT_{i+1,j,k} + BT_{i-1,j,k} + CT_{i,j+1,k} + DT_{i,j-1,k} + ET_{i,j,k+1} \\ &+ FT_{i,j,k-1} + G \end{aligned}$$

**Equation 3-197**

Therefore, the coefficient definitions will be;

$$A = \frac{\gamma_0 \tau_{i+}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \lambda_0}$$

**Equation 3-198**

$$B = \frac{\gamma_0 \tau_{i-}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \lambda_0}$$

**Equation 3-199**

$$C = \frac{j_{0a} \tau_{j+}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \lambda_0}$$

**Equation 3-200**

$$D = \frac{j_{0b} \tau_{j-}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \lambda_0}$$

**Equation 3-201**

$$E = \frac{j_c \tau_{k+}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \lambda_0}$$

**Equation 3-202**

$$F = \frac{j_c \tau_{k-} + \lambda_0 \psi}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \lambda_0}$$

**Equation 3-203**

$$G = \frac{-\frac{\Delta r^2 T'_{i,j}}{\alpha_{i,j,k}}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} - \lambda_0} \quad \text{Equation 3-204}$$

Substituting in a value of 1 for all of the  $\tau$  components to emulate constant material properties gives;

$$A = \frac{\gamma_0}{2j_c + 2\gamma_0 + 2 - \lambda_0} \quad \text{Equation 3-205}$$

$$B = \frac{\gamma_0}{2j_c + 2\gamma_0 + 2 - \lambda_0} \quad \text{Equation 3-206}$$

$$C = \frac{j_{0a}}{2j_c + 2\gamma_0 + 2 - \lambda_0} \quad \text{Equation 3-207}$$

$$D = \frac{j_{0b}}{2j_c + 2\gamma_0 + 2 - \lambda_0} \quad \text{Equation 3-208}$$

$$E = \frac{j_c}{2j_c + 2\gamma_0 + 2 - \lambda_0} \quad \text{Equation 3-209}$$

$$F = \frac{j_c + \lambda_0}{2j_c + 2\gamma_0 + 2 - \lambda_0} \quad \text{Equation 3-210}$$

$$G = \frac{-\frac{\Delta r^2 T'_{i,j}}{\alpha_{i,j,k}}}{2j_c + 2\gamma_0 + 2 - \lambda_0} \quad \text{Equation 3-211}$$

Equation 3-205 to Equation 3-211 shows that the new shell model formulation is identical to the old shell model formulation when the material properties are constant.

### Surface boundary condition

The substitution for the virtual above the work roll surface has previous been given as;

$$T_{i,j+1,k} = T_{i,j-1,k} + \frac{2\Delta r}{k_{i,j,k}} \dot{q}_{friction}(\theta) - \frac{2\Delta r h_{str}}{k_{i,j,k}} (T_{i,j,k} - T_{str[i,k]}) - \frac{2\Delta r h_{spr[i,k]}}{k_{i,j,k}} (T_{i,j,k} - T_{spr}) \quad \text{Equation 3-212}$$

Substituting Equation 3-212 into Equation 3-196 gives;

$$\begin{aligned}
& \left( j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} - \lambda_0(\psi_{k-} - \psi_{k+}) \right. \\
& \quad \left. + \frac{2j_{0a}\tau_{j+}\Delta r h_{str}}{k_{i,j,k}} + \frac{2j_{0a}\tau_{j+}\Delta r h_{spr[i,k]}}{k_{i,j,k}} \right) T_{i,j,k} \\
& = (\gamma_0\tau_{i+})T_{i+1,j,k} + (\gamma_0\tau_{i-})T_{i-1,j,k} + (j_{0b}\tau_{j-} + j_{0a}\tau_{j+})T_{i,j-1,k} \\
& \quad + (j_c\tau_{k+} - \lambda_0\psi_{k+})T_{i,j,k+1} + (j_c\tau_{k-} + \lambda_0\psi_{k-})T_{i,j,k-1} - \frac{\Delta r^2 T'}{\alpha_{i,j,k}} \\
& \quad + \frac{2j_{0a}\tau_{j+}\Delta r}{k_{i,j,k}} \dot{q}_{friction}(\theta) + \frac{2j_{0a}\tau_{j+}\Delta r h_{str}}{k_{i,j,k}} T_{str[i,k]} \\
& \quad + \frac{2j_{0a}\tau_{j+}\Delta r h_{spr[i,k]}}{k_{i,j,k}} T_{spr}
\end{aligned}$$

Equation 3-213

Which can be rearranged into;

$$\begin{aligned}
& \left( j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} - \lambda_0(\psi_{k-} - \psi_{k+}) \right. \\
& \quad \left. + \frac{2j_{0a}\tau_{j+}\Delta r}{k_{i,j,k}} (h_{str} + h_{spr[i,k]}) \right) T_{i,j,k} \\
& = (\gamma_0\tau_{i+})T_{i+1,j,k} + (\gamma_0\tau_{i-})T_{i-1,j,k} + (j_{0b}\tau_{j-} + j_{0a}\tau_{j+})T_{i,j-1,k} \\
& \quad + (j_c\tau_{k+} - \lambda_0\psi_{k+})T_{i,j,k+1} + (j_c\tau_{k-} + \lambda_0\psi_{k-})T_{i,j,k-1} - \frac{\Delta r^2 T'}{\alpha_{i,j,k}} \\
& \quad + \frac{2j_{0a}\tau_{j+}\Delta r}{k_{i,j,k}} (\dot{q}_{friction}(\theta) + h_{str}T_{str[i,k]} + h_{spr[i,k]}T_{spr})
\end{aligned}$$

Equation 3-214

Using  $H_0 = \frac{2j_{0a}\tau_{j+}\Delta r}{k_{i,j,k}} (h_{str} + h_{spr[i,k]})$  and  $H_1 = \frac{2j_{0a}\tau_{j+}\Delta r}{k_{i,j,k}} (\dot{q}_{friction}(\theta) + h_{str}T_{str[i,k]} + h_{spr[i,k]}T_{spr})$  gives the following coefficient definitions.

$$A = \frac{\gamma_0\tau_{i+}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} - \lambda_0 + H_0} \quad \text{Equation 3-215}$$

$$B = \frac{\gamma_0\tau_{i-}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} - \lambda_0 + H_0} \quad \text{Equation 3-216}$$

$$C = 0 \quad \text{Equation 3-217}$$

$$D = \frac{j_{0a}\tau_{j+} + j_{0b}\tau_{j-}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} - \lambda_0 + H_0} \quad \text{Equation 3-218}$$

$$E = \frac{j_c\tau_{k+}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} - \lambda_0 + H_0} \quad \text{Equation 3-219}$$

$$F = \frac{j_c \tau_{k-} + \lambda_0 \psi}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \lambda_0 + H_0} \quad \text{Equation 3-220}$$

$$G = \frac{-\frac{\Delta r^2 T'}{\alpha_{i,j,k}} + H_1}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \lambda_0 + H_0} \quad \text{Equation 3-221}$$

### Symmetry boundary condition

As with the core model, the symmetry condition can be emulated using a simple substitution;

$$T_{i-1,j,k} = T_{i+1,j,k} \quad \text{Equation 3-222}$$

Leading to the coefficient definitions;

$$A = \frac{\gamma_0 \tau_{i+} + \gamma_0 \tau_{i-}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \lambda_0} \quad \text{Equation 3-223}$$

$$B = 0 \quad \text{Equation 3-224}$$

$$C = \frac{j_{0a} \tau_{j+}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \lambda_0} \quad \text{Equation 3-225}$$

$$D = \frac{j_{0b} \tau_{j-}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \lambda_0} \quad \text{Equation 3-226}$$

$$E = \frac{j_c \tau_{k+}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \lambda_0} \quad \text{Equation 3-227}$$

$$F = \frac{j_c \tau_{k-} + \lambda_0 \psi}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \lambda_0} \quad \text{Equation 3-228}$$

$$G = \frac{-\frac{\Delta r^2 T'}{\alpha_{i,j,k}}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b} \tau_{j-} + j_{0a} \tau_{j+} - \lambda_0} \quad \text{Equation 3-229}$$

### Convection boundary condition

The substitution to give the virtual point created by a convection boundary condition is given in Equation 3-230.

$$T_{i+1,j,k} = T_{i-1,j,k} - \frac{2h_{conv} \Delta z}{k} (T_{i,j,k} - T_f) \quad \text{Equation 3-230}$$

Substituting Equation 3-230 into Equation 3-196 gives;

$$\begin{aligned} & \left( j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} - \lambda_0 + \frac{2\gamma_0\tau_{i+}h_{conv}\Delta z}{k} \right) T_{i,j,k} \\ & = (\gamma_0\tau_{i-} + \gamma_0\tau_{i+})T_{i-1,j,k} + (j_{0a}\tau_{j+})T_{i,j+1,k} + (j_{0b}\tau_{j-})T_{i,j-1,k} \\ & + (j_c\tau_{k+})T_{i,j,k+1} + (j_c\tau_{k-} + \lambda_0\psi)T_{i,j,k-1} - \frac{\Delta r^2 T'}{\alpha_{i,j,k}} + \frac{2\gamma_0\tau_{i+}h_{conv}\Delta z}{k} T_f \end{aligned}$$

**Equation 3-231**

Equation 3-231 can be used to provide the coefficient definitions, using  $\alpha_{rbc} =$

$$\frac{2\gamma_0\tau_{i+}h_{conv}\Delta z}{k}.$$

$$A = 0$$

**Equation 3-232**

$$B = \frac{\gamma_0\tau_{i-} + \gamma_0\tau_{i+}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} - \lambda_0 + \alpha_{rbc}}$$

**Equation 3-233**

$$C = \frac{j_{0a}\tau_{j+}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} - \lambda_0 + \alpha_{rbc}}$$

**Equation 3-234**

$$D = \frac{j_{0b}\tau_{j-}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} - \lambda_0 + \alpha_{rbc}}$$

**Equation 3-235**

$$E = \frac{j_c\tau_{k+}}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} - \lambda_0 + \alpha_{rbc}}$$

**Equation 3-236**

$$F = \frac{j_c\tau_{k-} + \lambda_0\psi}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} - \lambda_0 + \alpha_{rbc}}$$

**Equation 3-237**

$$G = \frac{-\frac{\Delta r^2 T'}{\alpha_{i,j,k}} + \alpha_{rbc}T_f}{j_c(\tau_{k+} + \tau_{k-}) + \gamma_0(\tau_{i+} + \tau_{i-}) + j_{0b}\tau_{j-} + j_{0a}\tau_{j+} - \lambda_0 + \alpha_{rbc}}$$

**Equation 3-238**

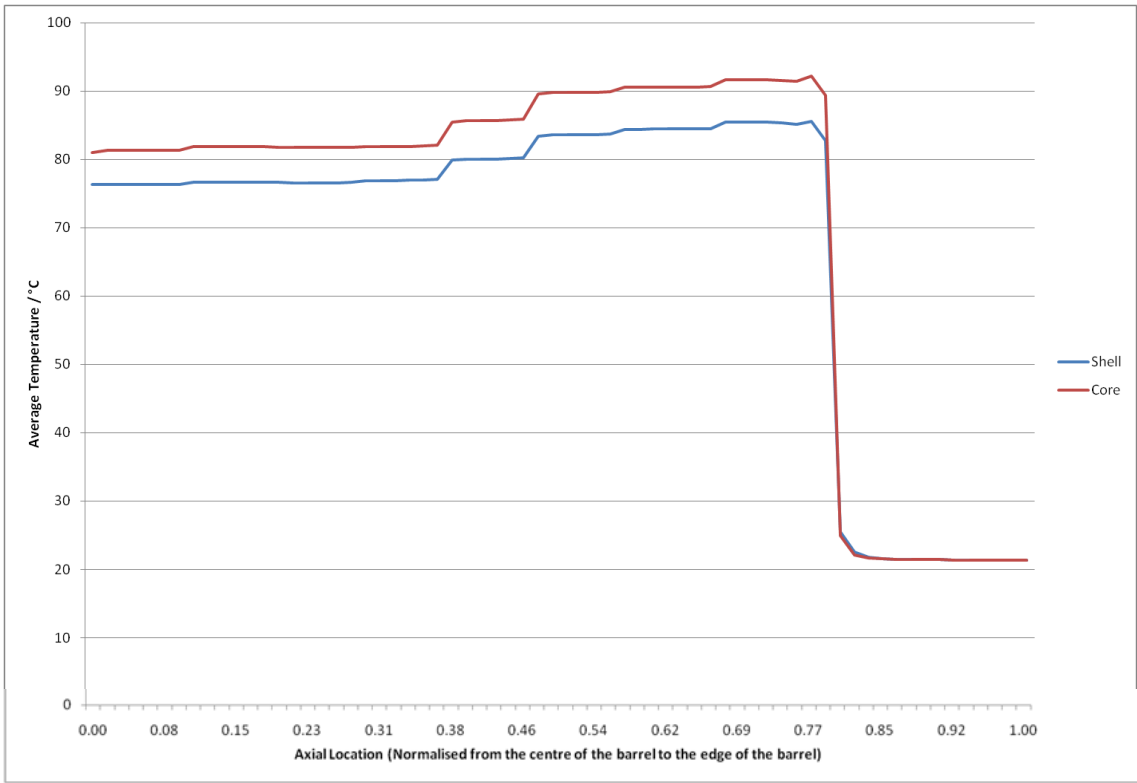
## Testing

The testing of the old and new core models uncovered a difference of 2.5°C at the end of the journal and on the centre line. This was due to a small error in the original core model code. That small error aside, the two models average  $1.99 \times 10^{-13}$ °C difference when the material properties are held constant, proving that the model is functioning properly. The new shell model agreed with the original shell model to  $1 \times 10^{-10}$ °C, a slightly higher error due to the higher number of calculations being performed allowing more errors to accumulate. Both errors are far from significant.

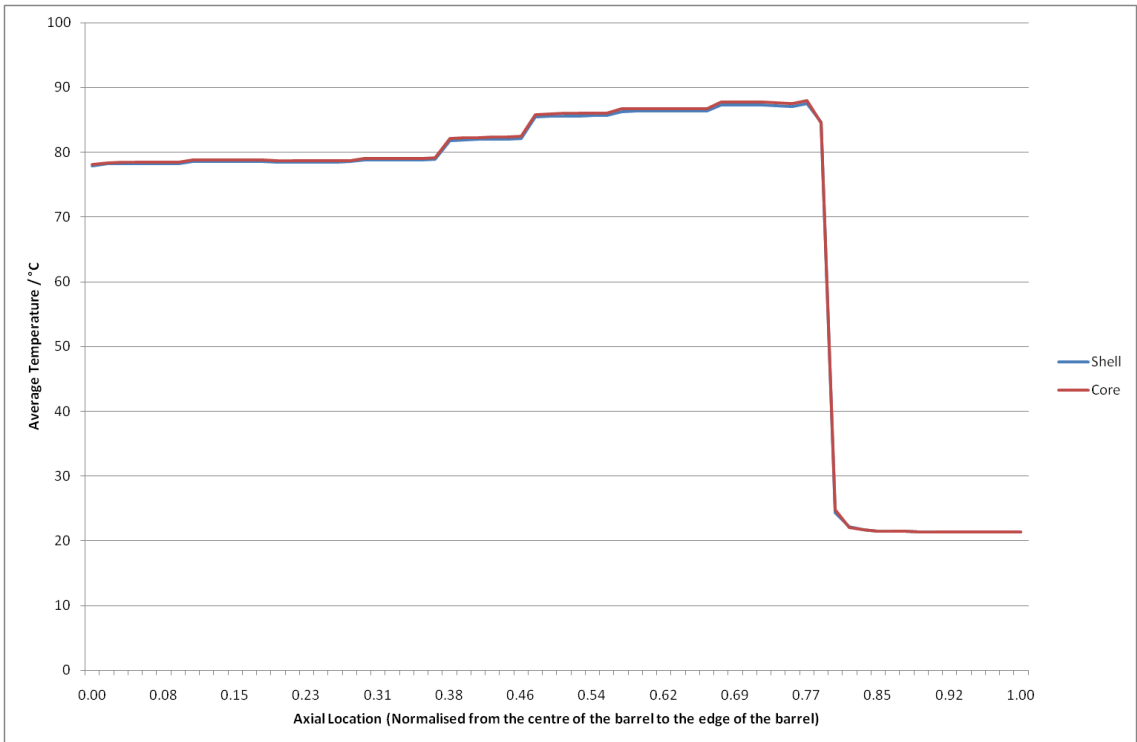
Element heat balances were performed on the shell model to ensure that it was conserving energy. The element whose results were given most scrutiny was an element just below the surface, where the radial heat gradient is at its highest, and in the roll bite, where the angular heat flow is at its highest. The heat flow at this location balanced to 12W, compared to 9.7W for the same element with constant material properties and 10KW being the largest component of heat flow through the element. This is a close enough agreement to be confident that the new models are actually conserving energy properly.

Figure 3-39 and Figure 3-40 compare the circumferentially averaged temperatures at the surface, varying in the axial direction from the centre of the barrel on the left to the edge of the barrel on the right. The difference between the core and shell models is drastically reduced in the new models. Figure 3-41 shows the change in temperature in the radial direction, from the interface between the core and shell models to the work roll surface. The shape of the two curves is not identical in this case, unlike when using constant material properties.

A potential explanation for the shape of the curve in Figure 3-41 is that the conductivity is much higher when the temperatures are high, so within the roll bite more heat is being conducted away to material below the surface. However, conductivity is reduced when temperatures are low, so under intensive cooling the ability of the surface material to conduct the heat from below the surface is reduced. On the scale of the core model resolution these features are too small to notice, and the difference in temperature is very small. However, this could be an interesting effect to quantify in a future study.

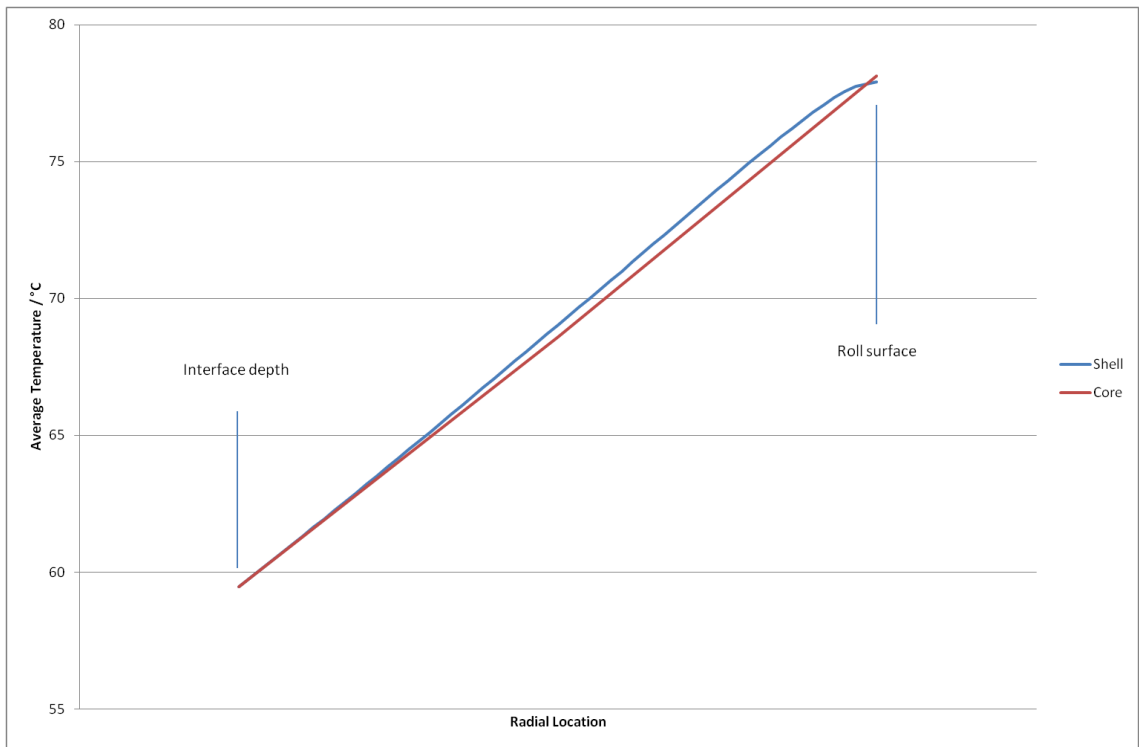


**Figure 3-39: Old core and averaged shell model temperatures varying axially with temperature dependent material properties**

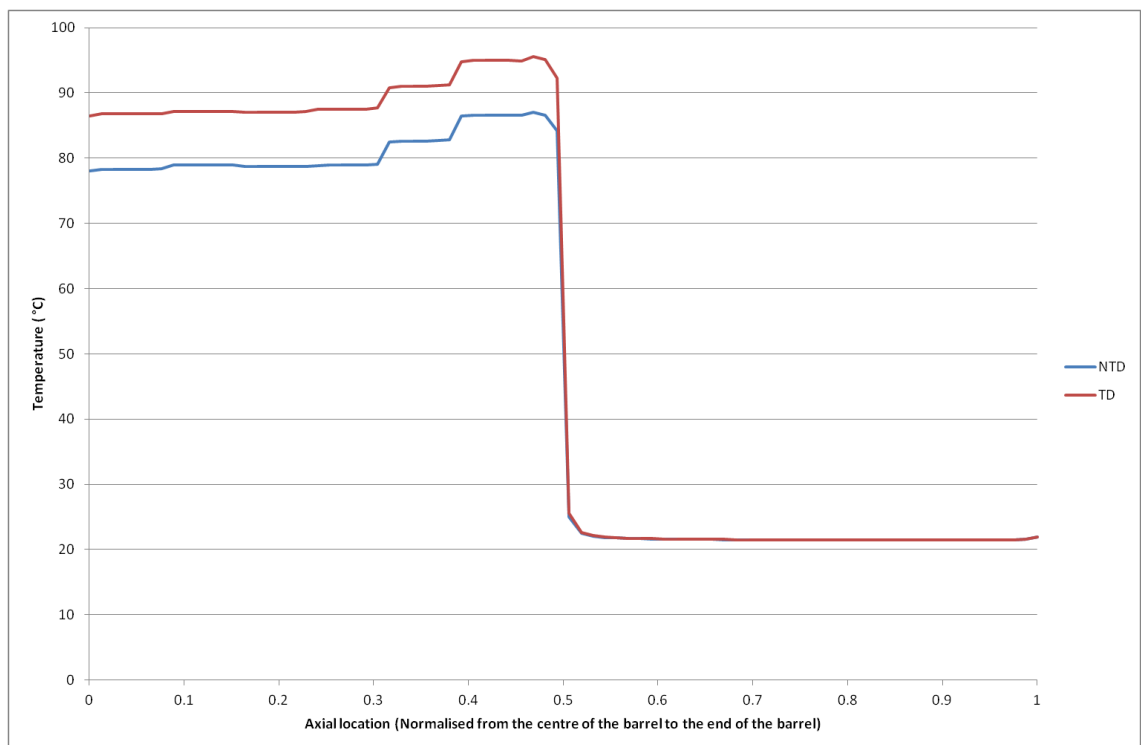


**Figure 3-40: new core and averaged shell model temperatures varying axially with temperature dependent material properties**





**Figure 3-41: Radial change in average temperature in the core and shell models**



**Figure 3-42: Comparison of axial surface temperature distribution between the new model with temperature dependent material properties (TD) and the old model without temperature dependent material properties (NTD).**

Figure 3-42 shows the difference between a model implementing temperature dependent material properties and a model which does not. The figure was generated using the data in

Appendix D.3. The increase in conductivity at higher temperatures results in significantly higher temperatures at the surface.

### 3.7. Conclusions

---

A considerable amount of time was invested in the thermal models during this project, with significant updates to both models. Initially the coefficient array storage for the solver was rearranged to improve efficiency in the solver loop, then the centre symmetry condition was expanded to allow for asymmetrical rolling conditions. Initial tests with ABAQUS revealed errors in the shell model, leading at first to adjustments to the work roll surface boundary conditions and then implementation of a term to emulate the effects of the change of temperature with time for the shell model. Further testing showed that the resolution of the shell model was insufficient to accurately define the temperature distribution near the surface, suggesting that the shell model resolution should be increased by a factor of 30. The increase in the resolution of the shell model resulted in a large increase in runtime, so additional efficiency improvements were considered, though none were implemented due to the required development time.

During a trip to the IJmuiden RD&T department, an interest in the effect of temperature dependent material properties was expressed, so a rearranged form of the solver equation was developed to allow the material properties to vary with temperature. A second trip to the IJmuiden RD&T department yielded a new equation for the heat transfer coefficient in the roll bite. The implementation of the new heat transfer coefficient changed the way the core and shell models interacted, resulting in the core model boundary conditions needing to reflect angular temperature variation. While introducing estimates of angular temperature variation to the core model, a lack of energy conservation was revealed in the thermal models which used temperature dependent material properties. This discovery prompted an adjustment of the mathematical underpinnings of the thermal models to include terms relating to the changing values of the material properties. The new models have been shown to conserve energy to the same level as the non temperature dependent models and compare very well with each other. An unusual feature of the radial temperature profile of the temperature dependent shell model has been discovered, causing the surface to be slightly cooler than expected, while the material just below the surface is slightly warmer than expected. This is caused by the increased conductivity of the roll material at high temperatures, conducting energy efficiently under the roll bite where temperatures are high, but not so efficiently under the cooling sprays where temperatures are low.

Despite all the changes to the models, the majority of the testing and validation work performed on the White model<sup>3</sup> can be applied to the current models. This is due to the fact that most of the testing concerned internal heat flow, resolution dependency and numerical stability. The only update which threatened the applicability of this testing and validation was the application of temperature dependency to the material properties. The new models were tested against the original models while using constant material properties, showing that the only differences were on the scale of rounding errors. Comparing element energy balances against each other showed that, even though the models using temperature dependency are giving different overall results, they are still conserving energy to a similar scale as the original models.

## 4. The Stress Model

---

### 4.1. Introduction

---

In the previous chapters the thermal models were introduced. This chapter will describe the models used in the next phase of the analysis to predict the stress distribution. As the literature review concluded, developing a custom modelling solution would have been prohibitively complicated, so third party software was required. The ABAQUS software package was chosen due to its established use within Tata and Cardiff University, its prominence in the field of simulation software and also to preserve the experience gained from previous modelling work performed by Daniel White.

The rest of this section (section 4.1) will be dedicated to introducing the concept of finite element modelling, the ABAQUS software package and the overall modelling strategy. Section 4.2 describes the processes by which the models are constructed, while section 4.3 discusses the boundary conditions. The precise numerical values used in the model, along with overall validation and results, can be found in Chapter 5.

#### 4.1.1. Finite Element Modelling

The finite element (FE) method is a way of approximately representing the behaviour of a continuously varying substance using small areas or volumes with constant (or constantly varying) properties. The method can be applied in a number of ways depending on the properties and geometry of the item being modelled. For example, single dimensional line elements whose deflection response reflects a constant cross-section are a very efficient way of modelling structural frames. In the case of this project the component being modelled is a three dimensional solid with significant dimensions in all three directions. The appropriate element type for this model is therefore a 3D Solid element as shown in Figure 4-1.

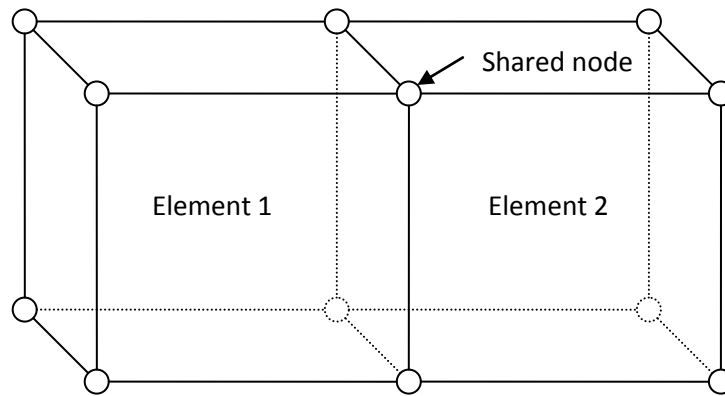


Figure 4-1: Two 3D solid elements

For a volume with a curved edge, a large number of these “linear” elements would be required before the straight element edges would conform closely to the curved edge of the original volume. In such cases “quadratic” elements can be the better option. In a quadratic element each edge is defined by three nodes, one at each end and one in the middle. The edge is therefore given by the circular arc which passes through all three points. While this can give accurate conformity with fewer elements it also increases processing time, as calculations need to be performed for each node. In practice a greater number of linear elements can out-perform a lesser number of quadratic elements.

The deflection response of each element to loading will be a complicated three dimensional problem, but can be simplified by stating that each element can only be loaded by direct forces at the nodes. This requires that large numbers of elements are used in areas of highly variable loading but allows a solution to the deflection problem to be achieved. Solving each element individually is not possible since even when some boundary conditions are known the element will be affected by the behaviour of the surrounding elements. Instead the equations for every element are considered together as a set of simultaneous equations which are linked by shared nodes at element corners.

The result of solving the simultaneous equations will be a deflection value (in three dimensions) for each node. With the deflections of each corner of an element known, the strain tensor can be evaluated and used with the material’s stress-strain relationship to determine the stress experienced by the material over that volume.

#### 4.1.2. The ABAQUS Package

ABAQUS is a powerful software suite, three main components of which are used in the project.

- a) ABAQUS CAE – responsible for creating and organising all of the inputs and interacting directly with the user
- b) ABAQUS Standard – the processing component, accepts inputs and delivers results
- c) ABAQUS Viewer – exclusively responsible for organising, displaying and formatting all model results

Figure 4-2 describes a standard model development process.

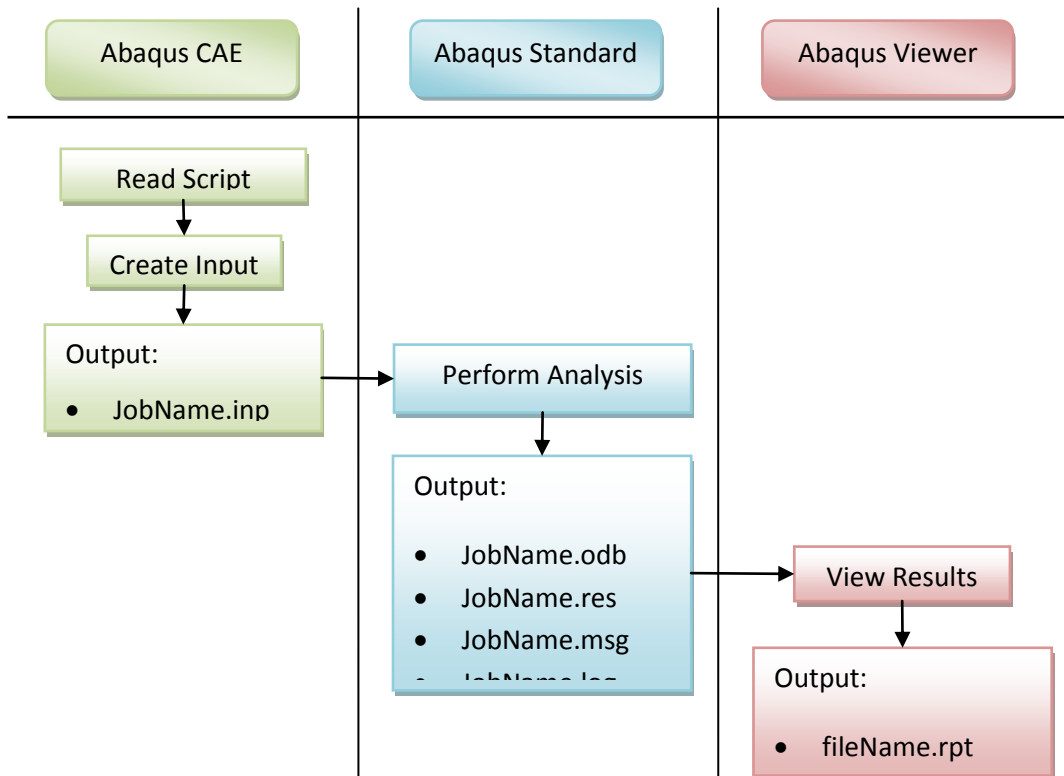


Figure 4-2: Process flow chart

## ABAQUS CAE

The first step in Figure 4-2 represents the input of a python script, a file containing a list of instructions for ABAQUS CAE to follow, written in the programming language “Python”. This allows the model construction process to be automated or (with a particular kind of python script called a “journal file”) progress on an existing model to be resumed by repeating the steps performed up to that point. Not only are the journal files useful in saving disk space by reducing saved file size requirements, they also provide an easy trial-and-error method for determining the python code required to perform an action in ABAQUS CAE.

In the second step ABAQUS CAE converts the model definition from its own internal types to a text file format called the “input file”. This is the format used to communicate all the

model information (except in user subroutines, see section 4.3) with ABAQUS Standard which performs the analysis. If the software user is directly interacting with ABAQUS CAE at this point then he/she will usually allow ABAQUS CAE to run ABAQUS Standard and ABAQUS Viewer, rather than operating each component manually. The same can be done from the scripting interface, though for numerous reasons (see Appendix B) it is better to control ABAQUS Standard directly.

The result of the first and second actions is the creation of an input file for the third step. This file can be opened and edited in order to add commands unavailable within ABAQUS CAE or can be fed unaltered into ABAQUS Standard.

### *ABAQUS Standard*

---

The user has very little control over ABAQUS Standard once it has started an analysis. It only communicates via input files, the arguments with which it is initially opened and an output stream which updates the user on what it is currently doing. If an error is encountered the details of this error are made available in a text file (ending in either .dat or .msg), otherwise the calculated model results are saved in the output database file (.odb). If the model results are from one stage in a multi-stage analysis then the restart file (.res) contains the information ABAQUS Standard will need to restart the analysis using the output of that step as the input of the next step. As it performs all of the actual analysis work this part of the ABAQUS package takes significantly longer to run than any other.

### *ABAQUS Viewer*

---

Although technically a separate program to ABAQUS CAE, ABAQUS Viewer acts as an additional module in ABAQUS CAE. The sole function of this component is to open output database files and display the results. Various plotting functions, section views and other results manipulations are available and any of these functions can be activated from the scripting interface and the resulting view saved to an image file. However, due to the complexity and inflexibility inherent in working with scripting commands between programs it is also useful to export the important information out of the ABAQUS specific database and into a more neutral format. This is most efficiently achieved using “field output reports”, which print the values of specifically requested variables at every point in the model to a text file (.rpt). The use of this text file will be covered further in Appendix C.

### 4.1.3. ABAQUS User Interface

To ensure that the project software is useable by the widest range of Tata personnel possible, the ABAQUS CAE user interface is not shown to the user unless specifically requested. This reduces the requirement for users to understand the inner workings of ABAQUS. However, this interface has been used extensively in the development of the stress models and is presented here as a means of explaining how ABAQUS CAE is organised.

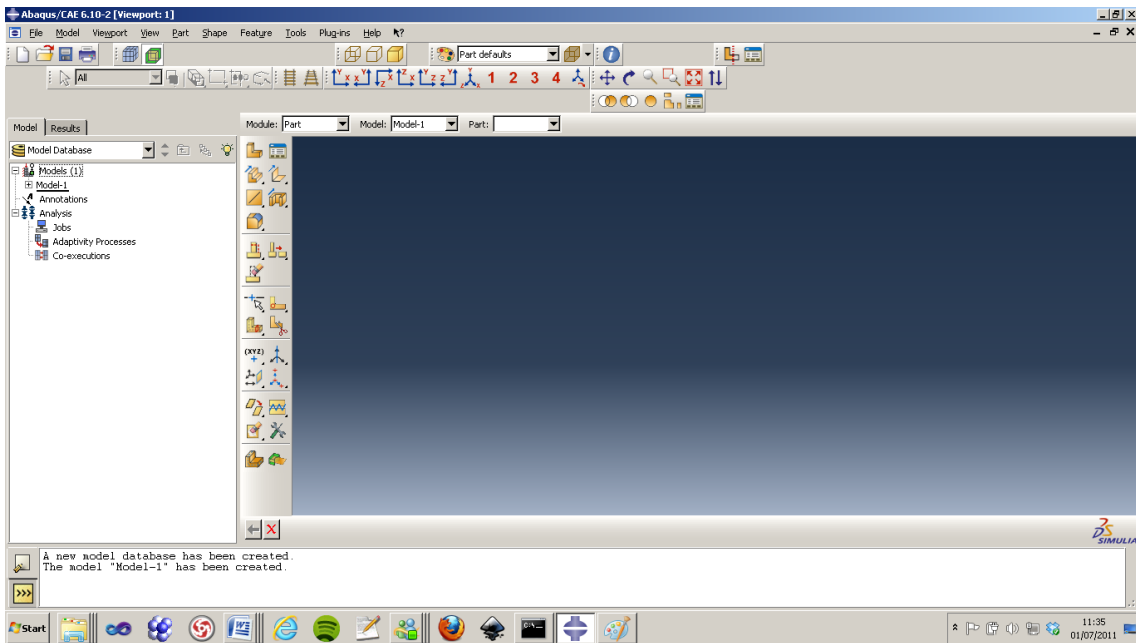
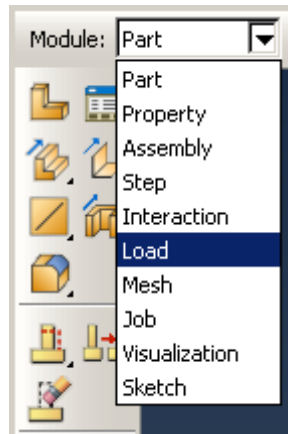


Figure 4-3: ABAQUS CAE user interface

Figure 4-3 gives an example of the user interface when a new file is created. The area to the right with a blue gradient background is the “Viewport” and shows three dimensional rendered images or “views” of selected model components. As the file has only just been created this view is currently empty. At the far left of the screen is a tree view which is currently on the “Model” tab and contains the “Models”, “Annotations” and “Analysis” headings. This provides quick and easy access to any component, condition or property of any model currently loaded. Between these two controls is a context sensitive toolbar which can be used to perform the majority of tasks available to CAE. As usual in Windows-based software, the menu bar along the top of the screen gives access to the most exhaustive list of functions.

The modules available to the central toolbar are shown in Figure 4-4 and break the modelling process into a series of logically separate tasks.





**Figure 4-4: Module selection**

The “Part” module is responsible for drawing the model geometry. Each “part” is a whole area or volume which can logically be defined as an individual entity. For example, when modelling a whole bicycle, a pedal would be a part, as would a gear block or a brake lever. A work roll is a solid object and therefore not reducible to multiple parts, so gains very little from this functionality. However, more complex models can benefit greatly from the facility to construct components in isolation and assemble them later.

The “Property” module is used to apply material properties to and specify the dimensionality of the model parts.

In the “Assembly” module the model is put together from the previously created parts. The same part can be used more than once, each copy being referred to as an “Instance” of that part. This module contains functions to move, align and merge or cut instances in order to create the fully assembled model.

The “Step” module creates “step” items, abstract modelling objects which represent different types of analysis. Within this module the choices between transient or static behaviour, deformation or thermal analysis must be made. Most of the choices are tailored towards fine control of transient behaviour or the interactions between contacting bodies, so the selection of a “Static/General” analysis type and a direct solution are sufficient for this project. This module is also responsible for field output requests, options which enable different types of output to be saved after the analysis has completed. Field output requests allow the user to reduce the size of the results files considerably by prompting ABAQUS to delete extraneous information.

The “Interaction” module is not dealt with in this project, as the work roll consists of only one part. If multiple parts were required then commands in the interaction module would allow the relationship between these parts to be specified and controlled.

The “Load” module is where the boundary conditions are applied. Available boundary conditions include but are not limited to;

- Restraint conditions – Prescribed deformations to prevent or enforce rotational or translational deflection of the part. Enforced deflection can be used as a form of loading but is most commonly used to restrain unwanted motion.
- Point forces – A load vector which acts directly on a given node.
- Pressure – A distributed load, applied to a surface or volume and translated into nodal forces by ABAQUS prior to analysis.
- Pre-defined fields – e.g. temperature distributions or initial stress fields. These can be applied to specific features or to the whole model.

The “Mesh” module is responsible for meshing, the process of breaking a large, often geometrically complicated component into small and geometrically simple and regular elements. The process can become quite involved and will be dealt with in detail in the next section.

The “Job” module is the final module in the analysis chain, being used to create the analysis requests which will be sent to ABAQUS Standard. The user can choose to “Submit” a model, i.e. send the information to ABAQUS Standard and await the model results, or to “Write Input” which will save the information to file without running an analysis. The final option is the more useful because, as previously mentioned, it allows editing of the input file to include functions unavailable in ABAQUS CAE.

The “Visualization” module opens ABAQUS Viewer in a tab in ABAQUS CAE, allowing all the tasks performed by ABAQUS to be accessed through a single interface.

The final module, “Sketch”, manages the two-dimensional drawings used to create and edit parts and instances.

#### 4.1.4. General Modelling Strategy

##### *Requirements*

---

In developing a strategy to create a valid work roll stress model it is important to precisely capture the model requirements.

- a) To give a full stress distribution for the whole roll

Regardless of which area of the work roll is of most interest, the elastic response of the work roll as a whole will have an effect on the results. Without a well known analytical solution to provide valid boundary conditions, no focused solution can be attempted without a full-scale model. A linear-elastic response will be assumed for simplicity. This assumption can be adjusted at a later date if the elastic limit is found to be significantly exceeded.

b) To provide a fine resolution near the surface of the roll, especially around the roll bite

Near the surface of the roll the spatially rapid temperature fluctuations provide very localised stress generating expansions and contractions. These fluctuations are likely to be small in scale but could be great in magnitude, and they require a large number of elements to establish the results accurately.

c) To model thermal, mechanical and residual loading

To provide a detailed stress picture this project combines the thermal loading from contact with the hot strip with the effects of mechanical loads caused by the rolling process and the residual stress profile from manufacture.

d) To accept data from the thermal models

For the model to be relevant it must be based on the same data used to generate the thermal profiles, so that errors do not arise due to mismatches in these quantities. The two models are uncoupled. The thermal model calculates the temperature profiles and ABAQUS uses the temperature distribution calculated for the roll to define the expansion of each element and thus arrive at the corresponding stress distribution.

e) To be easy to use by personnel untrained in programming or the use of complicated third-party software applications

One of the project goals is to create a tool that can be used by a wide range of personnel. To this end the model must be automated to de-skill the process and improve accessibility.

f) To achieve mesh independence

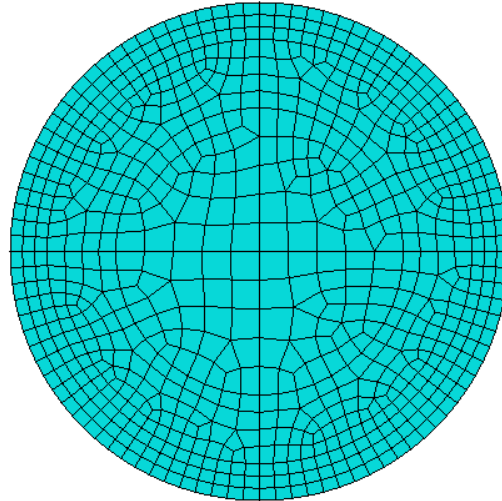
The test of a good finite element mesh is that its resolution can be changed without impacting the results. This should be achieved and observed to ensure that the model structure itself is not introducing significant errors.

*a) & b) - To give a full stress distribution for the whole roll and provide a fine resolution at the surface*

---

To meet requirement a) it is necessary to fully model a work roll which, allowing for variability between rolls, is roughly four metres long and over half a metre in diameter. To meet requirement b) the model must have a high enough resolution to discern the differences caused by fluctuations in the loading whose spatial dimensions are measured in millimetres. For a three dimensional component and on a conventional computer it is not possible to achieve both these aims with a single model.

The solution to this apparent impasse is a technique called sub-modelling. The process begins with a global model, which simulates the general behaviour of the body as a whole. Such a model is illustrated in two dimensions for simplicity in Figure 4-5 below.

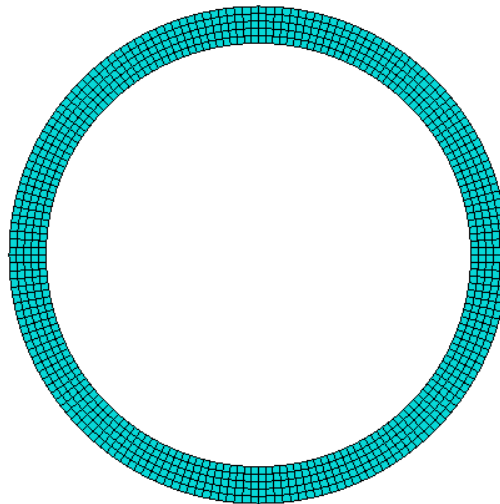


**Figure 4-5: A global model**

The global model has relatively uniformly sized elements, allowing the automatic meshing tools to create a smooth mesh with few poorly shaped (non-square) elements. Four sided elements provide the best possible accuracy when they are square in shape. Elements that depart significantly from this ideal shape are "poorly shaped". There is some compromise in element shape required to render a circular shape from quadrilateral elements, but the benefits to output accuracy of square elements outweigh the geometric versatility of

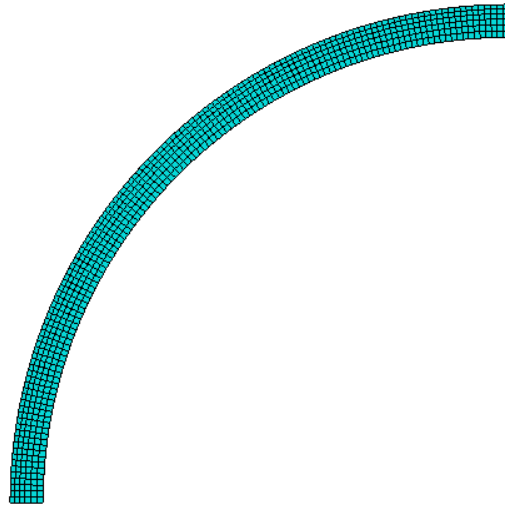
triangular elements. The global model has a coarse resolution, allowing the whole volume of the work roll to be simulated with only moderate computational load.

At the centre of the work roll the global model's resolution is perfectly acceptable for capturing the slow changes in stress intensity. Near the surface of the roll, however, sharp fluctuations in temperature and large mechanical loads applied over small areas will generate stress features on a far smaller scale than the global model can accurately resolve. To better define this area a second model is introduced.



**Figure 4-6: A sub-model**

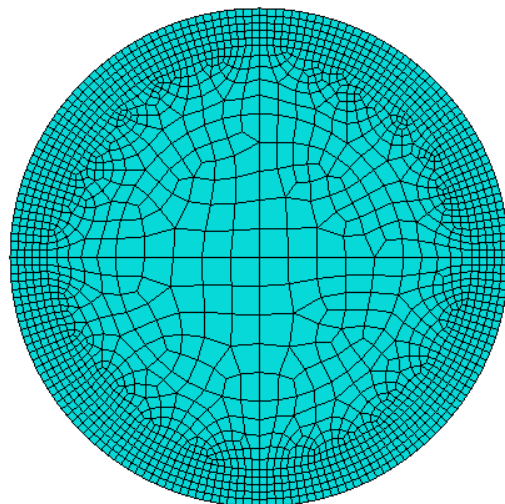
Figure 4-6 shows a model whose outer radius overlaps with the outer radius of the global model, but whose inner radius lies inside the global model. The displacement of each node on the inner radius is forced to be the same as the displacement of the equivalent point in the global model, thus preserving the influence of the material which is not considered by this sub-model. Without the coarse internal mesh to consider and with a smaller volume to cover, the elements can be smaller and well structured throughout the model. This has the effect of improving the precision of the results in the area near the surface, where precision is most important.



**Figure 4-7: A second sub-model**

While the sub-model described in the previous paragraph provides a sufficiently high resolution for observing the effects of most circumferential fluctuations, the surface elements are still quite large compared to the size of the roll bite itself. The model shown in Figure 4-7 solves this problem by implementing a second iteration of the same technique, basing a sub-model on the results of the previous sub-model. With the modelled volume further reduced the elements can be made small enough to track the very fast changes occurring within the roll bite.

The benefits of the sub-modelling approach can be seen by comparison with Figure 4-8 below.



**Figure 4-8: A single model with a high resolution at the surface.**

Figure 4-8 illustrates a single model compromise between requirement a) and requirement b), i.e. the whole of the roll is considered and there is a high resolution at the surface. This is

achieved by drastically changing element size between different parts of the roll in order to get a practicable balance between resolution and model complexity. The problems this model encounters and which are avoided by the use of sub-models are;

- Inaccuracy – Caused by elements not being square and some elements having longer edges on one side than another.
- Very long run times – Caused by large numbers of elements, e.g. the same resolution near the surface as the first sub-model but having to model the material below the sub-model inner radius as well.
- Reduced resolution – Increasing the resolution to the same level as the second sub-model would make the single model larger than any standard computer is capable of processing.

#### *c) – To model thermal, mechanical and residual loading*

---

The thermal stresses are only a part of the stress experienced by a work roll during normal operating conditions. Both the mechanical loading of the roll and the residual stresses remaining from manufacture will have an impact. This impact could increase maxima and minima or even change a component from compressive to tensile, or vice versa. When considering the effects of stress on the work roll material performance these issues need to be considered.

All of the components stipulated in this requirement are applied using user subroutines. An ABAQUS user subroutine is a file containing code written in the Fortran programming language. The beginning and end of the file is stipulated by ABAQUS to give the name of the subroutine and the information which will be supplied to it and must be returned by it. The user is free to fill the space between with any legal Fortran code they wish. This gives the user the flexibility to create any relationships they desire, or to import data from files or any other sources accessible to the Fortran language. The subroutines used are described in section 4.3.

#### *d) – To accept data from the thermal models*

---

The communication between the thermal and stress models incorporates many different kinds of data. The most obvious is the temperature profile used to thermally load the stress models, but no less important are the work roll geometry and material properties, loading

magnitudes from the hot mill database, the correct folder to output all the results files into, the size of the roll bite, etc. All this information is transferred by file, either directly written into the python script instructions from which the models are constructed or into data files which are opened by user subroutines.

These and other programming considerations are discussed in Appendix C.

*e) – To be easy to use by personnel untrained in programming or the use of complicated third-party software applications*

---

The end user of the software developed in this project will be interested in the thermal and stress response of work rolls, but not necessarily in simulation software. Therefore requiring extensive interaction with ABAQUS might be counterproductive. To streamline the simulation process and ensure continuity in the user experience, the ABAQUS user interface is hidden from the user and controlled from behind-the-scenes with a “scripting interface”.

The scripting interface shortcuts all manual user input (such as clicking on buttons or drawing shapes on the screen) by providing a file containing a list of instructions which give ABAQUS all the information it needs to build the model. The instructions necessary to simulate a work roll will vary, depending on the size and shape of the roll and its material properties. Therefore, rather than using one set of instructions for everything, the script file is created by the thermal model software, allowing it to change dynamically and automatically. By interacting with a simple setup window in the thermal model, the user can specify material properties and analysis types. The thermal model will then do all the work of applying those settings to the stress model. More detail on how this is accomplished can be found in Appendix C.

*f) – To achieve mesh independence.*

---

There are many ways for a finite element model to provide untrustworthy results. The most basic is a failure of the software to accurately reflect reality, which is guarded against to a large degree by the use of well respected third party software. The next most likely error would be incorrectly applied boundary conditions, which can be minimised by careful consideration and disproved by coherence with experimental results. However, before any of these issues can be considered it is important to ensure that the elements used are the best



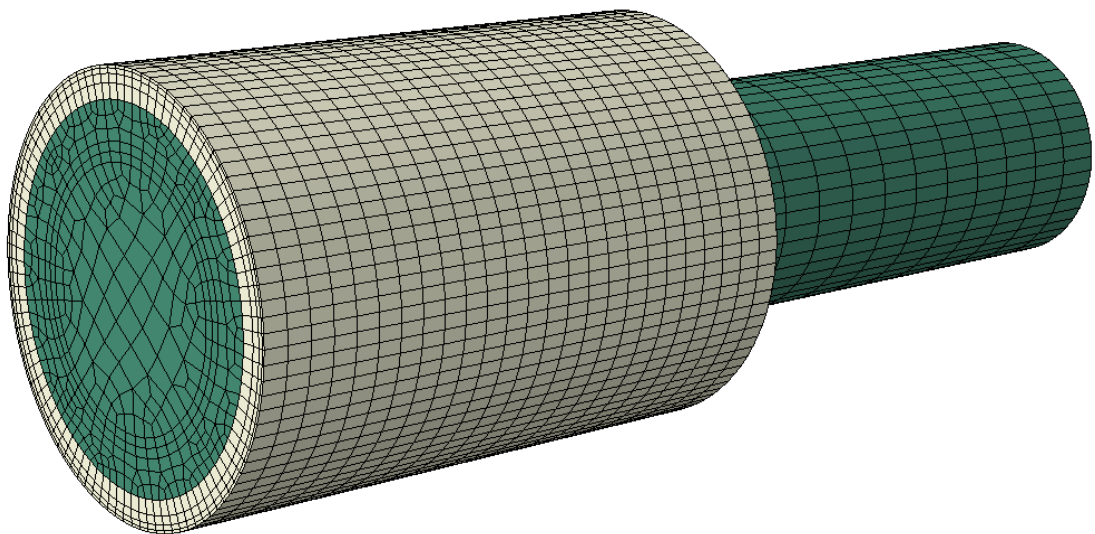
shape possible and of appropriate size. Problems in a mesh can generate wild deviations that can compromise model results, even if all other aspects of the model are perfect.

The easiest way to affirm mesh quality is to scale up the element count and prove that further refinements of resolution do not produce further refinements in the results. Unfortunately in the models proposed in the following section the capacity to increase resolution is hampered by hardware constraints. Therefore a two-dimensional radial-circumferential model is introduced in section 4.2.4 which can attain a far greater resolution than the main, three-dimensional models. Although not an accurate representation of the stress state of a work roll, the two-dimensional model can assess the influence of circumferential temperature variations and the resolution required to capture those influences.

## 4.2. Model Construction

---

### 4.2.1. Global Model



**Figure 4-9: A global model**

An example global model is illustrated in Figure 4-9 above. Since the purpose of this model is to capture the overall trend of the stress response at a distance from any stress generating features the resolution does not need to be high. However, the resolution must certainly be high enough to define those features which lie within its area of responsibility. This example makes use of symmetry either side of the roll centre to reduce the volume of the roll, allowing for a slightly higher resolution. The symmetrically reduced type of model will be

the type discussed in the following section as the reduced and non-reduced versions are very similar and the reduced version is the more complicated.

## Geometry

---

Figure 4-10 shows the key measures used to define the shape of the work roll. While most of the figure is self explanatory it is worth pointing out that the “length” parameter refers only to the barrel, excluding the journals. The “shell depth” parameter gives the depth of the interface between the hardened shell and softer core materials. This is not to be confused with the location of the interface between the thermal “shell” and “core” temperature models, which is decided by thermal effects.

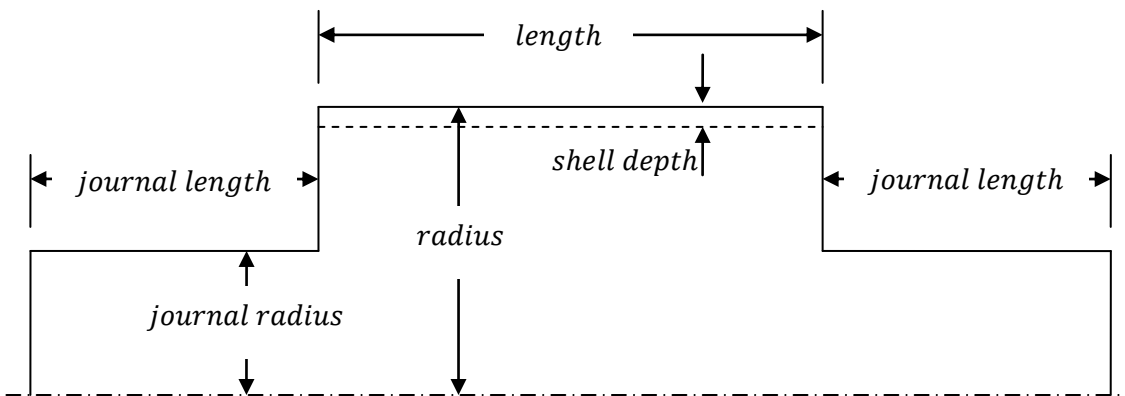


Figure 4-10: Work roll generating shape and dimensional notation

The shape of the global model is chosen to conform with the shape of the thermal models rather than the actual work roll, although there is a good similarity. The failings of the chosen shape are;

- poor representation of the journals

The only mechanical failures around this area of the work roll are always either due to bearings seizing up and the drive motors tearing a journal off, or stress-corrosion. Since the corrosion aspect of the latter failure mechanism is beyond the scope of this model and the bearings are a separate issue altogether, fine detail in the journal area is not necessary.

- lack of detail at the edge of the barrel

The stress raising effects of any sudden change in the profile of a component are well enough known throughout the scientific community as to not need introduction. The very sudden change in radius from the barrel to the journal would infer a large stress concentration. In reality there is always a slight reduction in radius at the barrel edge to prevent crack propagation due to stress concentration effects. This reduction is not included

in the model as it would complicate both thermal and stress models in an area where failure causes are well understood and already mitigated against.

- no camber on the work roll surface

The surface of the work roll is not perfectly flat but curved so that the radius in the centre is less than the radius at the barrel edge. This camber has been ignored as it is very small (typically less than  $1\text{mm}^{31}$ ) and none of the boundary conditions require that the surface profile is exact.

Figure 4-11 shows the sketch used to create the outline of the work roll and the three dimensional part created by rotating that sketch around the central axis. Both these and the following images were created from the ABAQUS user interface for illustrative purposes only, the end user is presented with a limited range of update messages during this process. As only half of the roll is being considered the barrel length will be  $length/2$ .

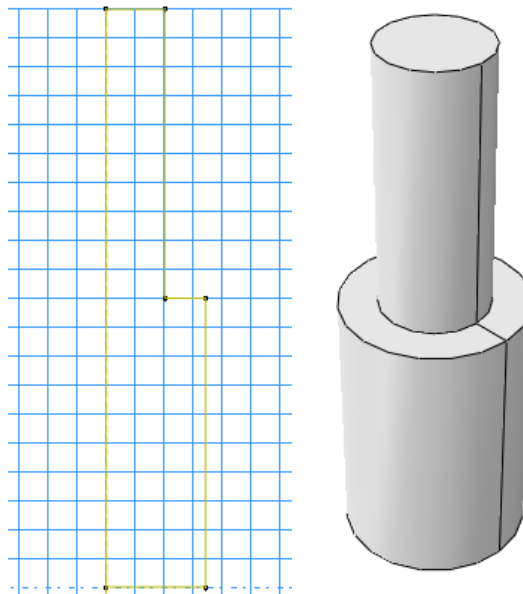
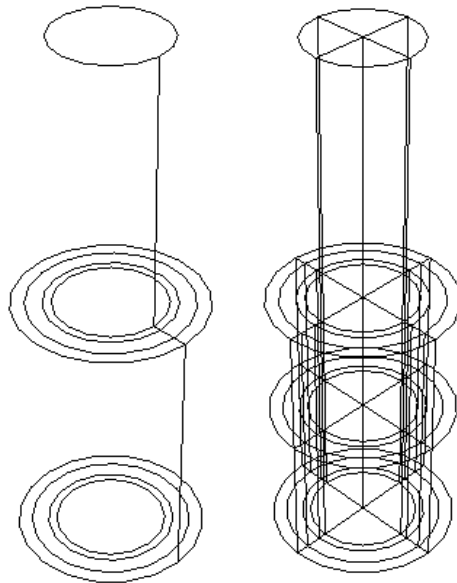


Figure 4-11: Drawing the basic geometry (left) and rotation to create a solid part (right)

In Figure 4-12 the geometry is split up first radially on the left, then circumferentially and axially on the right (surfaces are completely transparent, only edges are displayed). There are a number of reasons for this. Firstly, at least one radial partition would be required in order to distinguish the shell material from the core material. An axial partition halfway up the barrel is positioned to line up with the edge of the strip, which is useful for controlling mesh density and applying boundary conditions. The remaining partitions simplify the meshing process, breaking up the internal volume of the part into volumes of constant cross-section and creating a number of edges which allow the meshing tools to be deployed effectively. The importance of this will be discussed later. The circumferential partitions are particularly

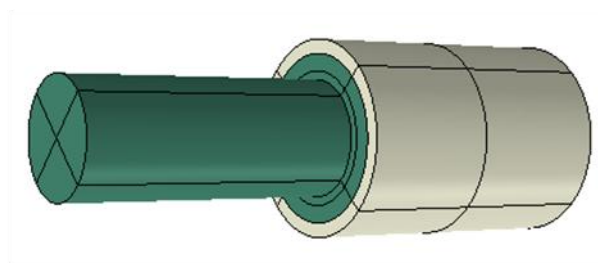
necessary given that ABAQUS's automatic meshing tools do not work well with annular cross-sections.



**Figure 4-12: Partitioning to facilitate the meshing process and separate material sections**

### *Material Properties*

---



**Figure 4-13: Material selection**

The material parameters for the core and shell are both included in the input script, so material definitions can be created and assigned to the relevant parts of the model. These material definitions include mechanical properties, such as Young's modulus and Poisson's ratio, as well as thermal properties, such as the coefficient of thermal expansion. Figure 4-13 shows the separate application regions, with the shell material highlighted in grey.

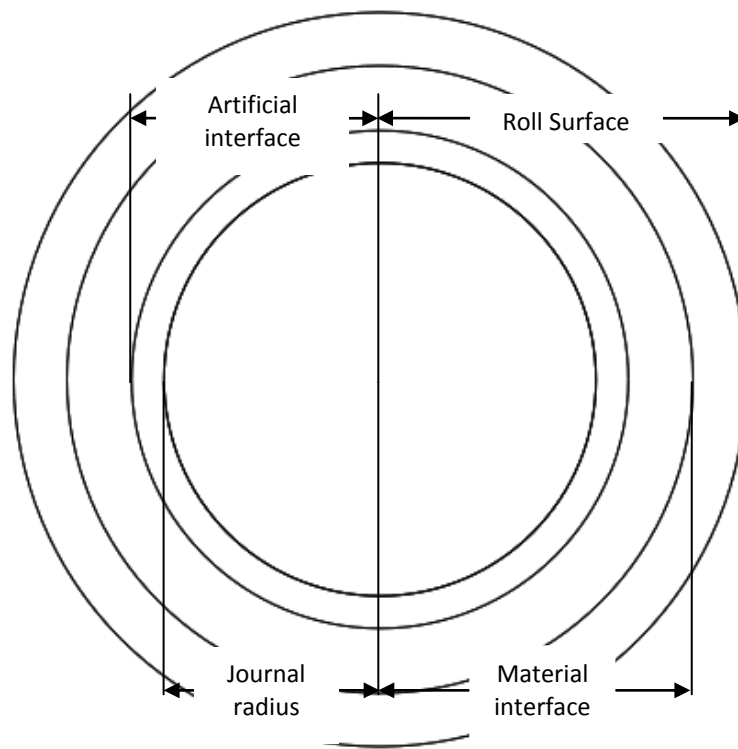
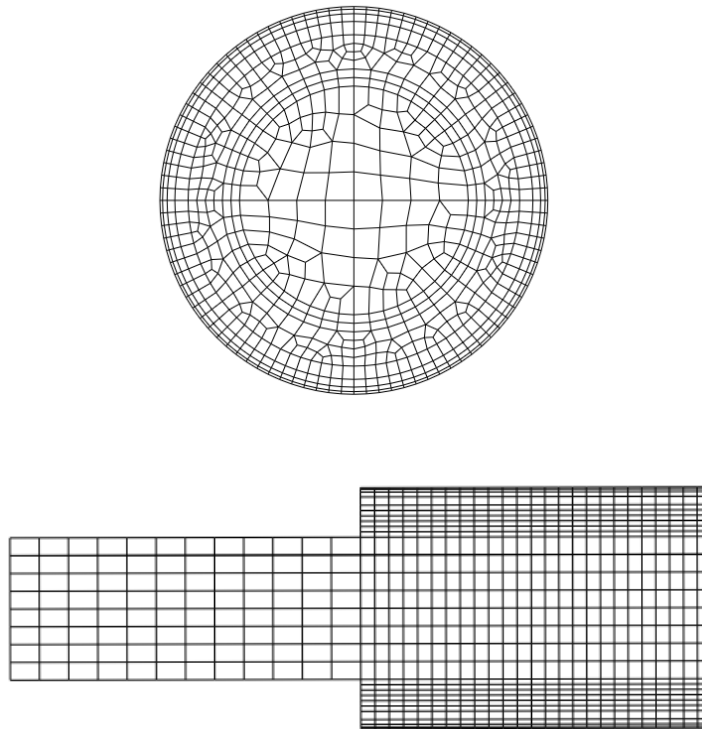


Figure 4-14: principle radii of a work roll model

Figure 4-14 shows an ABAQUS screen-shot detailing the four most important radii involved in creating a finite element model of a work roll. Starting at the outer radius and moving inwards these circles represent;

- The surface of the work roll
- The material interface between the shell and core material
- A control surface allowing the meshing strategy to be controlled
- The radius of the work roll bearing

To mesh this shape these curves need to be 'seeded' or broken up into multiple smaller curves. The size of these smaller curves dictates the resolution of the results, as they represent the distance between each calculation point or "node".



**Figure 4-15: Mesh structure in the radial-circumferential (top) and radial-axial (bottom) planes**

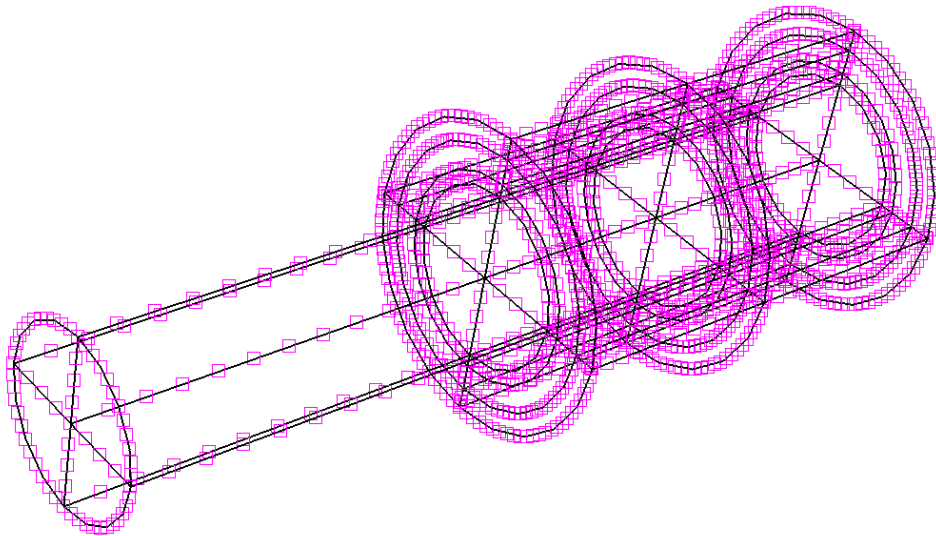
Figure 4-15 illustrates the mesh structure to be used by the global model, although the precise details will change dependent on the geometry of the work roll being studied. In the radial-circumferential plane the approach to creating the mesh can be broken into two distinct categories. At the surface and just outside the radius of the journal there are two areas of very uniform meshing. Between the two areas of uniform meshing and at the centre of the roll there are areas of relatively uneven meshing. This scheme provides;

- Even and dense meshing resulting in well resolved results near the surface
- Uneven meshing where the number of elements per edge needs to be different
- Even meshing in the intermediary area, because the mesh density does not need to change
- Uneven meshing where the profile is unsuitable, i.e. a quarter circle cannot be meshed regularly with square elements

In the radial-axial plane the mesh is relatively uniform due to the simplicity of the geometry in this direction. The only variation is a reduction in the mesh density in the journals, to reduce computational load by reducing resolution in an area of little interest, and an increase in mesh density near the surface to improve resolution where resolution is key.

The 3D elements produced are all extrusions of the element shapes in the radial-circumferential plane, and so are six sided "bricks".

Figure 4-16 highlights the seed positions required to generate the mesh in Figure 4-15, showing the breaks in the existing curves that will serve as corner points for elements once the meshing process is complete.



**Figure 4-16: Seeding constraints**

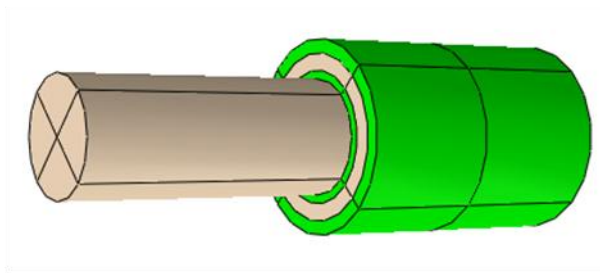
## *Meshing*

---

Figure 4-17 shows the area of application for the different meshing schemes used to create the mesh. The green shading covers the evenly meshed areas and represents the "Structured" meshing scheme, while the grey shading covers the unevenly meshed areas and represents the "Bottom-Up" meshing scheme.

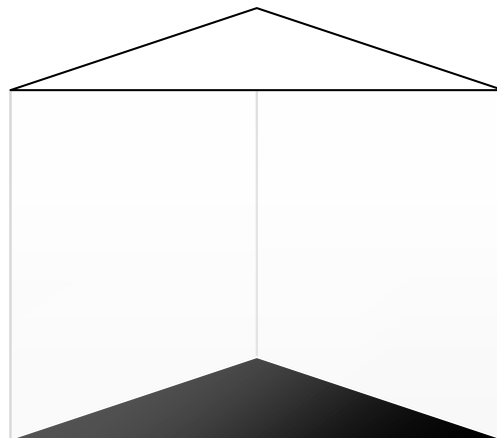
The structured scheme is the default meshing scheme as it is guaranteed to give a uniform and therefore highly accurate mesh. However, if the same face mesh cannot be used on all opposing faces then the structured scheme will fail to produce any mesh at all.

In cases where the "Structured" scheme is inappropriate the "Swept" meshing scheme will often provide a solution, requiring only that the cross-section be similar in one direction. The tool will automatically choose a sweeping direction and will normally pick the extrusion or rotation direction used to form the part. In this case the sweeping direction would be the circumferential direction, but should be changed to the axial direction. This will allow the tool to generate a face mesh on the radial-circumferential plane, where the meshing constraints are most stringent, and sweep the resulting mesh through the volume in the axial direction to create the volume mesh.



**Figure 4-17: Meshing strategy**

Unfortunately the automatic meshing tools can be temperamental and can fail to operate properly in numerous ways. A safer approach has been found to be the “Bottom-Up” meshing scheme, which allows a manual approach to the meshing process. The “Sweep” version of the “Bottom-Up” meshing scheme allows the user to specify the source side, target side and contact sides. Considering Figure 4-18, the source side is the white face and represents the cross-section of the swept area for which the complicated mesh must be made. The target side is the black face and represents the opposite end of the volume. The connecting sides are the three grey sides which control the size, shape and location of the cross-section at every point between the source and target sides.



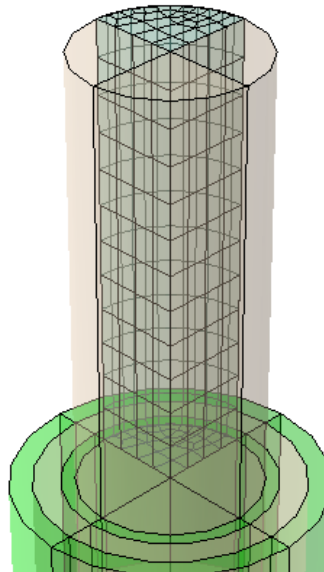
**Figure 4-18: Selecting sweeping faces**

Although some extra work is involved in using the “Bottom-Up” meshing scheme it is much more reliable. As the whole process has been automated the software user will not notice the added complexity of the process but will benefit from the improved stability.

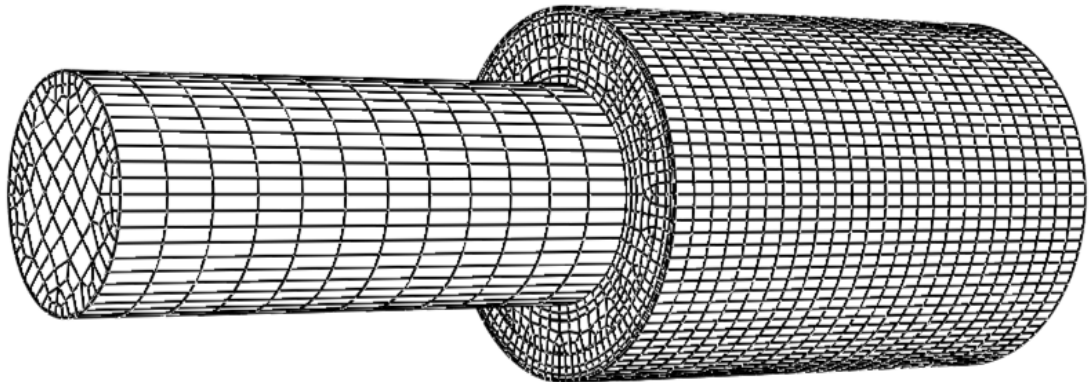
Figure 4-19 gives an example of a model with a single meshed cell. The top face mesh is fairly complicated, being a difficult shape to mesh with square elements, and is therefore the source face. The bottom (target) face is meshed identically to the source face. The source face is swept through the volume, creating a new “slice” of mesh at regular intervals. The



nature of this process means that the meshes on the connecting faces will be very regular. The fully meshed part is shown in Figure 4-20.



**Figure 4-19: One meshed cell**

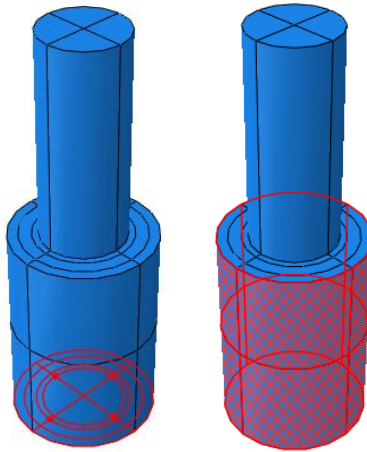


**Figure 4-20: Fully meshed part**

### *Boundary Conditions*

---

The final step in creating the global model is to apply the boundary conditions. The conditions applied to the model will vary depending on the type of analysis requested by the user, the options including any combination of thermal, mechanical and residual loading, and whether the model should be symmetrically reduced. Highlighted in red on the left in Figure 4-21 are the faces subject to a symmetry condition in this symmetrically reduced model.



**Figure 4-21: Areas of application for the boundary conditions**

On the right side of Figure 4-21 are the highlighted regions of application for the mechanical loads, which are restricted to the surface of the roll barrel. The precise area of effect of each of the loads is defined in the user subroutines which apply the load profile. This is much easier to accomplish mathematically than by manipulating the part geometry to isolate a suitable set of faces at the roll surface.

The temperature profile and residual stresses are both applied as “predefined fields”, meaning that they are calculated ahead of time and need to be stated for every point in the model. The process of calculating the temperature profile was covered in the previous chapter. The derivation of the residual stresses will be explained section 4.3.3. The process of applying the temperature and residual stress distributions is described in sections 4.3.1 and 4.3.3 respectively.

### 4.2.2. Surface Model

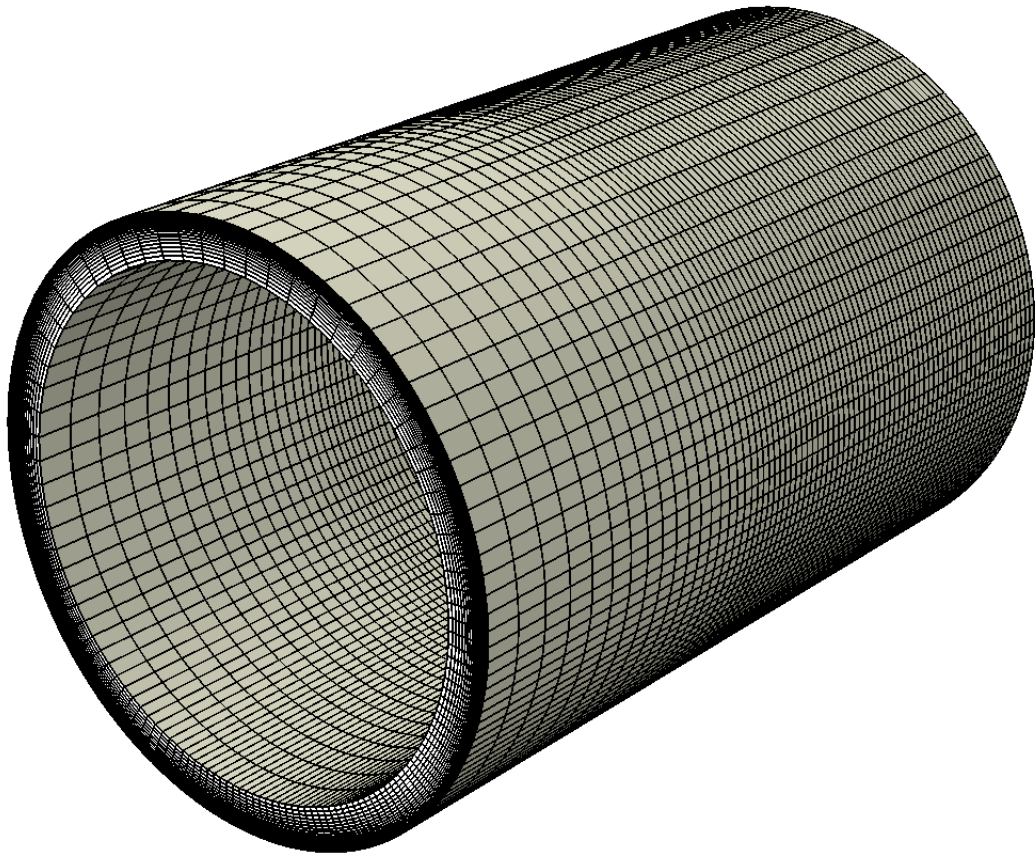


Figure 4-22: A surface model

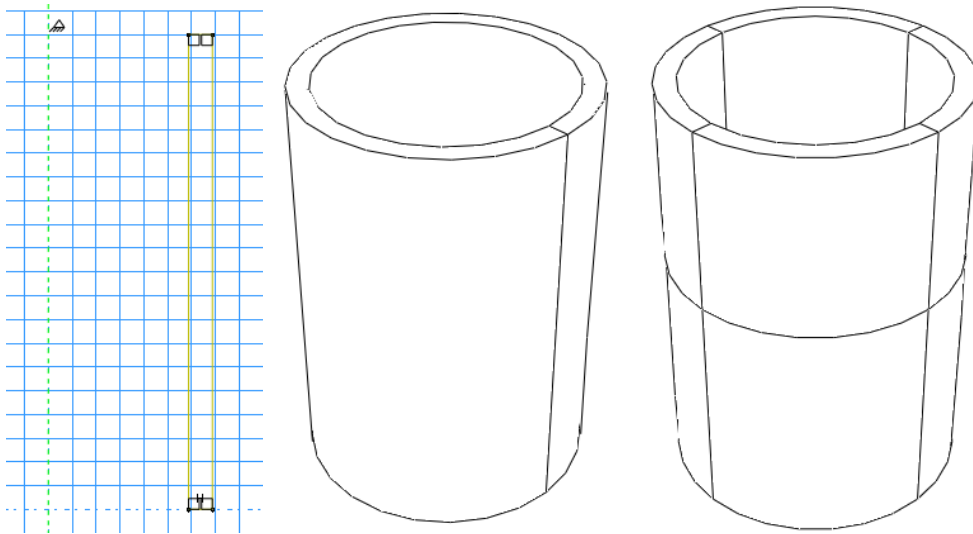
Figure 4-22 shows an example surface model. Because the sub-modelling process relies on deformation fields rather than specific nodal values the number of elements at the inner surface of the surface model does not need to match the number of elements at the equivalent position in the global model. This allows the inner surface resolution to match the outer surface resolution, resulting in a very well shaped and well structured mesh.

### *Geometry*

---

Figure 4-23 shows the three stages required to produce the surface model geometry. Following the same pattern as the global model this involves drawing the outline, rotating the outline to form the three dimensional part and partitioning the part to create meshing control surfaces. The process is greatly simplified by not having to incorporate the material below the material interface. The one radial-circumferential partition allows the area inside the strip width to be isolated from the area outside the strip width. The four axial-radial partitions

break up the annular top face to allow the meshing tools to function correctly, as well as providing edges to vary mesh density.



**Figure 4-23: Drawing the surface model outline (left), rotation to create a solid part (middle) and partitioning to aid meshing (right)**

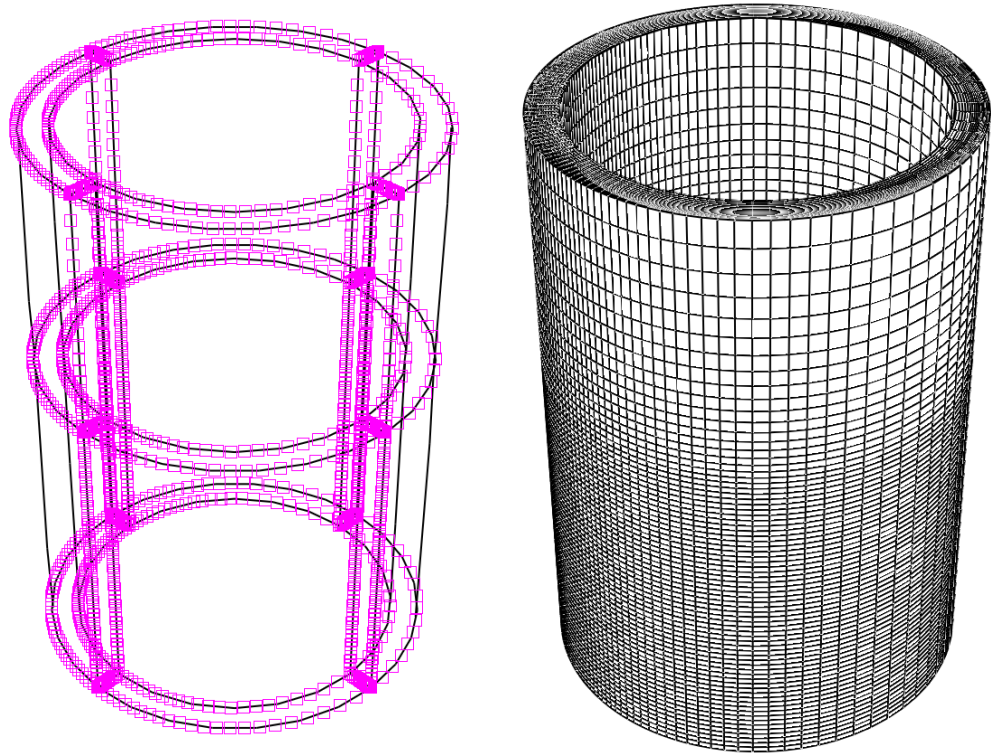
### *Seeding & Meshing*

---

Liberated from the core material, the surface model mesh density can be varied to a much greater extent without compromising on element squareness. Figure 4-24 illustrates the seeding constraints and resulting part mesh. The left side of the model as illustrated in Figure 4-24 is in contact with the strip and is therefore given the greater proportion of the mesh density. This concentrates the resolution where the temperature fluctuations and contact loads are at their most extreme. The meshing compromise reduces the resolution under the backup roll contact patch, which can be justified by recalling that basic contact mechanics should describe the interaction of those two bodies well enough that this model need not spend unnecessary resolution on the issue.

In the axial direction the resolution is considerably finer within the range of the strip width than towards the outer edge of the roll. This concentrates resolution where the roll is loaded and reduces resolution where there are few external stimuli. Both the previously mentioned changes in mesh density are controlled using sections of one-directional bias, gradually reducing the density to ease the transition between the different mesh areas.

In the radial direction the mesh spacing is very small, providing a high resolution to capture the response to the large changes in temperature over very small changes in depth. The radial seeding is also biased, keeping the resolution higher near to the surface.



**Figure 4-24: Seeding constraints (left) and fully meshed part (right)**

### *Boundary Conditions*

---

Figure 4-25 shows the three areas of application used to define the boundary conditions of the surface model. The left image shows the surface used to define the symmetry condition, giving the location of the work roll centre plane. The middle image highlights the surfaces used for the sub-model boundary condition, those surfaces which would be inside the global model and must therefore take their deflections from the global model results. The right image indicates the mechanically loaded area. The precise area of loading will be applied mathematically through the user subroutines, as was the case in the global model. In fact, the same user subroutine can be used for the purpose, as the surface of the surface model is technically the same as the surface of the global model and hence mathematically identical, even if resolved to a finer degree.

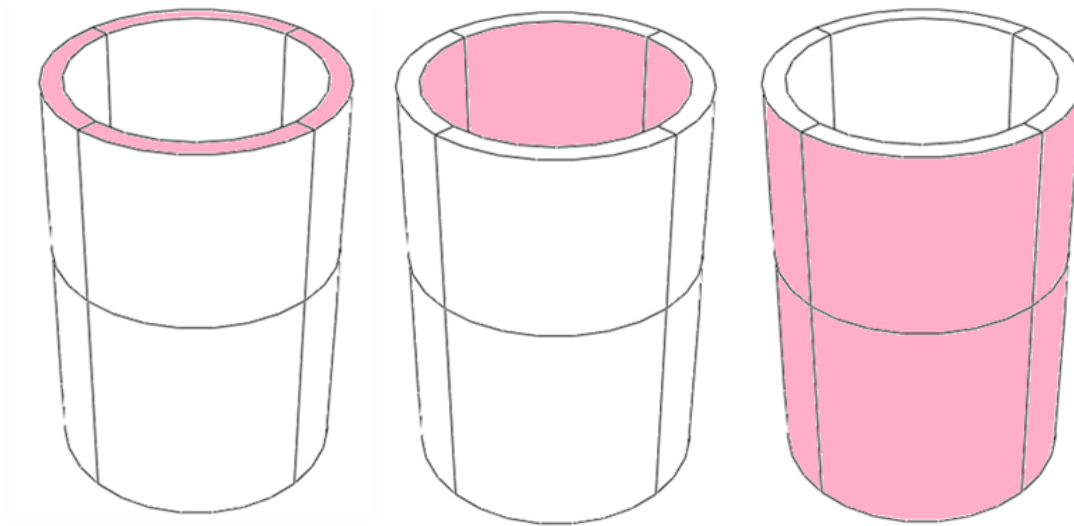


Figure 4-25: Areas of application of the symmetry (left), sub-model (middle) and mechanical (right) boundary conditions

### 4.2.3. Bite Model

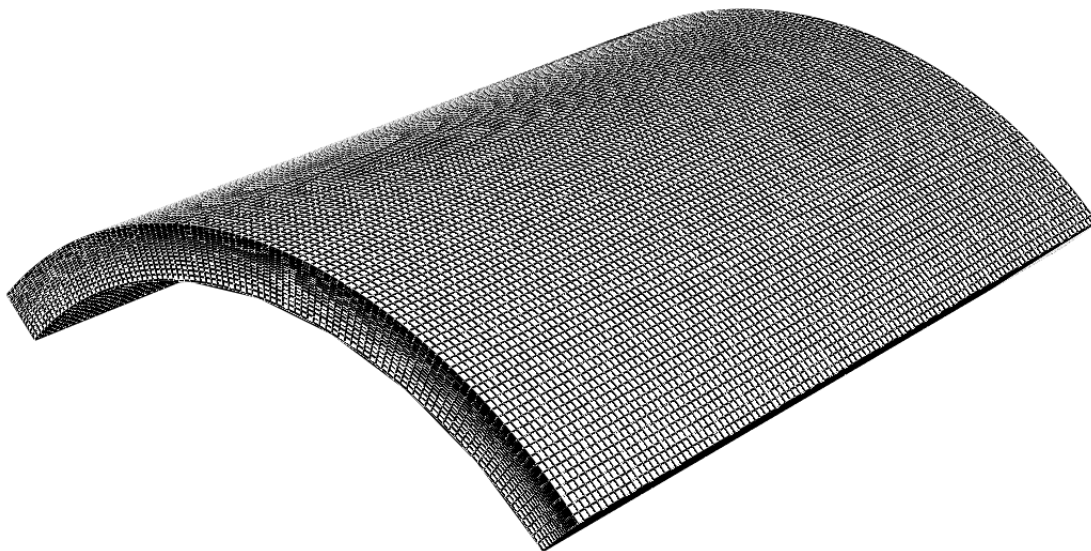
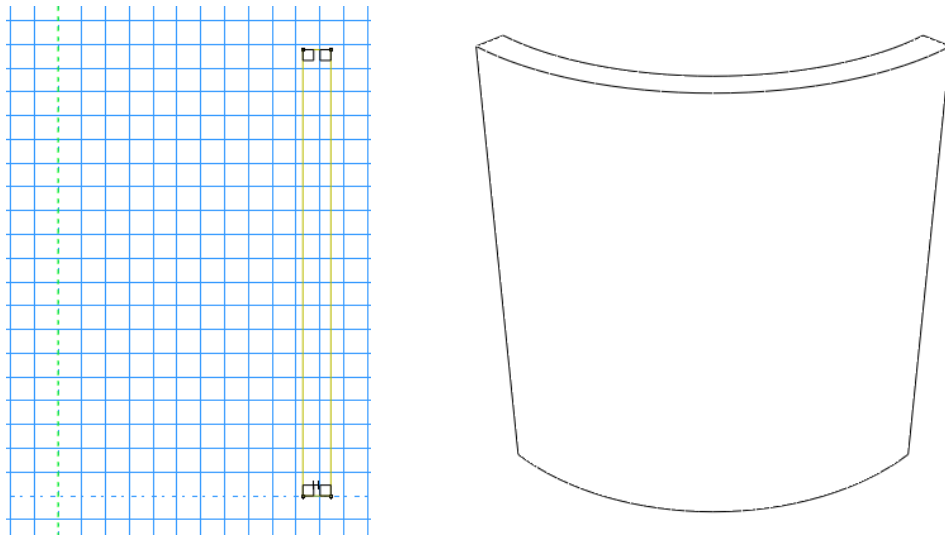


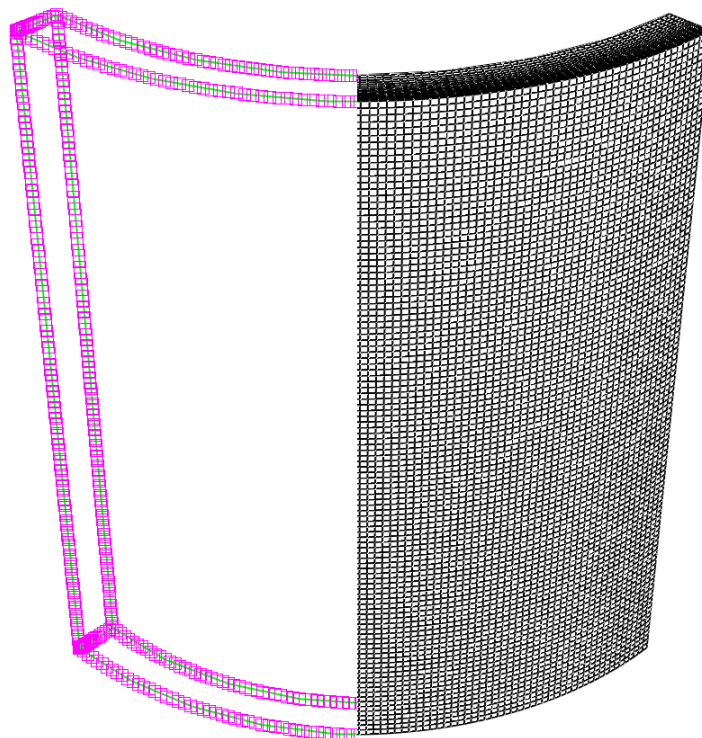
Figure 4-26: A bite model

Figure 4-26 shows a bite model, the highest resolution three-dimensional model, used to determine behaviour in and around the roll bite. Mesh density is relatively uniform throughout the model to capture fine detail from a variety of locations, whether in the roll bite, just before the roll bite, along the edge of the contact area, etc. The model is reduced radially to the heat penetration depth as calculated in the thermal models<sup>3</sup>, circumferentially to a 90 degree arc with the roll bite in the middle and axially to be slightly (10cm) greater than the strip width.



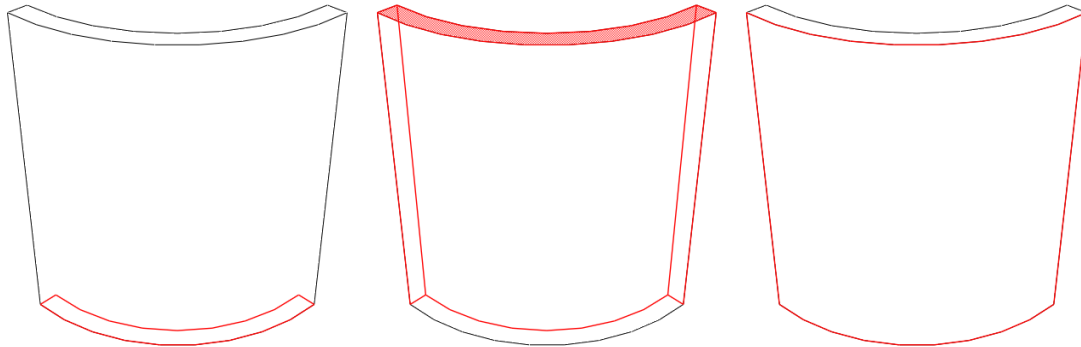
**Figure 4-27: Drawing the bite model outline (left) and rotation to create a solid part (right)**

Figure 4-27 shows the geometry creation process.



**Figure 4-28: Bite model with seeding illustrated on the left and meshing on the right**

Figure 4-28 gives a combined image showing the seeding and resultant mesh of the bite model. As previously mentioned the density is kept relatively constant except for a slight bias in the radial direction keeping density high at the surface. The resulting mesh is very fine and very regular.



**Figure 4-29: Areas of application of the symmetry (left), sub-model (middle) and mechanical (right) boundary conditions**

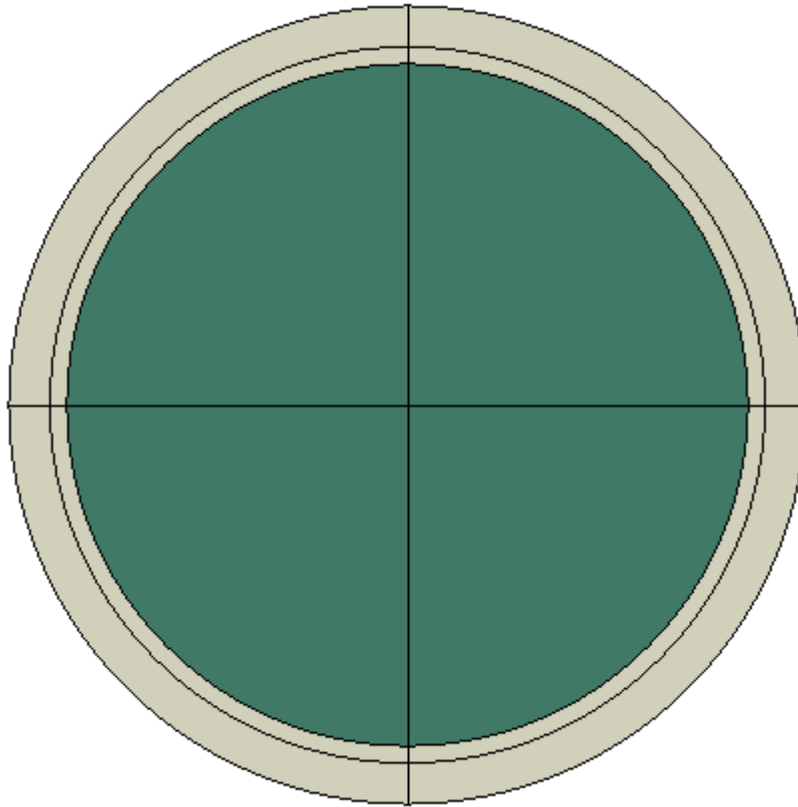
Figure 4-29 shows the area of application for the boundary conditions of the bite model. Because the bite model is reduced both radially, axially and circumferentially then sub-model boundary conditions are required on radial, axial and circumferentially oriented faces. The only faces not affected are the symmetry face, which is subject to its own condition, and the surface face, which still represents one of the work roll's external faces.

#### 4.2.4. Verification Model

The verification model is a special case, being free of the requirement of the other models to accurately reflect the work roll stress condition. The purpose of the verification model is to test the sensitivity of the results to the resolution level, and thus establish the meshing requirements that lead to mesh independent stress evaluations. Naturally this is a compromise situation, the nearer the verification model stays to reality the more directly applicable it will be to the other models, and hence the better verification it will provide.

The reduction from three dimensions to two is a natural step, releasing the vast majority of the consumed resources for a single simplification. The two dimensions chosen are the radial and circumferential dimensions, because the temperature profile changes most severely in these directions. By basing the model at the centre of the roll (the location of the symmetry plane in reduced models) the stress state of the work roll should be emulated to the greatest possible degree, as in most cases axial influences will balance at this point.



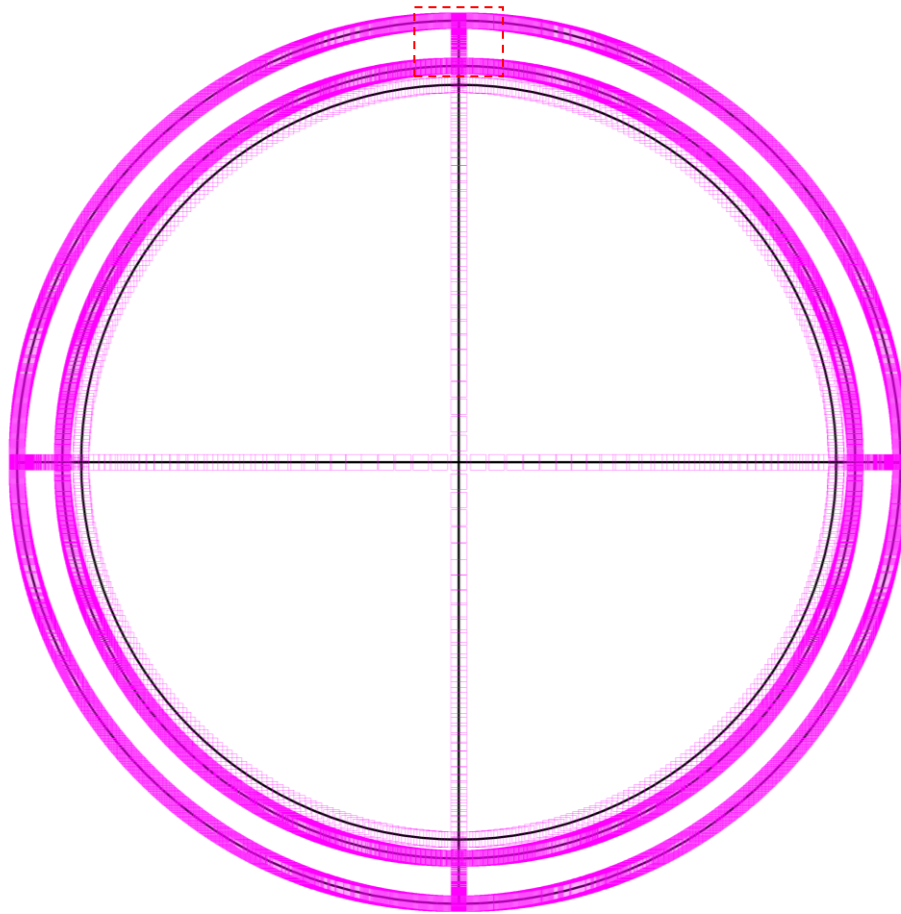


**Figure 4-30: Part geometry and material definition**

Figure 4-30 shows the geometry of the verification model and the location of the material interface. While the principles behind the choice of partitions in the verification model are similar to those used in the cross-section of the global model, there are some notable differences.

First, the control partition is outside the material interface rather than inside. This is because the resolution of this model will be so much finer, a regular zone with equal seeding on opposite faces can be applied within a short range of the surface. The smaller gap to the second partition can then be used to step down the mesh density in a controlled manner.

Second, there is no third partition. This is not necessary in this model as without the axial dimension there is no need to model the journals. The meshing tools are perfectly capable of meshing this area with a very gradually coarsening mesh, so there is no need to add any artificial complications.



**Figure 4-31: Seeding of the verification model (dashed area shown in Figure 4-32)**

Figure 4-31 gives an overview of an example seeding of the verification model. While the exact scale of this seeding is impossible to discern from the image, a significant increase in seed number over the three-dimensional models should be readily apparent. There are actually 3072 seeds around the circumference of this model, compared to the equivalent of 512 in the bite model. The use of reduced dimensionality along with the coarsening of the mesh near the centre of the roll therefore allows the kind of resolutions which can conclusively prove resolution independence. Figure 4-32 gives an illustration of the density of the mesh generated from the seeding shown in Figure 4-31.

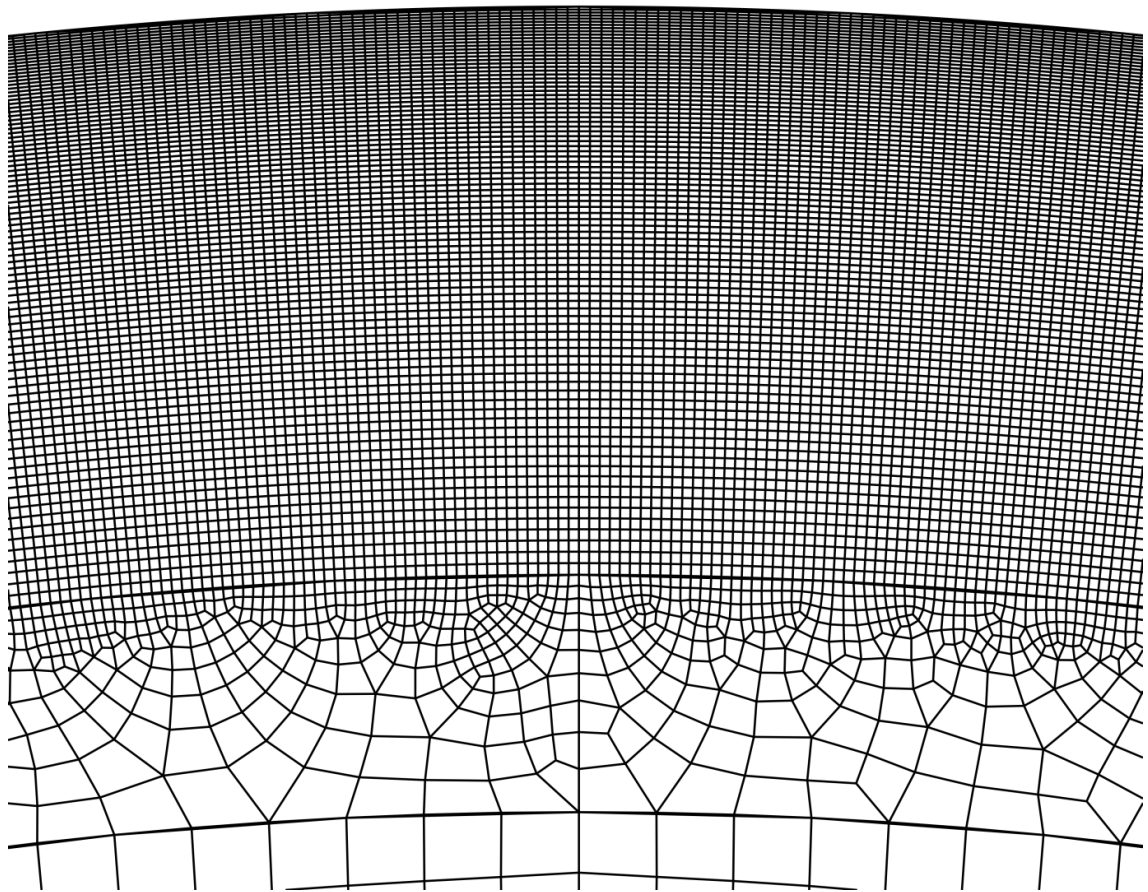


Figure 4-32: Zoomed in view of the verification model mesh for the region identified in Figure 4-31.

### *Boundary Conditions*

---

The boundary conditions for this model are much simpler than those for the three dimensional models. Firstly the centre point is restrained in the x and y directions to provide a known coordinate for the model, as shown in Figure 4-33 Secondly the bottom-most point in the model is restrained from moving in the x direction, preventing the model from rotating freely.

Both of these restrains act to prevent free-body motion and exert no actual force, therefore have no impact on the stress distribution. The temperature distribution is applied through a user subroutine and will be discussed in section 4.3.1. The choice of plane strain or plane stress assumptions for the behaviour in the third dimension is considered in Chapter 5 (Page 213).

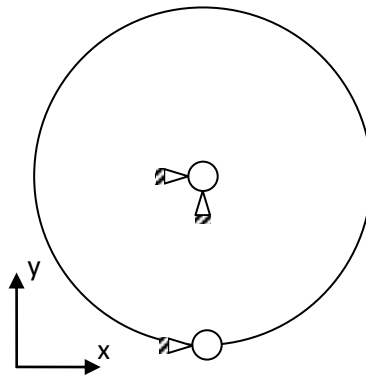


Figure 4-33: Restraint conditions on the 2D verification model

### 4.3. Boundary Conditions

---

#### 4.3.1. Thermal Loading

In the previous section the various work roll models and sub-models were assigned their material properties, so ABAQUS is fully equipped to transform a temperature profile into a resultant stress profile based on differential material expansion. All that remains is to introduce the user-subroutine that can import the temperature values from the thermal models. Conveniently the subroutine and associated C# functions developed in the White model<sup>3</sup> are still applicable and can be transplanted almost entirely intact. The following description is included for completeness, the full code is explained in Appendix C.

#### *Data Files*

---

The thermal models are written in the C# programming language, while the user subroutines are written in FORTRAN. The problem of passing information between the two programs could possibly be solved by direct interaction but it is far simpler to work with data files. This also functions as a further implementation of the concept that any stage of the analysis process should be performable in isolation and without relying on the chain of analyses beforehand being unbroken. Any necessary data can be saved to file and the process suspended for an indeterminate length of time, without sacrificing the capability to perform further analyses. Furthermore, both C# and FORTRAN work well with plain text files (.txt), allowing the use of straightforward writing and reading algorithms and direct inspection and editing by knowledgeable users.

Before the temperature data is considered there are certain variables which must be transferred in order for the user subroutine to be able to define where in the temperature data any given point is located. They are;

- The total length of the work roll and the barrel radius – These give the outer dimensions of the work roll for comparison against the outer dimensions of the temperature data files. This common reference is needed to ascertain where a set of coordinates lies within the work roll.
- Node counts in all directions and in both models – Required so that the FORTRAN code knows how many values to read from the file. FORTRAN does not have the kind of features available to higher languages, such as indications that the end of a file has been reached or dynamically adjustable array sizes, so these eventualities must be prepared for ahead of time.
- Mesh spacing for the core and shell models – These quantities are required to translate the coordinates of any point into a positional index within the temperature results.
- Control variable dictating whether interpolation should be used – this is an integer value which carries the value “1” if interpolation should be used, or “0” if the nearest temperature value should be used. This will always be “1” but control has been left parameterized in case of unforeseen developments.
- Initial temperature – In the White Model the temperatures are imported as differences from the initial temperature, i.e. a final temperature of 23 degrees in a model with 20 degrees as its initial temperature would be imported as 3 degrees. It was therefore necessary to pass the initial temperature to the user subroutine so that the true temperature could be applied to the model.

Since there are two temperature models with separate scope and dimensionality it is best to use two files to store the temperature data. The variables mentioned above are each added into one of the two files based on whether the variables in question are stored in the core model C# object or the shell model C# object.

### *Core Model Data*

---

The first and simplest file is the core model data file, as the core model data is two dimensional and so the quickest and easiest to write. The first line of the file contains the

variable values of the work roll length, barrel radius, axial node count, radial node count, core axial mesh spacing, core radial mesh spacing, shell radial mesh spacing, interpolation choice and initial temperature, each separated by a space (" ") character. This format is not arbitrarily chosen, if FORTRAN reads a line from a file and finds a series of values separated by spaces there are set conventions for updating a series of variables to reflect those values.

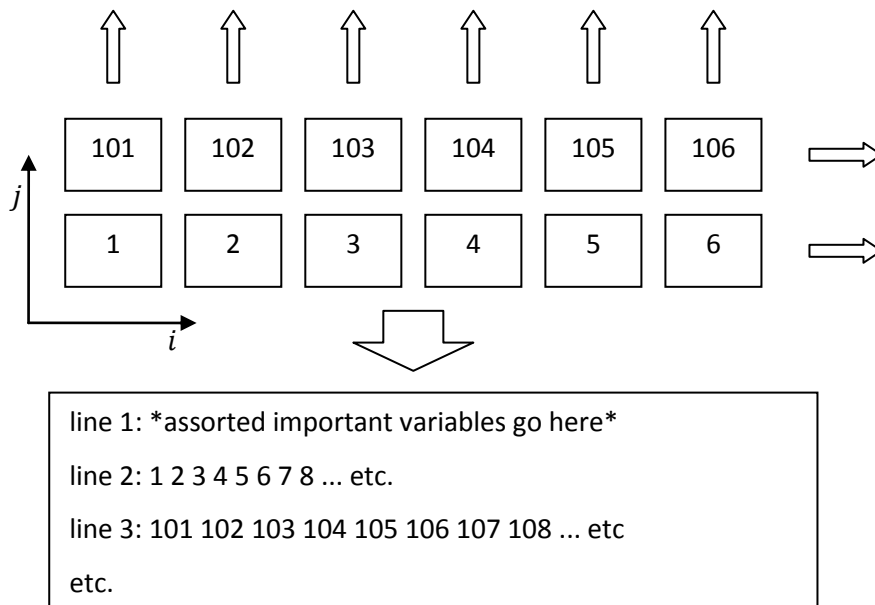


Figure 4-34: Example conversion from a two dimensional model to a data file

The structure of the data file is a simple tabular format, as illustrated in Figure 4-34. The first line of the temperature data consists of the temperature value of the node at index  $(i, j) = (1,1)$ , followed by a space, followed by the temperature value of the node at index  $(2,1)$ , and so on until the maximum value of  $i$  is reached. Then the next line starts with the temperature value of the node at index  $(1,2)$  and follows the same pattern. The file is complete once the maximum value of  $i$  is reached for the maximum value of  $j$ .

### Shell Model Data

The first line of the shell model temperature data file contains the values of the shell model radial, axial and circumferential node counts and the circumferential mesh spacing, in the same format as the first line in the core model temperature data file.

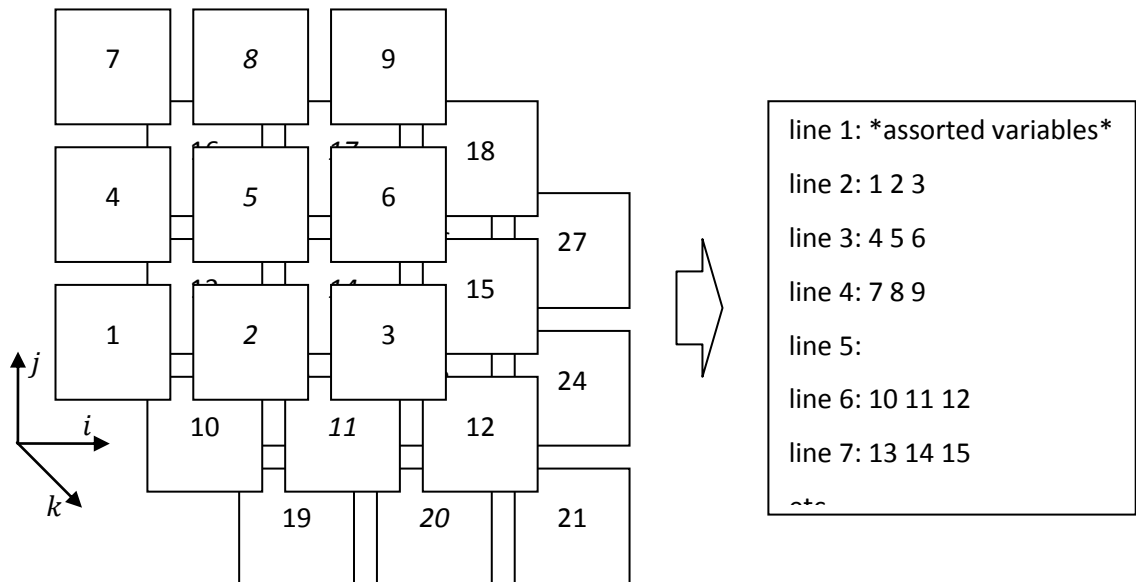


Figure 4-35: Example conversion from a three dimensional model to a data file

The structure of the three dimensional data file is one extra logical step beyond the two dimensional data file. Considering Figure 4-35, the first three lines of the temperature data follow the pattern established in the previous file, incrementing  $i$  until the maximum value of  $i$  is reached, then returning  $i$  to 1, incrementing  $j$  and repeating. However, in this case when the maximum value of  $j$  is reached an empty line is inserted,  $k$  is incremented and  $i$  and  $j$  are returned to 1, before the whole process starts again. The extra space is not strictly necessary from a programming perspective, since mathematics will be used to count rows and specify when a new  $k$ -index is required. However, from a human perspective the added space lends a lot more clarity to the data file and adds very little complication to the process of reading or writing.

### User Subroutine

ABAQUS applies a temperature field to a finite element model one node at a time, passing through all the nodes in the model until every node has been assigned a temperature. This means that the UTEMP subroutine will be called multiple times to supply individual temperatures at specific points in space. In order to keep the processing cost of loading the data files to a minimum, the variable values used in the subroutine are retained between calls. A temperature is returned to ABAQUS by giving the  $TEMP$  variable a value, when the subroutine ends ABAQUS will use whatever value  $TEMP$  is currently holding. The process flow describing this subroutine is given in Figure 4-36.

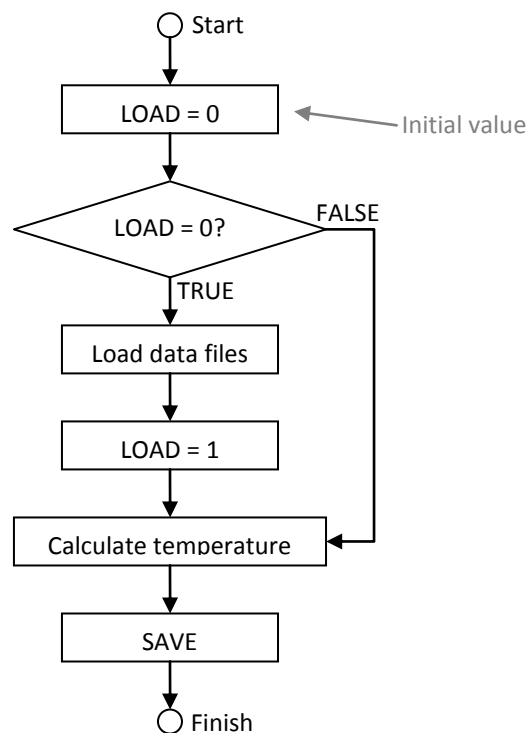


Figure 4-36: UTEMP Process overview

The initial value for *LOAD* is only set the first time the subroutine is called, initial values are not applied if saved values for variables already exist.

### Load the Data

---

The process of loading the data files has been condensed down to one step, where in reality there are two separate actions to load the two data files. Each action begins by reading the line of variable values at the top of the file directly into a series of internal variables, then moves on to consider the temperature data set.

Fortran reads text files a line at a time and can employ an embedded loop mechanism to read the values in a line into an array. The process is essentially the reverse of the process followed in Figure 4-34 and Figure 4-35 to write the data files. The embedded loop counts from a starting value to an ending value and at each step copies a value from the line into the array, using the current count as the coordinate index for the array. Depending on whether the file is two dimensional or three dimensional, one or two external loops count between the ranges of the other dimensions, as shown in Figure 4-37.

The *I* variable counts through each variable in a line of the data file, the *J* variable counts through each line on a two dimensional plane and the *K* variable counts through each two



dimensional slice of a three dimensional volume. An extra “read line” statement in the  $K$  loop bypasses the empty space between planes of data in the three dimensional data file.

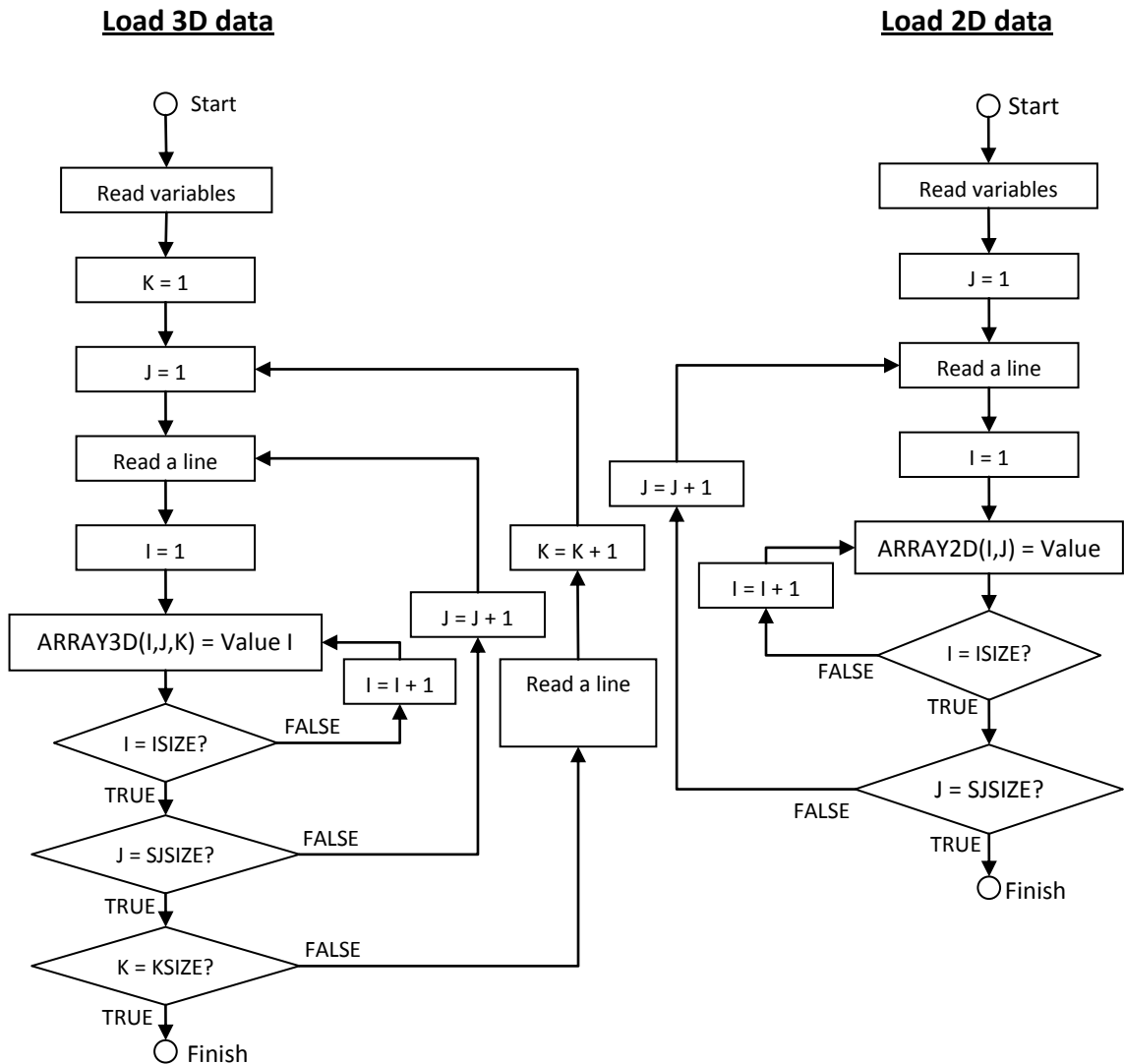
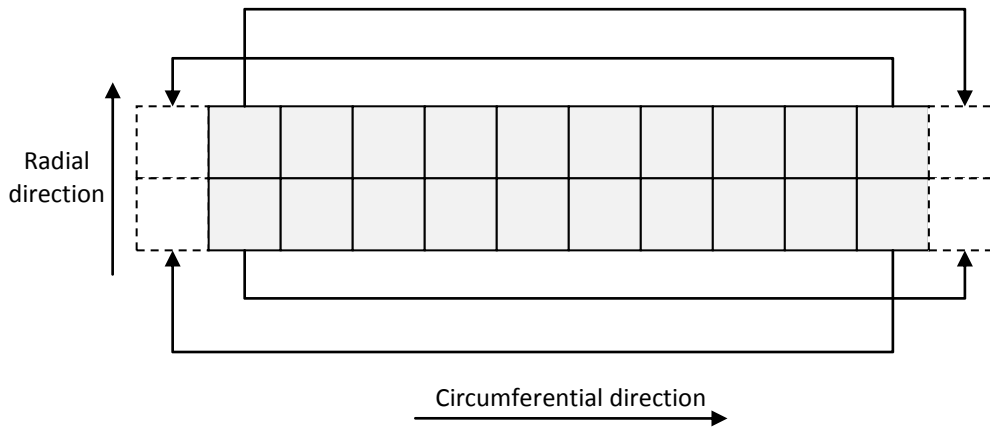


Figure 4-37: Flow diagrams for loading 3D and 2D data files

Following the two processes illustrated above the two dimensional and three dimensional data are both loaded. However, the three dimensional data is stored in an essentially shapeless rectangular block form but represents the surface material from a cylinder. An interpolation routine will be introduced later, which can provide a temperature for any point within the shell volume based on the temperature values surrounding it. However, if the point in question is just beyond the left edge of the example data set in Figure 4-38, a legitimate place for the point to be, there will be no second value to interpolate to. Rather than introduce logic to catch this eventuality at the calculation stage, the data set is extended by one space at each circumferential end and the value from the opposite end is copied across.



**Figure 4-38: Wrap around in the shell temperature data**

### *Calculate Temperature*

---

The “calculate temperature” step from Figure 4-36 can be expanded into the simplified flow diagram in Figure 4-39. An important feature of this process is the conditional statement that RC, the radial coordinate, is greater than the shell depth. This controls whether the three dimensional temperature data is needed, bypassing the calculation if the point is outside the scope of the shell model. The 2D temperatures could be incorporated into the logical statement if at any point in the future processing time becomes a critical constraint.

Loading the coordinates involves copying the coordinate values from an array supplied by ABAQUS into individual variables, making them easier to work with. ABAQUS works in a rectangular reference frame so the coordinates need to be converted to the cylindrical polar reference frame used by the thermal models.

Since the axial direction is the same in both cylindrical polar and rectangular reference frames the axial coordinate can remain as it is. The radial and angular components are calculated as follows.

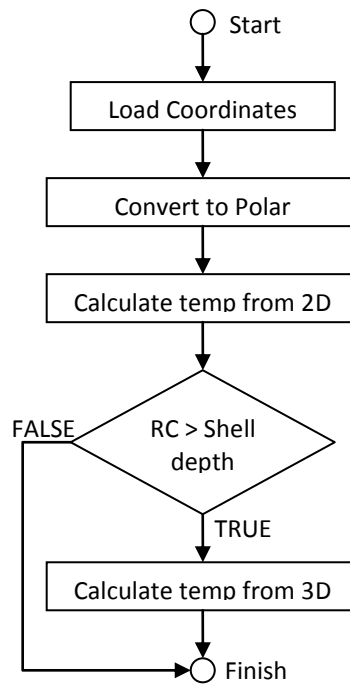


Figure 4-39: Simple flow diagram for calculating the temperatures

$$r = \sqrt{x^2 + y^2} \quad \text{Equation 4-1}$$

$$\theta = \cos^{-1} \frac{x}{r} + 1.34 \quad \text{Equation 4-2}$$

$$\theta = \cos^{-1} \frac{-x}{r} + \pi + 1.34 \quad \text{Equation 4-3}$$

Where Equation 4-2 should be used if  $y$  is greater than zero and Equation 4-3 should be used if  $y$  is less than zero. The arbitrary looking addition of 1.34 moves the roll bite (in the thermal models the bite entrance is at  $\theta = 0$  and extends in the positive  $\theta$  direction) away from the FE model partitions required by the meshing process. The edge of a meshed volume is far more likely to contain what inconsistencies there may be in a mesh, so rotating the temperature field in this way keeps the area of interest away from the areas of risk.

### *Two dimensional interpolation*

---

Figure 4-40 gives an example situation of a point in space which does not exactly line up with any point in the temperature data. The location of this point can be described as being within an area fully described by the location of four corner points;  $Y_1, Y_2, Y_3$  and  $Y_4$ . Subtracting the coordinate of the point  $Y_1$  from  $(z, r)$  gives the values of  $a$  and  $c$ , which in turn allows the calculation of  $b$  and  $d$  as the mesh spacing is known.

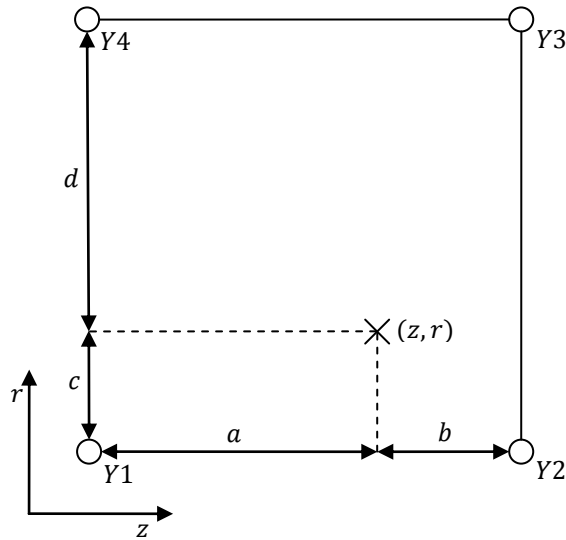


Figure 4-40: Interpolation between four points in two-dimensional space

The distance of the point between Y1 and Y2 can be made into a ratio as follows.

$$T = \frac{a}{a + b} \quad \text{Equation 4-4}$$

And similarly for Y1 and Y4.

$$U = \frac{c}{c + d} \quad \text{Equation 4-5}$$

With these ratios known the standard linear interpolation is given by;

$$TEMP = (1 - T)(1 - U)Y1 + T(1 - U)Y2 + TUY3 + (1 - T)UY4 \quad \text{Equation 4-6}$$

The three dimensional interpolation solution is slightly more complicated but follows the same approach. The naming convention is given in Figure 4-41. The first steps are much the same, subtracting the coordinate of Y1 from  $(r, z, \theta)$  gives the values of  $a, c$  and  $e$ , allowing the calculation of  $b, d$  and  $f$  respectively. Equation 4-4 and Equation 4-5 can still be used to find the  $T$  and  $U$  ratios, while one extra equation provides the ratio for the third direction.

$$V = \frac{e}{e + f} \quad \text{Equation 4-7}$$

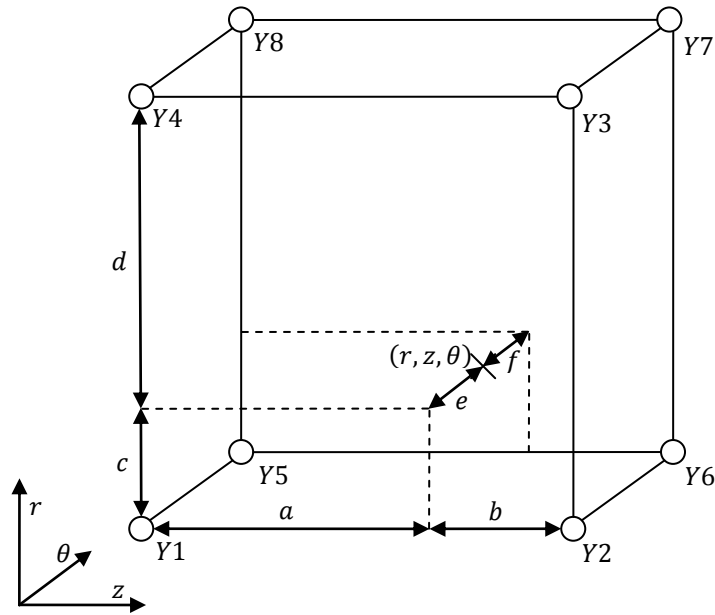


Figure 4-41: Interpolation between eight points in three-dimensional space

A two dimensional interpolation in Equation 4-6 can be performed for both the Y1-Y2-Y3-Y4 plane and the Y5-Y6-Y7-Y8 plane, since the coordinate projected onto each plane only varies in the  $\theta$  direction.

$$P1 = (1 - T)(1 - U)Y1 + T(1 - U)Y2 + TUY3 + (1 - T)UY4 \quad \text{Equation 4-8}$$

$$P2 = (1 - T)(1 - U)Y5 + T(1 - U)Y6 + TUY7 + (1 - T)UY8 \quad \text{Equation 4-9}$$

The final temperature can be obtained through the linear interpolation between these two points.

$$TEMP = (1 - V)P1 + VP2 \quad \text{Equation 4-10}$$

### Validation Model Temperatures

There are numerous time saving alterations that could have been made to the treatment of the thermal subroutine for use with the validation model. Being two dimensional, limited to the radial-circumferential plane, the temperature data required is limited to a one-dimensional slice of the core model and a two dimensional section of the shell model data. However, given the limited run time of the full scale process these optimisation measures were ignored to free up development time.

The two dimensional interpolation scheme as it currently stands is suitable for use near the surface. Below the surface a simple one-dimensional interpolation scheme is required.

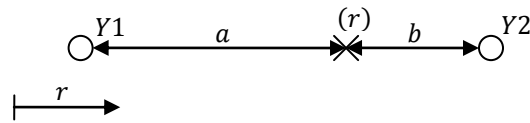


Figure 4-42: Interpolation between two points in one-dimensional space

Equation 4-4 can be used again to give the ratio of the distance between Y1 and Y2. Temp is then given by;

$$TEMP = (1 - T)Y1 + TY2 \quad \text{Equation 4-11}$$

### 4.3.2. Mechanical Loading

#### *Normal Loading*

---

To achieve the desired reduction in strip thickness in a rolling stand a large force is applied to the backup rolls. This leads in turn to contact pressures between the backup roll and the work roll and between the work roll and the strip. The load case is illustrated in Figure 4-43. The user subroutine DLOAD is used to apply these loads to the surface. This subroutine will only be called once per node, so both load conditions must be accounted for. Figure 4-44 gives a simplified view of the subroutine structure.

Technically the true case of the first conditional statement need not lead into the second conditional statement, as a surface point in the roll bite will never also be in the backup roll contact patch. However, this structure minimises the amount of nesting (loops or conditional statements within loops or conditional statements) keeping the code readable and is safe from all but the most self-evident of errors. The “Load Coordinates” and “Convert to Polar” steps are identical to those performed in the previous section so will not be described again.

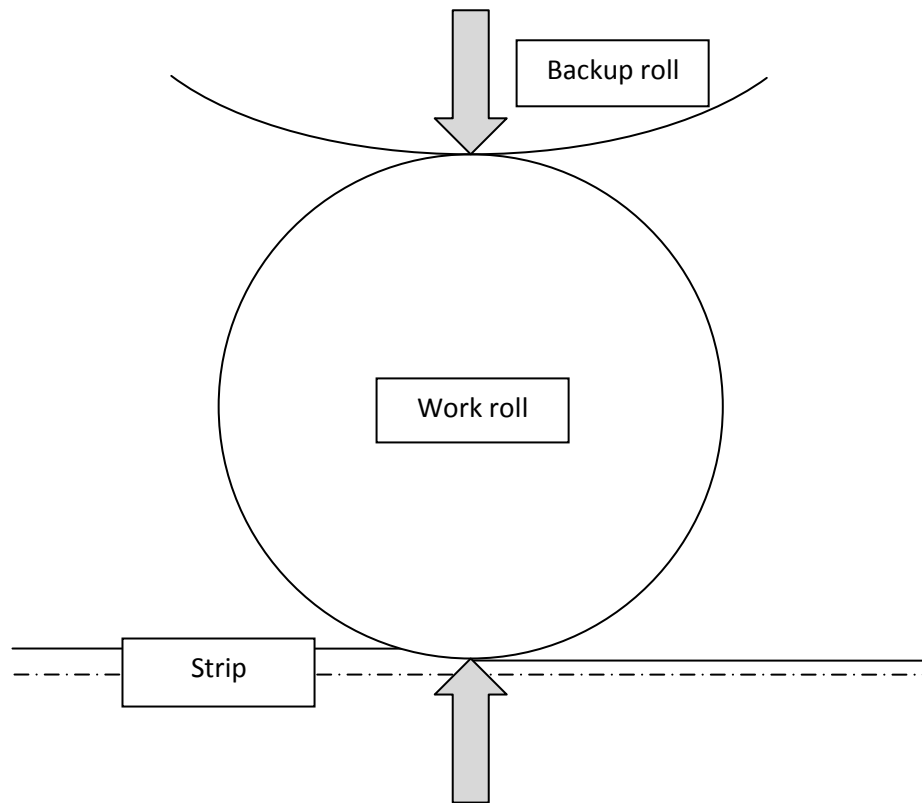


Figure 4-43: Work roll normal loading

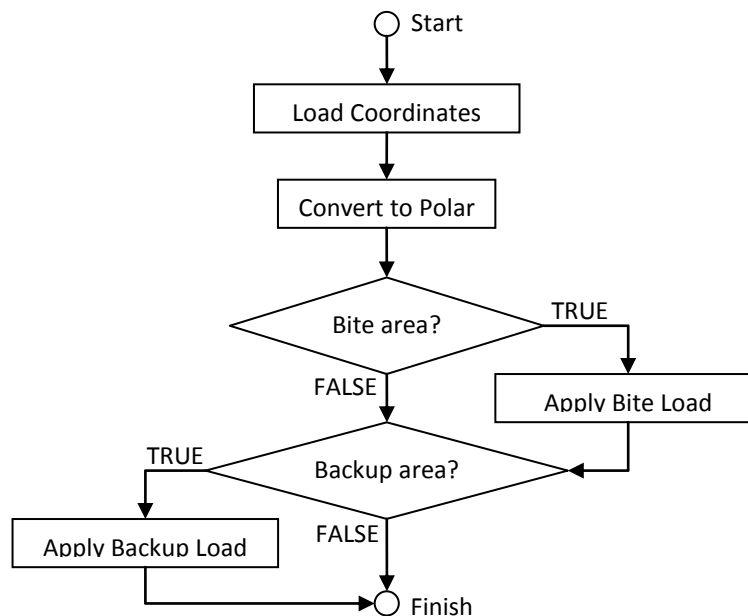


Figure 4-44: DLOAD Process overview

### Hertzian Contact

The contact between the backup roll and work roll is best represented as a Hertzian line contact, whose dimensions and peak load are given by Equation 4-12 and Equation 4-13.

$$a = \sqrt{\frac{8 R w'}{\pi E'}} \quad \text{Equation 4-12}$$

$$P_0 = \sqrt{\frac{E' w'}{2 \pi R}} \quad \text{Equation 4-13}$$

In these equations  $a$  is half the contact width,  $p_0$  is the maximum pressure and  $R$ ,  $w'$  and  $E'$  are given by;

$$R = \frac{R_1 R_2}{R_1 + R_2} \quad \text{Equation 4-14}$$

$$w' = \frac{w}{L} \quad \text{Equation 4-15}$$

$$E' = \frac{1}{\frac{1 - \nu^2}{E_1} + \frac{1 - \nu^2}{E_2}} \quad \text{Equation 4-16}$$

Where  $R_1$  and  $R_2$  are the work roll and backup roll radii,  $w$  is the total load,  $L$  is the length of the contact,  $E_1$  and  $E_2$  are the Young's Moduli of the work roll and backup roll and  $\nu$  is the Poisson's Ratio of both materials. Almost all of these details are available on the hot mill database. The exceptions are the material properties, which must also be provided by the user to create the material property elements applied to the stress models. After evaluating Equation 4-12 and Equation 4-13 the process in Figure 4-45 gives the instantaneous pressure for a given coordinate.



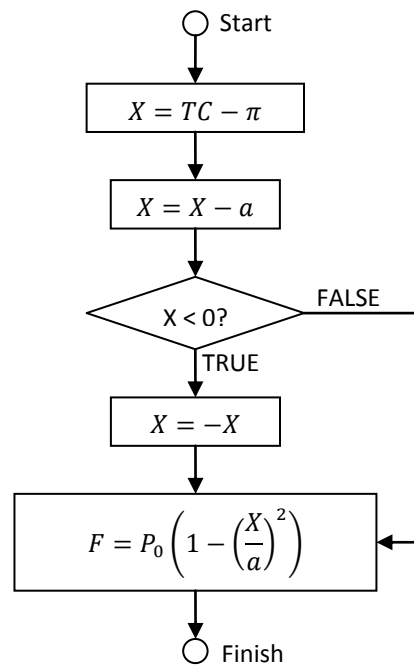


Figure 4-45: Process flow for finding the instantaneous pressure at the angular coordinate TC

In Figure 4-45  $X$  represents the distance from the contact centre. Following the “Load coordinates” and “convert to polar” steps of Figure 4-44 the angular (Theta) coordinate  $TC$  has been adjusted so that the entrance to the roll bite is at  $TC = 0$  and the rolling direction is positive. Because the contact sizes are small compared to the earlier models’ mesh sizes the backup contact size is set identical to the bite size for the global and surface models. This means that the entrance to the backup roll contact will be  $\pi$  radians away from the roll bite entrance, at  $TC = \pi$ .

The first part of the process therefore sets  $X$  equal to the distance of  $TC$  from the start of the backup contact. The second part subtracts half the contact width, making  $X$  equal to the distance of  $TC$  from the centre of the backup contact. The conditional statement changes  $X$  to positive if it is negative, essentially making  $X$  an absolute value. Finally the local instantaneous pressure is assigned to  $F$ , to be returned to ABAQUS, using the standard equation.

$$P = P_0 \left( 1 - \left( \frac{X}{a} \right)^2 \right)$$

Equation 4-17

### Roll Bite

With the size of the roll bite and the total force applied to the strip available on the hot mill database there is little more than the distribution of the normal pressure that needs to be

calculated. An initial study was carried out using a basic Hertzian distribution, with a Hertzian load and profile calculated then adapted to provide the correct total force when applied over the roll bite dimensions retrieved from the hot mill database. The impact of this load on the stresses at the surface was minimal, with the most influence being far below the surface material where the precise distribution of the load is less important than its overall magnitude. With this being the case the distribution has been left in a modified Hertzian form, so that development time could be focused on more influential aspects of the load case. The evidence leading to this decision is addressed in Chapter 5.

### Shear Loading

As the strip thickness changes through the roll bite the strip accelerates, while the work roll rotates at a constant speed. The difference in speed results in a shear force acting to oppose the slippage, i.e. away from the neutral point, where the strip and work roll are moving at the same speed. This scenario is illustrated in Figure 4-47. Hwang et al<sup>16</sup> give simulated results of the roll-strip interface for a non-slipping contact regime, results shown in Figure 4-46.

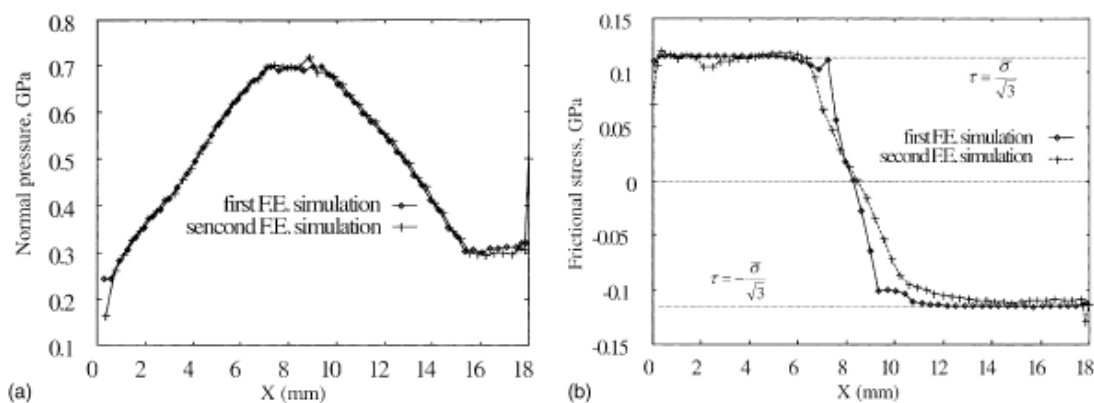


Figure 4-46: Normal pressure (left) and shear stress (right) in the roll bite

In reality the situation in the roll bite will be a mix of slipping and sticking-type behaviour, however this relationship sets a worst-case scenario for the roll bite. After comparing results using a range of worst-case scenarios, described in Chapter 5, it was found that the shear stresses made very little difference to the overall stress profile. Given the limited return on development time, the shear loading was left in this simplified state.

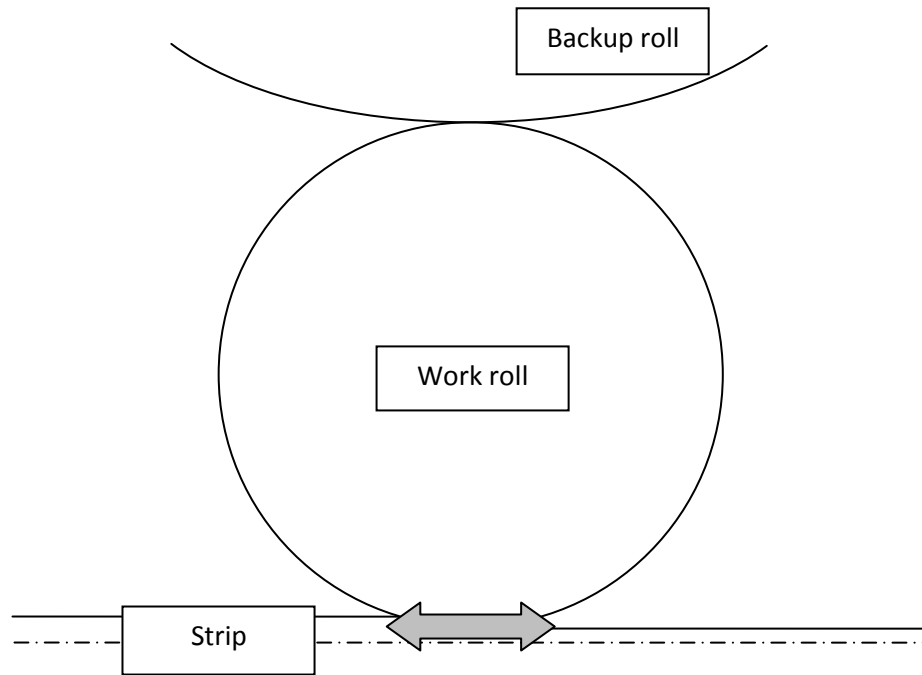


Figure 4-47: Work roll shear loading

### Work Roll Bending

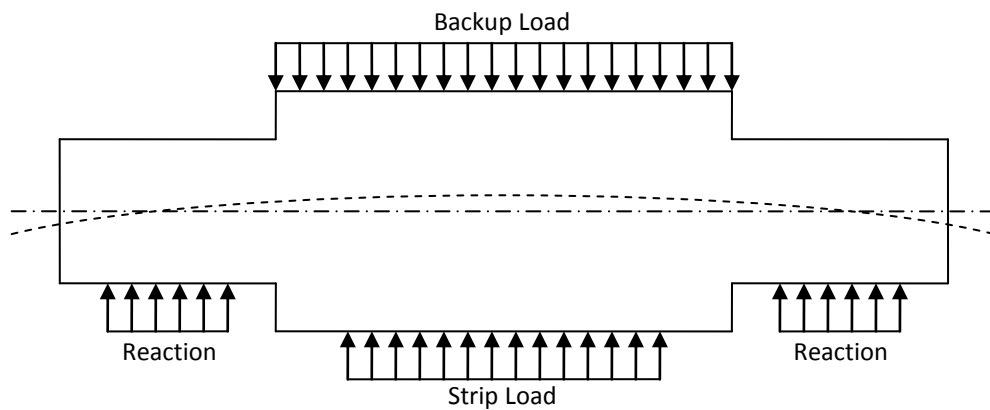


Figure 4-48: Work roll bending

Under normal rolling conditions a great deal of pressure is being exerted on the middle of the work roll as it presses down on the strip. As shown in Figure 4-48, the loading applied by contact with the backup roll overlaps the reaction force. As a result the centre line of the work roll tends to be deformed away from straight (the unevenly dashed line in Figure 4-48) to a curved profile (the evenly dashed line in Figure 4-48). The Work Roll Bending system takes control of this deformation by adjusting the magnitude of the reaction forces at the two journals either side of the work roll barrel in Figure 4-48.

Applying Work Roll Bending to the finite element model is complicated by the fact that the journals are the most convenient location for the restraints against free body rotation. This makes applying forces to the journals illegal, since deflections are already prescribed (only deflections or forces can be applied to any one node, not both, since any value of force will result in a specific deflection, and vice versa). Instead, the effect of Work Roll Bending can be applied by fully restraining deflection in the journals and increasing the force applied through the "backup load" without changing the force applied through the "strip load". The journals are restrained from moving, so the "reaction" forces ABAQUS applies to maintain their position will automatically scale to account for any offset between the "backup load" and the "strip load", thereby providing the Work Roll Bending.

The theory defined above was implemented using recorded loading data from the hot mill database, and was shown to be insignificant, as discussed in section 5.3.5 (Page 250). For completeness, work roll bending is still applied in the current models.

#### 4.3.3. Residual Loading

The problem of simulating the effects of residual stresses is compounded by the scarcity of experimental data and the complexity of the processes leading to the residual stress state. Modelling the manufacturing process would involve prohibitively complex materials modelling during the casting and heat treatment processes and plastic deformation during the grinding and surface preparation phases. However, the experimental data is not complete enough to provide complete results either.

The most applicable data available was published in an article in 2009 by Hosseinzadeh<sup>32</sup>. A part of the data acquisition of the project involved the deep hole drilling technique being used to measure a residual stress component in a work roll. Unfortunately the roll in question was hollow rather than solid. Despite the lack of exact similarity, the data presented in Hosseinzadeh represents the best estimate of the residual stress state of a work roll currently available. It is used in this work as the basis for a comparison between stress states of the work roll with and without an estimated residual stress field, to gauge the likely scale of effect of the real residual stress field compared to other effects.

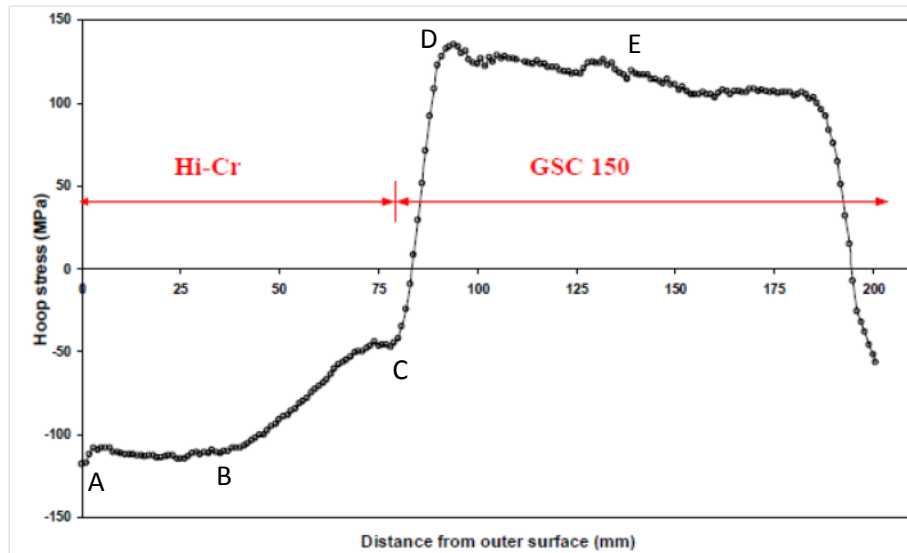


Figure 4-49: Deep hole drilling stress results from a work roll

Figure 4-49 shows the results of the deep hole drilling test. The roll was new so the recorded stresses are representative of a work roll fresh from the manufacturers. This project was the first to provide such detailed, direct measurements of residual stress at such a range of depths.

The supervisor of the PhD student who performed this work recommended a technique which they had found successful in other fields of work. The technique involves setting a limited range of recorded values in the initial stress field of an ABAQUS model and allowing ABAQUS to adjust the stress field to achieve an internal equilibrium. The known values would be changed along with all the others, so the process must be repeated after resetting the known values until those values stop changing.

For the current study the residual hoop stress profile between the points annotated as A,B,C,D and E on Figure 4-49 was taken as a guide to a supposed residual stress distribution in the solid workroll. The profile A-B-C was applied to the material shell portion of the work roll radius and this radial scaling was also used to apply profile C-D-E so that the hoop stress for the outer portion of the roll was specified based on Figure 4-49.

### Data Files

In order to follow the ABAQUS balancing route the residual data needs to be transferred to ABAQUS. The easiest way to do this is to follow the example of the thermal loading and use data files. The user subroutine which applies stresses to nodes is SIGINI, which is called for every node in the model in turn. Like the other subroutines the only important information

ABAQUS provides is the coordinate of the current node. Unlike the other subroutines SIGINI requires an array of six parameters to be returned in order to construct a complete stress tensor.

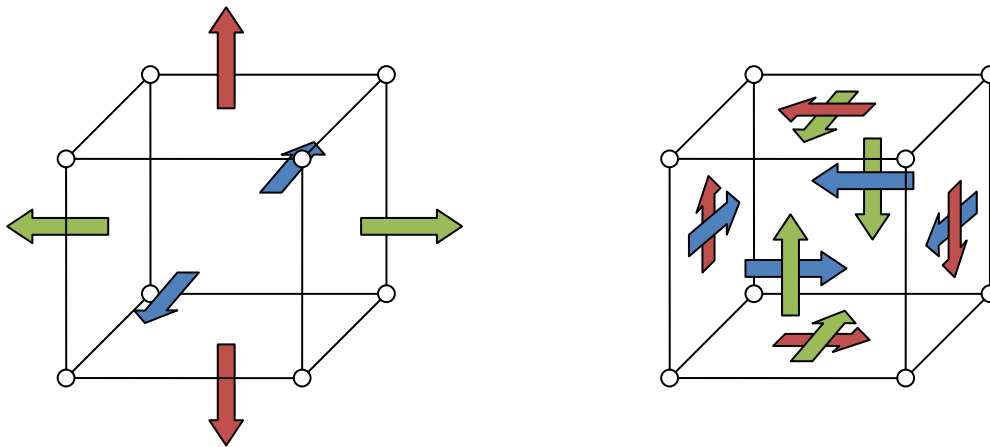


Figure 4-50: Components of a stress tensor

Figure 4-50 shows all the components of a stress tensor. The left of the figure illustrates the normal stresses, the loads which define whether and to what degree the material is in tension or compression. The right of the figures gives the shear stresses, those loads which act perpendicular to the normal stresses and parallel to the face on which they are imposed. To meet the requirement for equilibrium, the shear stresses on opposite faces and acting along the same axis must be equal, to prevent translation of the element. This generates a moment around the element centre, so the shear stress pair acting at ninety degrees (which must also be equal and opposite) must act in the opposing direction by the same magnitude to counteract the moment created by the other pair. The colour coding of the stresses identifies those stresses whose magnitude must therefore be identical, meaning that three normal stresses and three shear stresses are required to completely describe the stress state of the element.

The data file will therefore need to provide the radial, circumferential (hoop) and axial stresses, followed by the radial-hoop, radial-axial and hoop-axial shear components (e.g. radial-hoop refers to the shear stress on the face subject to the radial normal stress, acting in the hoop direction). Since only one of these components will be known, the other components must be able to change. To allow for this a master-slave file system is applied.

<u>master</u>	<u>slave</u>
line 1: S11 S22 S33 S12 S13 S23 Rad	line 1: S11 S22 S33 S12 S13 S23 Rad
line 2: x -100 x x x x 0.35	line 2: 0 -100 0 0 0 0 0.35
line 3: x -100 x x x x 0.32	line 3: 0 -100 0 0 0 0 0.32
line 4: x -50 x x x x 0.3	line 4: 0 -50 0 0 0 0 0.3
line 5: x 120 x x x x 0.28	line 5: 0 120 0 0 0 0 0.28
line 6: x 100 x x x x 0.1	line 6: 0 100 0 0 0 0 0.1
line 7: x x x x x x 0	line 7: 0 0 0 0 0 0 0

**Figure 4-51: Example master and slave data files**

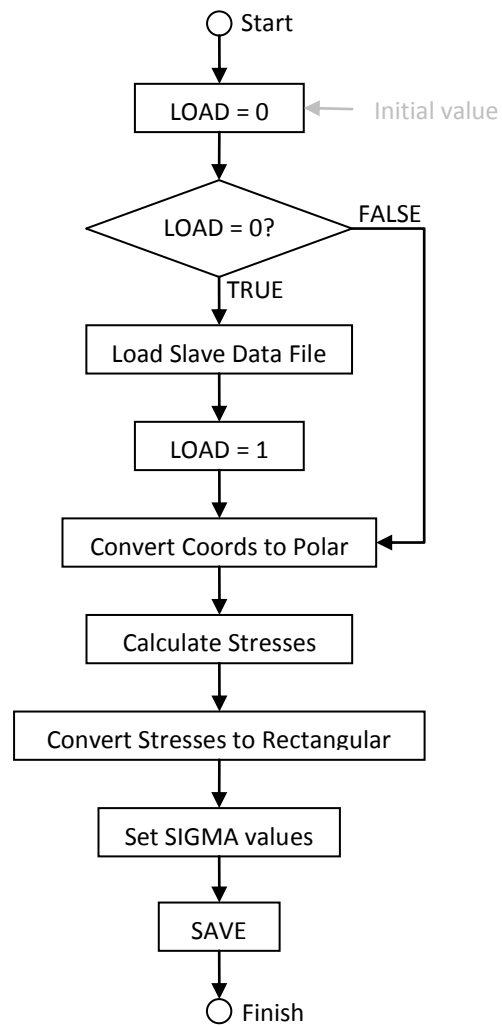
Figure 4-51 gives examples of the master and slave file formats. The first line of both files gives the headings for each column, separated by spaces, and are present to improve readability. The master file is written first, each line stating the value of every known quantity or an “x” for unknown quantities, at a single radial position. The slave file is then created from the master file, substituting in “0” values for “x” characters. This slave file can be updated to reflect the latest estimate at the full stress profile, while the master file retains a record of which values should be kept constant and which can be altered.

The flexibility of this system allows for known values to be changed and new components added in the future to reflect new breakthroughs in measurement techniques or understanding.

### *User Subroutine*

---

From the structure of the SIGINI subroutine shown in Figure 4-52 it can be observed that the same process is used as in the UTEMP subroutine, to ensure that the data file need only be loaded once. The coordinates supplied by ABAQUS must be converted to the polar reference frame in order for the radial position to be ascertained. This process has been discussed in the thermal subroutine section (section 4.3.1) and will not be repeated. The calculation of the stresses is a simple one-dimensional interpolation between the data points provided. This is also similar to the process described in the thermal subroutine section and an explanation has therefore been omitted too.



**Figure 4-52: SIGINI process flow**

Converting the resultant stress tensor from a rectangular reference frame to a cylindrical polar reference frame is not as straightforward as converting the coordinates. This is due to the interplay between the normal stress and shear stress components.



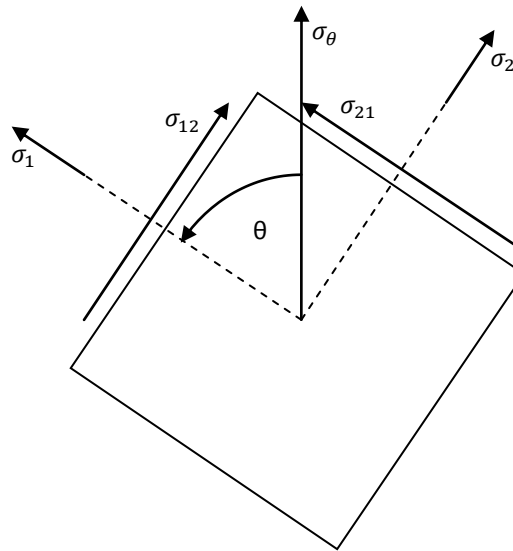


Figure 4-53: Rotation of a normal stress tensor component

Figure 4-53 shows a normal stress component as defined in the cylindrical polar reference frame and the tensor components of the rectangular reference frame whose values will be affected by the magnitude of that original component. When the effects of the other original tensor components are introduced the situation becomes too involved for basic geometry and is better served by matrix algebra. Gere and Weaver<sup>33</sup> describe the use of rotation matrices.

$$B = RAR' \quad \text{Equation 4-18}$$

Where  $B$ ,  $R$  and  $A$  are 3x3 matrices.  $A$  is the matrix in its original form,  $R$  is a rotation matrix and  $B$  is the rotated form of  $A$ . Equation 4-18 can be expanded to give;

$$\begin{bmatrix} \sigma'_{11} & \sigma'_{12} & \sigma'_{13} \\ \sigma'_{21} & \sigma'_{22} & \sigma'_{23} \\ \sigma'_{31} & \sigma'_{32} & \sigma'_{33} \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix} \begin{bmatrix} R_{11} & R_{21} & R_{31} \\ R_{12} & R_{22} & R_{32} \\ R_{13} & R_{23} & R_{33} \end{bmatrix} \quad \text{Equation 4-19}$$

Using this notation  $\sigma_{11}$ ,  $\sigma_{22}$  and  $\sigma_{33}$  represent the normal stresses in the 1, 2 and 3 directions respectively, and  $\sigma_{12}$ ,  $\sigma_{13}$  and  $\sigma_{23}$  are the corresponding shear stresses.  $\sigma_{21}$ ,  $\sigma_{31}$  and  $\sigma_{32}$  represent the opposing shear stress couples (the same-coloured shear stresses in Figure 4-50) and are therefore identical to  $\sigma_{12}$ ,  $\sigma_{13}$  and  $\sigma_{23}$  respectively.

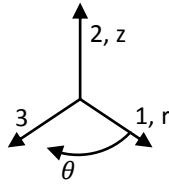


Figure 4-54: ABAQUS standard axes

Figure 4-54 shows the axis convention used in ABAQUS. By default direction 2 is set as the axis for rotation and so represents the axial direction. To keep the mathematics straightforward the original 1, 2 and 3 directions will refer to the radial ( $r$ ), axial ( $z$ ) and hoop ( $\theta$ ) components respectively, so that axis 2 and the axial direction remain the same. The re-ordering from radial, hoop, axial to radial, axial, hoop is performed in the user subroutine.

Equation 4-19 can be expanded to give the following set of equations.

$$\begin{aligned} \sigma'_{11} = & R_{11}^2 \sigma_{11} + R_{12}^2 \sigma_{22} + R_{13}^2 \sigma_{33} + 2R_{11}R_{12} \sigma_{12} + 2R_{11}R_{13} \sigma_{13} \\ & + 2R_{12}R_{13} \sigma_{23} \end{aligned} \quad \text{Equation 4-20}$$

$$\begin{aligned} \sigma'_{22} = & R_{21}^2 \sigma_{11} + R_{22}^2 \sigma_{22} + R_{23}^2 \sigma_{33} + 2R_{21}R_{22} \sigma_{12} + 2R_{21}R_{23} \sigma_{13} \\ & + 2R_{22}R_{23} \sigma_{23} \end{aligned} \quad \text{Equation 4-21}$$

$$\begin{aligned} \sigma'_{33} = & R_{31}^2 \sigma_{11} + R_{32}^2 \sigma_{22} + R_{33}^2 \sigma_{33} + 2R_{31}R_{32} \sigma_{12} + 2R_{31}R_{33} \sigma_{13} \\ & + 2R_{32}R_{33} \sigma_{23} \end{aligned} \quad \text{Equation 4-22}$$

$$\begin{aligned} \sigma'_{12} = & R_{11}R_{21} \sigma_{11} + R_{12}R_{22} \sigma_{22} + R_{13}R_{23} \sigma_{33} + (R_{11}R_{22} + R_{12}R_{21}) \sigma_{12} \\ & + (R_{12}R_{23} + R_{13}R_{22}) \sigma_{23} + (R_{11}R_{23} + R_{13}R_{21}) \sigma_{13} \end{aligned} \quad \text{Equation 4-23}$$

$$\begin{aligned} \sigma'_{13} = & R_{11}R_{31} \sigma_{11} + R_{12}R_{32} \sigma_{22} + R_{13}R_{33} \sigma_{33} + (R_{11}R_{32} + R_{12}R_{31}) \sigma_{12} \\ & + (R_{12}R_{33} + R_{13}R_{32}) \sigma_{23} + (R_{11}R_{33} + R_{13}R_{31}) \sigma_{13} \end{aligned} \quad \text{Equation 4-24}$$

$$\begin{aligned} \sigma'_{23} = & R_{21}R_{31} \sigma_{11} + R_{22}R_{32} \sigma_{22} + R_{23}R_{33} \sigma_{33} + (R_{21}R_{32} + R_{22}R_{31}) \sigma_{12} \\ & + (R_{22}R_{33} + R_{23}R_{32}) \sigma_{23} + (R_{21}R_{33} + R_{23}R_{31}) \sigma_{13} \end{aligned} \quad \text{Equation 4-25}$$

These equations can be simplified significantly by noting that, while the 1 and 3 rectangular dimensions do not match the  $r$  and  $\theta$  dimensions, the 2 and  $z$  dimensions are coherent.

$$R = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \quad \text{Equation 4-26}$$

Equation 4-26 gives the rotation matrix required to rotate a matrix through  $\theta$  degrees around the 2-axis. No matter the angle required  $R_{22}$  will always be 1 and  $R_{12}$ ,  $R_{21}$ ,  $R_{23}$  and  $R_{32}$  will always be 0. Substituting in these values gives the following simplified form.

$$\sigma'_{11} = R_{11}^2\sigma_{11} + R_{13}^2\sigma_{33} + 2R_{11}R_{13}\sigma_{13} \quad \text{Equation 4-27}$$

$$\sigma'_{22} = \sigma_{22} \quad \text{Equation 4-28}$$

$$\sigma'_{33} = R_{31}^2\sigma_{11} + R_{33}^2\sigma_{33} + 2R_{31}R_{33}\sigma_{13} \quad \text{Equation 4-29}$$

$$\sigma'_{12} = R_{11}\sigma_{12} + R_{13}\sigma_{23} \quad \text{Equation 4-30}$$

$$\sigma'_{13} = R_{11}R_{31}\sigma_{11} + R_{13}R_{33}\sigma_{33} + (R_{11}R_{33} + R_{13}R_{31})\sigma_{13} \quad \text{Equation 4-31}$$

$$\sigma'_{23} = R_{31}\sigma_{12} + R_{33}\sigma_{23} \quad \text{Equation 4-32}$$

Equation 4-27 to Equation 4-32, along with Equation 4-26, give the complete stress tensor rotated from cylindrical polar coordinates to the rectangular coordinate supplied by ABAQUS. The SIGMA parameter is then loaded with these values and returned to ABAQUS.

### *Automation*

---

While originally built into the full stress model, the iteration process required to give a full residual stress state takes a long time to converge on a solution. Given that the residual stresses are associated with a specific roll, not a particular point in the process history, the decision was made to un-link the residual calculation from the main stress simulation. Instead a residual stress result can be calculated for a specific work roll and the results then imported to multiple analyses without the required overhead of running the residual model for each instance.

Figure 4-55 gives the process flow for the residual stress model. A dedicated windows form provides the user with controls for specifying the master file setup and either directly supplying or loading roll geometric and material properties from a properties file generated by the thermal models.

The judgement of whether a change is significant is based on the total amount of change in the slave data file when it is updated. If the total change is above a set limit then the process is repeated, until the total change caused by an update falls below this value. Defining the specific value of the limit is dealt with in Chapter 5.

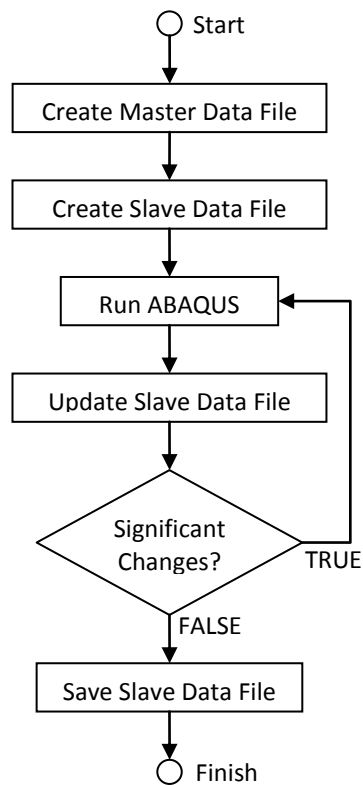


Figure 4-55: Process flow for calculating residual stresses

#### 4.4. Conclusions

---

While the FE model inherited from the previous project provided a useful tutorial, the current models had to be built from scratch following an update to a later version of ABAQUS. A nested series of specialised models was developed. A global model provides a low resolution image of the whole roll, a surface model provides a medium resolution image of the surface of the roll barrel and a bite model provides a high resolution image of the area around the roll bite. A detailed two-dimensional model was also introduced for the purpose of establishing stable circumferential and radial results, to aid in determining whether the three-dimensional models were capable of achieving a high enough resolution to accurately define the stress profile near the surface.

The boundary conditions of the models were introduced, along with functional descriptions of the user subroutines used to apply them. The completed models can import temperature data from the thermal models and, using data downloaded from the hot mill database, provide loads at the work roll surface based on measured process conditions. A process has also been developed to import measured residual stress data and to provide estimates for any unspecified components by using ABAQUS to enforce internal equilibrium.

## 5. Validation Testing Results

---

### 5.1. Introduction

---

In the previous two chapters the thermal and stress models were introduced. The supporting evidence for decisions made during the model design process, which was withheld from previous chapters for simplicity, is presented in this chapter. Section 5.2 gives the results of two investigations performed using the thermal models. Section 5.3 looks in detail at the sub-modelling concept and proof of its suitability to the work roll thermal problem, along with comparison studies and treatment of boundary conditions. The validation was carried out using the constant material properties versions of the core and shell models.

NB – All of the results in this chapter were generated using the data sets given in Appendix D.1 or Appendix D.2, two data sets that are identical in all but length and reduced to a very small size to allow quick and easy debugging.

### 5.2. Thermal Model Investigations

---

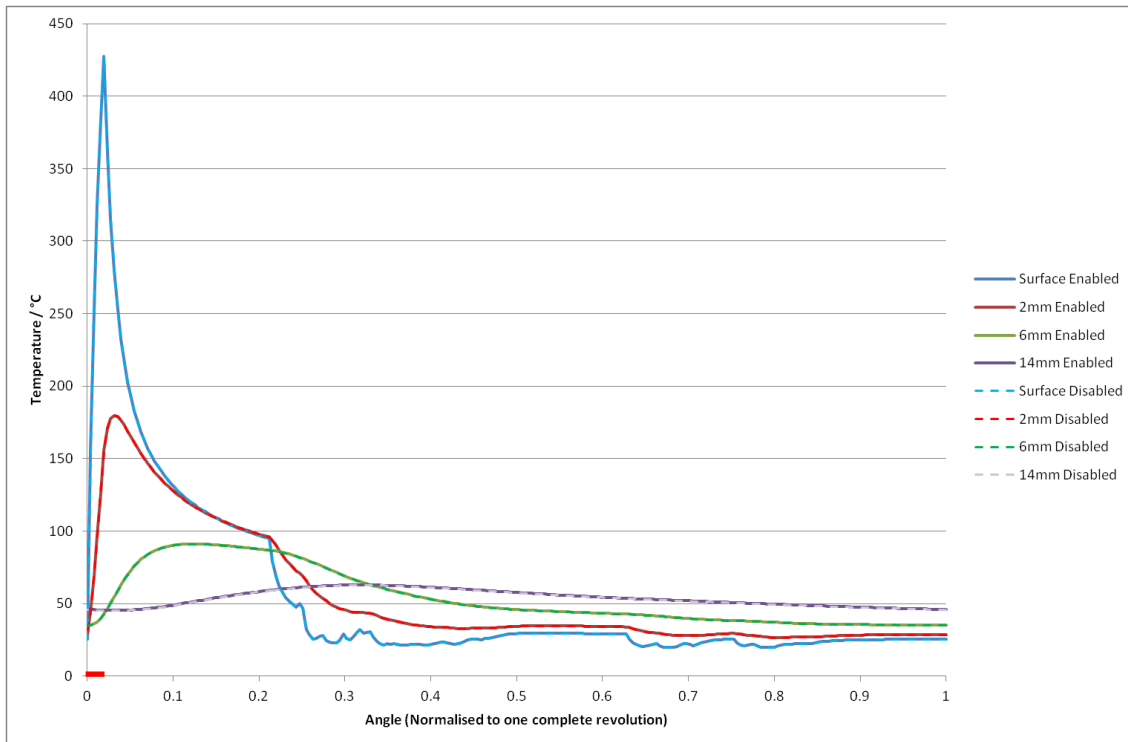
#### 5.2.1. Testing circumferential conduction

Having possession of a fully three dimensional model presents the opportunity of testing one of the principle assumptions of Abbaspour and Saboonchi<sup>21</sup>, i.e. that circumferential conduction is negligible and can therefore be removed from consideration in the governing heat flow equation. It would be useful to know whether this assumption is creating a different temperature variation than could realistically be expected.

Recalling Equation 2-69 from Chapter 2 (the governing heat flow equation for the shell model), the reduced conduction assumption can be made by eliminating one of the conduction terms.

$$0 = \alpha \left( \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial r^2} + \frac{\partial^2 T}{\partial z^2} \right) - \omega \frac{\partial T}{\partial \theta} \quad \text{Equation 5-1}$$

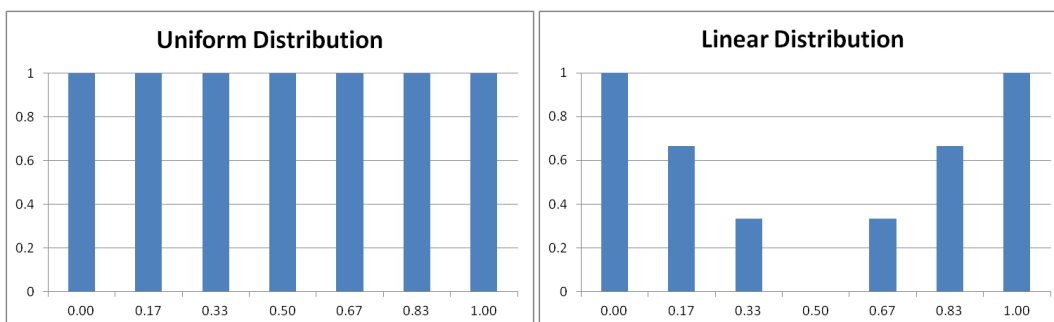
Working through the requisite substitutions and rearranging yields a set of equations identical to Equations 2-71, 2-72 and 2-73, except that  $j_c = 0$ . Implementing this substitution yields a shell model which assumes no angular conduction.



**Figure 5-1: Shell model results with and without circumferential heat conduction enabled, at different depths below the roll surface. The temperature profiles with conduction are nearly identical to those without. Roll bite location is indicated by thick red line on the abscissa.**

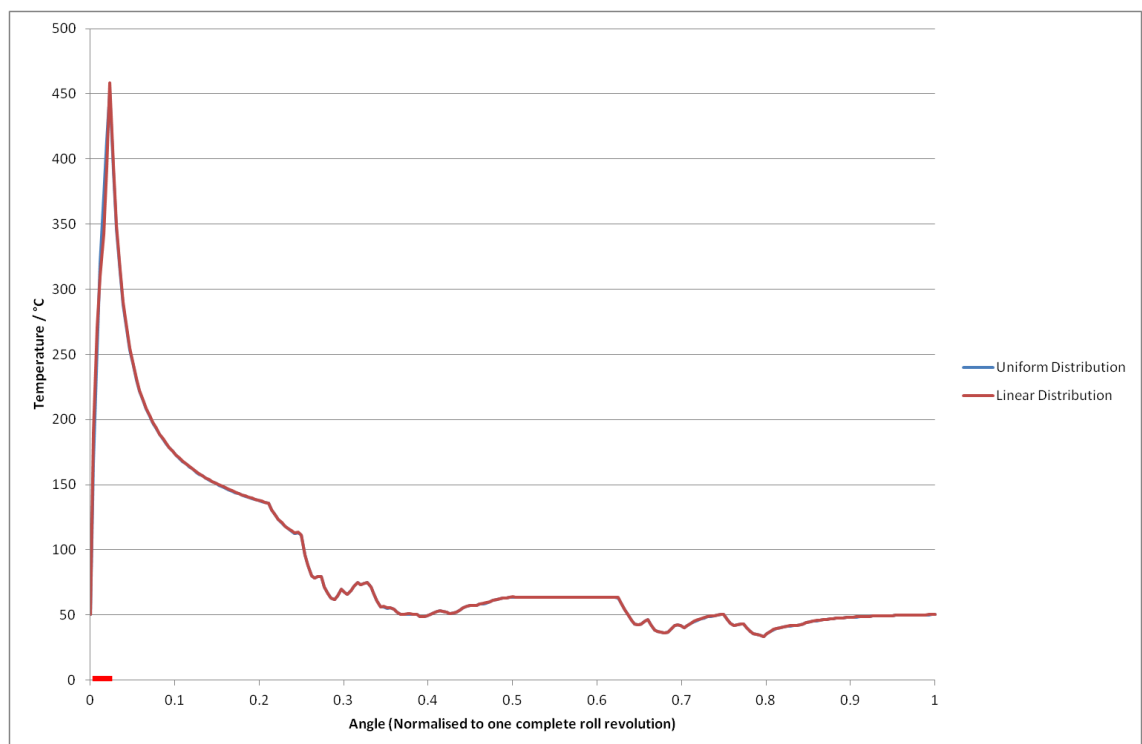
Figure 5-1 shows a set of shell model results with and without the circumferential conduction term enabled. Each line represents the circumferential temperature profile at a certain depth below the work roll surface, i.e. "Surface" is at the surface, "2mm" is 2mm below the surface, etc. The dotted and solid lines are indistinguishable, suggesting that omission of the circumferential conduction term makes practically no difference, justifying the assumptions of [21].

### 5.2.2. Roll bite behaviour



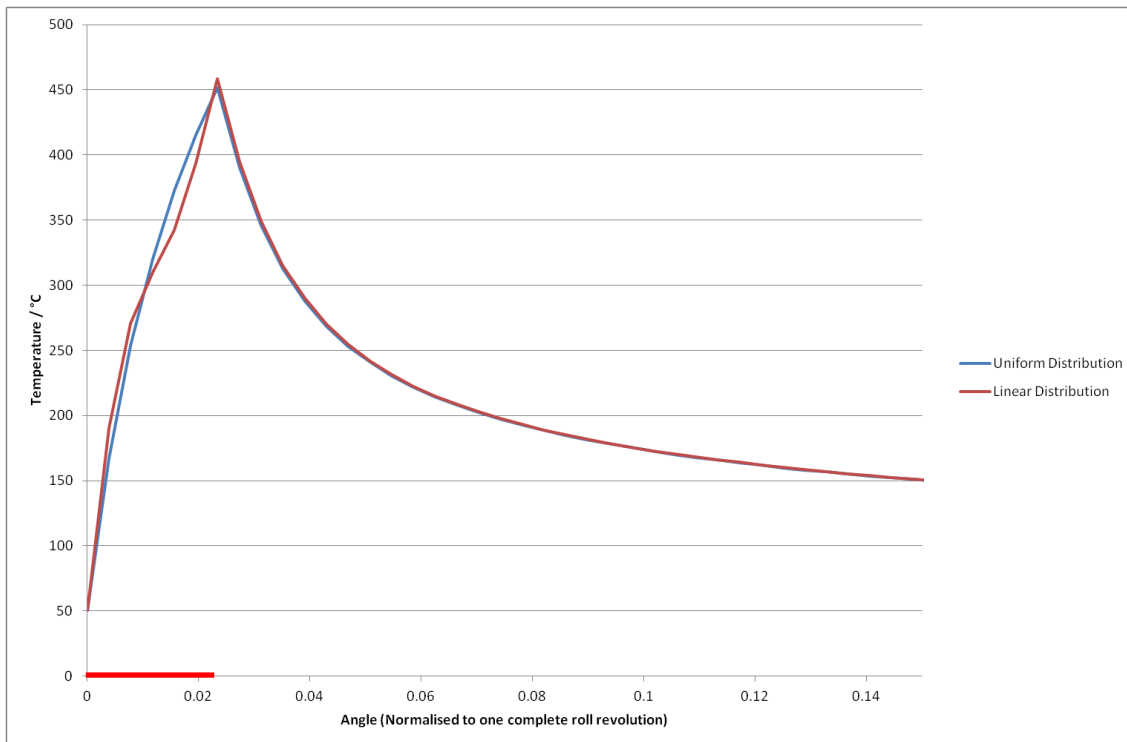
**Figure 5-2: Flat and linear heat transfer distributions (Normalised to the length of the roll bite)**

One of the assumptions made by the thermal model is that the distribution of frictional heat generation through the roll bite is uniform. The real friction heat generation will be related to the slipping speed between the work roll and the strip. At the entry to the roll bite the strip will be moving slower than the work rolls, the difference in speed drawing the strip into the roll bite and preventing the strip from sticking or buckling in a way that might cause a cobble (i.e. failure of the strip to pass through the roll bite, resulting in the strip piling up in the gap between stands and causing lots of damage and down time). As the strip passes through the roll bite, its thickness is reduced under the pressure exerted by the rolls. This reduction in thickness causes the lengthening of the strip, resulting in acceleration. At a position known as the "neutral point" the strip speed overtakes the work roll speed. Between the neutral point and the exit of the roll bite the strip thickness is further reduced, and therefore continues to accelerate. Estimating the neutral point as being in the centre of the roll bite (inaccurate but sufficient for the purposes of this investigation) and plotting the difference in speed between the work roll and strip gives the V-shaped distribution shown in Figure 5-2.



**Figure 5-3: Surface temperature profiles for different friction models. Roll bite location indicated by the thick red line on the abscissa**

Figure 5-3 shows the temperature profiles created using the two friction heating models given in Figure 5-2. The profiles are identical for the majority of the surface..



**Figure 5-4: Detail view of the temperature profiles for different friction models. The location of the roll bite is indicated by the thick red line on the abscissa.**

Figure 5-4 focuses on the small portion of the surface temperature profiles from Figure 5-3 where a significant difference is visible. The reality of heat generation in the roll bite will doubtless include many other factors, such as normal pressure distribution and plasticity. However, the extreme linear distribution given in Figure 5-2 provides a worst-case scenario for comparison to the uniform heating distribution. Even using this worst case scenario, the change to the shape of the temperature profile is only slight, and can be expected to reduce if a more realistic distribution was selected. This supports the use of a simple linear distribution, since the potential impact of using a different profile is minimal.

### 5.3. Stress Model Validation

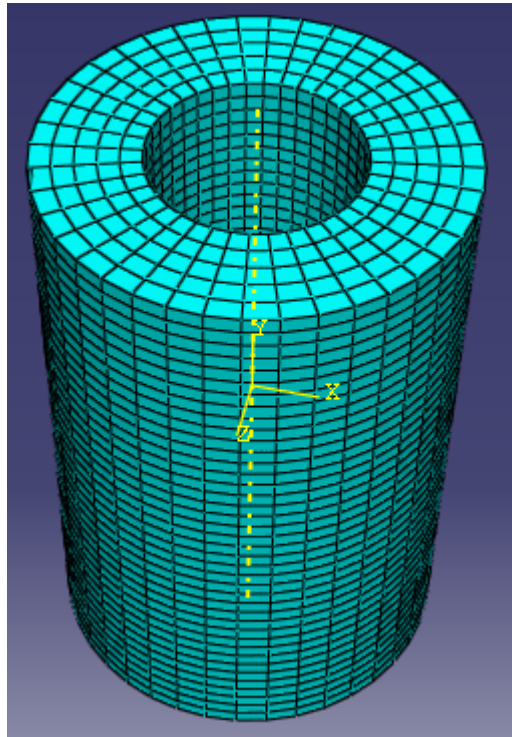
---

#### 5.3.1. Sub-Modelling proof

Because the variation of temperature with radius and depth can be spatially rapid in the bite region, it is clear that a finely resolved FEA mesh will be needed in that area. However, a fine mesh is unlikely to be justified over most of the roll. ABAQUS provides the means to make sub models with finer resolution and this was identified as a promising way forward. However, before breaking the stress modelling task into smaller sub-tasks to enable greater mesh

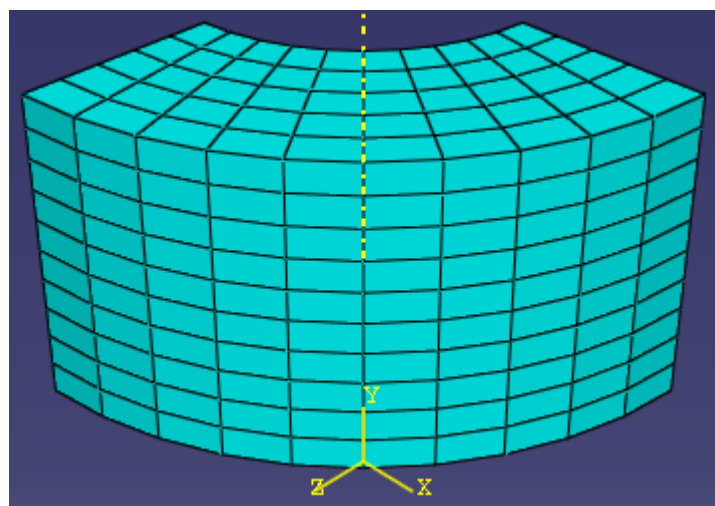


density in areas of importance, a trial was performed to assess the viability of the sub-modelling process.



**Figure 5-5: Test global model**

Figure 5-5 shows the volume used to test the theory of sub-modelling. The shape was chosen to bear some resemblance to the work roll geometry but simplified to allow a stable, well-structured mesh to be produced with minimum development time.

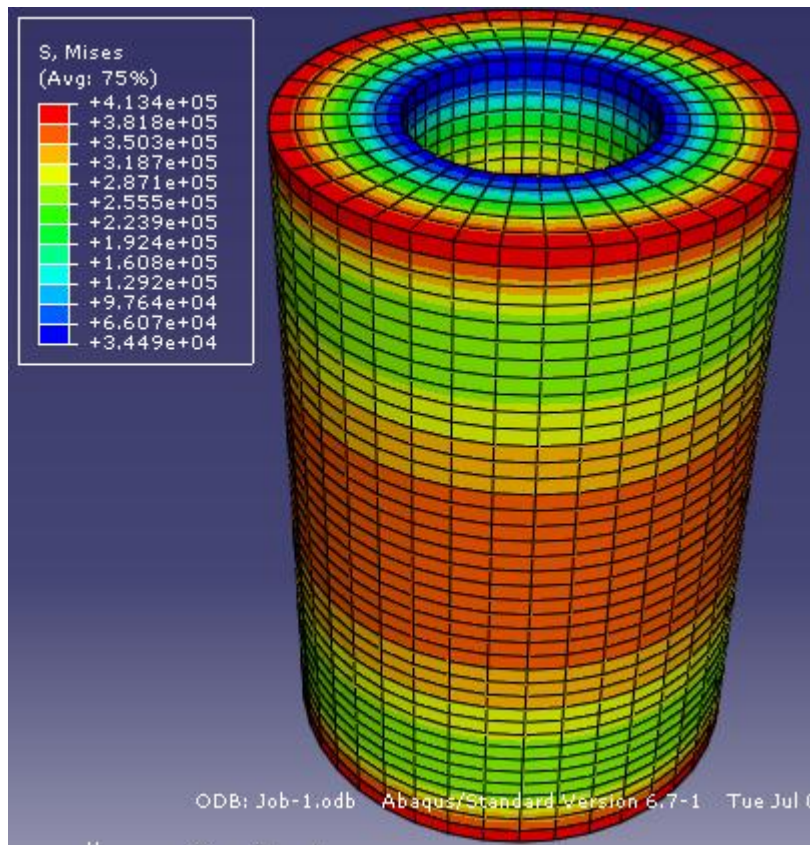


**Figure 5-6: Test local model**

Figure 5-6 shows the first test local model. The volume of this model corresponds to a ninety degree section of the top quarter of the global model and is intentionally given exactly

the same mesh density and structure as the global model. Since every aspect except the nature of the two models is identical, the two models should ideally give identical results.

To keep the test model as relevant to the work roll model as possible, the test models were loaded thermally with the profile,  $T = 3r + 20$ , where  $T$  is the temperature of any point in the model and  $r$  is the radial coordinate. This gives a linear and uniform increase in temperature from the inner to the outer radius of the model, the temperature difference giving differential expansion and resulting in a stress field.



**Figure 5-7: The global test model's Von-Mises stress result**

Figure 5-7 shows the global test model's stress field. The stress is highest at the edges, due to stress raising effects at the edges. These stress peaks are effectively superimposed on a more general tendency towards higher stresses around the middle of the cylinder, where axial stresses are highest.

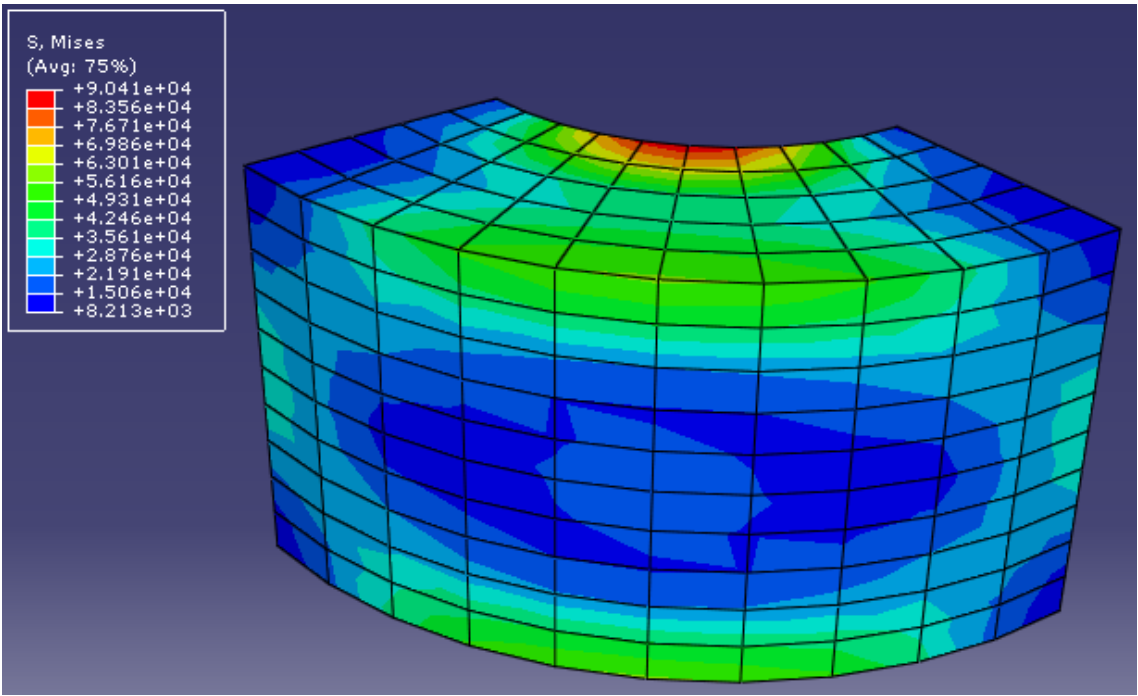


Figure 5-8: Local test model Von-Mises stress result without sub-model boundary condition

Figure 5-8 shows the stress field created in the local model by applying the temperature profile but not implementing the sub-modelling technique. Since the local model defines a very different geometry it reacts to the same temperature profile in a very different way, as would be expected.

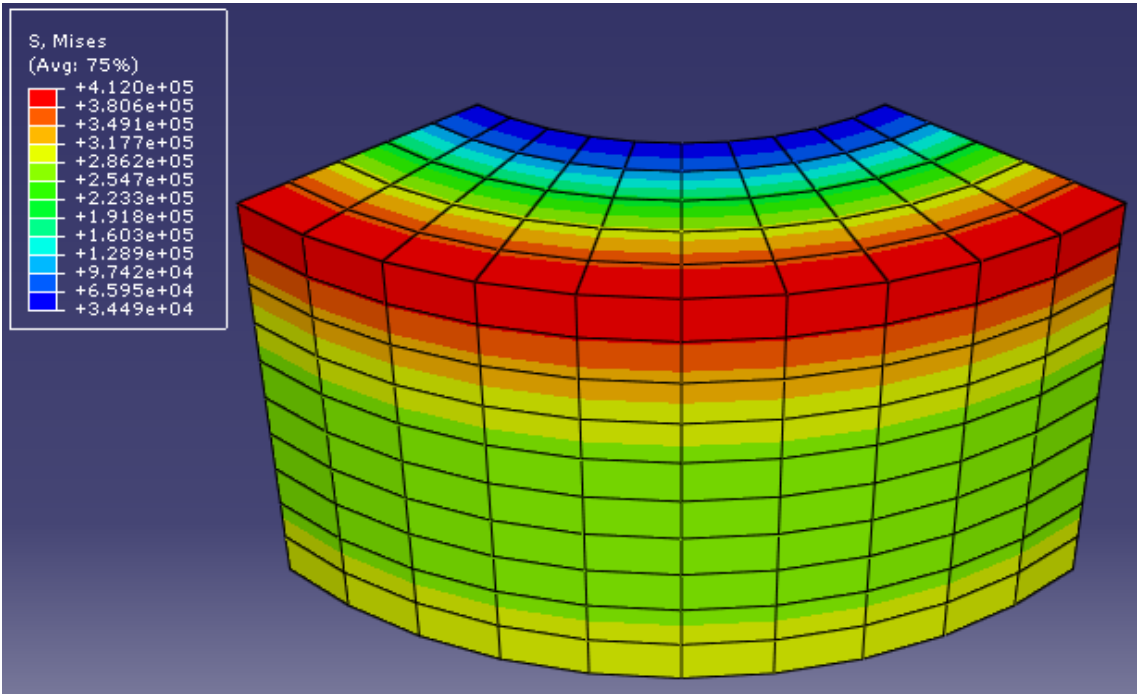


Figure 5-9: Local test model Von-Mises stress result with sub-model boundary condition

Figure 5-9 shows the stress field after the sub-model boundary condition has been applied to the "internal" faces, i.e. those faces which would be created by cutting the local model out of the global model. These newly created faces cannot be treated like normal "external" faces, since they represent the transition between the local and global models, not the transition between the modelled component and the surrounding environment. This is achieved by prescribing the same deflections to nodes on the "internal" faces as are found in the results of the global model. As Figure 5-9 shows, this has the effect of causing the local model to behave as though it is a part of the global model as far as deflection is concerned, rather than an isolated component. Figure 5-9 and Figure 5-7 seem to describe a very similar response, suggesting that the sub-model boundary conditions are successfully emulating the behaviour of the global model.

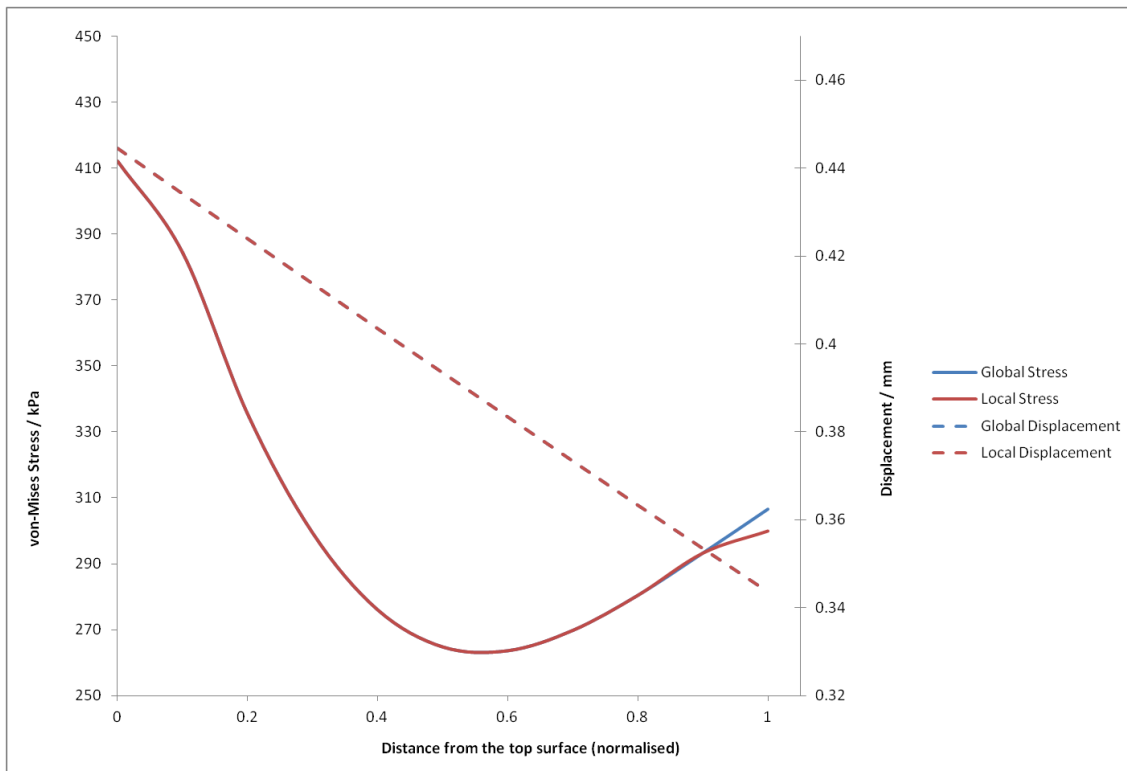
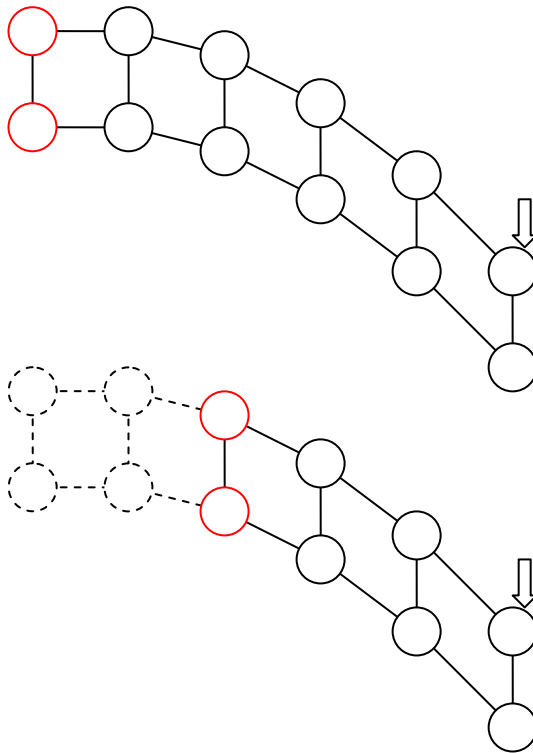


Figure 5-10: Comparison between global and local test model radial result profiles

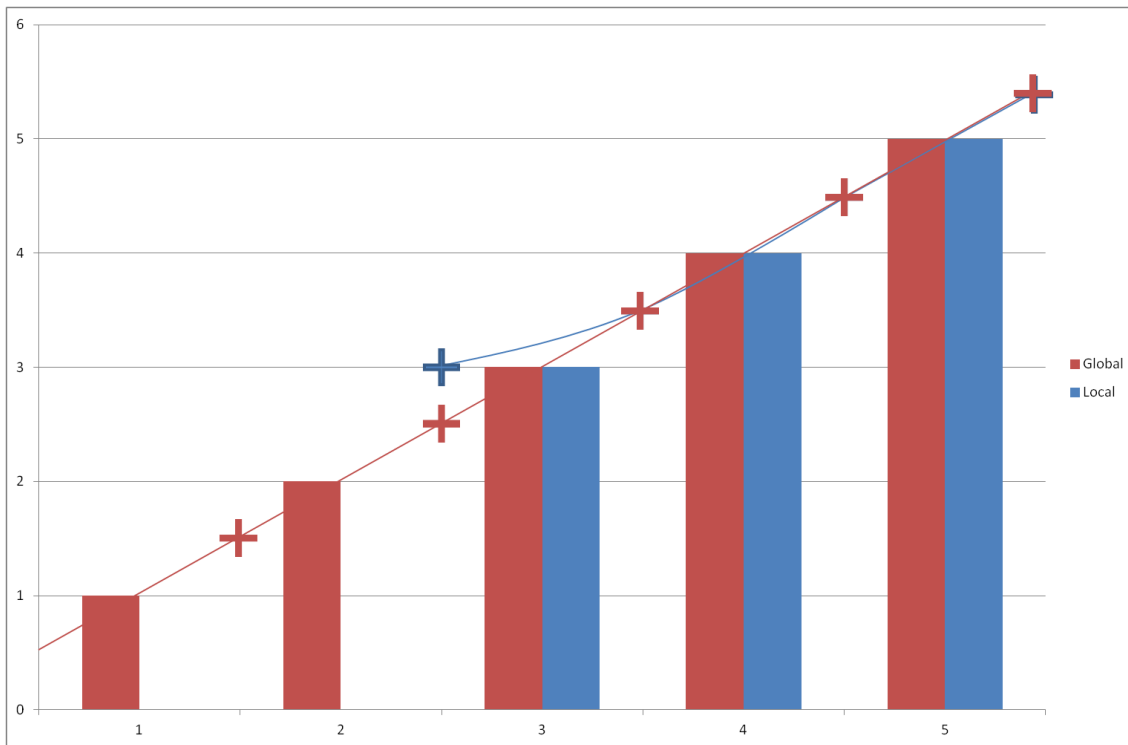
Figure 5-10 gives a clearer comparison between the two figures, comparing the stress and deflection results along the outer radius, following a path from the top edge to the bottom edge of the local model. The deflection results are identical. This is also the case for the von-Mises stress results, except for a slight kink on the approach to the interface between the local and global models. To explain this feature it is necessary to examine what is happening at the sub-model boundary condition.



**Figure 5-11: Example loaded global (top) and local (bottom) model meshes**

Figure 5-11 gives a very simplified example of a situation in which a global and local model are being used. In this example the local model is an identical replica of a part of the global model. The highlighted nodes in the global model are subject to a restraint-type boundary condition, restricting them from moving in any way. At the opposite end of the global model a vertical force is applied to deflect the whole structure. In the local model the highlighted nodes represent the sub-model boundary condition, where the displacement and rotation of the nodes is specified to be the same as an equivalent position in the global model. The same load that was applied to the global model is also applied to the local model. The two models should produce the same deflected shape.

Problems are introduced when the stresses are calculated. When Abaqus calculates stress in a reduced-integration linear element, the type of element used by the stress model, the stress value is based on the deflected shape of the whole element. The stress for a particular node is then obtained by taking the average of the stresses from each element that the node is connected to. Considering the highlighted nodes in the local model, the elements to the right have a known deflected shape but the outlined elements to the left are not actually included in the model. As such, when the averages are taken these missing elements are ignored and the stress field appears significantly different.



**Figure 5-12: Element and nodal stresses in the example model**

Figure 5-12 presents a fictitious set of stress values, with the bars representing element stresses and the crosses representing nodal stresses. The global model's stress profile is quite simple, a linear increase in stress from the left to the right end of the model. The local model's elemental stresses are identical to the global model's elemental stresses, and the nodal stresses are also generally correct. The only inconsistency occurs at the interface between the local and global models, where the global model uses the average of elements 2 and 3 to calculate the nodal stress, while the local model can only use the value of element 3.

Recalling Figure 5-10 the unexpected feature on the right side of the stress results appears to be the product of a similar lack of information. As such it can be stated that the sub-modelling technique works perfectly for situations where the local model is at the same resolution as the global model. However, care should be taken near to the sub-model boundary condition, since nodal stress results in this area are likely to be inaccurate.

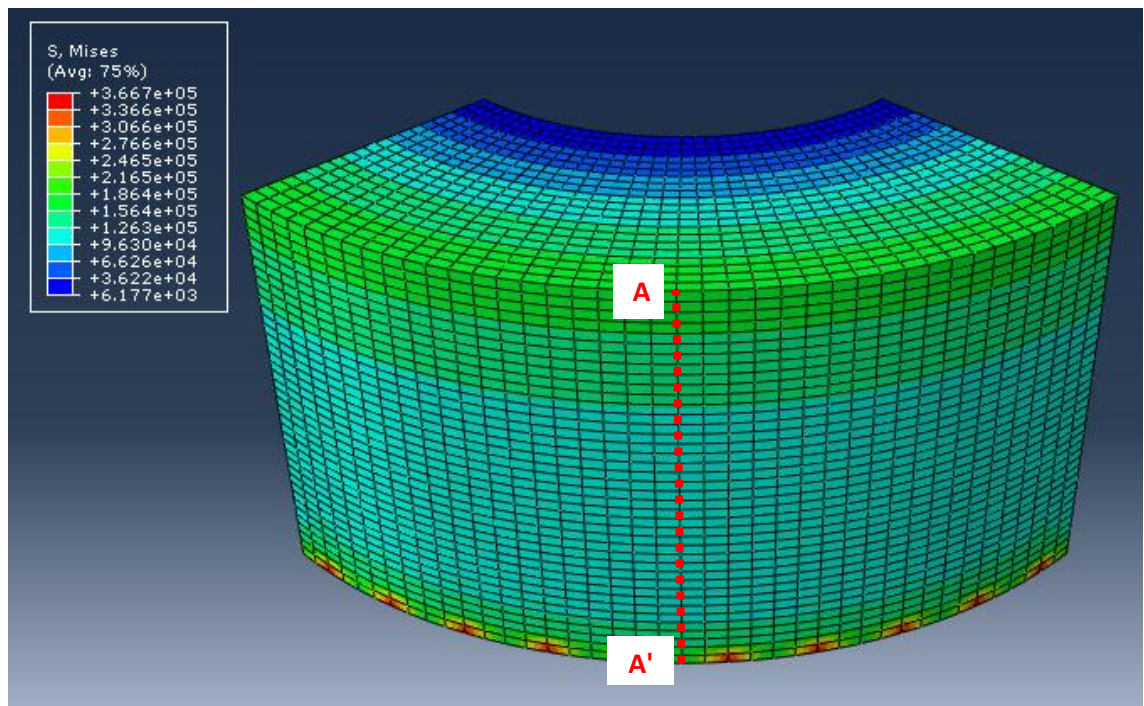


Figure 5-13: High resolution local model

In order for the sub-modelling technique to be of use, it must be able to work effectively when the local model is at a higher resolution than the global model. Figure 5-13 shows a high resolution local model used to test this scenario. There is regular oscillation at the sub-model boundary condition on the lower surface, with each peak coinciding with the centre of an element in the global model. This is due to the linear nature of the global model results giving the highest accuracy at the nodes but the lowest accuracy at the centre of the element.

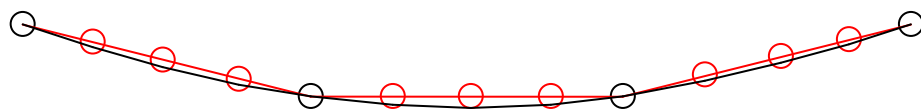
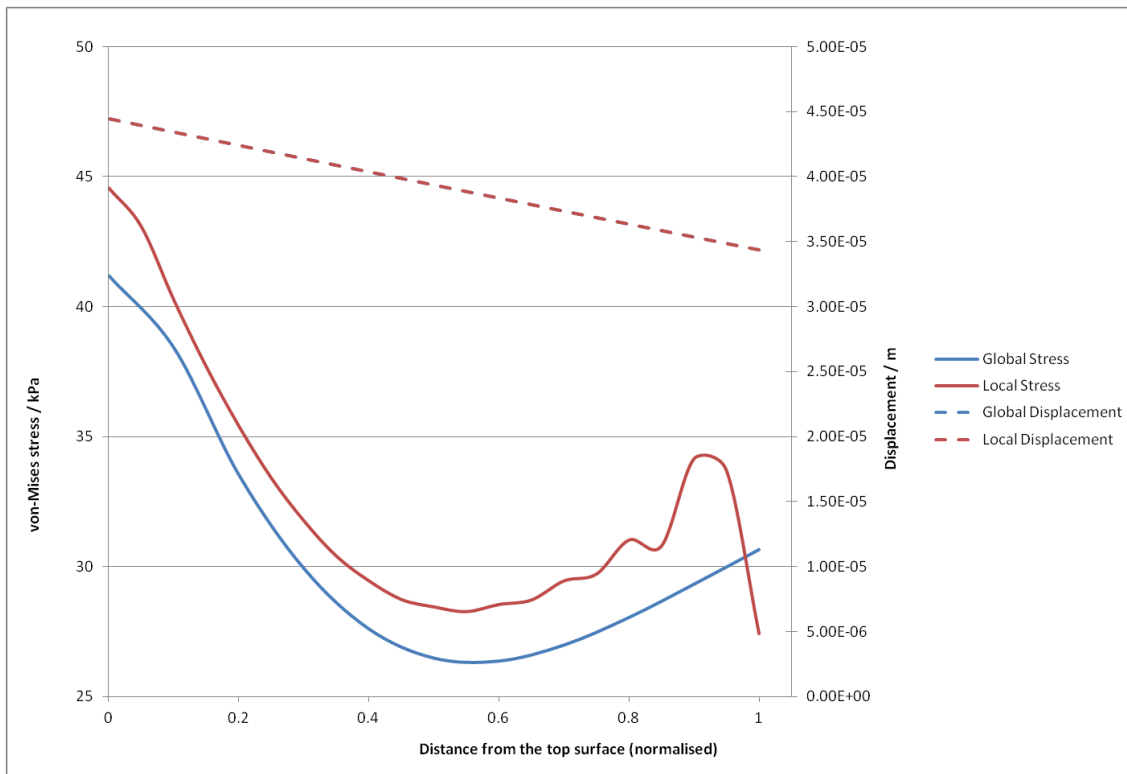


Figure 5-14: Predicting the displacement of local model nodes from a lower resolution global model

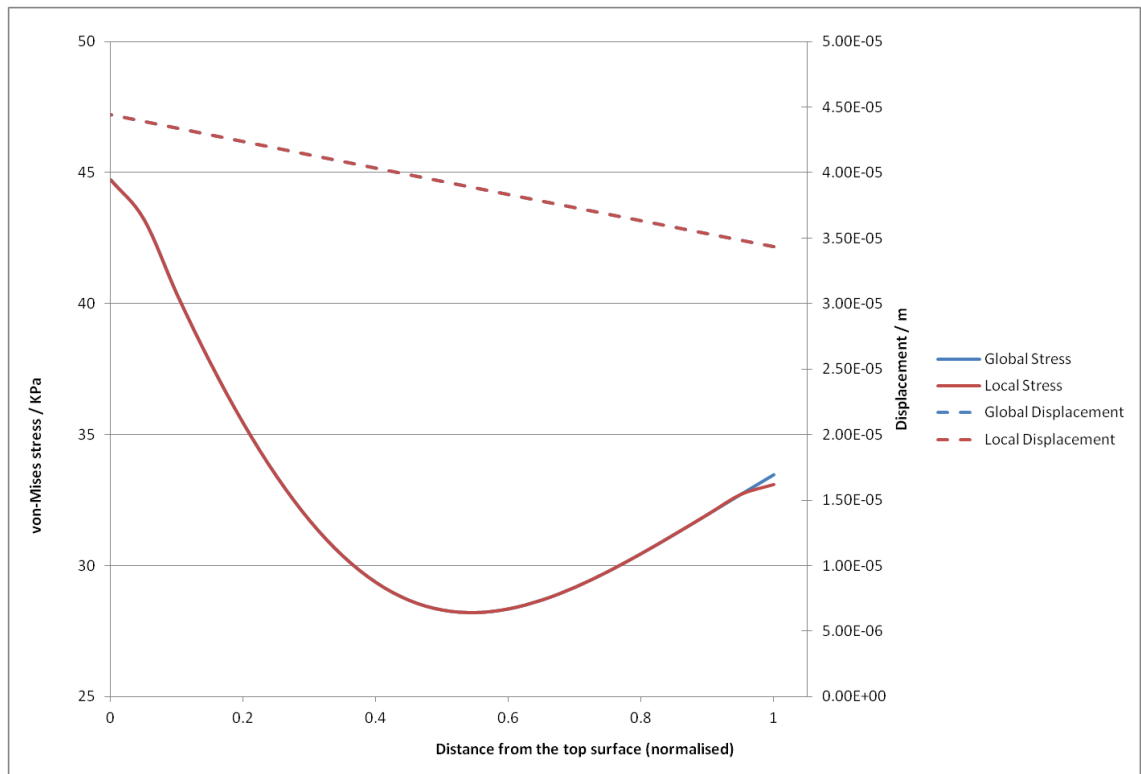
Figure 5-14 illustrates the interpolation problem. The black line represents the ideal shape of the surface, while the red line represents the actual shape of the surface as calculated by linear interpolation from the four global model data points. If the local model has four times the mesh density of the global model then extra nodes must be added, shown here in red. The difference between the red line and the black line at each of the nodes will result in extra expansion or contraction of an element corner, which will incur extra stresses in those elements. While this oscillatory behaviour is present in Figure 5-13, it decays within the space of a couple of elements.



**Figure 5-15: Comparison of stress and displacement in a low resolution global model and high resolution local model. Comparison is along line AA' of Figure 5-13**

Figure 5-15 shows the stress and displacement results from a high resolution local model and a low resolution global model, both following the line AA' in Figure 5-13. The global and local model displacements are identical, but the stress results are very different. The local model stress starts out lower than the global model stress at the sub-model boundary (the right end of the graph) but quickly exceeds it, fluctuating severely. The oscillations die down approximately half way through the local model before settling down, but at a significantly higher level of stress than the global model.

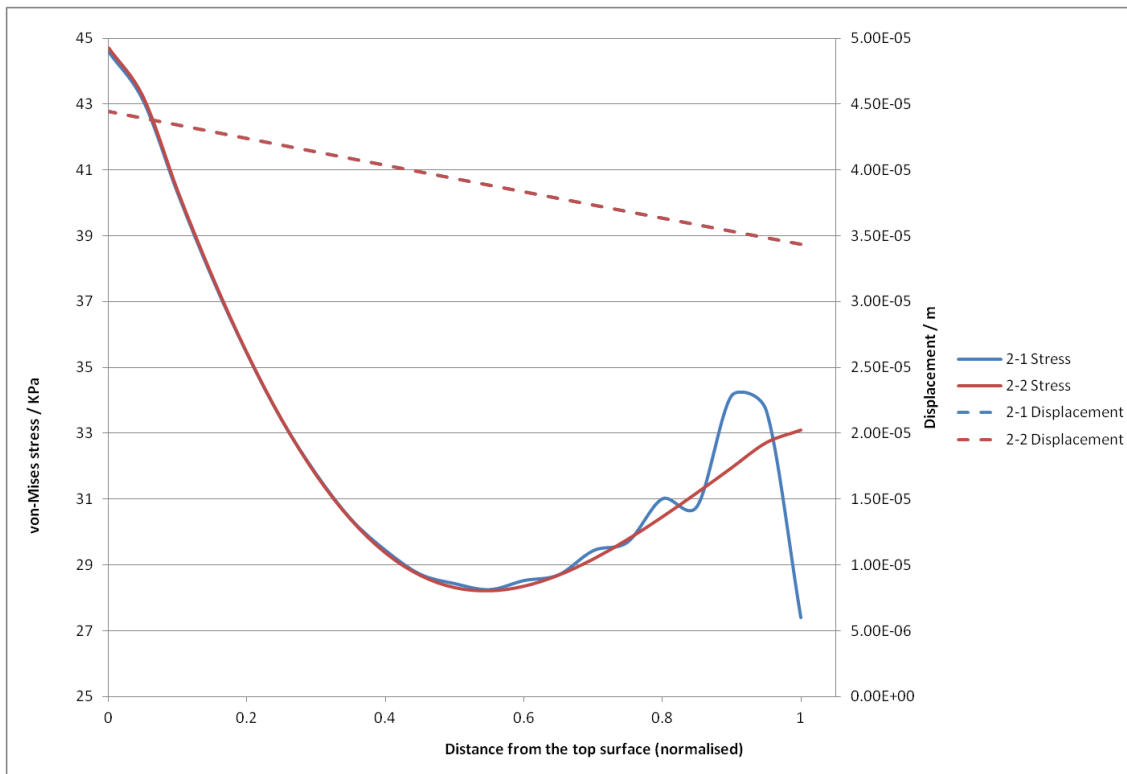




**Figure 5-16: Comparison of stress and displacement in high resolution global and local models. Comparison is along line AA' of Figure 5-13**

Figure 5-16 shows the same comparative stress and displacement plots as Figure 5-15, but with the global model resolution increased to match the resolution of the local model. The displacement results coincide exactly, as with the previous figure. However, in this example the stress results also agree very well, excepting a small feature at the sub-model boundary of the type determined to be caused by an averaging error when calculating the nodal stresses.

Since the global and local models agree on the stress result when both models are using a higher resolution, it would appear that the original low resolution global model was simply at too low a resolution to accurately define the stress profile. The difference in stress displayed in Figure 5-15 is therefore caused by the difference in resolution, not any fault in the operation of the sub-model boundary condition.

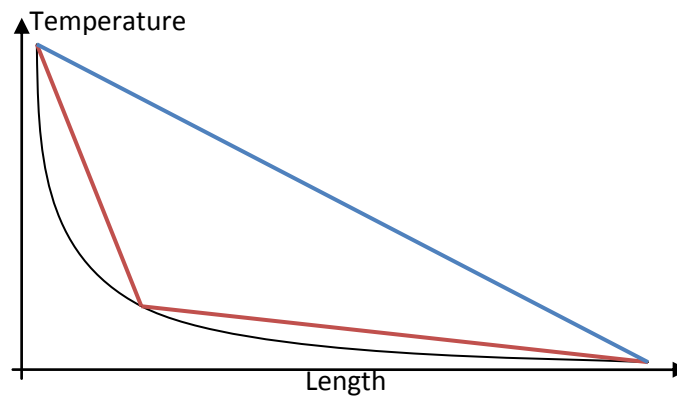


**Figure 5-17: Comparison of stress and displacement in the local models from Figure 5-15 and Figure 5-16. Comparison is along line AA' of Figure 5-13**

Figure 5-17 compares the high resolution local model results from Figure 5-15 and Figure 5-16. "2-1" refers to the local model which uses a low resolution global model to supply the displacements for its sub-model boundary conditions. "2-2" refers to the local model which uses a high resolution global model to supply the displacements for its sub-model boundary conditions. As in previous figures, the displacement results lie directly on top of each other. While the stress behaviour of "2-1" is very unstable near to the sub-model boundary condition, away from the sub-model boundary condition the two sets of results agree very well.

The significance of this finding is that a high resolution local model does not necessarily need to be informed by the displacements of a high resolution global model to predict accurate stress results, as long as a reasonable distance is maintained from any sub-model boundary conditions.

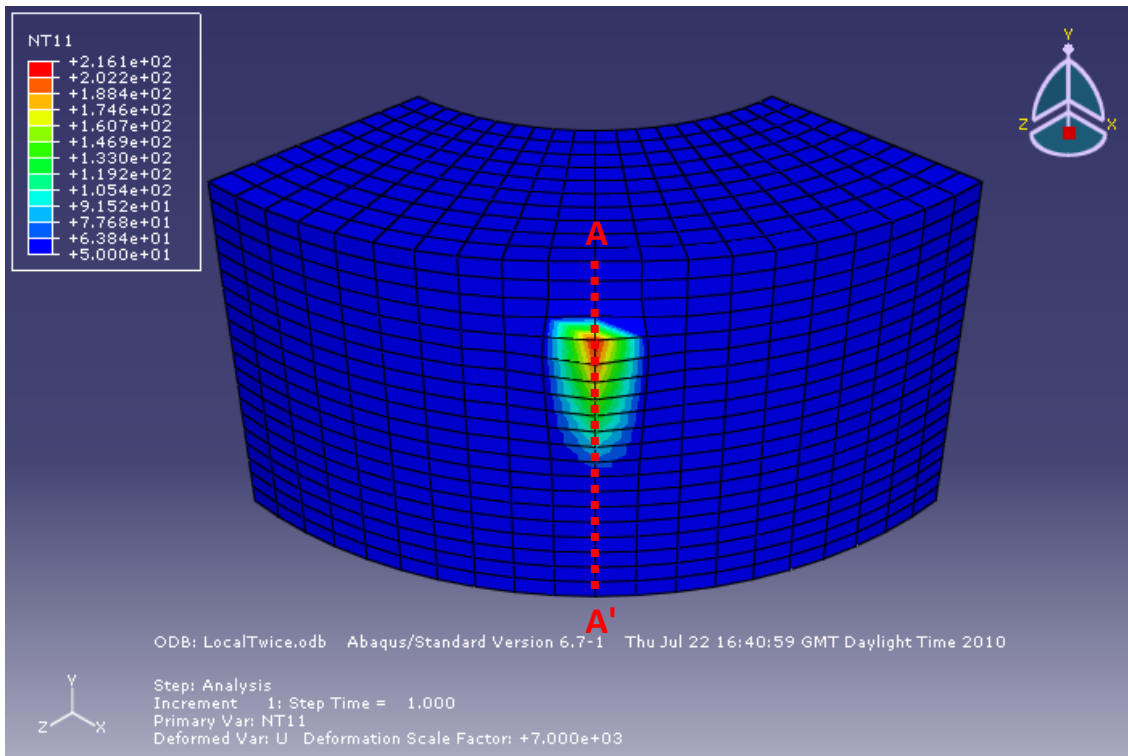
The last consideration in judging the suitability of the sub-modelling technique for the task of modelling a work roll is the effect on a local model of using a global model with an under-defined thermal feature.



**Figure 5-18: Effect of element size on reproduction of thermal features**

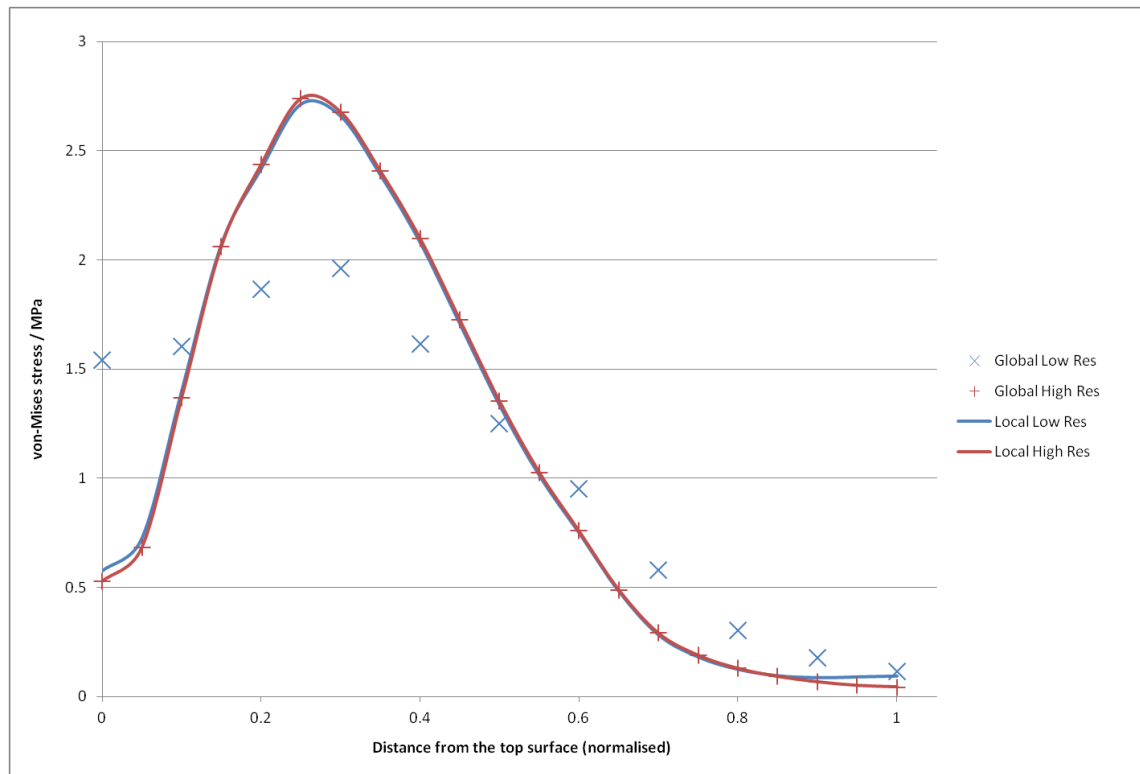
Figure 5-18 shows the effect of interpolation on an ideal temperature profile when element sizes vary. The black curve in the figure represents the ideal temperature profile, while the red and blue lines represent the actual temperature profiles in a two element and a one element model respectively. The area under the blue profile is greater than the area under the red profile, meaning that the model using the blue profile will contain more thermal energy overall. If this kind of imbalance affects the outcome of the local model then the sub-modelling technique could be very difficult to use effectively.

The test designed to assess the impact of this effect uses a series of four models. The first two models are global models, one with a high resolution (Global High Res) and one with a low resolution (Global Low Res). These two models show the difference caused by increasing the resolution. The third model will be a high resolution local model which uses the results of the Global Low Res model for its sub-model boundary conditions (Local Low Res). If the sub-modelling technique is successful then the Local Low Res model's results should be similar to the Global High Res model's results. The final model is a high resolution local model paired to the Global High Res model (Local High Res), which should give identical results to that model. Any differences between the Local High Res model and the Global High Res model will highlight problems with the sub-model boundary conditions. All of the models use a temperature of 5°C throughout with a linear temperature profile applied along an edge, i.e. a line of nodes, varying from 5°C at the bottom to over 200°C at the top.



**Figure 5-19: Modelling a concentrated thermal feature**

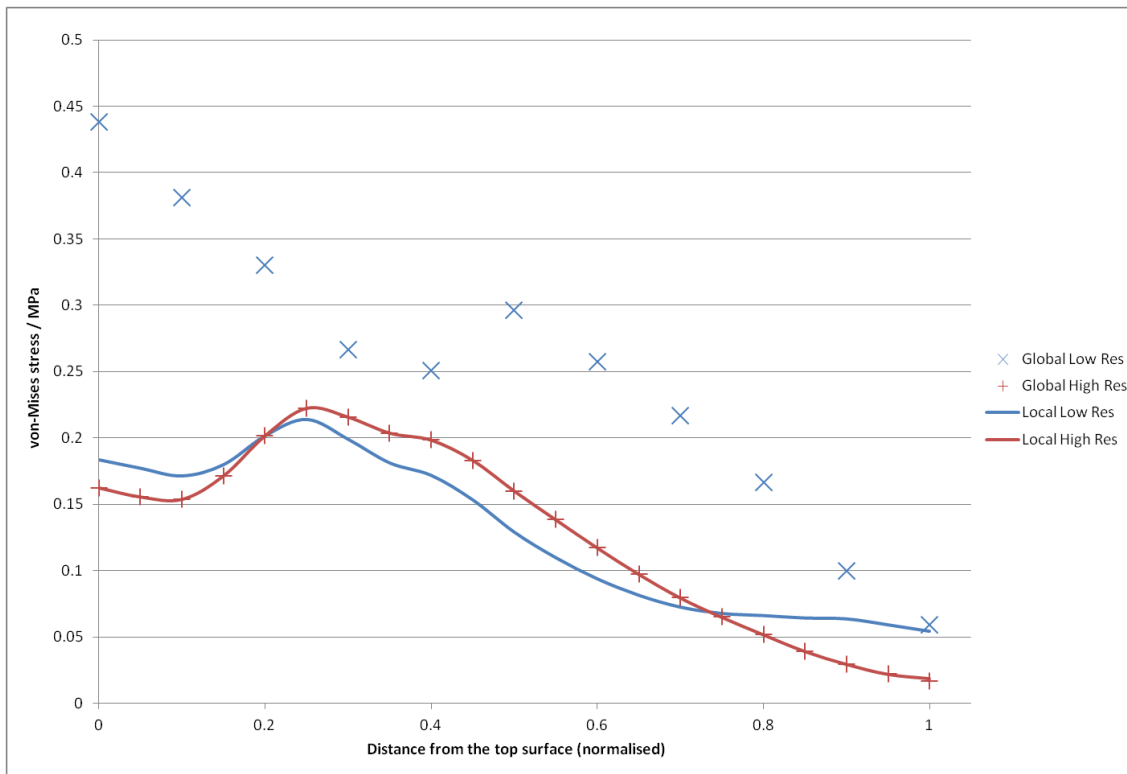
Figure 5-19 shows the small thermal feature applied to the Local High Res model. Ideally all of the models would produce the same stress results, since the same temperature profile is applied to each model in the same way. However, the higher resolution models are made up of smaller elements, which means that the distance between nodes is smaller, so the spatial temperature gradient between a node in the middle of the thermal feature and a node just to the side of it is steeper. The steeper thermal gradients of the higher resolution models will result in larger stresses.



**Figure 5-20: von-Mises stress in a set of global and local models. Comparison is along line AA' of Figure 5-19**

Figure 5-20 compares the surface stress profiles of the four models, following the path of A-A' given in Figure 5-19. The two global models are represented by the "+" and "x" symbols. The stress distribution of the Global Low Res model is slightly diminished compared to the Global High Res model, with lower peak stresses but higher minimum stresses. Recalling Figure 5-18, the greater thermal gradients very near to sharp features are better captured by the higher resolution model, resulting in higher stresses. However, away from the thermal feature the lower resolution model has the steeper temperature gradient, resulting in the lower resolution model having the greater stresses. It is this kind of behaviour which is causing the "Global Low Res" model in Figure 5-20 to have higher stresses at a distance from the thermal feature.

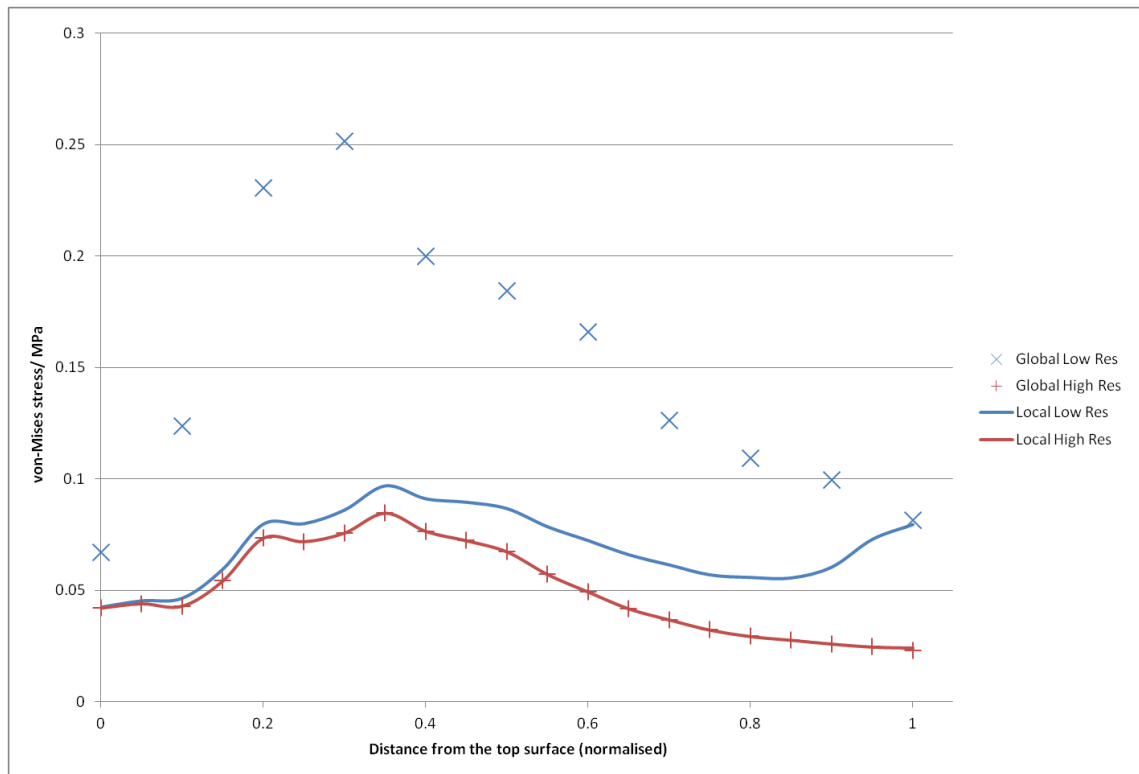
The results of the Local High Res model (high resolution local model tied to a high resolution global model) are almost indistinguishable from the results of the Global High Res model. This shows that the sub-model boundary condition is still working properly when the resolutions of the local and global models are identical. The Local Low Res data (high resolution local model tied to a low resolution global model) starts with very similar stresses to the Global Low Res model at the sub-model interface (right hand side of Figure 5-20). However, in the space of three node points the Local Low Res results are very similar to the Global High Res results, and remain similar across the rest of the profile.



**Figure 5-21: Von-Mises stress in a set of global and local models. Comparison is along a line perpendicular to line AA' of Figure 5-19, halfway between the inner and outer radius.**

Figure 5-21 shows a similar set of results to Figure 5-20, but from below the surface of the test volume, halfway between the inner and outer radius. The predicted stress profiles from each of the global models are much less similar in this instance, with variation in both the shape and the overall magnitude of each profile. The increased stresses in the Global Low Res model are caused by the extra overall thermal energy at the surface, due to the rougher temperature interpolation, resulting in a greater overall expansion and therefore higher stresses.

The Local High Res model is very similar to the Global High Res model. Again, this shows that the sub-model boundary condition is applied correctly and working properly. The Local Low Res model starts at a much higher stress than the Global High Res model is predicting at the sub-model boundary condition (right hand side of Figure 5-21). The difference is greater than the starting difference in Figure 5-20 and so it takes longer for the Local Low Res and Global High Res results to begin to look similar. Even though the Local Low Res model is clearly emulating the behaviour of the Global High Res model much more than the behaviour of the Global Low Res model, the disparate starting conditions have affected the level of agreement between the two high resolution models.



**Figure 5-22: Von-Mises stress in a set of global and local models. Comparison is along a line perpendicular to line AA' of Figure 5-19, running along the inner radius.**

Figure 5-22 shows the same plot as the previous two figures, but considering a path running along the inner radius. The global models' results are significantly different and also start from very diverse initial values at the sub-model boundary condition (right hand side of Figure 5-22). The Local High Res model still matches perfectly with the Global High Res results, so the sub-model boundary condition is still working properly at this location.

The Local Low Res model only manages to achieve an identical result to the Global High Res model at the top surface, with significant differences below the surface. As with the previous figure, the Local Low Res model is displaying much more similarity with the Global High Res results than with the Global Low Res results which were used to provide its boundary conditions. However, the stress profile of the Local Low Res model has become distorted, due to the enormous difference between the Global High Res and Global Low Res results at the sub-model boundary condition.

In all three tests the high resolution and low resolution global models showed different stress distributions, showing that resolution dependency was being effectively modelled. In all three tests the Local High Res model showed that the sub-model boundary condition had been applied correctly and was resulting in the right behaviour. From the three tests, the Local Low Res model showed that a high resolution local model would always mimic the behaviour of a high resolution global model, even when a low resolution global model was used to provide

the displacements for the sub-model boundary condition. However, the tests also showed that the ability of the Local Low Res model to produce an accurate representation of the results of the high resolution global model was dependent on the stresses of the low resolution global model being similar to the stresses of the high resolution global model at the sub-model boundary. Any differences in predicted stress at the sub-model boundary will result in inaccuracies which can propagate through the whole local model stress profile.

Therefore, the sub-modelling technique can be used to concentrate resolution around the roll bite in a work roll model, but only so long as the global model which provides its boundary conditions is reasonably accurate at those locations where the boundary condition values will be taken.

### 5.3.2. 2D validation model

The 2D validation model was designed solely to test the importance of resolution in defining the circumferential stress profile. The aim was to discover whether a measure of resolution independence was being achieved, or whether the ceiling on the resolution of the 3D mesh, as set by current hardware limitations, would prevent accurate results from ever being achieved. The design of this model is discussed in detail in Chapter 4, and consists of a two-dimensional area equivalent to the cross-section of the roll at the centre symmetry plane.

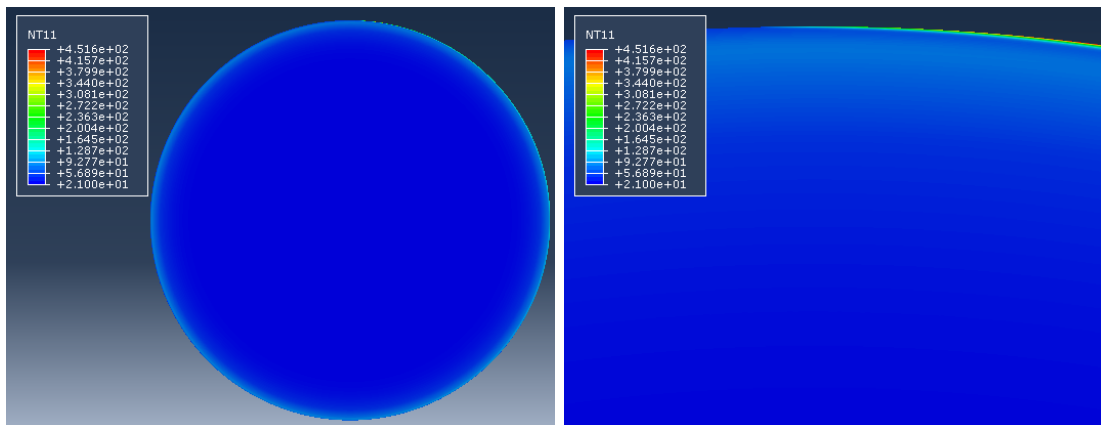
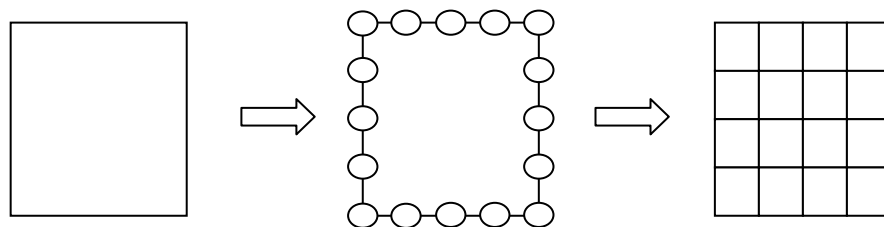


Figure 5-23: A high resolution temperature profile applied to the 2D verification model, overview (left) and detail (right)

Figure 5-23 illustrates the temperature profile applied to the 2D verification model. The strip contact is at the top of both images, with cold material visible at the surface on the left side of the detail image and being heated as it moves to the right and passes through the roll bite. The temperature data is imported from real thermal model results, to keep the model as relevant to the three-dimensional models as possible. The data was produced from a very



early round in a rolling campaign, so the majority of the roll is still at or very close to the initial temperature of 21°C. The figure highlights how shallow the volume experiencing significant angular temperature variation is. The interpolation routine described in Chapter 3 was introduced to ensure that the smoothness of the profile was maintained at higher resolutions, since step changes in temperature could seriously affect the stresses generated by thermal expansion.



**Figure 5-24: Geometry (left), seeding (middle) and meshing (right)**

To test the resolution dependence of the stress results from the 2D verification model, the model must be meshed at a number of different resolutions. The process of meshing in ABAQUS begins with "seeding", i.e. breaking each of the edges in the geometry into a collection of smaller edges. In Figure 5-24, a seed of 4 is applied to the original geometry (the seed count refers to the number of elements along each edge created by the seeding, not the number of points), with each individual point representing a break in the edge which will become the location of a boundary between two elements in the resulting mesh. As such, the resolution of a mesh can be specified by reference to the seed which is applied to each edge.

<b>Trial</b>	<b>¼ Circumferential Seed</b>	<b>Surface Radial Seed</b>
1	128	15
2	256	30
3	512	60
4	768	90

**Table 5-1: trial resolutions**

Table 5-1 gives the set of seeds used to test the resolution dependence of the stress results from the 2D verification model. The circumference was split into quarters so that the automatic meshing tools could function effectively, hence the stipulation of ¼ circumferential seed. The radial mesh spacing is constant near the surface then grows steadily wider towards the centre. The most important aspect of the radial mesh density is the density near to the surface; therefore it is this measure which is given above.

The first three tests give a steady doubling of the mesh density, while the final test takes the resolution to the maximum level possible with the available hardware.

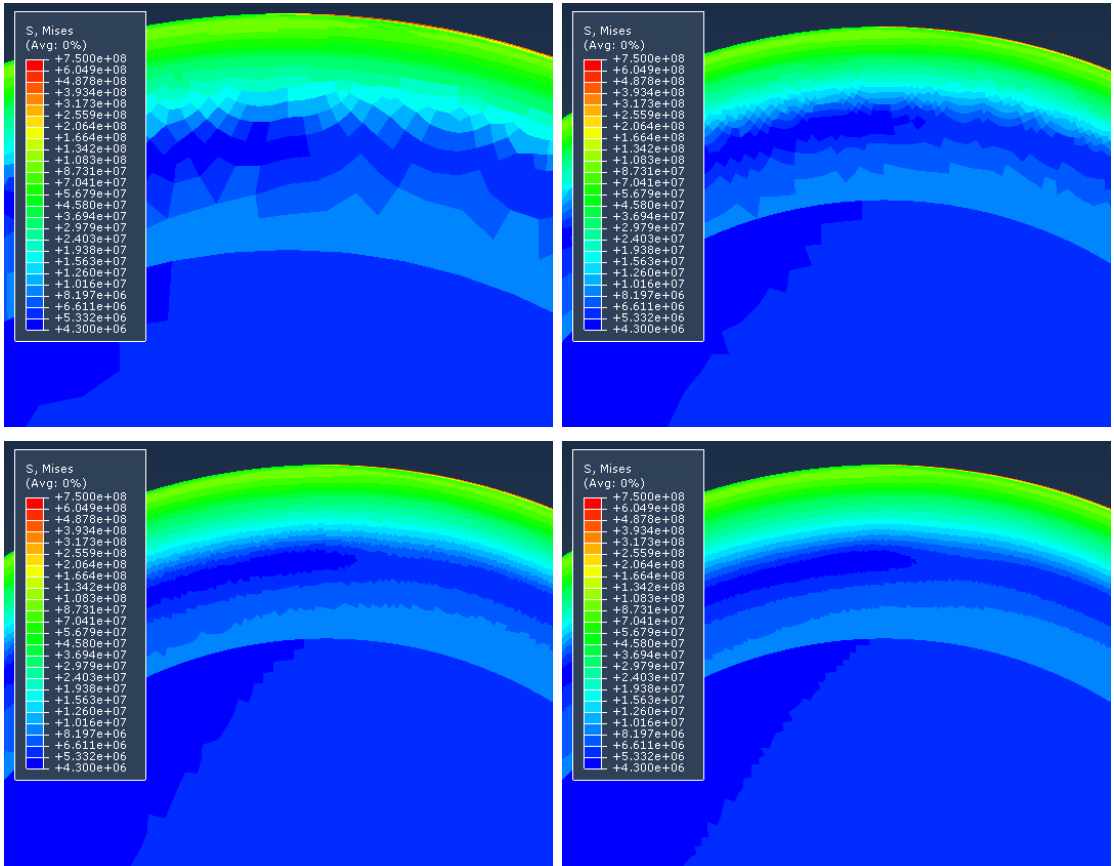


Figure 5-25: Von-Mises stress results from tests 1 (top left), 2 (top right), 3 (bottom left) and 4 (bottom right)

Figure 5-25 shows a set of results from all four tests, using the plane strain assumption (i.e. semi-infinite solid, so no deflection in the third dimension). The character of each of the results is very similar, although the resolution of test four is clearly much better than test one.

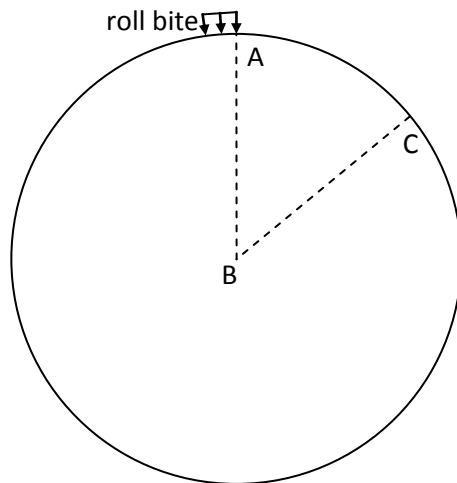


Figure 5-26: highlighted locations of radial profiles in the 2D verification model

A more precise comparison can be made by focusing on the radial variation in stress at the end of and fifty degrees after the roll bite, as indicated by lines AB and CB in Figure 5-26.

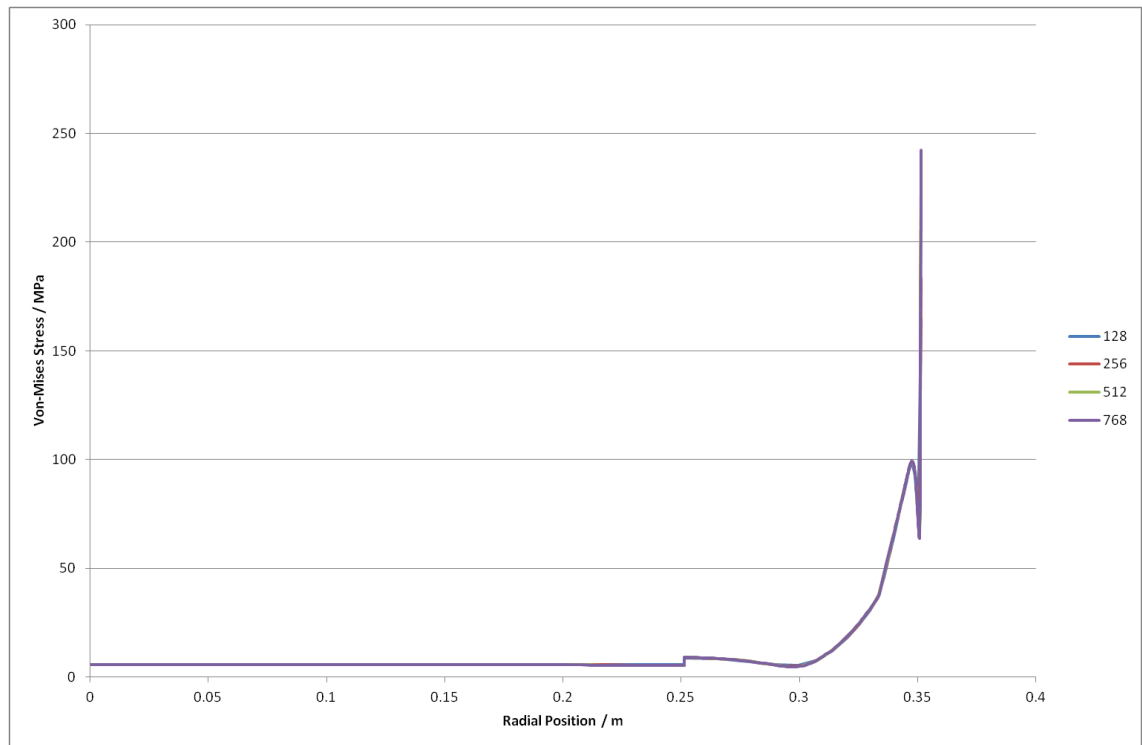


Figure 5-27: Radial stress profile below the end of the roll bite

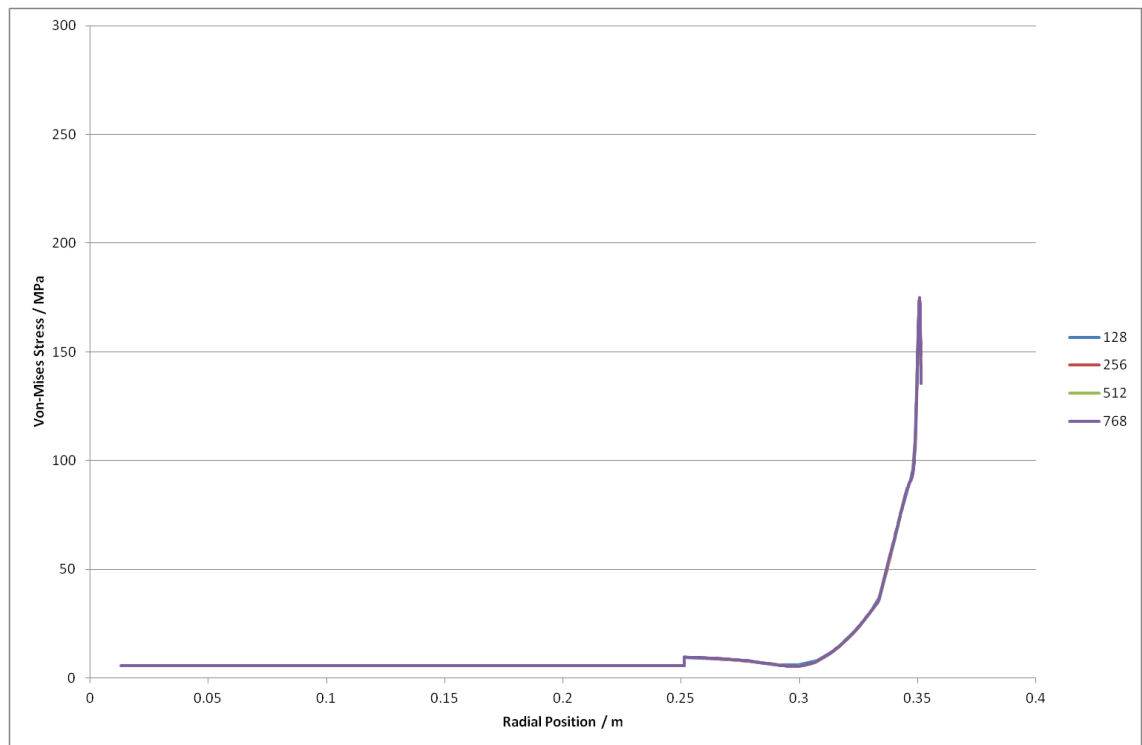
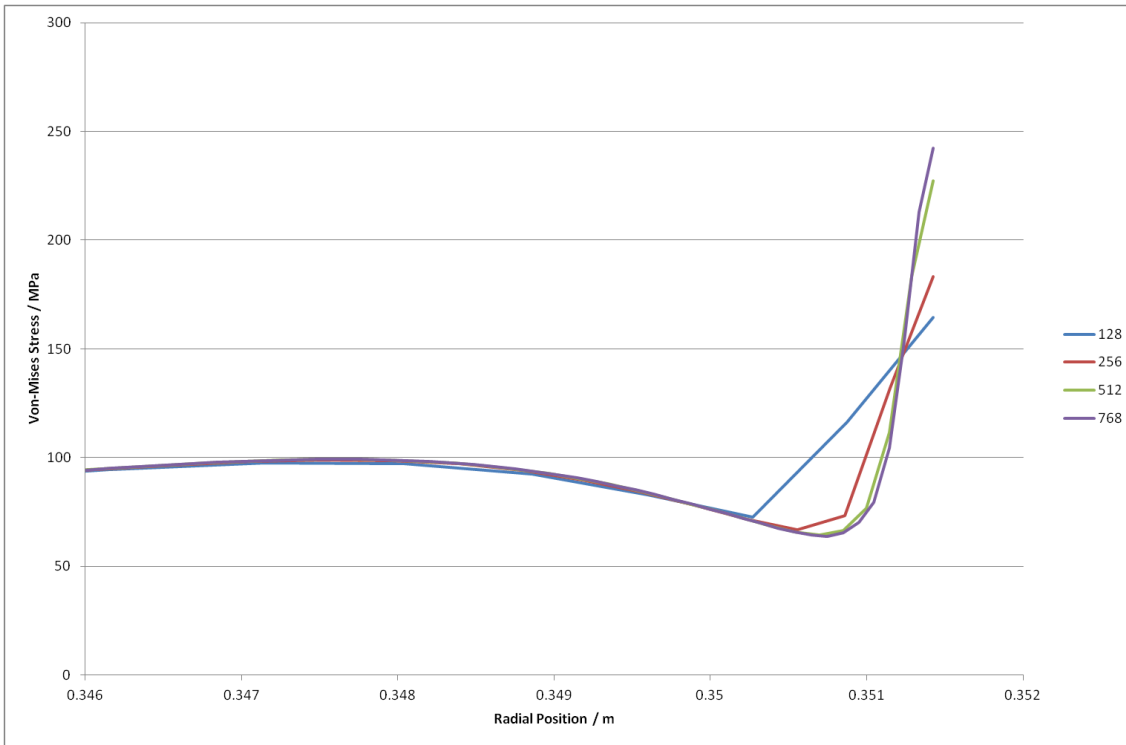
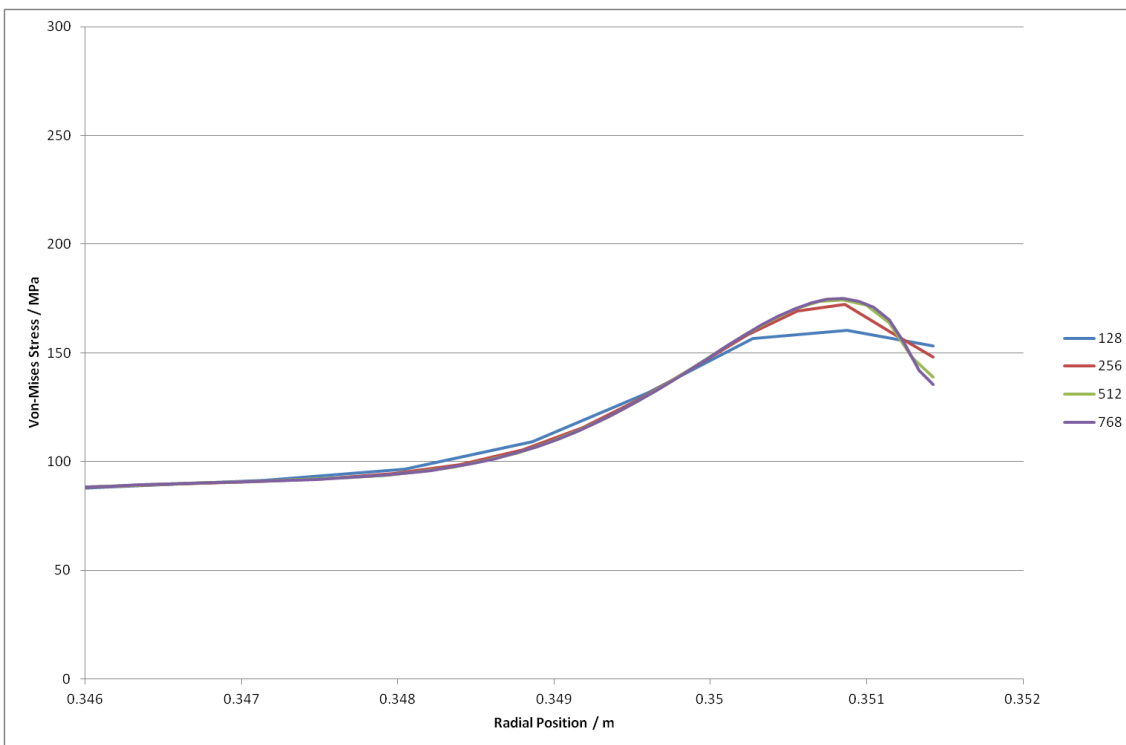


Figure 5-28: Radial stress profile fifty degrees from the end of the roll bite



**Figure 5-29: Detail view of the radial stress profile below the end of the roll bite**



**Figure 5-30: Detail view of the radial stress profile fifty degrees beyond the end of the roll bite**

From Figure 5-27 and Figure 5-28 the models all look identical, showing very little resolution dependency. At a higher level of magnification the effect of the increase in resolution can be seen with greater clarity. Figure 5-29 and Figure 5-30 show that higher

resolutions are required to reach a fully converged solution at the surface, with radial resolution being key in defining the emerging near-surface features.

### 5.3.3. Validating the 3D model

With a stable result from the 2D verification model, different forms of the three dimensional models can be compared.

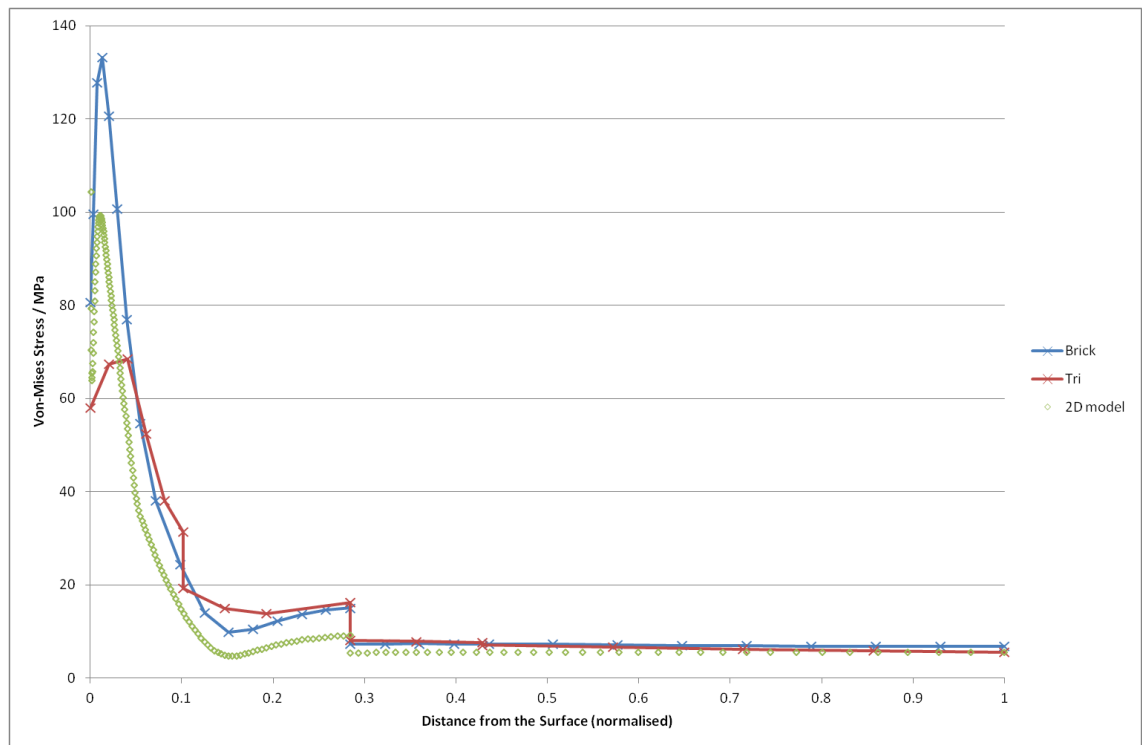


Figure 5-31: Radial stress profiles fifty degrees beyond the end of the roll bite using different element shapes

Figure 5-31 compares the two dimensional model against two potential three dimensional model meshes, each using a different set of element shapes. Cross-sections of the two three dimensional meshes are shown in Figure 5-32, with the two-dimensional verification model mesh presented for comparison. The model entitled "brick" uses rectangular sectioned elements throughout the model, making use of the element shape's greater accuracy but suffering some mesh distortion from forcing the rectangular elements to conform to the circular roll shape. The model entitled "tri" uses wedge-shaped elements, prisms made of two triangular faces separated by three rectangular faces. The triangular profile makes these elements very efficient for rapidly increasing or decreasing mesh size and conforming to complicated geometries.

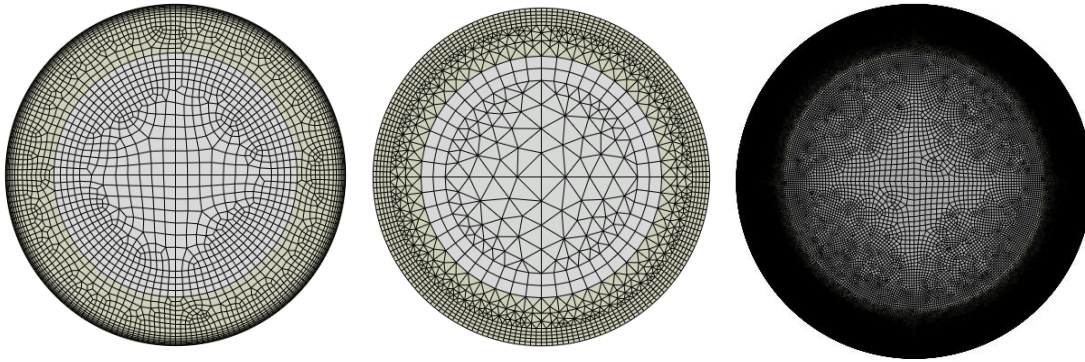


Figure 5-32: mesh cross-sections for the brick (left) and tri (middle) three-dimensional models, compared to the two-dimensional validation model (right)

Figure 5-31 shows that neither model perfectly replicates the two dimensional model's results, but that the "brick" model at least presents a similar character. Figure 5-33 and Figure 5-34 show individual stress components to further illustrate the better shape of the "brick" model results.

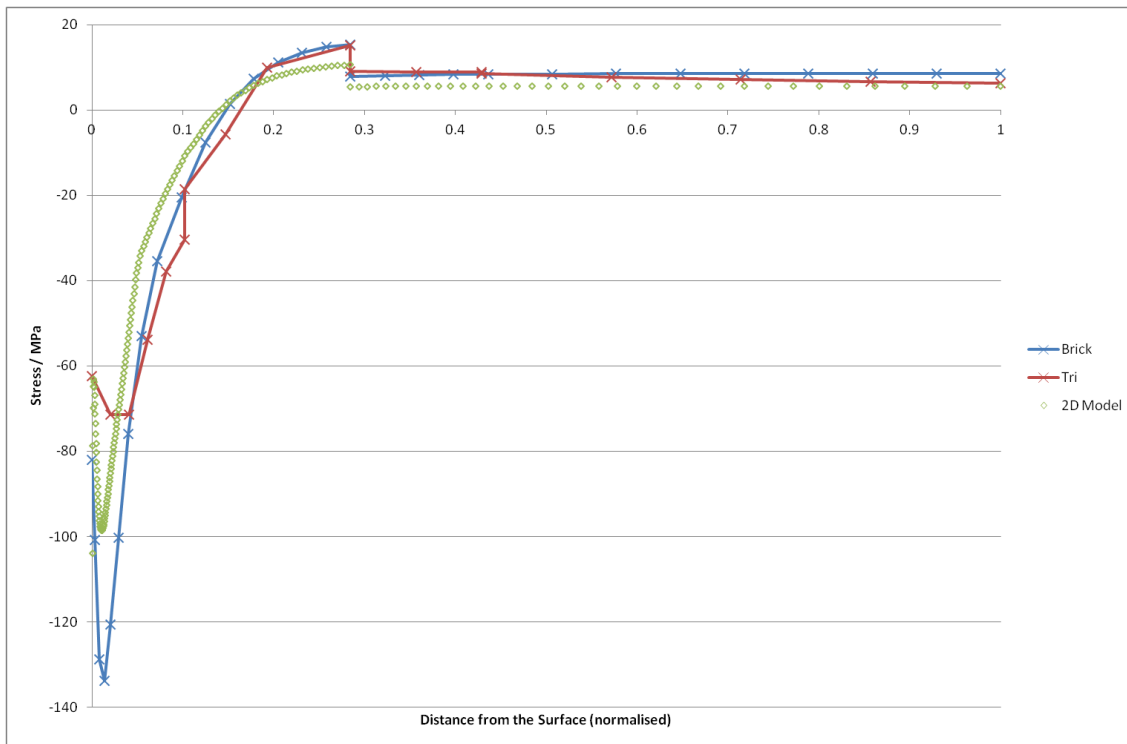


Figure 5-33: Hoop stress profiles fifty degrees from the end of the roll bite using different element shapes

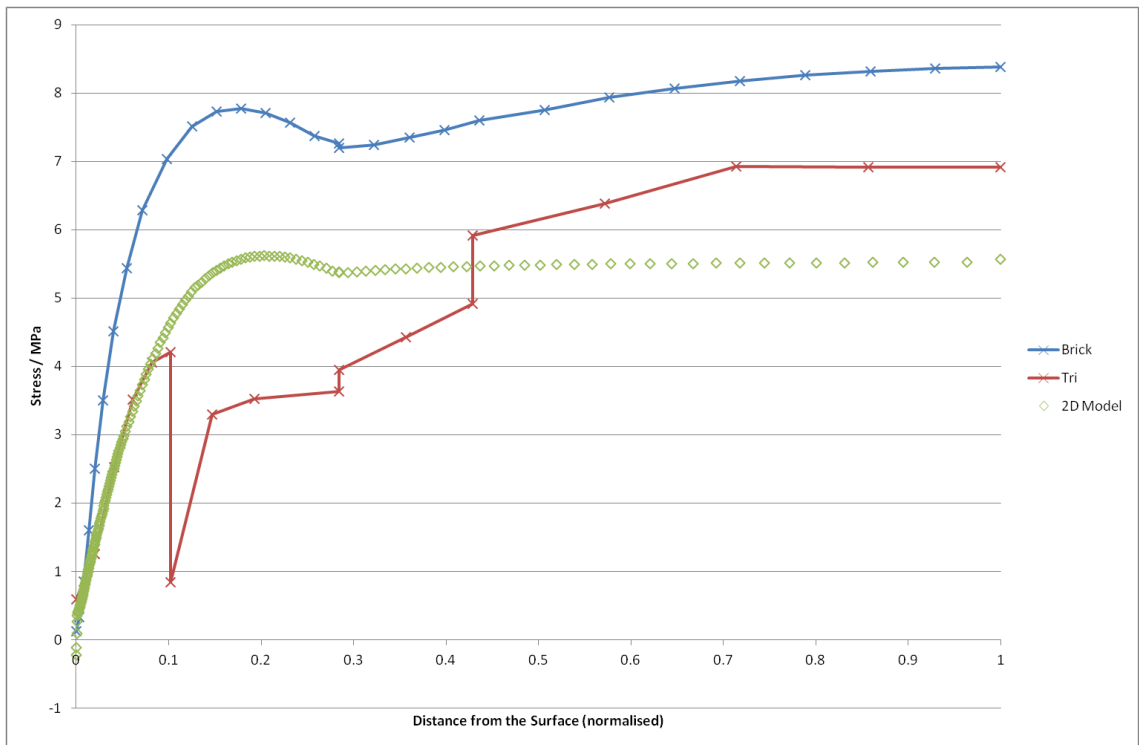


Figure 5-34: Radial stress profiles fifty degrees from the end of the roll bite using different element shapes

While the previous figures show the "brick" mesh to be the superior, the difference between the brick and two dimensional models is large enough to require some explanation. Figure 5-35 provides an insight into why the two dimensional model could not accurately predict the stress profile of the three dimensional model.

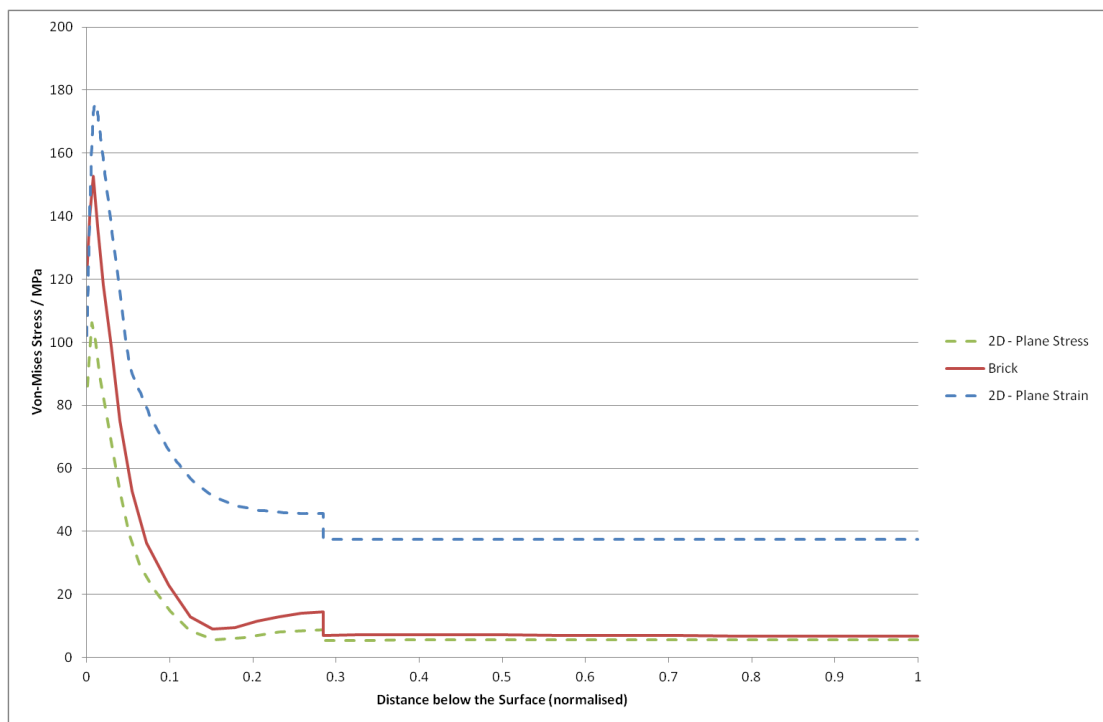
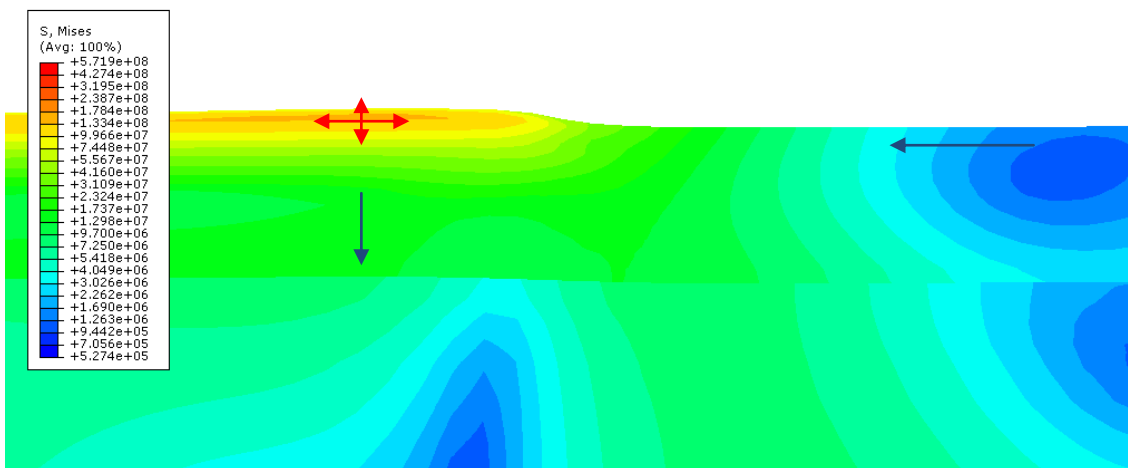


Figure 5-35: The difference between using a plane stress and plane strain assumption

The difference between the two two-dimensional results is due to a difference in the simplifying assumption used to allow the third dimension to be ignored. The "plane strain" model assumes that there is no deflection at all in the third dimension. This assumption simulates a slice from an infinitely long solid, i.e. as much pressure to expand or contract as is being applied to any one point in the model will be balanced by identical pressures along the entire third dimension. This creates a highly restrained system, where any pressure build up can only be relieved by deflection in two of the three dimensions, hence the higher stresses. The "plane stress" model assumes that there is no stress at all in the third dimension. This assumption simulates an infinitesimally thin model, where no stress can build up in the third dimension as there is no surrounding material to constrict any element's ability to expand. This creates the antithesis of the plane strain system, hence the lower stresses.

The three dimensional model lies unevenly between the two, closer to the plane stress result near the centre and nearer the plane strain result near the surface. This behaviour can be understood by considering Figure 5-36.



**Figure 5-36: Radial-Axial view of Von-Mises stress and deflected shape at the work roll surface**

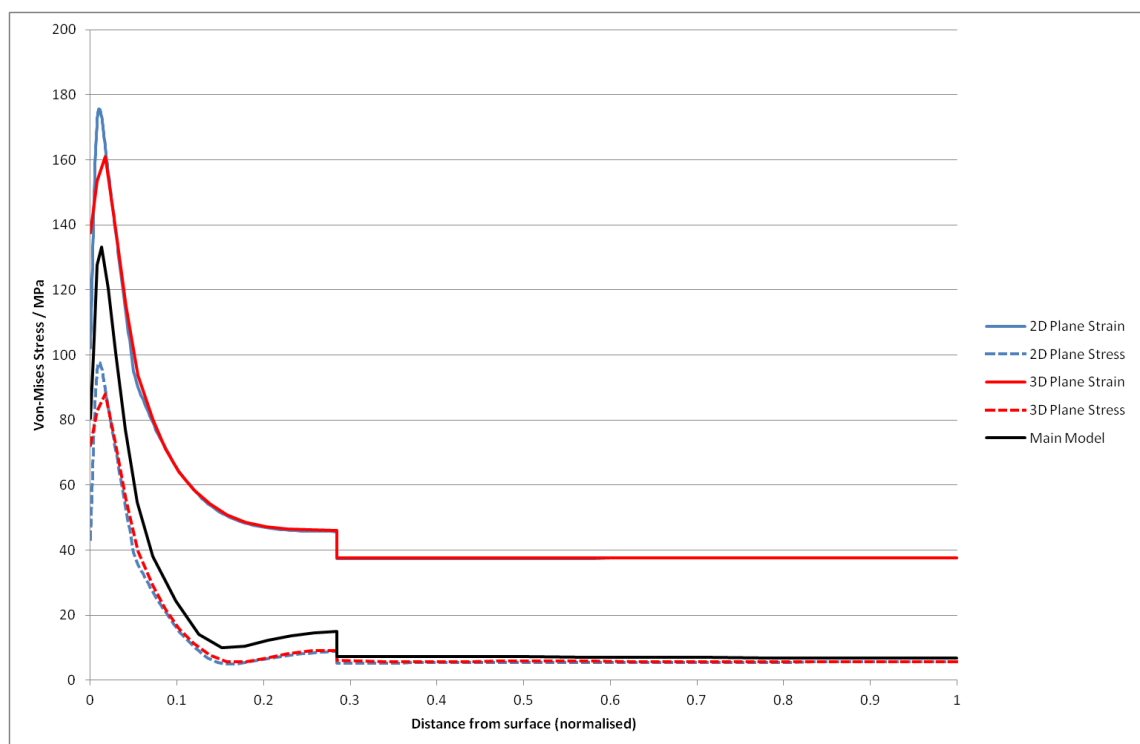
Figure 5-36 shows the stresses below the surface of the work roll. The left side of the image is within the strip width and the right of the image is outside the strip width. The angle of the roll at this position is several degrees beyond the roll bite exit, hence the sub-surface stress peak. The step change in stress which runs below and approximately perpendicular to the surface is caused by a difference in material properties between the hardened shell and softer core material.

In the situation given in Figure 5-36, the material near the surface and within the strip width is very warm and hence tends to expand. Below the surface the material is colder, so tends to pull the surface material towards the roll centre line. The two-dimensional model is perfectly capable of modelling this kind of behaviour. The axial expansion of the strip of hot



elements is also restrained by the colder core material, which is not driven to expand as far as the surface. The cold core and barrel ends act together as a clamp to restrain the axial expansion, resulting in a surface bulge as the less restrained radial direction accommodates the pressure to expand. The two-dimensional assumption that best fits this scenario is plane strain, since the ability of the surface material to expand is restrained.

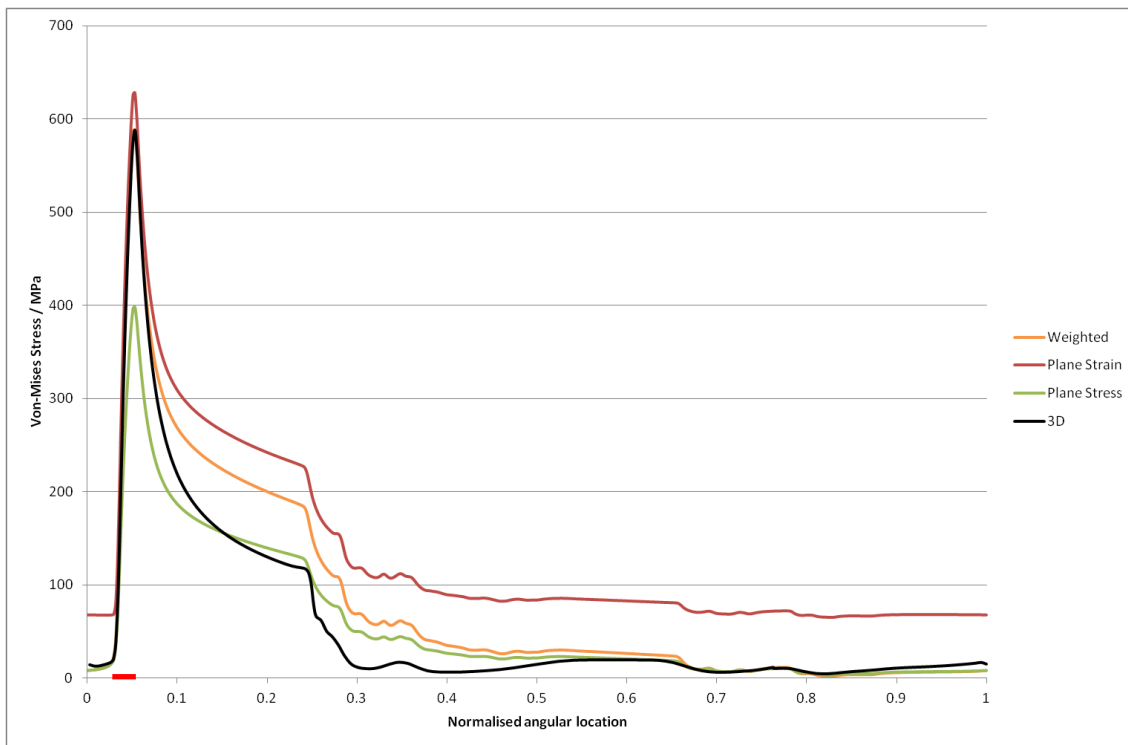
Nearer the centre of the work roll the temperatures in the axial direction are relatively uniform. There is a little influence from the high temperature elements at the surface, but there are many more cool elements than warm elements so the effect is diluted. At this location the appropriate two-dimensional assumption would be plane stress, since the whole core is expanding or contracting together with relatively little restriction.



**Figure 5-37: Comparison of 2D models and their 3D equivalents**

The previous discussion explained why the results of the three dimensional model might not agree perfectly with the results of either the plane strain or plane stress two-dimensional model. To test this explanation and ensure that the three-dimensional model's elements were not simply misbehaving, the three dimensional model was adjusted to reflect the 2D simplifying assumptions. To reflect plane stress a single element thickness model was created, called 3D plane stress. 3D plane stress was modelled on the original three dimensional model, representing a radial-circumferential cross-section through the centre plane. Next, to reflect the plane strain assumption, a restraint condition was applied to the whole of the original three dimensional model, preventing any axial movement.

Figure 5-37 shows that these assumptions alone were enough to nearly perfectly replicate the two-dimensional model results. This discounts the possibility that an unknown error in either model was distorting the results or that the three dimensional elements were simply behaving inappropriately. The figure does, however, confirm that higher resolutions than the original three dimensional model was capable of are required to accurately define the stress profile near the surface.



**Figure 5-38: Surface stresses using a weighted plane strain assumption. The location of the roll bite is indicated by the thick red line on the abscissa.**

If a two-dimensional model could be used to accurately predict the stress profile at the centre plane of the work roll, such a model could provide valuable information very quickly. Therefore it was worth checking that a weighted average approach might not yield a more useful result. The theory, suggested by Jan De-Roo of the IJmuiden RD&T department, was that the temperatures used with the plane strain assumption should be offset by the average temperature of the whole centre plane. This reflects that similar temperatures are experienced along the axis and that the roll is not semi-infinite, so will not be fully restrained at the ends. The stresses induced by this scheme will be tensile where the temperature is colder than the average and compressive where the temperature is warmer.

Figure 5-38 shows the results of this scheme, alongside the results from the three dimensional model and the original plane stress and plane strain assumptions. The coherence is certainly much better, with the peak stress matching with the three dimensional model and

the minimum stress coming very close (although matching better with the plane stress model than with the three dimensional model). The largest departure is after the roll bite and at the beginning of the first cooling zone.

Depending on the level of accuracy or the amount of information required, Figure 5-38 shows that a suitably adjusted plane strain model could be used as a stand-in for a fully three dimensional model.

#### 5.3.4. Locating sub-model boundaries

The previous sections showed that the three dimensional model resolution is insufficient near the surface, but that the requirement for resolution decreases below the surface, making sub-modelling a potential solution. The sub-model validation section showed that the sub-modelling technique is valid for sharp thermal features as long as the results are stable at the sub-model boundary condition. Therefore, the first sub-model (the surface model) must cover the surface of the work roll, down to a depth,  $d$ , below the surface where the original three-dimensional model (the global model) gives accurate results.

Figure 5-39 shows a set of radial von-Mises stress results from a collection of global models using different mesh densities, with the roll surface on the left side of the figure and the roll centre on the right side. The series names reflect the number of elements around the circumference at the surface. There is a compromise to be made in choosing the depth of the sub-model boundary condition. The nearer the boundary condition is to the work roll surface, the greater mesh density and therefore higher resolution can be provided at the surface. However, the further the boundary condition is from the work roll surface, the lower the global model resolution can be, and therefore the shorter the global model run time.

$d$  was chosen to be 5cm, which is the same as the assumed thickness of the work roll's shell material<sup>34</sup>. The results of all three models from Figure 5-39 were well defined at this depth, yet all diverged after a short distance. Furthermore, an internal surface will always exist at this radius, so the deflection values used for the sub-model boundary condition will never suffer from interpolation errors.

Figure 5-40 illustrates the chosen pairing of global model and surface model, using the same format as Figure 4-39. The 96 element model was used to provide the boundary conditions. The surface model agrees very well with the high resolution global model, and provides a much smoother, denser mesh while taking 45 minutes to run, rather than an hour.

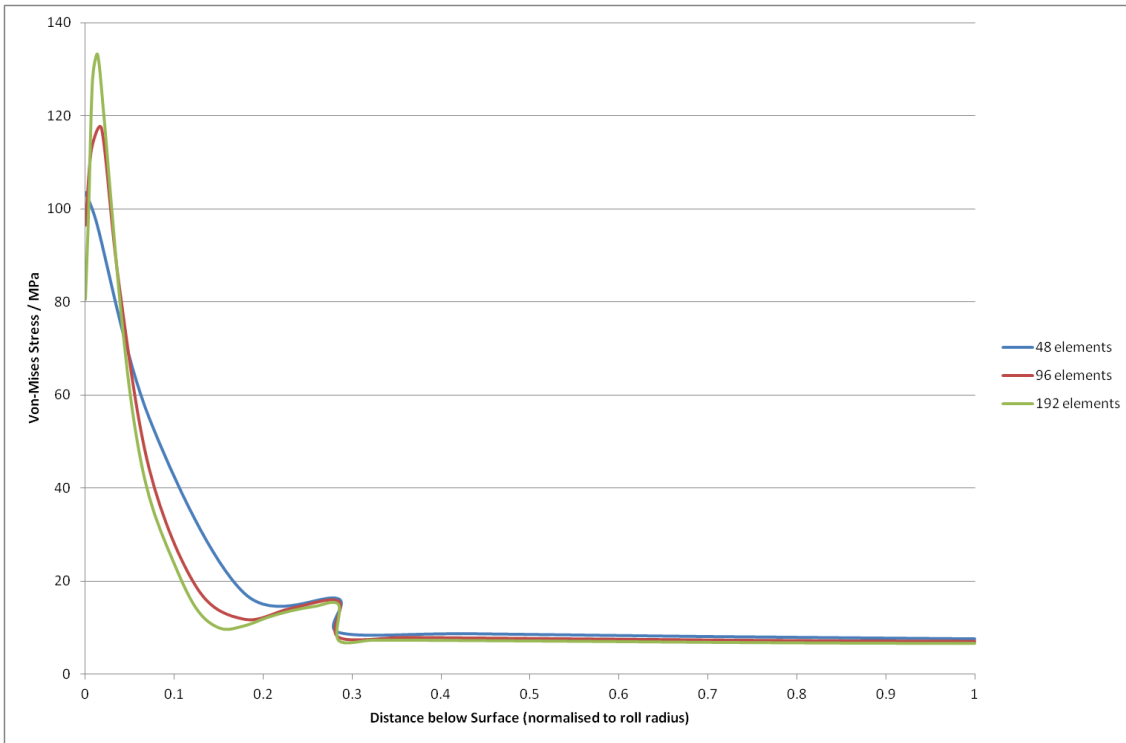


Figure 5-39: Radial variation of Von-Mises Stress for global models of varying resolution

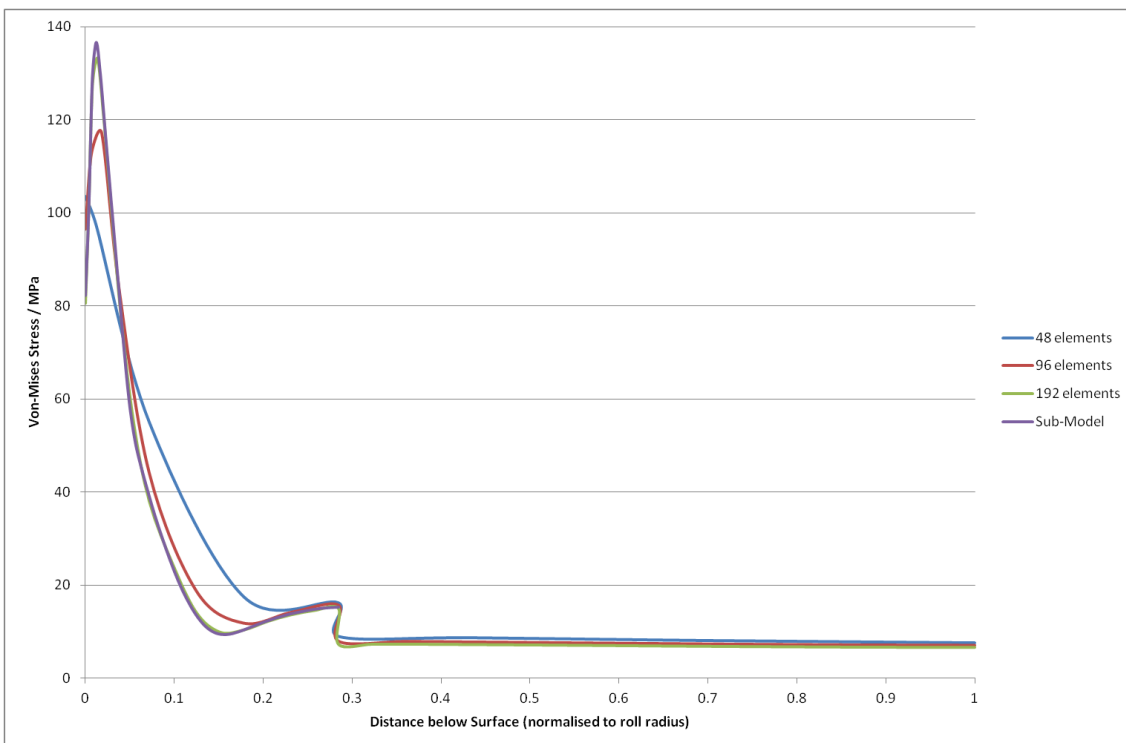
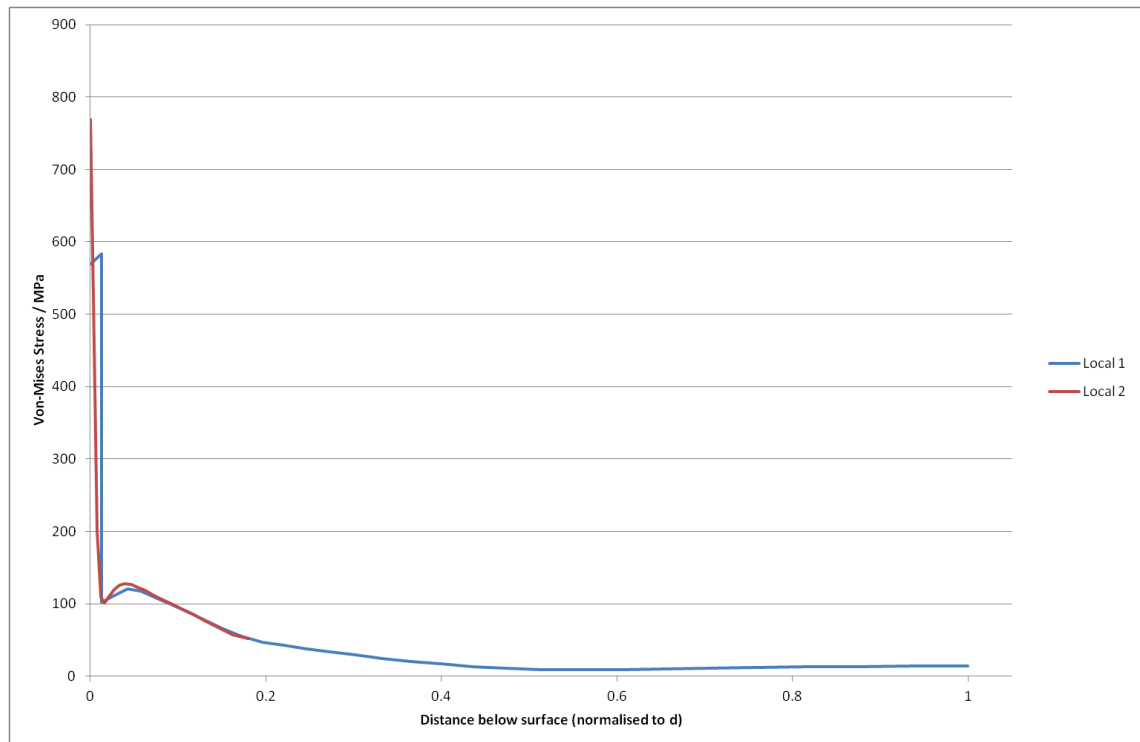


Figure 5-40: Radial variation of Von-Mises Stress for the surface model, compared to global models of various resolutions



**Figure 5-41: Radial variation of Von-Mises Stress for the bite model, compared to the surface model**

Very near to the roll bite, the temperatures change so quickly that a very fine mesh is required to accurately predict their effect on the stresses. A second sub-model (the bite model) was introduced to account for those areas that the surface model cannot quite define. The bite model concentrates on the area around the roll bite, its narrow focus allowing for a very high resolution. Figure 5-41 shows the effect of adding the bite model, where local1 gives the radial stress profile from the first sub-model and local2 gives the radial stress profile from the second sub-model. The local2 results give a much smoother stress distribution and a well defined peak at the surface.

Together, the three models provide an effective level of accuracy that hardware constraints would render impossible in a single model and for a shorter run-time, as shown in Table 5-2 below. The "radial mesh density" column gives a measure of the radial mesh density at the surface of the model, normalised to the density of the global model, e.g. adding one local model provides a 6.1 times increase in radial mesh density at the surface. For reference, the computer used to generate the data in Table 5-2 was a Toshiba Tecra M11-11M, running Windows 7 Professional 32-bit, with 4GB ram and a 2.67GHz Intel Core i7 Processor.

Model type	Runtime	radial mesh density
High-res 3D model	62.5 mins	1
Med-res 3D + Local model	44.9 mins	6.1
Med-res 3D + 2 local models	63.5 mins	16.8

Table 5-2: Changes in run-time and effective resolution of different model combinations

### 5.3.5. Mechanical loads

Since the beginning of the project, conventional wisdom has always stated that the effect of thermal loading outweighs the effect of mechanical loading at the roll surface. However, to be sure that no important factors are being overlooked and to get a full idea of sub-surface stress behaviour, representative mechanical loads were also implemented. An additional benefit is that comparisons can be made between the thermal and mechanical effects to validate the view that mechanical stresses are insignificant at the surface.

#### 1. Normal load

---

The size of the roll bite is readily available from the hot mill database and is used in the thermal models. Hertzian geometry for a line contact can be used to estimate the size of the contact patch between the work roll and backup roll.

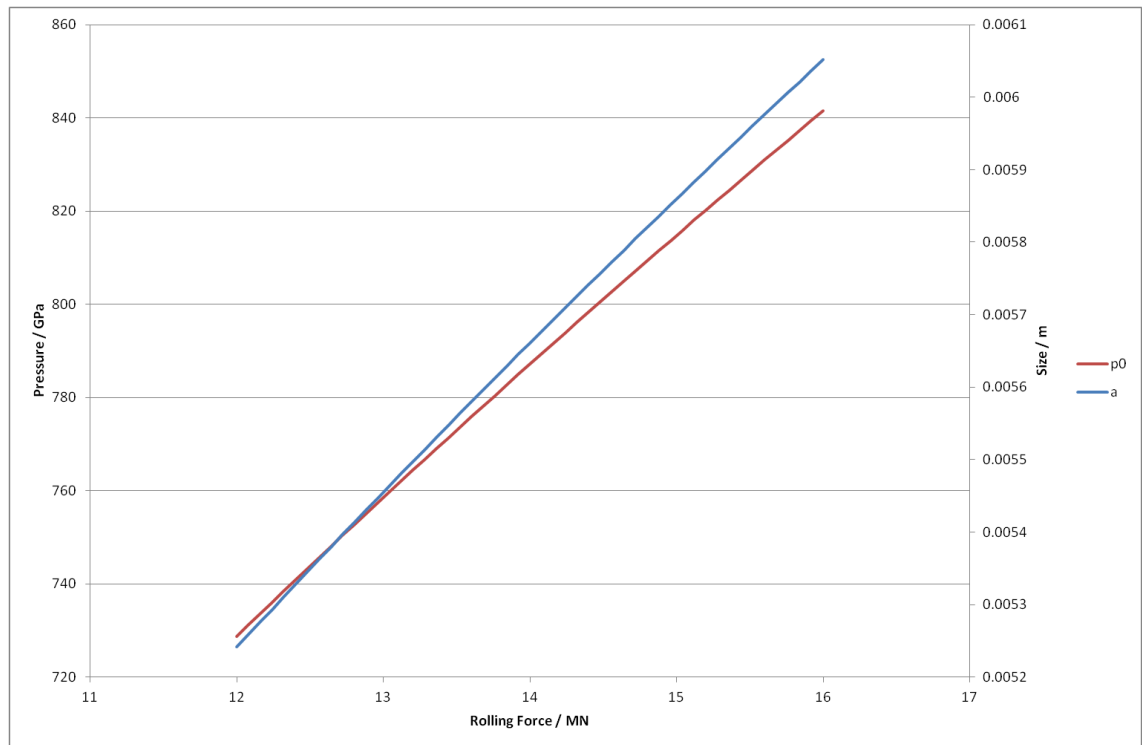
$$a = \sqrt{\frac{8 R w'}{\pi E'}} \quad p_0 = \sqrt{\frac{E' w'}{2 \pi R}} \quad \text{Equation 5-2}$$

Where  $a$  is half the contact width,  $R$  is the contact radius,  $w'$  is the load per unit length and  $E'$  is the combined Young's Modulus of the two contacting materials.  $E'$ ,  $R$  and  $w'$  can be attained using the following formulae.

$$\frac{2}{E'} = \frac{1 - \nu^2}{E_1} + \frac{1 - \nu^2}{E_2} \quad w' = \frac{w}{L}$$

$$R = \frac{R_1 R_2}{R_1 + R_2} \quad \text{Equation 5-3}$$

Where  $R_1$  and  $R_2$  are the work roll and backup roll,  $E_1$  and  $E_2$  are the Young's Modulus of the work roll and backup roll,  $w$  is the total load and  $L$  is the length of the contact between the work roll and backup roll.

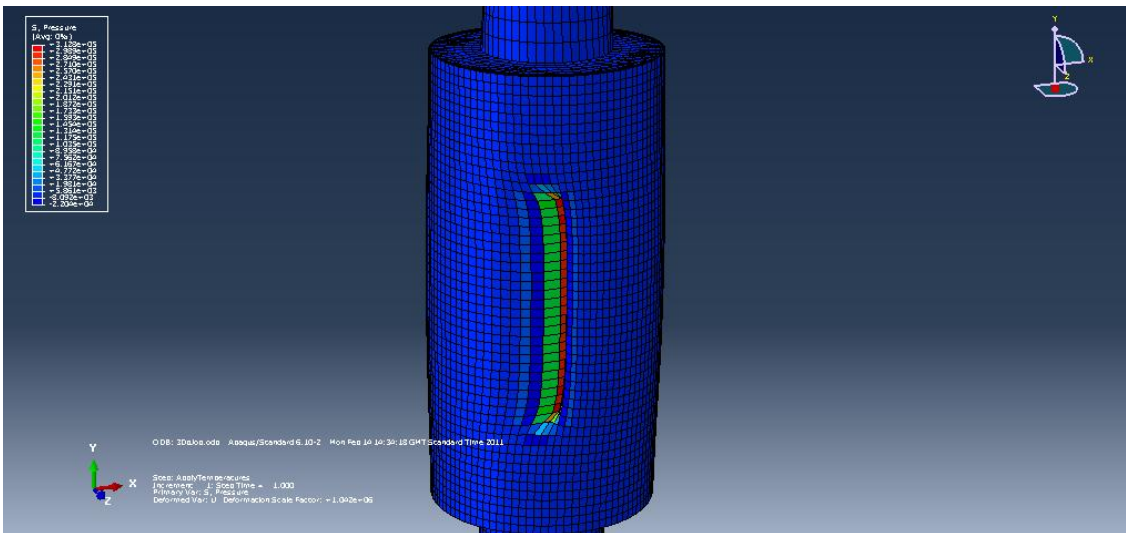


**Figure 5-42: The effect of rolling force on pressure and contact dimensions**

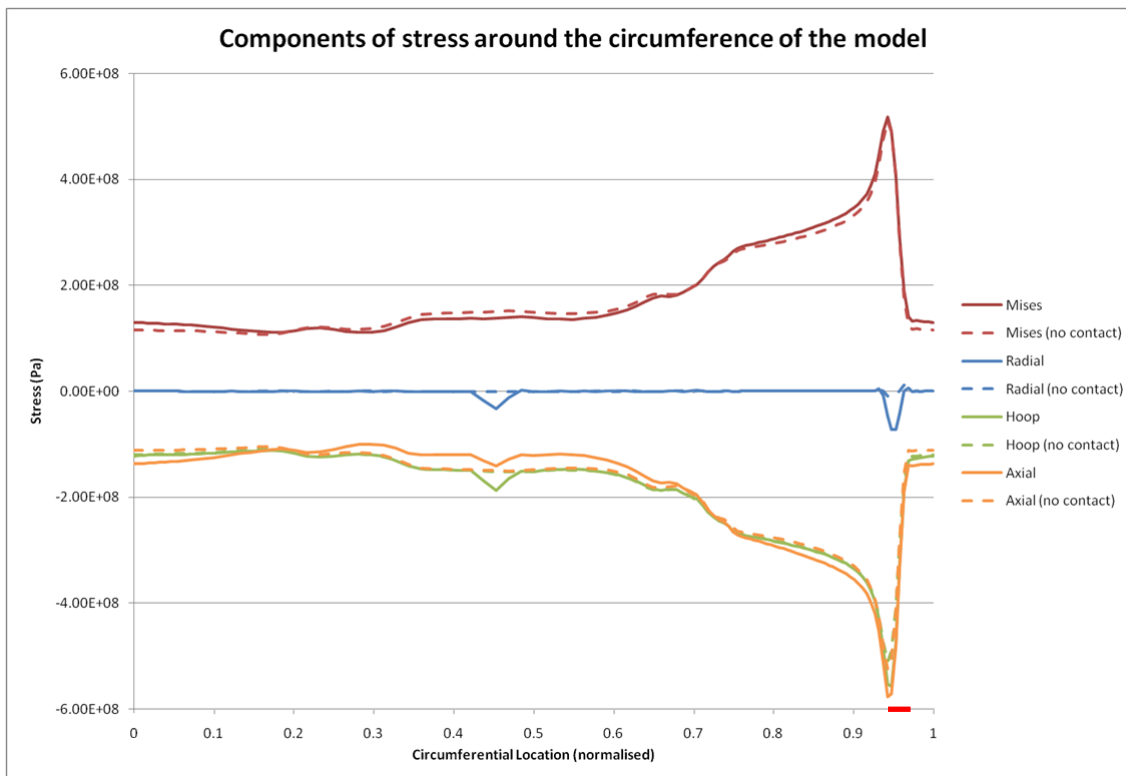
Figure 5-42 was created using standard values of 0.33m for  $R_1$ , 0.68m for  $R_2$ , 225GPa for  $E_1$ , 225GPa for  $E_2$ , a contact length of 2m and a range of loads varying from 12MN to 16MN. This covers the range of rolling loads typically experienced in the hot mill at Port Talbot.

The surface element are roughly 0.022m long in the global model, 0.011m in the surface model and 0.0055m in the bite model. Since  $a$  is only half the contact width, the figure gives a range of roughly 0.01-0.012m for the contact size. This does not give many elements in the contact area, meaning that results at the surface are likely to be inaccurate. However, if thermal loads really do dominate over mechanical loads then the effects of any errors should be negligible. Furthermore, below the surface, where the mechanical loads are more likely to dominate, the effects will be diffuse and therefore less demanding on resolution.

The additional benefits of using a Hertzian distribution are shown in Figure 5-43. When the resolution is small compared to the contact size then the roll bite and the backup roll contact patch might not necessarily line up. Any slight offset can result in the rotation shown in the figure below. Using a Hertzian pressure profile, the effects of any outlying points are greatly reduced compared to those at the centre of the contact patch. This removes any threat of off-centre forces applying unintended rotation to the work roll.



**Figure 5-43: Stress (false colour) and deformation of the work roll using constant pressure distributions**



**Figure 5-44: Surface stress profiles with and without normal loading, normalised to one complete roll revolution. Roll bite location is indicated by the thick red line on the abscissa**

Figure 5-44 shows the effect of adding normal mechanical loads to the thermal stress profile. The conventional view that the surface stresses are dominated by thermal effects appears to be justified. While there are noticeable differences, the range and general shape of the results are largely unaltered. NB – The direction of rotation in this figure is from right to left.



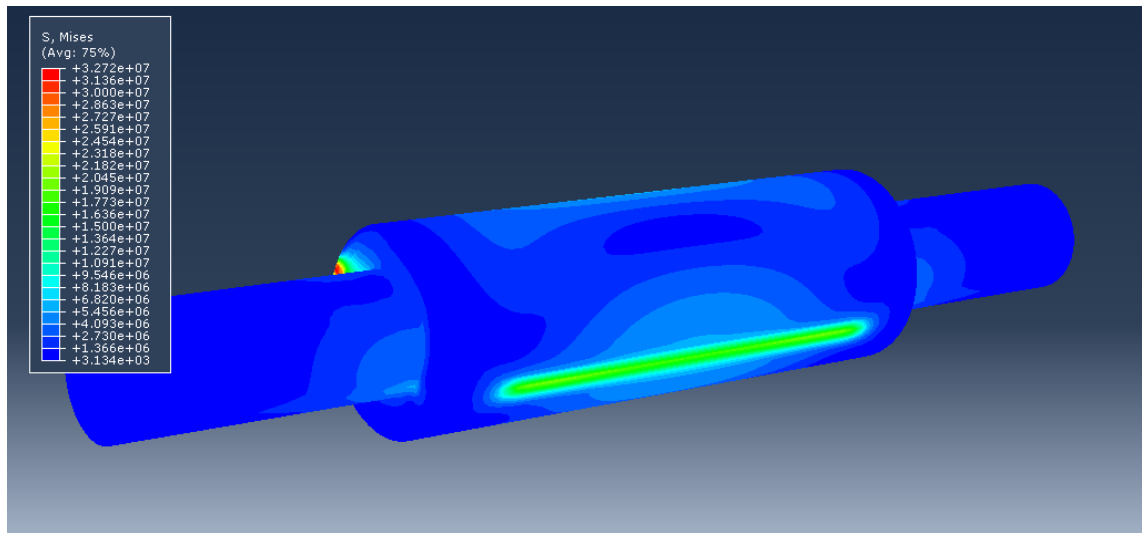


Figure 5-45: False colour Von-Mises stress plot of a purely mechanically loaded roll – strip contact patch

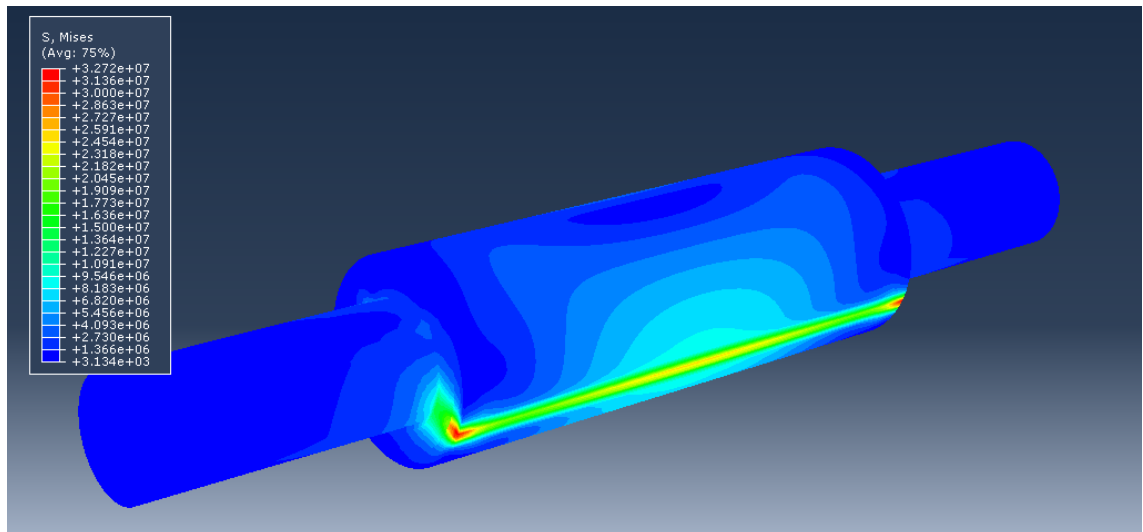


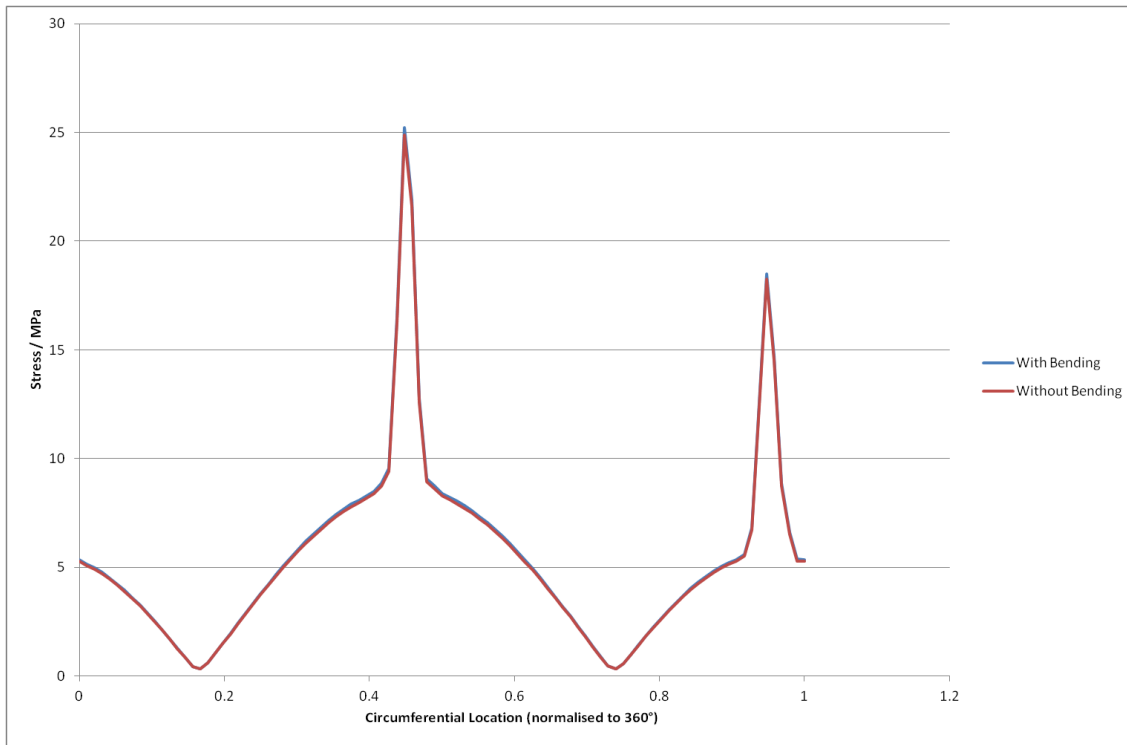
Figure 5-46: False colour Von-Mises stress plot of a purely mechanically loaded roll – backup roll contact patch

Figure 5-45 and Figure 5-46 show the effects of a purely normal load on the Von-Mises Stress state of the roll. In normal work roll geometry the barrel edges are chamfered, to prevent the stress raising effects of the barrel edge from causing cracks to develop. The peak stresses on the simplified roll geometry in Figure 5-46 show the value of this practice. The only other anomalies are some slight stress concentrations between the barrel and journals. More realistic modelling of the journals could have been implemented, but roll failure developing in this area is rare, and mechanisms for failure are either not related to the work roll (i.e. bearing failure) or derive from Stress-Corrosion cracking. It was judged that these failure mechanisms are beyond the scope of this project, and the anomalous results are far enough from the areas of interest (at and near the barrel surface) to have no significant influence.

## 2. Bending

---

The process of applying bending loads to a work roll is discussed in section 0, page 203.



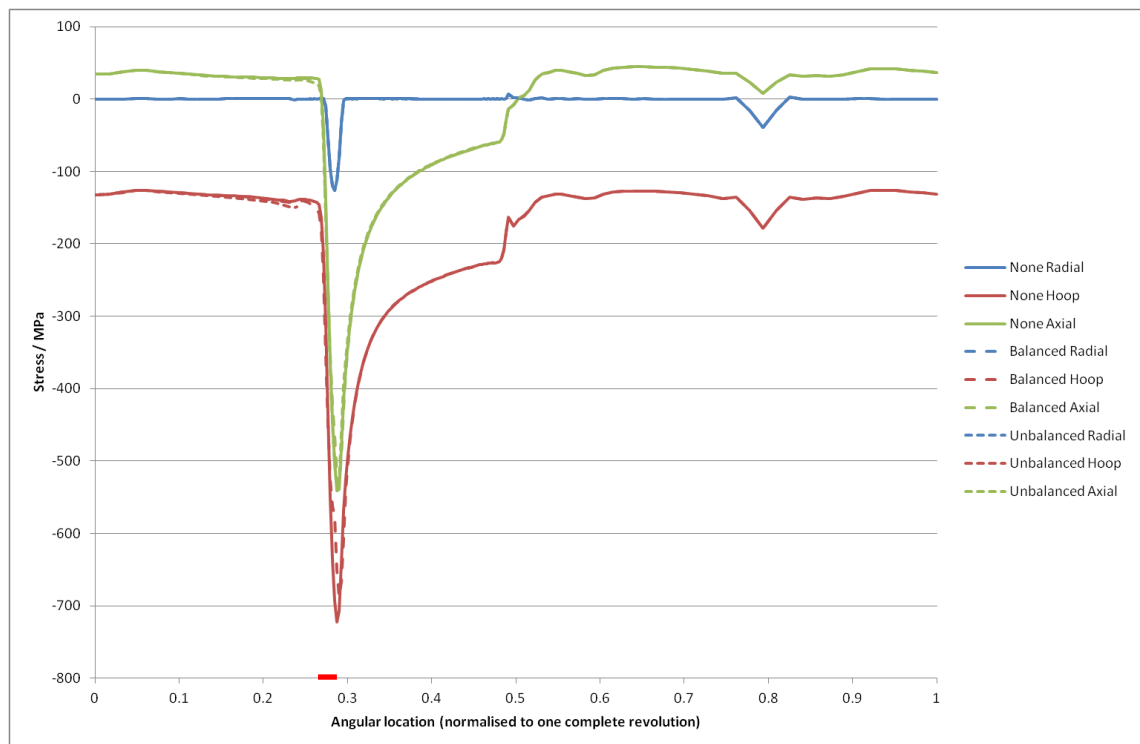
**Figure 5-47: Von-Mises Stress at the work roll surface**

Figure 5-47 shows the minimal effect of including work roll bending in the mechanical loading setup, using process data from a typical rolling schedule. The figure shows that the effects of bending on the work roll surface stress distribution are negligible.

## 3. Shear

---

Figure 5-48 compares the radial, hoop and axial stress components of three models which only differ in the shear stress distribution applied within the roll bite. Each distribution represents an extreme scenario, no shear stress, balanced stresses acting towards the centre of the roll bite or unbalanced shear stress acting entirely in one direction, as described in Chapter 4.



**Figure 5-48: Stress components at the surface under different shear stress profiles, roll bite is indicated by thick red line on abscissa**

The figure shows that, even in such extreme scenarios, the shear stress at the surface makes little difference to the overall stress profile.

### 5.3.6. Residual stresses

As mentioned in Chapter 4, the residual stresses are assigned based on the best currently available data. There are six components of stress, three normal components and three shear components. However, only one component,  $\sigma_{22}$ , is known, so ABAQUS is used to perform an iterative process of balancing the stresses to give an overall internal equilibrium<sup>i</sup>. A number of tests are required to ensure that this process is working correctly, the first of which shows that the applied components are acting in the desired direction.

---

<sup>i</sup> The technique was recommended by Prof David Smith of Bristol University, having been used successfully in the past by members of his department.

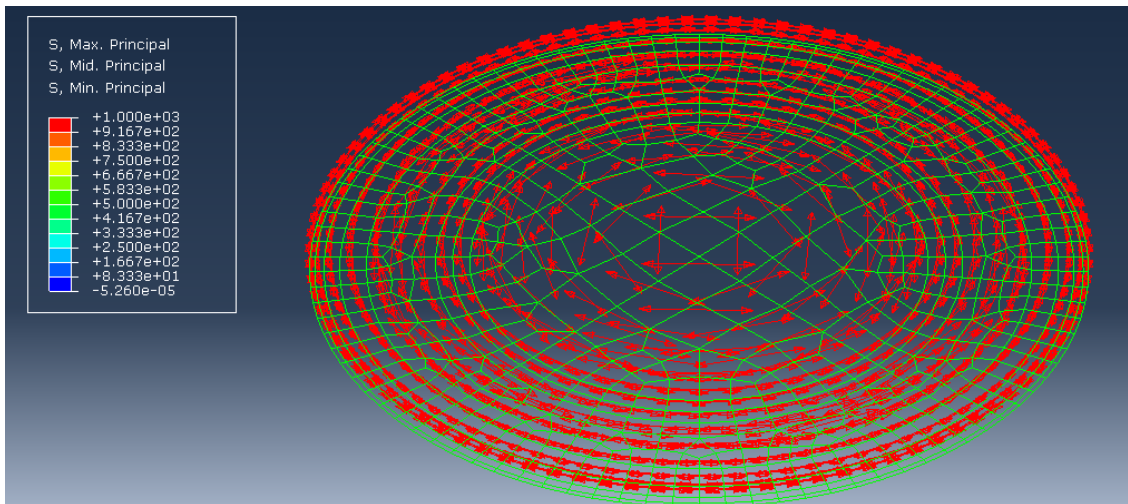


Figure 5-49: Vector plot of the hoop stresses (isometric projection)

Figure 5-49 shows a vector plot of the stresses in the work roll with  $\sigma_{22} = 1kPa$  applied throughout. This hoop stress was applied using the subroutine described in Chapter 4 (Page 204). Each of the arrows is pointing in the appropriate direction and all of them are the same length and colour, signifying that the sense and magnitude of the stress distribution has been applied correctly.

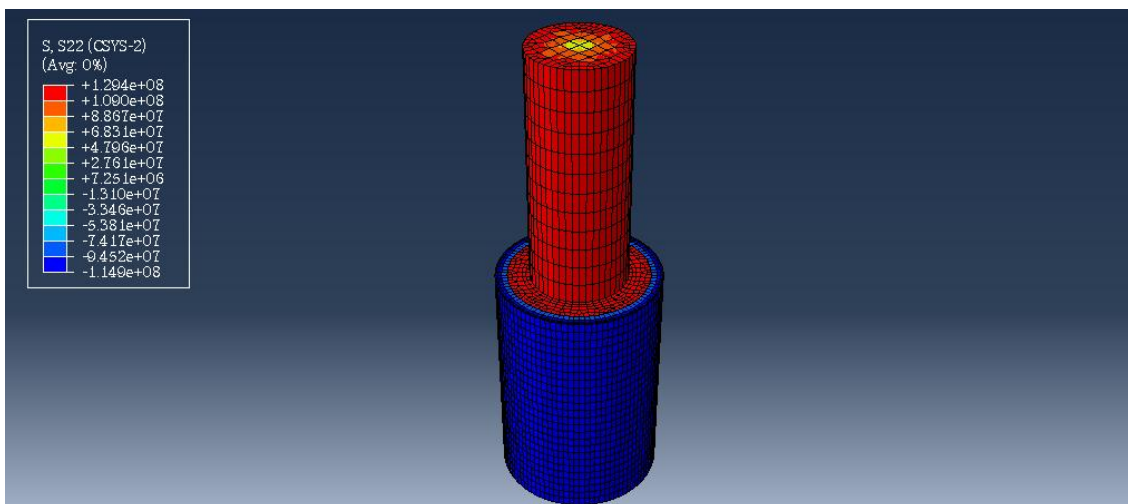


Figure 5-50: Applied stress distribution

Figure 5-50 shows a false colour plot of the hoop stress based on the residual stress data in Figure 4-49. The data does not fit exactly to the currently depicted work roll profile, the work rolls for the F6 stand are solid cylinders, not hollow cylinders. The main features of the residuals stress profile have been scaled to fit the new size as a best approximation, using the interface between the shell and core material as the interface between the compressive and tensile portions.

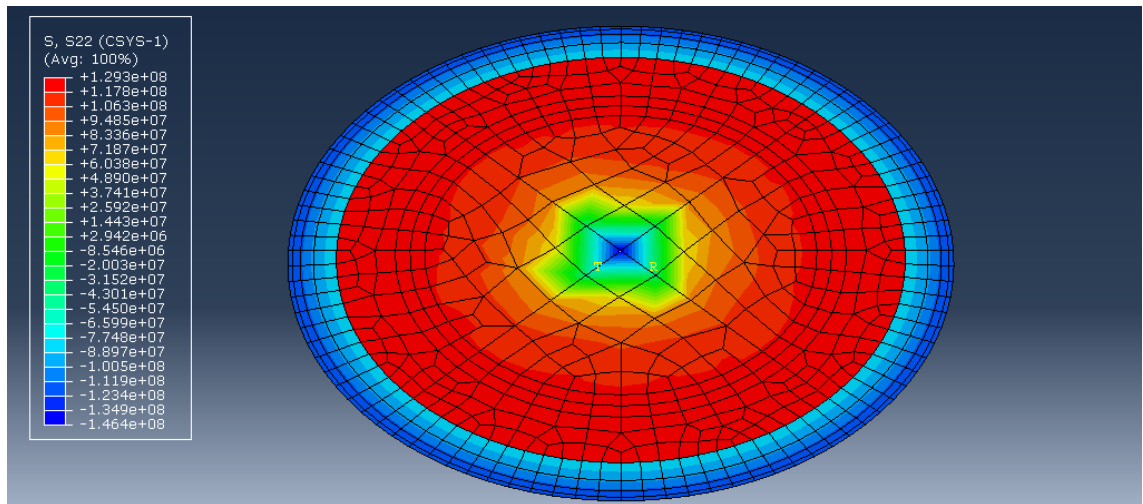


Figure 5-51: Balanced stress distribution (isometric projection)

While ABAQUS will provide estimates of the required stress component values to provide equilibrium, it will also alter the initial values. This is why the process must be iterative, saving the unknown values but resetting the known values and repeating until the known values remain the same. Unfortunately the hoop stress profile never fully converges on a comfortable solution, constantly seeking a very low stress in the centre, as shown in Figure 5-51. This is quite likely to be a symptom of resizing the stress profile without rebalancing the stresses, which would necessitate making arbitrary adjustments to eliminate a spurious result from an area of the roll which receives very little attention. This assumption can be tested by comparing two models, identical except for the stresses at the surface.

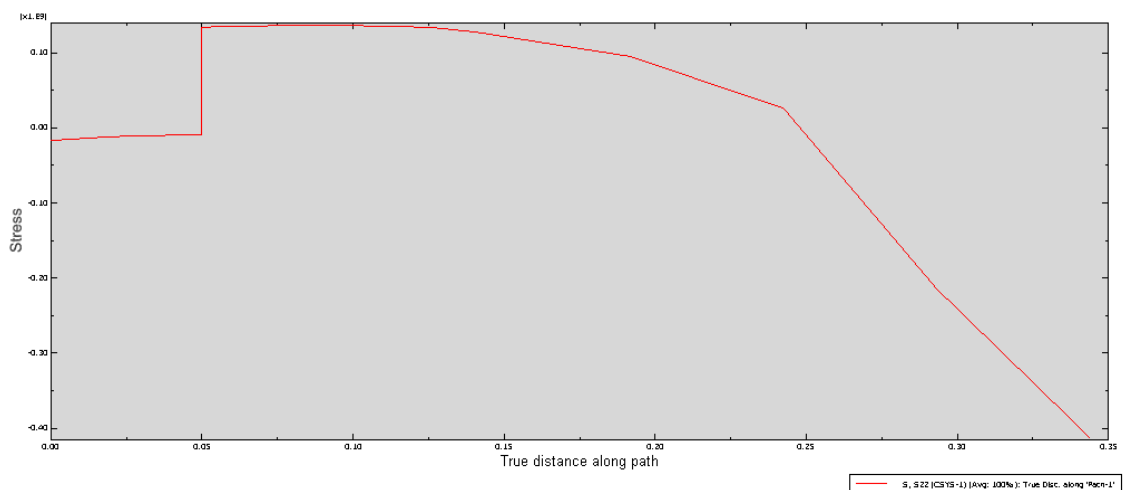
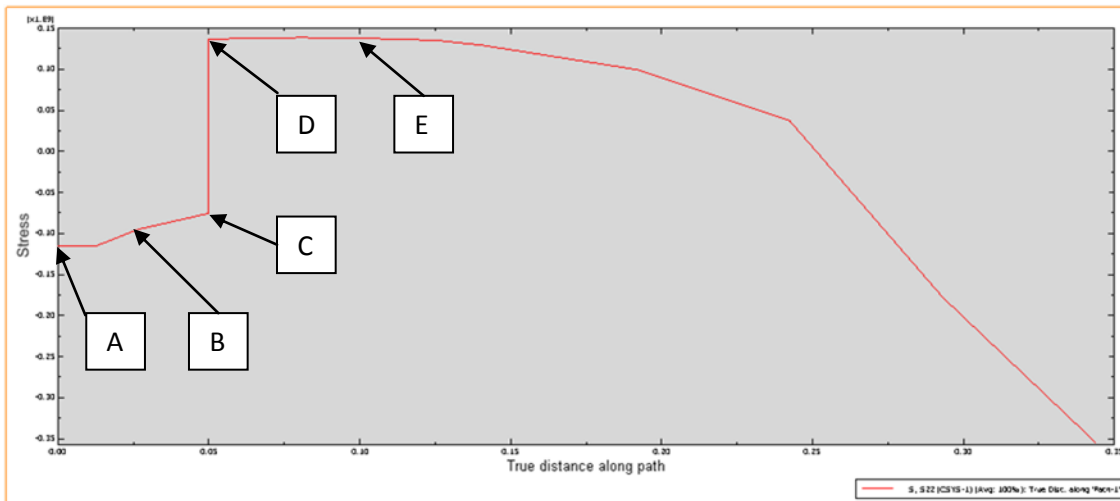
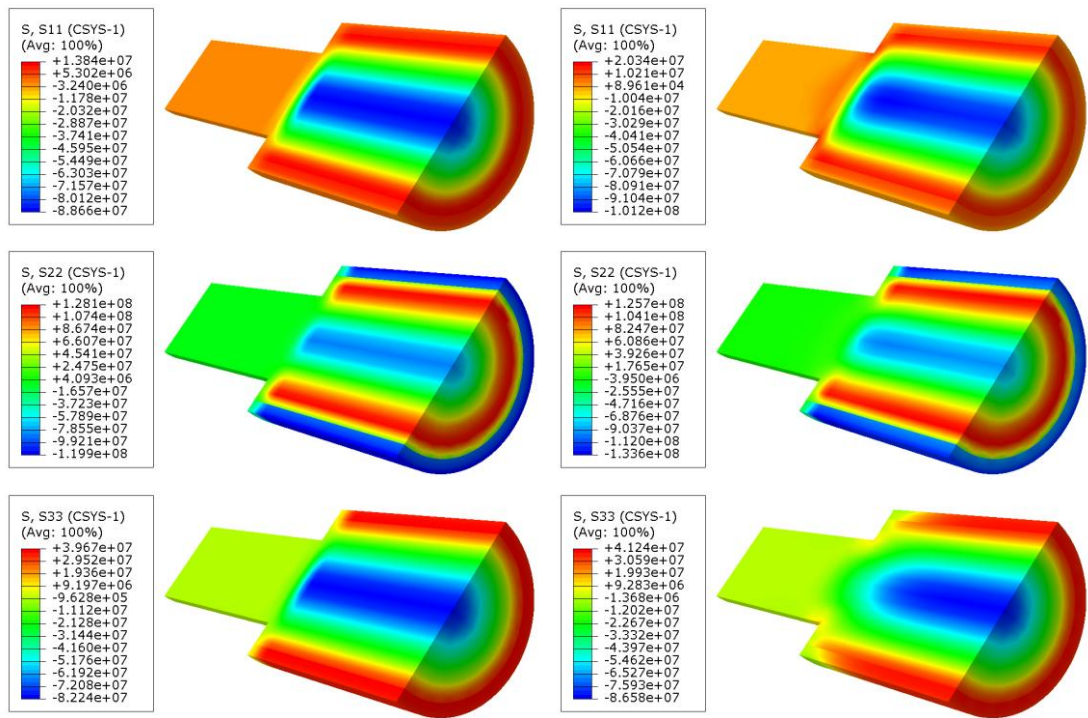


Figure 5-52: Hoop stress profile varying from the surface (left hand side) to the roll centre (right hand side) with reduced surface stresses,



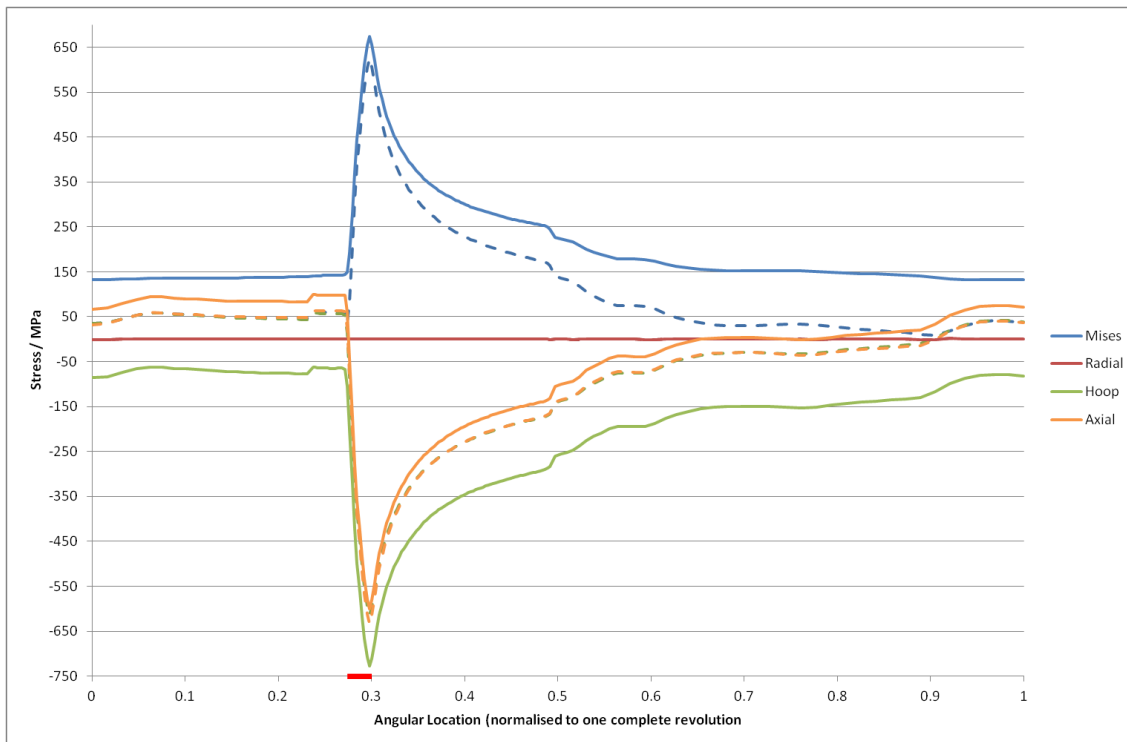
**Figure 5-53: Residual hoop stress profile varying from the surface (left hand side) to the roll centre (right hand side) with normal surface stresses. The hoop stress values at positions A, B, C, D and E were specified according to Figure 4-49 as described above.**

Figure 5-52 and Figure 5-53 show the radial change in hoop stress from the roll surface (left of the figure) to the roll centre (right of the figure), with and without compressive surface stresses respectively. The patch of very high compressive stress at the roll centre is closely related to the amount of compressive stress at the surface (the left of the figures) and, by association, the tensile band in the middle. Without evidence to suggest whether the surface compression should be higher or the sub-surface tension should be lower, there is no justification for choosing either option. Therefore the profile was left as it stands, on the understanding that it represents the best currently available estimate.



**Figure 5-54: Residual stress components as applied by the subroutine (left) and after ABAQUS balancing (right), where S11 = radial stress, S22 = hoop stress and S33 = axial stress.**

Figure 5-54 shows that despite best efforts, the residual stresses do not quite remain as they are applied. This is likely due to differences between the way the stresses are recorded and the way they are applied by the user subroutine. However, S22 (the hoop stress) only changes marginally and since the other components are estimates developed by ABAQUS it is clear that this is the best estimate of the residual stress field that can be applied. It is regarded as a reasonable approximation to the true residual stress field and is used to assess the error caused in the stress evaluation where residual stress has been ignored.



**Figure 5-55: Effect of the residual stresses on the thermal stress profile at the surface. Solid lines represent a combined residual and thermal loading. Dashed lines represent thermal loading only. The location of the roll bite is indicated by the thick red line on the abscissa.**

In Figure 5-55 the dotted lines represent the thermal stress profile, while the solid lines represent the combined effect of the thermal and residual stress profiles. The difference is significant, a 130MPa increase to the minimum von-Mises stress and a 30MPa difference to the peak von-Mises stress. The figure shows that the residual stress profile from manufacture can be an important component of the working stress profile of the roll.

## 5.4. Conclusions

---

With some manipulation of the shell model, this chapter has shown that the assumption of negligible circumferential conduction made by [21] is valid. A second study into the distribution of frictional heating through the roll bite showed very little difference for extreme differences in distribution, validating the constant distribution approach.

A series of tests validated the sub-modelling method for stress analysis showing that an accurate representation of the effects of a sharp thermal feature can be attained by the sub-modelling method, even when the parent model's resolution is insufficient in the area immediately surrounding the feature. The tests also showed that the accuracy of this method hinged on the parent model's results being suitably defined at the interface between the parent and sub-model.



The two dimensional model showed that very high resolutions were required to reach a truly resolution independent result, but also showed that the three-dimensional model was converging on a realistic solution as its resolution increased. Comparisons between the two dimensional and three dimensional models showed that the three dimensional models were insufficiently resolved, but also showed that the simplifying assumptions of the two dimensional model were invalid. An adapted version of the plane strain assumption was tested and proved to be accurate enough for rough analyses, although not necessarily accurate enough to provide fine details with accuracy.

Three three-dimensional models were established, with the resolution of each being optimised to give the best combined resolution with the lowest combined run time. The resulting model had twice the radial resolution in the roll bite and an order of magnitude increase in the radial resolution.

The mechanical loading of the work roll was considered, using simplified conditions to test its overall effect. The direct pressure of the roll bite and backup roll were found to add small features to the thermal stress distribution, affecting the axial stress more than the hoop stress. However, the direct pressure loading did not make a significant difference to either the peak or minimum stresses. The effects of shear loading inside the roll bite were tested by implementing two extreme scenarios and one scenario without shear loading, again revealing little significant impact. The effects of roll bending were barely perceivable even when only mechanical loads were applied. The combined results of the mechanical load testing reinforced the commonly held view that the thermal loads dominate over mechanical loads.

Finally a residual stress profile was developed, based on the best available measured values. Only a part of the full stress tensor can be measured using current technology, so ABAQUS was used to supply the remaining components by assuming internal equilibrium. The resultant residual stress profile was shown to make a considerable difference to the stress results.

## 6. Results

---

### 6.1. Introduction

---

With the thermal and stress models assembled and validated, the opportunity arose to experiment with input parameters and inspect the resulting changes to thermal and stress profiles. Due to time constraints, the scope of this numerical experimentation was focused on those areas which have been and are currently of interest to Tata.

- The effect of oxide layer formation on the work roll
- The effect of using temperature dependent material properties
- The effect of key events in the process history

The most recent studies in IJmuiden suggest that the work roll oxide layer thickness varies from  $0\mu\text{m} \rightarrow X\mu\text{m}$  within the roll bite<sup>35</sup>, where  $3 \leq X \leq 10$ . The extremes of this range,  $X = 3$  and  $X = 10$  were tested to assess the maximum possible impact of varying this parameter. The oxide layer in question is developed and abraded away within the roll bite, and its only effect on the modelling is to modify the heat transfer coefficients in the roll bite, according to Equation 3-62 in a way which depends on the value of  $X$  specified.

In order to generate a realistic process history, a data set was chosen which contained the minimum number of stoppages or other aberrant behaviour and had high rates of reduction in thickness (30-40%). The low number of stoppages gives a close-to-ideal set of rolling conditions without having to use artificial data, and can be edited to add in interesting events at a later date. The high rates of reduction will result in high rolling loads and therefore maximum heat generation, pushing the rolls to the boundaries of their operational tolerances.

From the data set mentioned above, four scenarios were studied in detail. Each scenario is comprised of a full transient two-dimensional thermal analysis, followed by a single static three-dimensional thermal and stress analysis at the end of a period of rolling defined by the scenario description given below.

1. Early – The period of rolling is only the second in the campaign, i.e. the second strip through the hot mill.
2. Normal – The period of rolling is towards the end of the campaign, when the rolls have had time to reach a stable thermal profile.
3. Mill Stop – The period of rolling immediately follows a long mill stop (15 minutes). In such cases the cooling sprays are turned off, to prevent over-chilling the roll.

4. Long Gap – The period of rolling immediately follows a long gap between strips (15 minutes). In this case the cooling sprays are left on, to assess the impact of not turning off roll coolant when the roll is warm and there is no rolling activity.

Scenarios 2, 3 and 4 all consider the same period of rolling, with the data referring to the preceding gap between strips adjusted to give the desired process conditions for each respective scenario.

For each scenario, four complete sets of results were created; using  $3\mu\text{m}$  and  $10\mu\text{m}$  oxide layer thickness on the roll surface, each with and without temperature dependent material properties.

NB – The results presented in this chapter were generated using the data set provided in Appendix D.3.

## 6.2. The scenarios

This section will illustrate the effect of each scenario on the average temperature profile in the work roll, both before and after the period of rolling which will be studied. Each of the images in this section is taken from a model with temperature dependent material properties and a  $3\mu\text{m}$  thick oxide layer. The layout of all the images is as shown in Figure 6-1 below which illustrates the core model, where  $r$  is the radial direction and  $z$  is the axial direction.

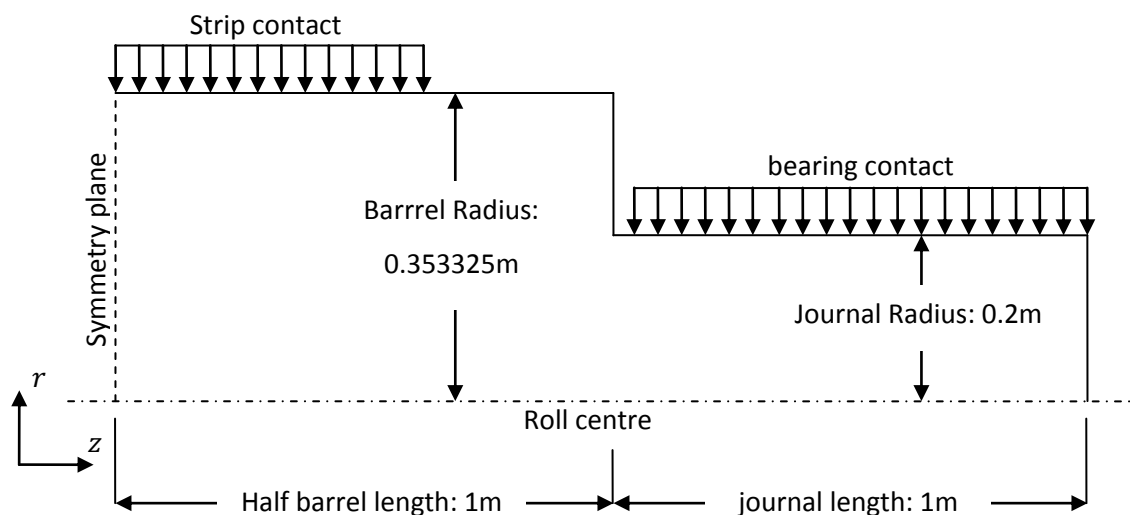


Figure 6-1: Core model temperature profile layout

### 6.2.1. Early

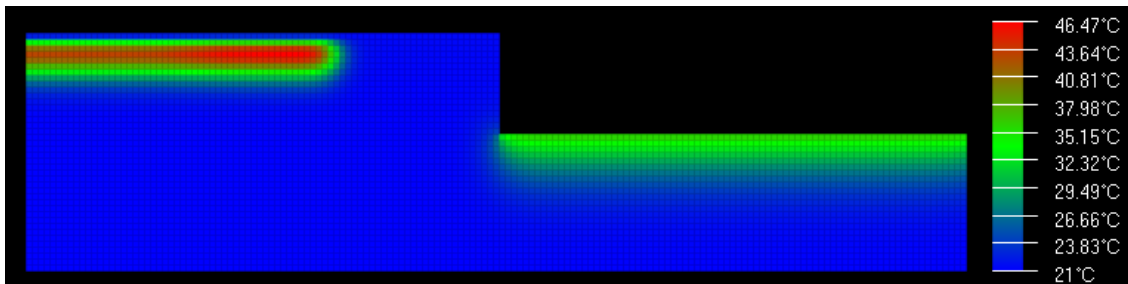


Figure 6-2: Core model temperature profile, just before the beginning of the rolling period. Axes illustrated and dimensioned in Figure 6-1.

At the very beginning of the period of rolling the temperature profile is as shown in Figure 6-2. Some heat from the first strip has conducted below the surface and has not been fully extracted by the cooling sprays in the short gap between strips, but on the whole the work roll is still at its initial temperature of 21°C. The chock which holds the roll journal is typically at roughly 40°C, hence the second area of heating at the journal surface.

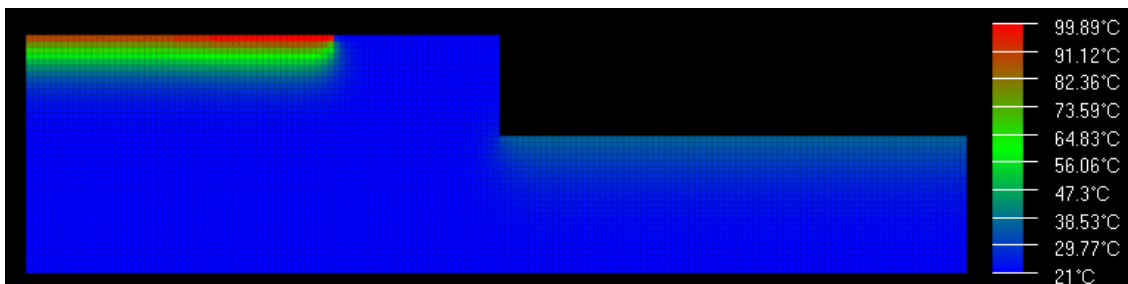


Figure 6-3: Core model temperature profile, at the end of the rolling period. Axes illustrated and dimensioned in Figure 6-1.

Figure 6-3 shows the temperature profile at the end of the period of rolling. The surface of the roll is now very warm from extended contact with the strip, but there is still relatively little heating below the surface. This scenario will show the effects of having very high temperatures at the surface when the internal temperatures are very low.

## 6.2.2. Normal

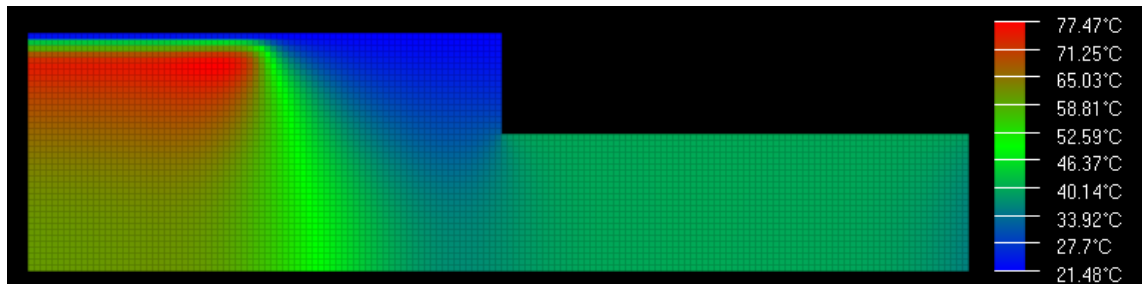


Figure 6-4: Core model temperature profile, just before the beginning of the rolling period. Axes illustrated and dimensioned in Figure 6-1.

Figure 6-4 shows the temperature profile just before a period of rolling, after much more time has passed. A considerable amount of heat has been conducted into the depths of the work roll, and temperatures have generally risen everywhere except at the edges of the barrel, which are intensively cooled but don't make contact with the strip.

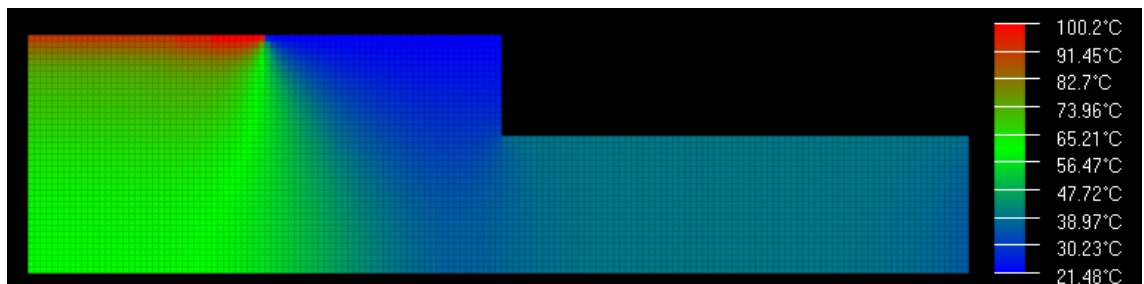


Figure 6-5: Core model temperature profile, at the end of the rolling period. Axes illustrated and dimensioned in Figure 6-1.

Figure 6-5 shows the temperature profile at the end of the period of rolling. The surface temperature is similar to that in the "Early" scenario, but the change in temperature with depth below the surface is much less pronounced. This scenario will act as a base result to which the other results can be compared.

### 6.2.3. Mill Stop

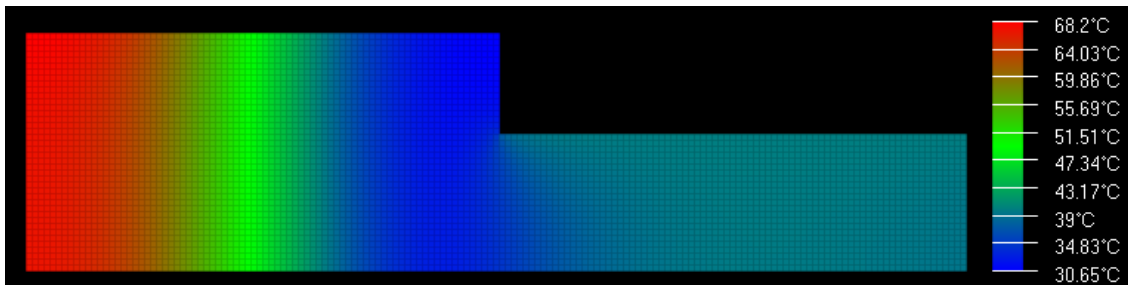


Figure 6-6: Core model temperature profile, just before the beginning of the rolling period. Axes illustrated and dimensioned in Figure 6-1.

Figure 6-6 shows the core model temperature profile at the end of a long mill stop. Since the cooling has been turned off, the only heat loss from the roll is from conduction to the ambient environment. The heat actually transferred to the environment is minimal, and the radial temperature differences concern relatively small volumes of material, so they can normalise fairly quickly. The dominant component of heat transfer becomes axial, where the large volumes of warm material at the centre of the roll cool down as the colder barrel ends heat up.

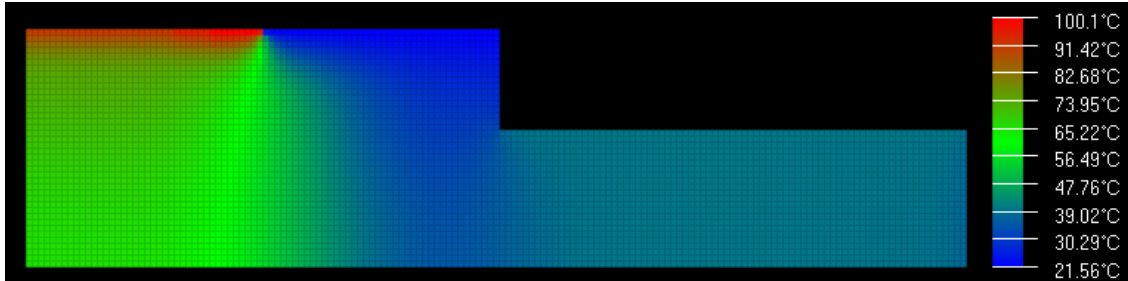


Figure 6-7: Core model temperature profile, at the end of the rolling period. Axes illustrated and dimensioned in Figure 6-1.

Figure 6-7 shows a striking resemblance to Figure 6-5. Since the cooling sprays were turned off during the mill stop there was very little heat lost from the work roll. Some axial rebalancing of the temperature distribution has had minimal effect on the overall form of the temperature distribution by the end of the period of rolling.

#### 6.2.4. Long Gap

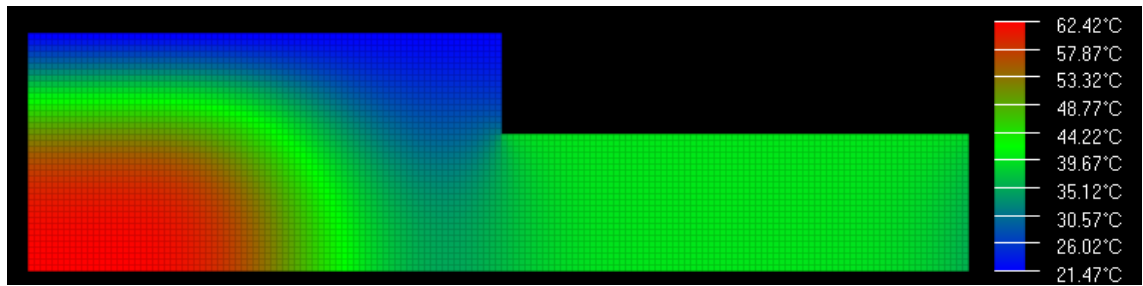


Figure 6-8: Core model temperature profile, just before the beginning of the rolling period. Axes illustrated and dimensioned in Figure 6-1.

Figure 6-8 shows the temperature distribution in the work roll after 15 minutes of cooling without any strip contact. The surface of the work roll has been chilled to just above the temperature of the coolant (21.4°C) and the cooling effect has penetrated to a significant depth.

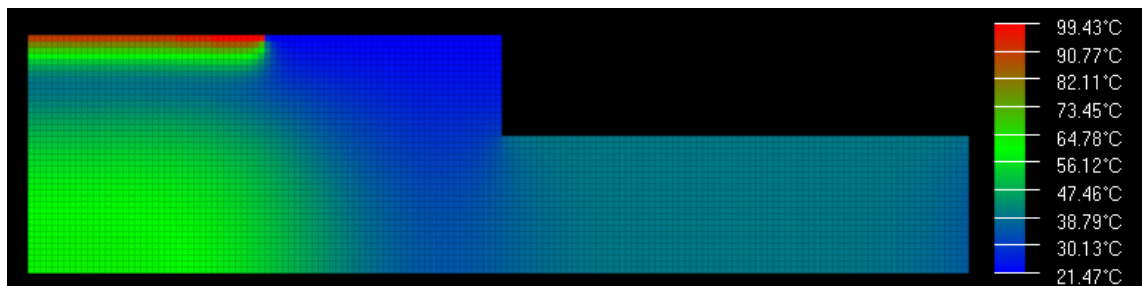


Figure 6-9: Core model temperature profile, at the end of the rolling period. Axes illustrated and dimensioned in Figure 6-1.

Figure 6-9 shows the effect of the following period of rolling on the temperature profile. The surface temperatures are practically identical to every other scenario, but the area just below the surface looks very similar to the "Early" scenario. However, the centre of the roll still retains a significant amount of heat. This scenario will show the effects of having a band of cold material between a warm surface and warm centre of a roll.

### 6.3. Temperature results

---

#### 6.3.1. Core model temperature profiles

Figure 6-10 shows all of the core model temperature profiles for the "Normal" scenario. Despite the large differences in overall temperature, the profiles are all very similar in shape, to the point that the figures are indistinguishable without reference to the legend. The core model temperature profiles for the other scenarios are omitted, since they follow the same trend.

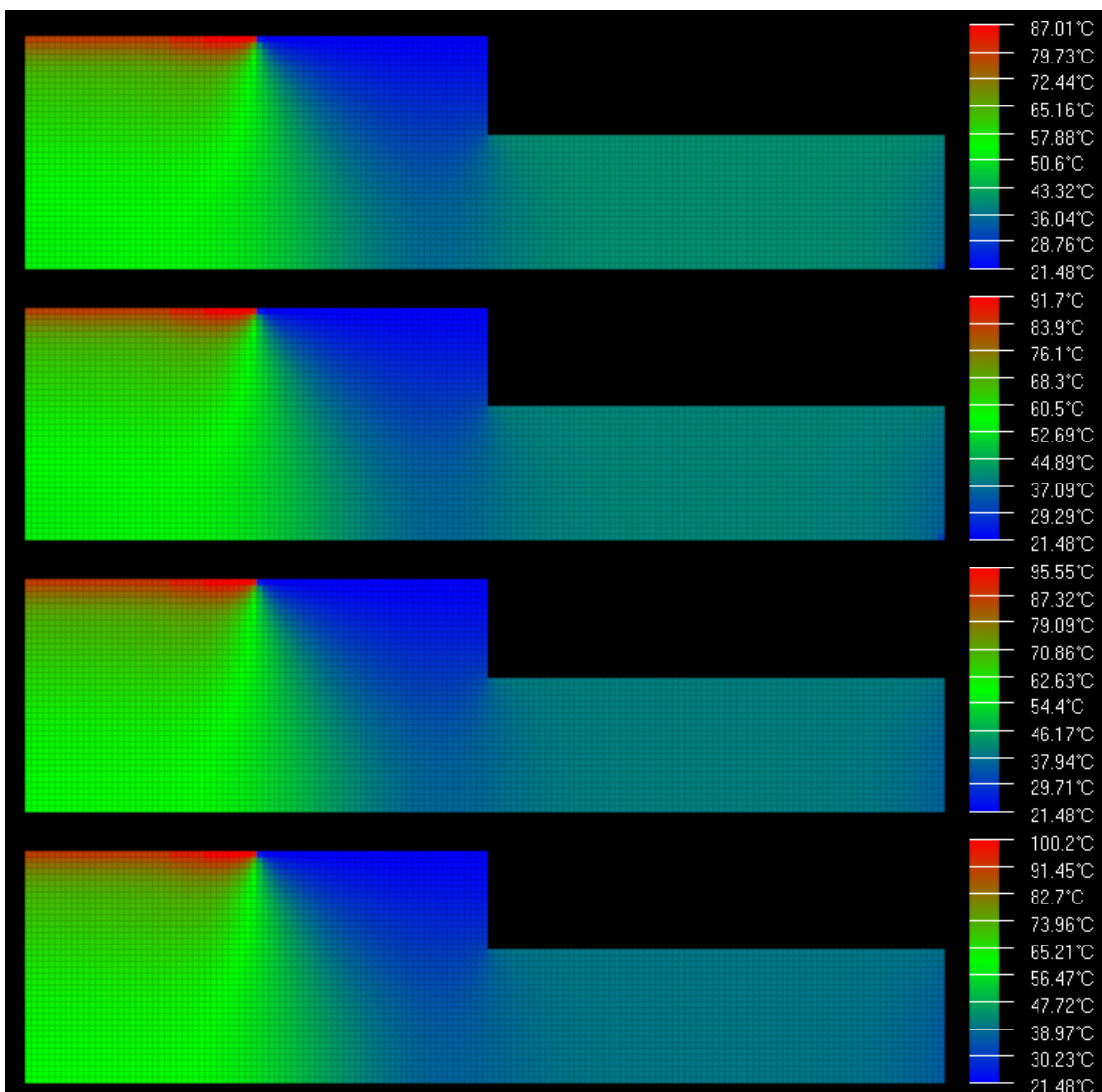


Figure 6-10: Comparison of core model temperature profiles for the "Normal" scenario using (from the top down); 10μm oxide thickness and constant material properties, 3μm oxide thickness and constant material properties, 10μm oxide thickness and temperature dependent material properties, 3μm oxide thickness and temperature dependent material properties. Axes illustrated and dimensioned in Figure 6-1.



### 6.3.2. Radial graphs

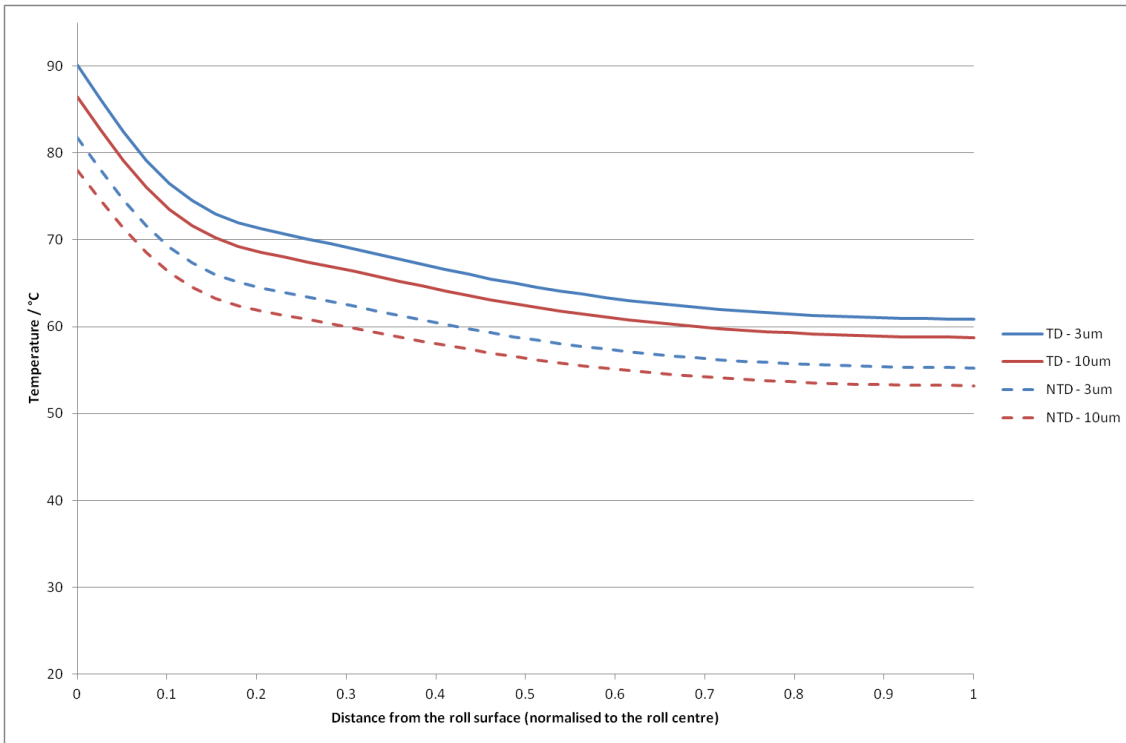
The previous section showed the temperature distributions for each of the different scenarios for the case of  $3\mu\text{m}$  oxide thickness and temperature dependent material properties. Figure 6-10 then showed that changing the oxide thickness and temperature dependency of the material properties resulted in changes in the magnitude of the core model's temperature profile but no significant change in its distribution. This section will look at the effect on the temperature distributions of using temperature dependent material properties and changing the thickness of the oxide layer. Graphs of temperature against radial location at  $z = 0$  (i.e. at the symmetry plane in Figure 6-1) will be used to compare the different settings, where the left hand side of the graph represents the temperature at the surface of the roll and the right hand side represents the temperature at the centre of the roll.

#### *Normal*

---

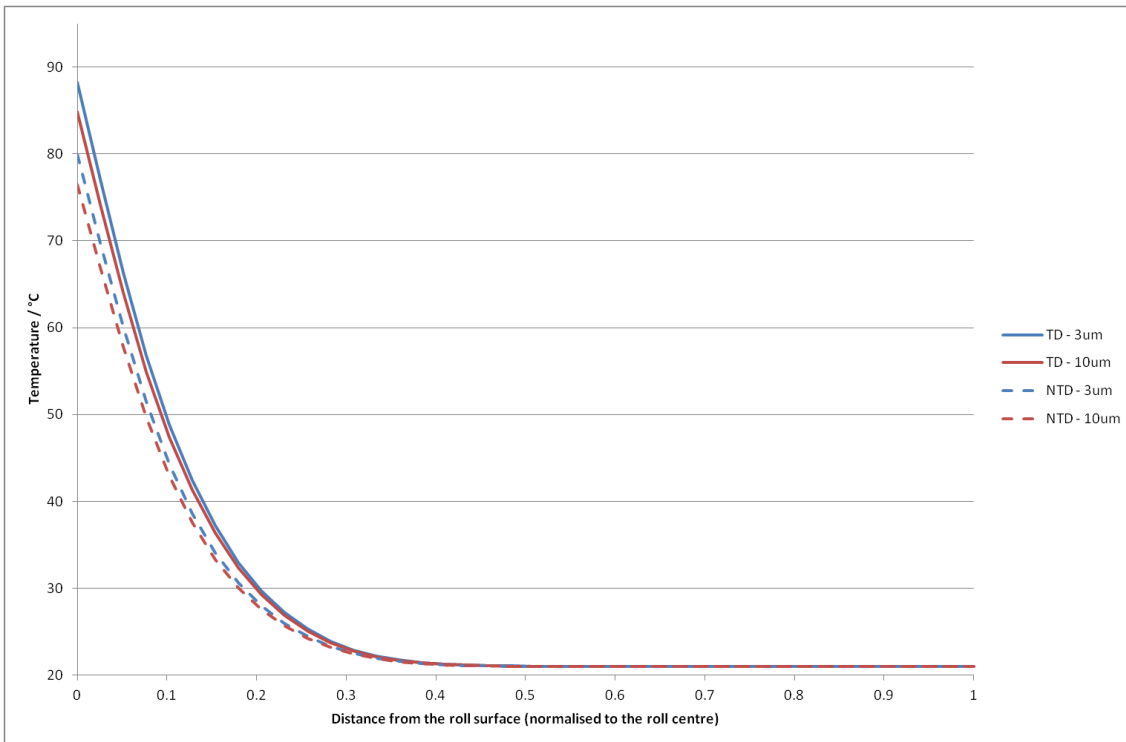
The four models illustrated in Figure 6-11 all start with a roll at a constant  $21^\circ\text{C}$  and the same properties, geometry and process data, excepting oxide thickness and temperature dependency. The difference between the four profiles is not the difference that occurs over the space of one period of rolling but the total difference between their entire thermal histories.

While the general shape of each result is largely the same, the magnitude of the temperature distributions is quite different. The models using temperature dependent material properties see an  $8^\circ\text{C}$  increase in average temperatures throughout, an increase of roughly 10%. Increasing the oxide layer thickness from  $3\mu\text{m}$  to  $10\mu\text{m}$  resulted in a  $3.5^\circ\text{C}$  drop in temperature throughout, also a significant difference.



**Figure 6-11: Change in circumferentially averaged temperature with radial position at the centre symmetry plane, with the specified settings in effect for the whole campaign**

*Early*

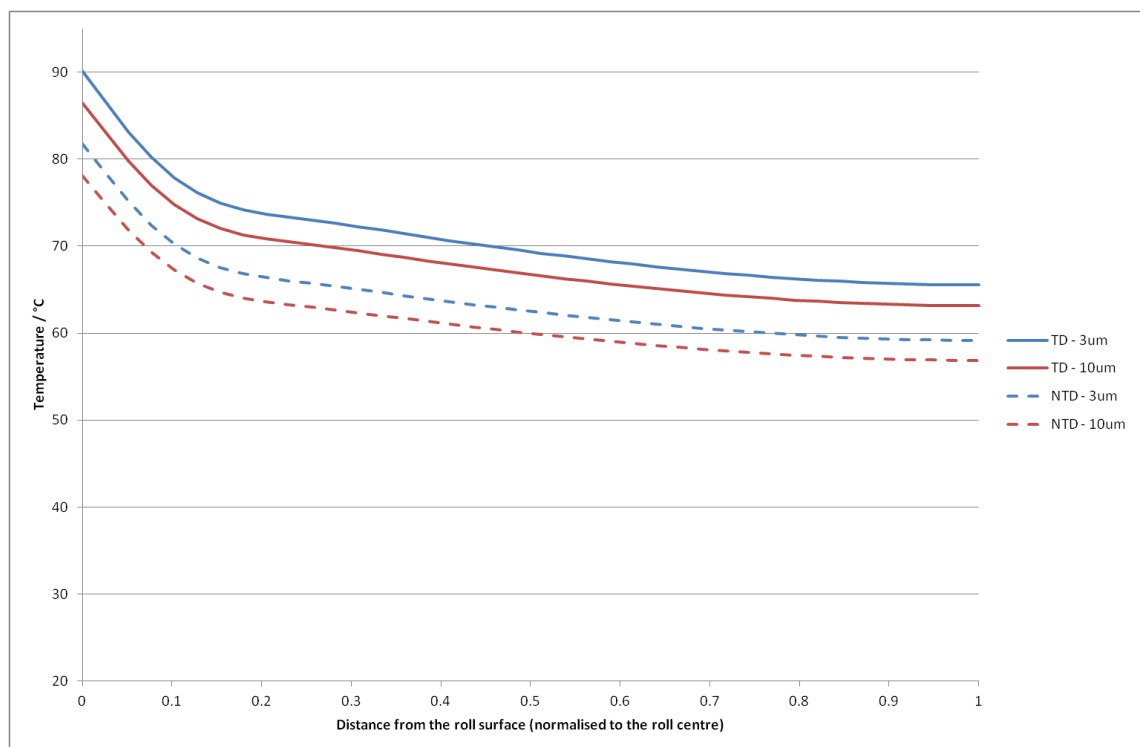


**Figure 6-12: Change in circumferentially averaged temperature with radial position at the centre symmetry plane, with the specified settings in effect for the whole campaign**

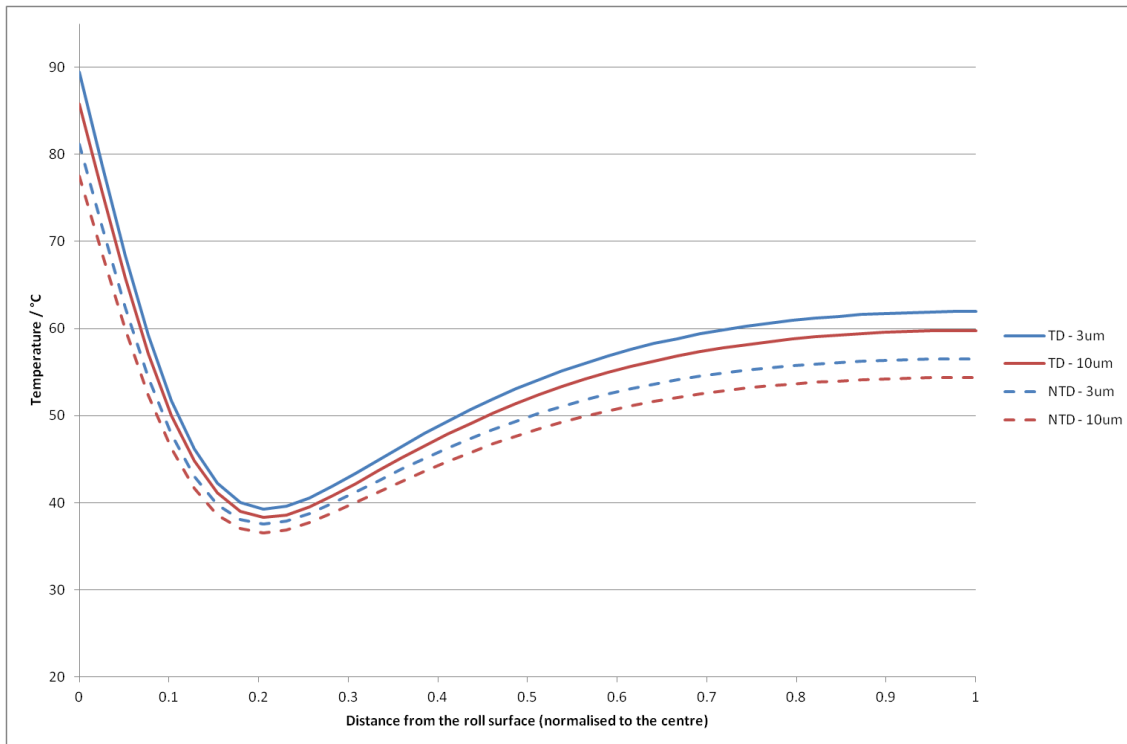
Figure 6-12 shows the difference in temperature between the four model setups at the beginning of a campaign. The temperature at the centre of the roll is identical for all four models, since there has not been enough time for any of the surface heat to conduct through to the centre. However, the models using temperature dependent material properties are warming up noticeably faster. This is probably due to the higher conductivity of the material at elevated temperatures, reducing the resistance to heat flow of the material just below the surface.

### Mill Stop

Figure 6-13 shows the radial temperature profile for the "mill stop" scenario. Other than the centre of the roll being 4°C warmer there is relatively little difference between this and the "normal" scenario.



**Figure 6-13: Change in circumferentially averaged temperature with radial position at the centre symmetry plane, with the specified settings in effect for the whole campaign**

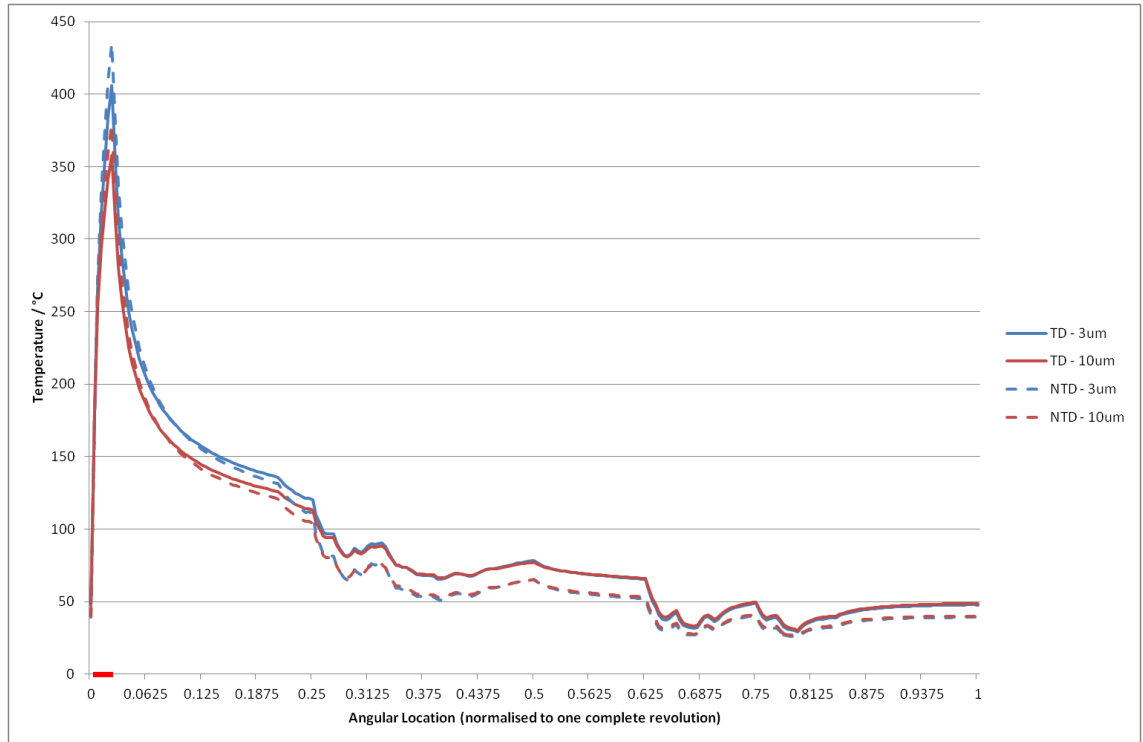


**Figure 6-14: Change in circumferentially averaged temperature with radial position at the centre symmetry plane, with the specified settings in effect for the whole campaign**

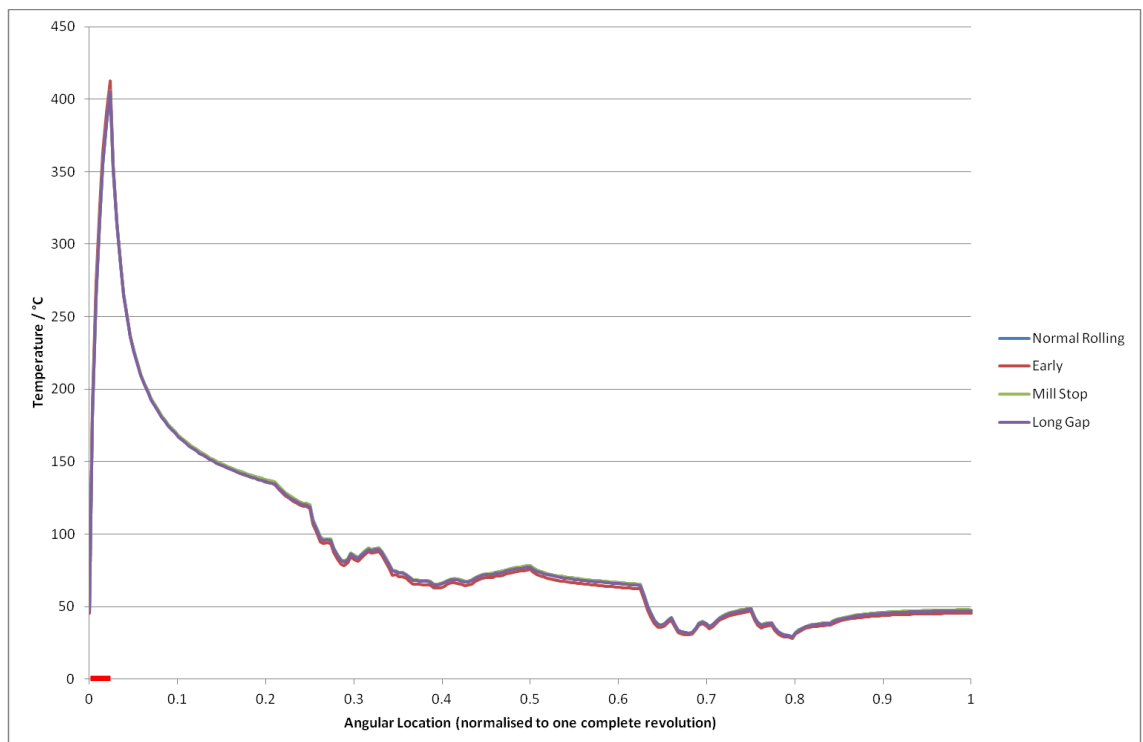
The effect of a long period of cooling can be seen in the large dip in Figure 6-14. As with the "Early" scenario, the temperature difference between the four models is not completely uniform throughout the radius. Again, this could be the result of higher conductivity of the temperature dependent models as they cool more rapidly due to the improved internal heat flow.

### 6.3.3. Circumferential Graphs

Figure 6-15 shows the circumferential temperature variation at the surface of the shell model during the "Normal" scenario. As could be expected, reduction in the thickness of the oxide layer removes some of the insulation from the work roll surface and results in an increase in peak temperature. However, the peak temperature and overall temperature difference in the models with temperature dependent properties is reduced, despite the increased average temperature. This is a product of the increased conductivity, allowing more heat to escape below the surface of the work roll and therefore reducing the surface temperature for the same level of heat flow.



**Figure 6-15: Circumferential temperature distribution at the surface. Roll bite location is indicated by the thick red line on the abscissa.**



**Figure 6-16: Comparison of circumferential temperature distributions at the surface. The location of the roll bite is indicated by the thick red line on the abscissa.**

Figure 6-16 shows why it is not necessary to replicate Figure 6-15 for each of the scenarios. The figure shows the circumferential surface temperature distributions for each of the four

scenarios, using temperature dependent properties and an oxide thickness of  $3\mu m$ . The profiles are almost identical, a trend which holds true for all of the other setting combinations too. This shows that the angular surface temperature profile at the end of a strip is almost entirely independent of the process history. It also shows that any difference between the stress results of the four scenarios is independent of the surface temperature profile.

#### 6.4. Stress results

Figure 6-17 and Figure 6-18 show how the stress profiles vary in the axial and radial directions, when a section is taken at approximately 90 degrees to the roll bite and at the roll bite, respectively. These models were produced using the temperature results generated using temperature dependent material properties and an oxide layer thickness of  $3\mu m$ . The models also include the effects of mechanical loading and the residual stress profile from manufacture. In both figures a log distribution is used to calculate the colour fringes. The main differences are that the "Long Gap" scenario has slightly higher stresses around the centre, while the "Early" scenario has noticeably higher stresses near the surface of the roll.

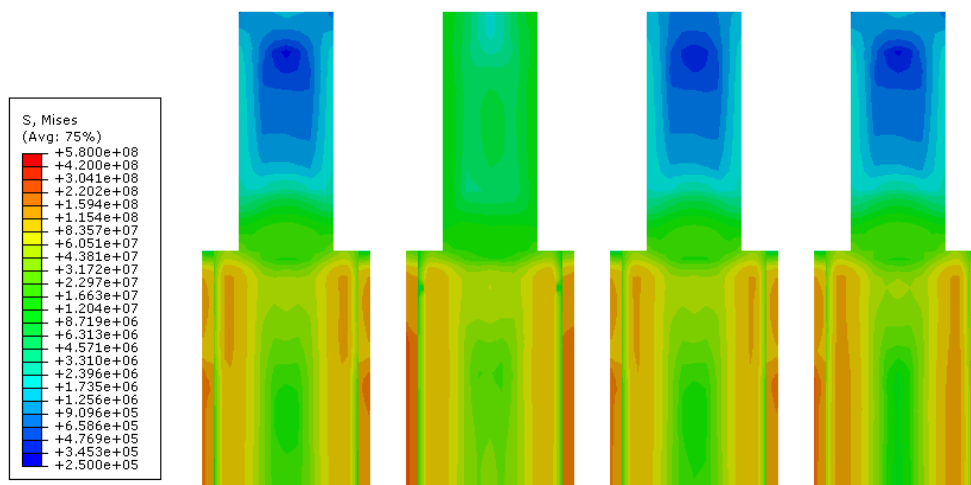


Figure 6-17: Axial-Radial von-Mises plots for (from left to right) Normal, Early, Mill Stop and Long Gap scenarios, at 90 degrees to the roll bite. Units are in Pascals, colour range from blue at 250KPa to red at 580MPa.

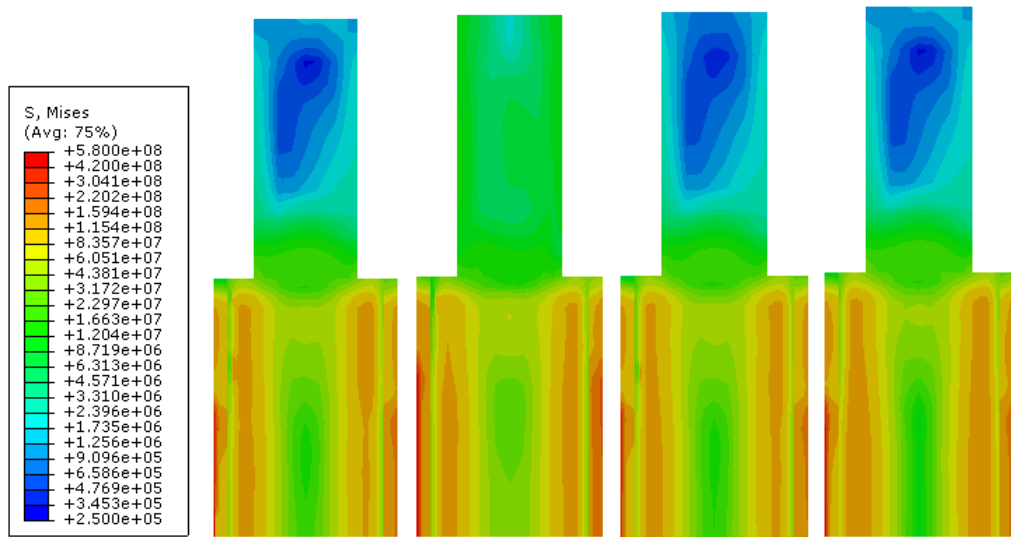


Figure 6-18: Axial-Radial von-Mises plots for (from left to right) Normal, Early, Mill Stop and Log Gap scenarios, at the roll bite (left side of each plot is under the strip contact, right side is under the backup roll contact). Units are in Pascals, colour range from blue at 250KPa to red at 580MPa.

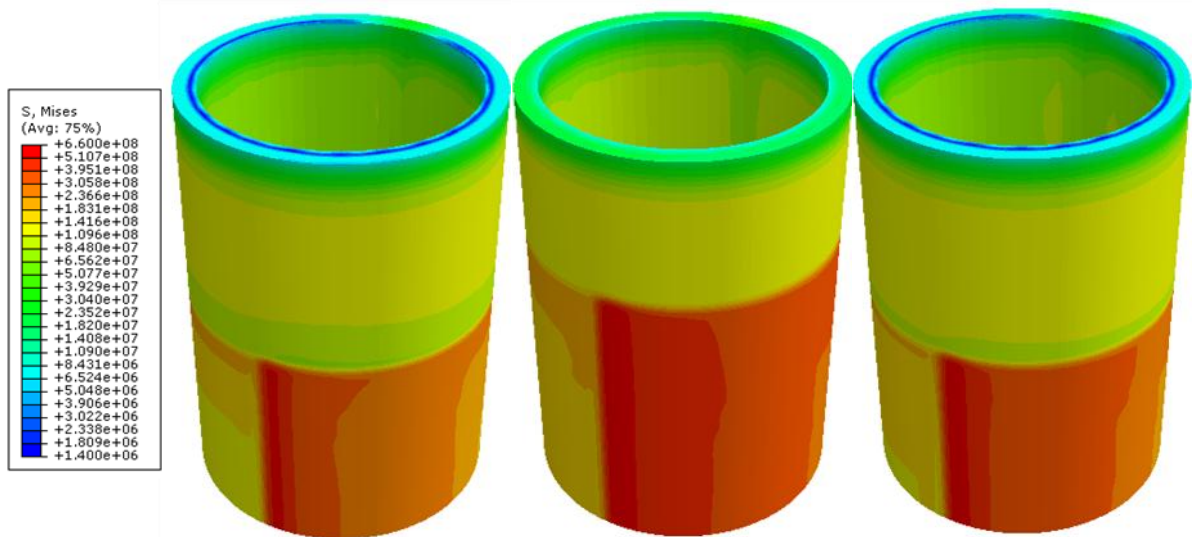


Figure 6-19: Surface model von-Mises stress plots for (from the left) the Normal, Early and Long Gap scenarios. Units are in Pascals, colour range from blue at 1.4MPa to red at 660MPa.

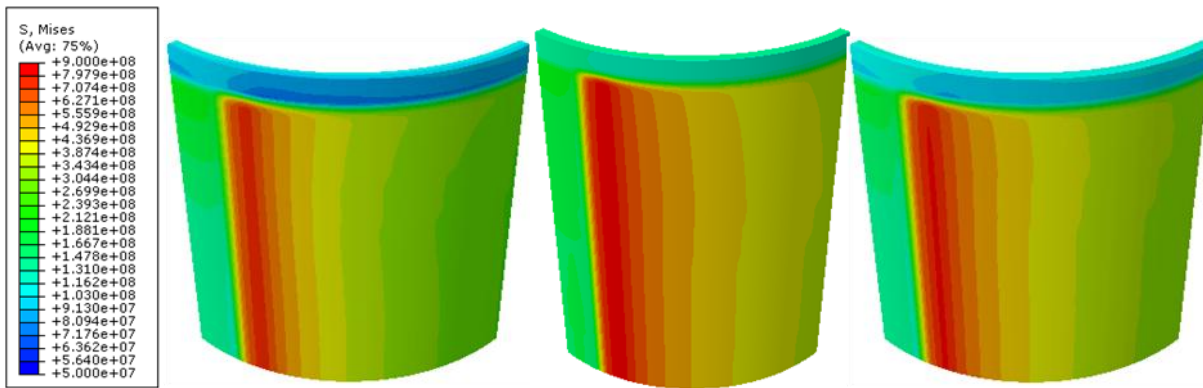


Figure 6-20: Bite model von-Mises stress plots for (from the left) the Normal, Early and Long Gap scenarios. Units are in Pascals, colour range from blue at 50MPa to red at 900MPa.

Figure 6-19 and Figure 6-20 show the circumferential stress variations in more detail. The figures ignore the "Mill Stop" scenario as it is almost identical to the "Normal" scenario (confirming that the current practice of turning off the roll cooling during extended periods of inactivity is advisable). Both sets of figures show that the "Early" scenario has the highest stresses, while the "Long Gap" scenario lies somewhere between the "Early" and "Normal" levels of stress.

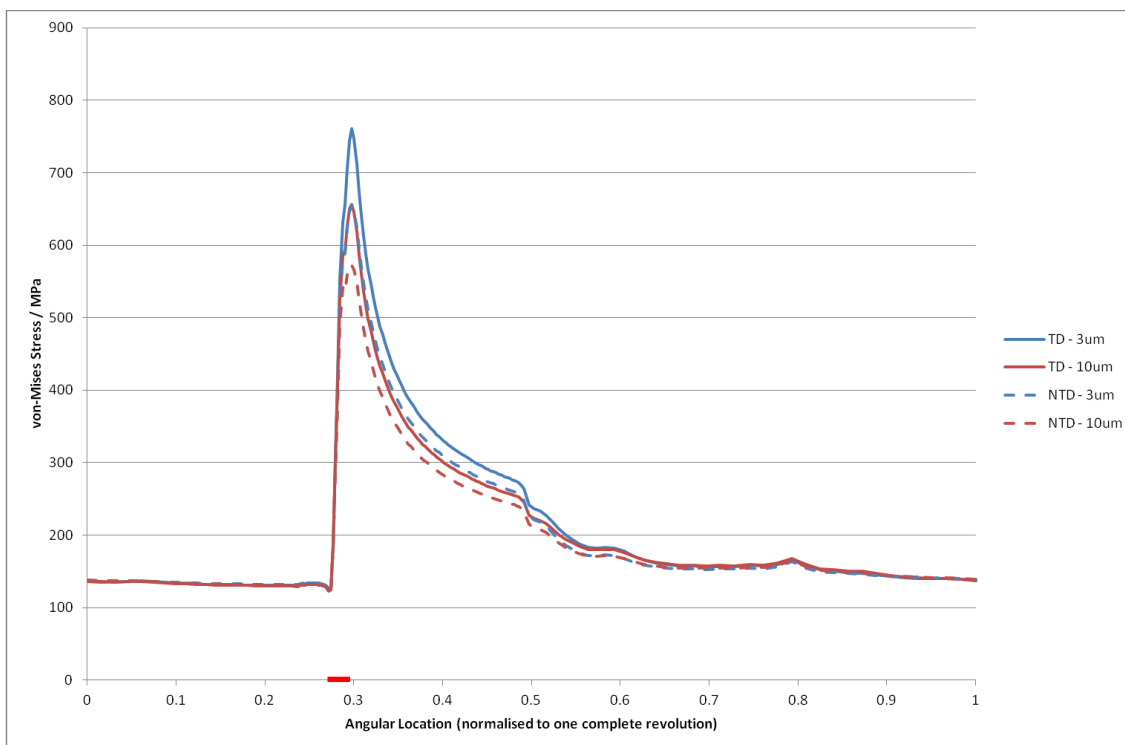


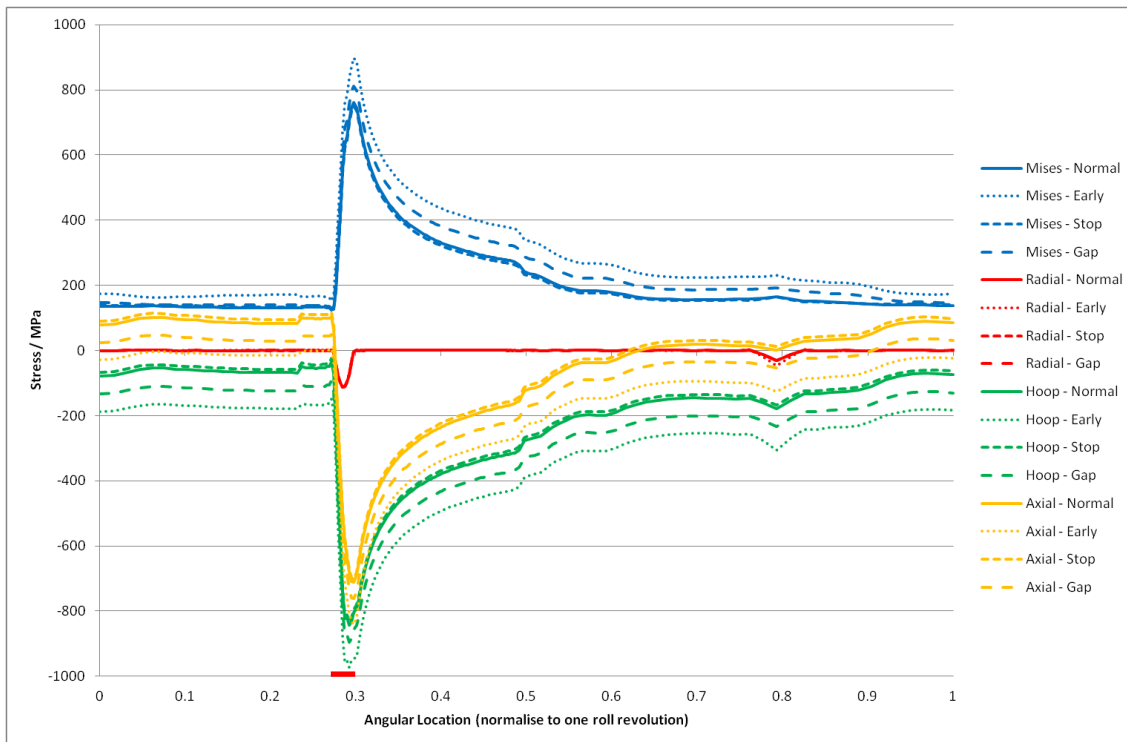
Figure 6-21: Circumferentially varying combined surface von-Mises stress distributions in the "Normal" scenario. Location of the roll bite is indicated by the thick red line on the abscissa.

Figure 6-21 shows the variation in von-Mises stress at the surface of the work roll for all four versions of the "Normal" scenario. The results are mainly taken from the surface model,



but bite model values are used where available. The figure shows that both the temperature dependency of the material properties and the thickness of the oxide layer on the work roll surface are significant factors. A thicker oxide layer reduces the temperature of the material, which in turn reduces the thermal expansion and resulting thermal stresses. Since the stress fluctuation is principally due to thermal effects, this reduction is significant. While temperature dependency actually reduces the peak surface temperature, it also reduces the Young's modulus and increases the expansion coefficient. The increased tendency towards expansion and reduced material stiffness results in a higher stress despite the lower temperatures.

When talking about the probability of early failure of a work roll, it is the difference between the maximum and minimum stress in a single revolution which is key. The resulting stress cycle will be the primary measure for determining the likely life of the work roll, based on fatigue considerations. The use of temperature dependent properties can be seen to add up to 100MPa to the magnitude of the stress cycle, so a model for predicting work roll life must take this effect into account. The range of expected oxide thicknesses makes a further 100MPa difference to the magnitude of the stress cycle, by changing the amount of heat transferred from the strip to the work roll. The first conclusion to be drawn from this is that accurate information about oxide layer thickness is required before an accurate prediction of work roll stresses can be achieved. The second conclusion is that an oxide layer can benefit the work roll by insulating the roll and therefore reducing the amount of thermal stress to which it is subjected. The effect of the oxide layer on the quality and mechanical properties of the roll surface is, of course, another question.



**Figure 6-22: Circumferentially varying combined surface von-Mises stress distributions for all models using temperature dependent material properties and an oxide layer thickness of  $3\mu m$ . Location of the roll bite is indicated by the thick red line on the abscissa.**

Figure 6-22 shows the effect on the stress distribution at the surface of changing the process history, all other factors being equal. The figure highlights an interesting property of the interaction between the stress distribution and the temperature distribution. Considering the axial and hoop components of the stress distribution, changing the scenario type has the effect of offsetting the entire stress distribution. Figure 6-16 shows that the circumferential temperature distributions for each of these models at the surface are identical. By contrast, only the angular component of the temperature profiles which generated the stresses shown in Figure 6-21 varied significantly. The resulting stress profiles had identical minimum stresses but different maximum stresses.

Therefore, it can be concluded that;

- Minimising the radial temperature gradient results in lower minimum stresses at the surface
- Minimising the angular temperature gradients results in smaller differences between the minimum and peak stresses at the surface

	von-Mises			Hoop			Axial		
	Max / MPa	Min / MPa	Range / MPa	Max / MPa	Min / MPa	Range / MPa	Max / MPa	Min / MPa	Range / MPa
Normal	761	126	635	-35	-844	810	104	-712	817
Early	897	152	745	-149	-974	825	5	-842	847
Mill Stop	751	129	622	-24	-835	810	116	-701	817
Long Gap	812	125	687	-91	-897	806	50	-762	812

Table 6-1: Maximum, minimum and range of stress for each scenario

Table 6-1 compares the maximum, minimum and range of stress components at the surface for each of the scenarios. The radial component was omitted since it makes up a very small percentage of the overall stress state.

The "Mill Stop" scenario shows identical ranges in its hoop and axial stress components to the "Normal" scenario, although it actually has lower maximum stresses, resulting in a smaller von-Mises fluctuation. This could be a result of the slight equalisation of the work roll temperatures below the surface.

While the range of the fluctuations in the hoop and axial stress components of the "Long Gap" scenario is slightly reduced, a general negative (compressive) shift of roughly 50MPa has led to a 50MPa increase in the range of the von-Mises stress cycle. This increase shows in numerical terms why leaving the cooling running for extended periods without a strip in the mill is bad for work roll life.

By far the highest peak stresses and stress ranges are given by the "Early" scenario, the highest component approaching 1GPa. The von-Mises stress cycle magnitude of 745MPa represents a 17.3% increase over the normal rolling conditions. This data indicates that the work roll is considerably more susceptible to damage when it is cold.

#### 6.4.1. Comparison to literature

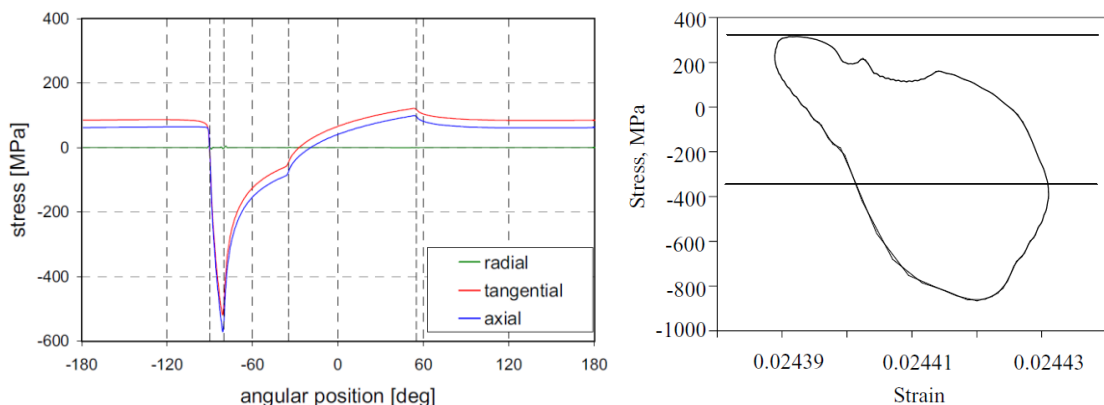


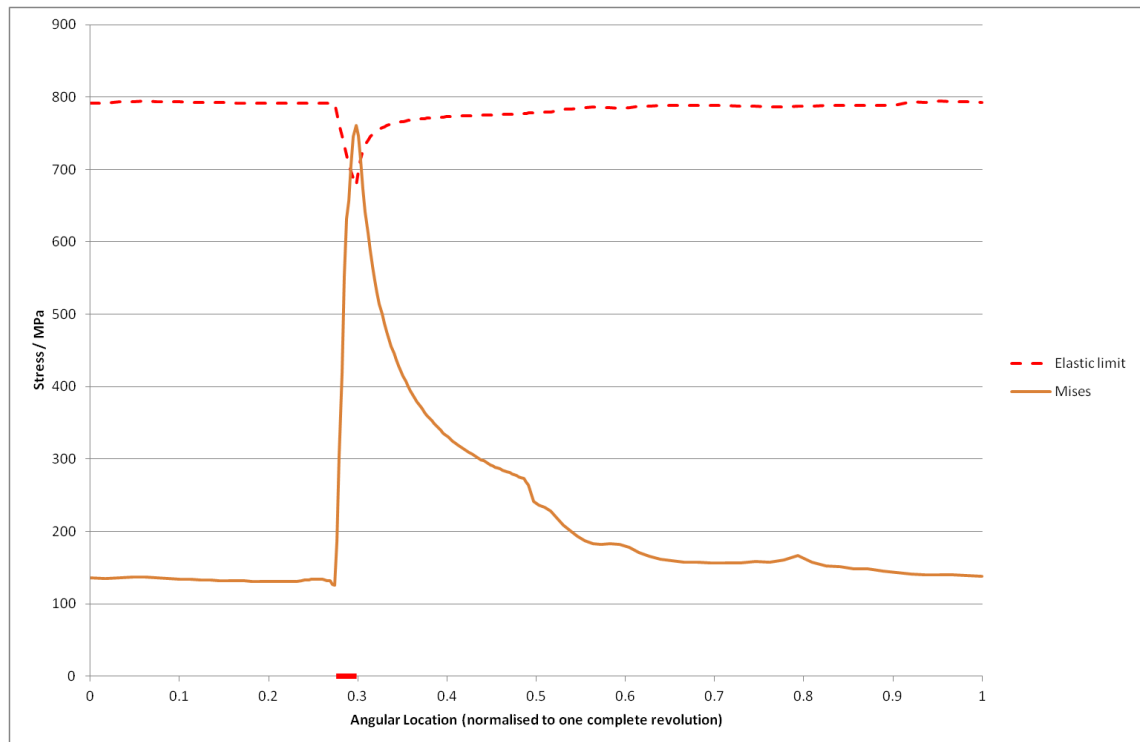
Figure 6-23: Stresses on the work roll surface<sup>22</sup>, and a stress cycle, based on an assumed temperature profile<sup>17</sup>

Literature on the stress fields in work rolls is currently quite thin, so there is relatively little material available for comparison. There is no directly measured experimental data, due to the aggressive nature of the rolling environment and the expense required to instrument a work roll. Figure 6-23 shows results from two of the most relevant articles.

The model developed by [22] was a methodological study, as mentioned in the literature review, so the setup was simplistic. The model did not include the residual stresses from manufacture or any mechanical loading, was two-dimensional and assumed a single set of material properties throughout the roll. The comparison between [22] and this study is therefore loosely qualitative at best. However, the general conclusion of near-zero radial stress at the surface agrees with this study's results. Also, when residual stresses and mechanical loads are ignored (as shown in Chapter 5, Page 248) the hoop and axial stresses are very similar to each other, as is the case in [22], and considering the difference in cooling conditions, the profiles of these stress components show a very similar shape. The overall magnitude of the stress fluctuations, roughly 700MPa, further supports the conclusion that the results of the current study are in the right area.

The model developed by [17] used an analytical relationship for the temperature profile, which predicted higher temperatures than the current study (A peak of over 600°C, rather than just over 400°C). The analytical temperature solution assumes a semi-infinite solid where the temperature tends to the initial temperature as the distance below the surface tends to infinity, emulating the "Early" scenario. Without any details about the assumed material properties a direct comparison is not possible, but the stress cycle of 1100MPa for an extreme temperature profile seems reasonable if the current study gives a difference of 847MPa in its dominant component.

## 6.4.2. Yielding study



**Figure 6-24: von-Mises stress at the surface compared to yield stress in the "Normal" scenario. The location of the roll bite is indicated by the thick red line on the abscissa.**

Figure 6-24 shows the von-Mises stress distribution at the surface and an estimate of the yield stress for the work roll shell material in the "Normal" scenario. This estimate is acquired from a combination of two sources, a figure for room temperature yield stress from the roll manufacturer, and a relationship for the change in Vickers hardness with temperature ( $Hardness = 650 - 0.1512T + 2.568 \times 10^{-4}T^2 - 1.198 \times 10^{-6}T^3$ ) from the Jmuiden RD&T department. For the value of yield stress given, 800MPa, the hardness is quite high ( $650 \frac{kg}{mm^2}$  at room temperature, which corresponds to nearly 2GPa). However, the decay of the assumed yield stress with temperature should be similar to the decay in the hardness with temperature. The estimate shown in the figure above used the relationship  $\sigma_{yield} = \sigma_{21} \times \frac{H_T}{H_{21}}$ , where  $\sigma_{yield}$  is the yield stress,  $\sigma_{21}$  is the yield stress at room temperature,  $H_{21}$  is the hardness at room temperature and  $H_T$  is the hardness at the given temperature.

Figure 6-24 predicts that there will be a level of plastic deformation in the roll bite, based on a comparison of the yield stress curve against von-Mises stress. This agrees with conventional wisdom in Tata that a certain amount of plastic deformation can take place in and around the roll bite.

## 6.5. Conclusions

---

The study described in this chapter was undertaken with three primary objectives;

1. To discover the effect of varying the oxide layer thickness on the surface of the work roll
2. To discover the effect of using temperature dependent material properties
3. To discover the effect of the thermal history of the work roll

The oxide layer thickness was found to be a significant factor in determining the heat flow between the strip and the work roll. The insulating effect of the thickest expected oxide layer, compared to the thinnest expected oxide layer, was found to create a 3°C difference in the circumferentially averaged temperature at the surface. This resulted in a significant 100MPa difference in the stress analysis. It can be concluded that a thicker surface oxide layer can be beneficial in minimising the temperature rise through the roll bite, although the effects of thicker oxide layers on surface quality and durability need to be assessed before any recommendations can be made.

The temperature dependence of the material properties of the work roll were found to have many effects on both temperature and stress distributions. The temperature dependent models circumferentially averaged surface temperatures were 8°C higher than non-temperature dependent models, although the peak surface temperatures were lower, possibly thanks to improved conductivity at higher temperatures. Despite the lower peak temperatures, the softening effects of the high temperatures, together with higher rates of thermal expansion at higher temperatures, combined to give a 100MPa increase over the stresses predicted by the non temperature dependent models. It is therefore very important to consider the variability of the work roll's material properties with temperature when calculating either the roll's temperature or stress distribution.

Comparison of the effects of varying the work roll's thermal history has provided numerical support for current rolling practices. A roll which is cooled during a fifteen minute mill stop will experience an 8.2% greater stress cycle per revolution in subsequent rolling than if the cooling were switched off. A roll which is entirely cooled to room temperature, or a fresh roll which has yet to pick up any heat from the strip, will experience a 17.3% greater stress cycle per rolling revolution. This aspect of the testing really highlighted the importance of establishing a level radial temperature profile as quickly as possible, since the longer the roll

maintains a high radial temperature difference, the more fatigue damage is likely to be inflicted.

By comparing a selection of the tests described above it was possible to draw another conclusion from the interaction between the temperature profile and the stress profile. The angular temperature profile decides the difference between the maximum and minimum stresses per revolution, while temperature differences in the radial profile offset the entire stress profile. This insight could help engineers target specific improvements to specific problems. If peak stresses are the problem, then minimising radial temperature differences will reduce them. If large stress fluctuations are the problem, then minimising angular temperature differences will reduce them.

Finally, the results from a normal rolling scenario were compared against a best estimate of the yield stress of the material, based partly on manufacturer's data. The comparison suggested that a small amount of plastic deformation could be expected in the roll bite, a finding which agrees with the commonly held view.

## 7. Conclusions & Recommendations

---

### 7.1. Conclusions

---

The focus of this project was split fairly evenly between the thermal modelling and stress modelling aspects. Minor adjustments were made to the existing core models on the advice of the previous project author, before a first foray into ABAQUS revealed inconsistencies in the shell model outputs. A lot of the project time was spent in tracking down the inconsistencies and picking apart the combination of errors which were causing them. Later visits to the IJmuiden RD&T department introduced the further development tasks of adding temperature dependency to the material properties of the work roll and of implementing a new relationship for the heat transfer coefficient in the roll bite. The resulting thermal models are significantly more robust and flexible. They also agree with each other very well under conditions where the varying material properties are fixed for comparison and predict similar thermal behaviour to comparable studies from the literature.

The stress modelling side of the project showed that, with current software and hardware limitations, it is not possible to accurately model the three-dimensional stress state of a work roll using a single model. However, a family of nested specialised three-dimensional "sub-model" models was shown to achieve the same effect as a much higher resolution single model for a fraction of the required resources. A detailed two-dimensional model was also shown to work with reasonable accuracy, given the right simplifying conditions, and provided further evidence that the rapid spatial variation seen in the stress profile was a genuine feature of the roll problem and not the result of numerical instability.

Each of the boundary conditions of the stress models were tested to assess their contribution to the overall stress profile. The thermal stresses were demonstrated to dominate over the other components, agreeing with the conventional wisdom within Tata. The mechanical stresses were split into individual components, normal pressure from contact with the backup roll, normal pressure and shear stress from contact with the strip, and work roll bending. While the normal pressures made a small difference to the stress profile, variation of the distribution of pressure across the contact areas made no significant difference. The shear stress in the roll bite also made very little difference to the results as far as the purposes of this project are concerned. The effect of work roll bending on roll stress was miniscule. A residual stress profile was assumed for the roll, based on the best available experimental data. The data available was in the form of hoop stress variation with depth so ABAQUS was used to "fill in the gaps" by enforcing internal equilibrium when the hoop stress



component was specified. Adding the residual stresses to the thermal stresses made a considerable difference, reinforcing the view that residual stresses are an important part of the total stress profile.

The study performed at the end of the project used the software developed throughout the project to extract real process data from the hot mill database and use that data to generate a series of thermal histories. A set of key times in each process history was then subject to a full thermal and stress analysis. The results showed that;

- Temperature dependency of material properties can make a significant difference to the predicted temperature and stress profiles.
- Oxide layer thickness plays a large part in determining the heat flow between the strip and work roll, and therefore in determining the work roll temperature and the magnitude of the incurred stresses.
- Circumferential temperature differences control the range of cyclic stress variations, radial temperature differences control the magnitude of the minimum stress.
- A small amount of plastic deformation is likely to occur in and around the roll bite

## 7.2. Recommendations

---

While every effort has been made to ensure that heat flows into and out of the models are reflective of real process conditions and that internal heat conduction is based on realistic material behaviour, unexpected developments left no time for experimental validation to be attempted. A serious validation study is highly recommended, so that the models can be used in a directly targeted manner.

The discovery that circumferential conduction is negligible has significant ramifications for the structure of the thermal models. If the shell model could be collapsed by one dimension it could be made transient, opening up numerous possibilities. For example, one such possibility could be the simulation of the period immediately after a strip enters the roll bite, a period known to be dangerous to work roll health.

Given the importance of the residual stress profile as part of the total stress state of the work roll, more information on residual stresses is desperately needed. A study of what the residual stresses are and how they evolve through the working life of the roll will provide part of the foundation for accurately predicting when rolls are likely to suffer damage. A study of how the residual stresses are created will potentially allow informed changes to manufacturing processes, producing longer-lived rolls.

Following the brief comparative study described in Chapter 6, it seems that oxide layer thickness is a parameter whose control could provide big benefits. Results suggest that a thick, stable oxide layer can help to protect a work roll from the extreme conditions of the hot rolling process. As such, a study into what the most beneficial oxide layer thickness is, in terms of minimising damage while ensuring good surface quality, and how to control that oxide layer thickness, through optimising process conditions and specifying roll chemistry, is highly recommended.

Finally, the results suggest that radial temperature differences can be very detrimental to work roll health by increasing peak stresses. This is already reflected in operating procedures that limit cooling when there is no strip in contact with the roll. Pre-heating the rolls would be one way to minimise radial temperature gradients, helping to reduce the chance of damage or failure early on. Alternatively, ensuring that more challenging grades are only attempted after the rolls have reached a stable working temperature is advisable.

## Appendix A. Software Users Manual

---

### A.1. Prerequisite software

---

To run the thermal models the following software is required:

- "Bens Work Roll Temperature Model.exe"
- .NET framework 4.0
- Zedgraph.dll – (should be included with the executable)

The models were developed on a computer running the Windows 7 Operating system. Other requirements may present themselves for computers running different operating systems. Further development of the models will require access to the source files ("Bens Work Roll Temperature Model.sln" and associated files) and Visual Studio 2010. Other IDEs (Integrated Development Environments) may also work, depending on their compatibility with VS2010 solutions.

To run the stress models the computer will also need:

- ABAQUS v6.10 – (If this version is not available then the script files will need to be adjusted)
- A compiler for Visual Fortran 10 or Visual Fortran 11 – (i.e. the version range specific to the version of ABAQUS being used)
- A compiler for Visual C++

The process of running the stress models involves running external applications from the C# software. If the software is run from a different operating system then this process may have to be adapted.

### A.2. Input files

---

Many of the data files required for the thermal and stress models can be produced from within the software. However, certain files are required at the outset:

- PropertiesFile.csv – This file contains the current model properties. NB any values replicated in the hot mill database are overwritten using the process data.
- FMSQLQuery.txt and RMSQLQuery.txt – These are the query files that are used to access the hot mill database.

- vstedenmap.csv – This is the cooling map used to create the heat transfer coefficients at the work roll surface.
- AngularProfile.csv – A file containing a set of values which represent the expected temperature profile in the circumferential direction, relative to the mean surface temperature. This is used by the core model to adjust heat flow to the cooling sprays and should be two nodes longer than the desired circumferential mesh size

All of the files listed above should be present in "C:\Bens Work Roll Temperature Model\Input\"

### A.3. The main menu

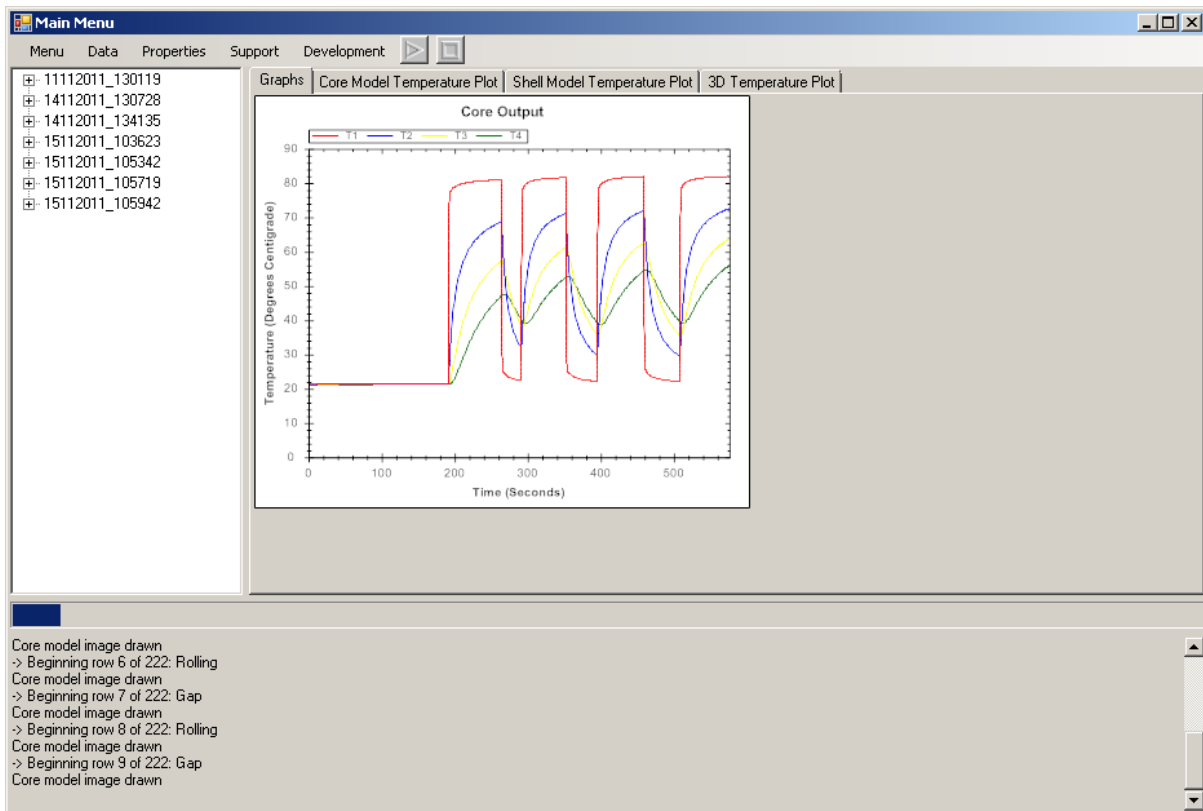


Figure A-1: The main menu

Figure A-1 shows the main menu of the software. At the bottom of the figure is the log box, a control which displays update messages to the user. Just above the log box is the progress bar, which gives the user a graphical indication of how much of the analysis has been done. Above the progress bar and on the left is the solution explorer.

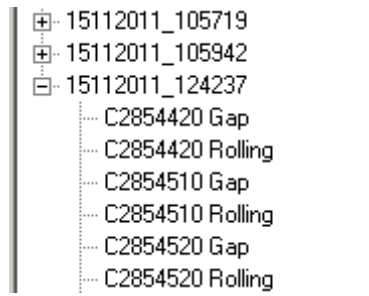


Figure A-2: The solution explorer

Figure A-2 shows the solution explorer in greater detail and with one of the solutions expanded. Each solution is identified by the date and time at which the simulation was performed, while each sub-item represents a stage in the analysis, i.e. the end of a gap between strips or the end of a period of rolling. Each stage of the analysis is labelled with the coil number of the strip which was being processed at the time.

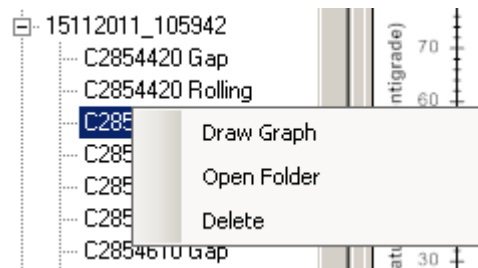


Figure A-3: Context menu

Figure A-3 shows the options available in the context menu (raised by right-clicking on any item) when a non-rolling stage is selected. The "draw graph" command draws different charts and figures, depending on whether a single stage or the parent solution is selected. For a single stage, a temperature history graph of the selected stage is drawn, as well as a 2D false-colour temperature plot of the core model at the end of stage. If the parent solution is selected, the temperature history for the whole solution is presented, as well as an animated control that allows the user to flick between 2D false-colour temperature maps that have been drawn using the same scale. "Open folder" opens the solution folder in windows explorer, for direct access to files. The "delete" command will delete the selected folder, along with its contents (There is a message box prompt to prevent accidental deletion).

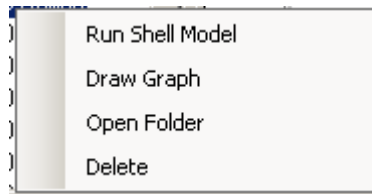


Figure A-4: Rolling stage context menu

Figure A-4 shows one of the context menus that is raised when a rolling stage result is right-clicked. This context menu corresponds to a rolling stage which only contains core model data. The new option, "Run Shell Model", loads the necessary information from the saved results files and runs the shell model.

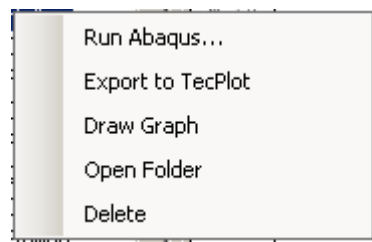


Figure A-5: Rolling stage context menu

Figure A-5 shows the context menu which corresponds to a rolling stage with both core and shell model data. The "Run Abaqus..." command opens the Abaqus form, allowing the user to set up and run a stress analysis. "Export to TecPlot" saves the temperature data in a format which can be read by the TecPlot software package, allowing for more advanced graphing and 3D results representation. With shell model results available, the "Draw Graph" command will also add a new control to the "Shell model temperature plot" tab of the main menu, allowing the user to flick through 2D false-colour temperature plots at various depths below the surface.

To the right of the solution explorer is the results pane, a tabbed control which is used to present results to the user. Different results will be available depending on how many different types of analysis have been performed at any given time. The graph shown in Figure A-1 represents the temperature history of four points at and near the roll surface and is shown during a core model simulation run.

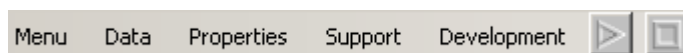


Figure A-6: Menu bar

At the top of the main menu form is the menu bar, as shown in Figure A-6. The "Menu" item contains non-model specific functions. The "Data" and "Properties" items open separate

windows for manipulating model data and properties. The "Support" and "Development" items open separate windows or run functions that convert awkward file types or perform error checks. Each of the functions contained in each menu item will be discussed in length later in this appendix. Finally, the play and stop buttons on the far right of the menu bar allow the user to launch an analysis or stop an analysis that is currently running.

#### A.4. SQL data and processing

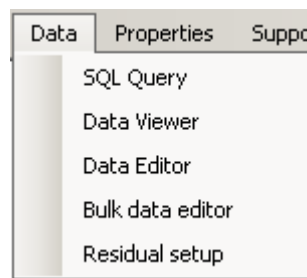


Figure A-7: Data menu items

Figure A-7 shows the items contained in the "Data" menu:

- SQL Query – A windows form which automates the process of extracting data from the hot mill database.
- Data Viewer – Supplanted by...
- Data Editor – An xml viewer and editor, allowing individual values to be adjusted and rows to be added or removed.
- Bulk data editor – A compact form which makes sweeping changes to very large data sets a little easier.
- Residual Setup – Generates residual stress profiles for the stress model.

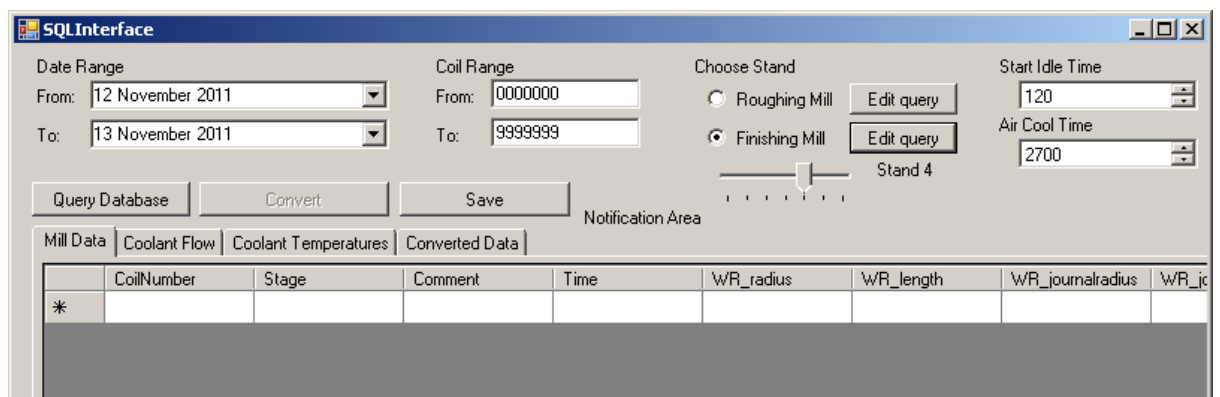


Figure A-8: SQL Interface

Figure A-8 shows the SQL Interface form. This form provides a simple interface for interacting with the hot mill database, automating the data retrieval and processing tasks. The range of process data extracted from the hot mill is defined by the date and coil range parameters. Any data extracted must have been recorded within the date range and also belong to a coil whose ID number lies within the bounds of the coil range parameters. So for a specific date range, set the coil range excessively wide, or for a specific coil range, set the date range excessively wide.

To the right of the date and coil range parameters is the group of controls which pick the stand that will be addressed. The radio-button controls labelled "Roughing Mill" and "Finishing Mill" select between the RMSQLQuery.txt and FMSQLQuery.txt query files, respectively. Since there are a number of stands in the finishing mill, a sliding control is provided to select between them if the finishing mill query is chosen. The "Edit query" buttons open up the query files in notepad for easy editing.

Finally, the "start idle time" and "Air cool time" values are used to create extra periods of thermal activity to simulate the time before and after the work roll is being actively used. With all of the options selected, the "Query Database" button will adjust the selected Sql query file based on the provided parameters and download the requisite information from the hot mill database. The downloaded data is displayed in data grid view controls in the first three tabs (entitled "Mill Data", "Coolant Flow" and "Coolant Temperatures") for perusal before any processing is performed. This gives the user a chance to check that the range of data is correct and there are no obvious errors present.

The "Convert" button becomes available when data is held in the three grid views. This button populates the "Converted Data" table, processing the data to make it digestible by the thermal and stress models. The "Save" button will then allow the user to specify a file name and destination directory for the resulting .xml file, which will be saved for later use.



CoilNumber	Stage	Comment	Time	WR_radius	WR_length	WR_journalradius	WR_journallength	WR_camber
2854420	0	Gap	190.6	0.353325	2	0.2	1	0
2854420	1	Rolling	72.6	0.353325	2	0.2	1	0
2854510	2	Gap	26.8	0.353325	2	0.2	1	0
2854510	3	Rolling	60.9	0.353325	2	0.2	1	0
2854520	4	Gap	42.99999999999...	0.353325	2	0.2	1	0
2854520	5	Rolling	64.4	0.353325	2	0.2	1	0
2854610	6	Gap	48.9	0.353325	2	0.2	1	0
2854610	7	Rolling	68.4	0.353325	2	0.2	1	0
2854710	8	Gap	34.3	0.353325	2	0.2	1	0
2854710	9	Rolling	68.1	0.353325	2	0.2	1	0
2854810	10	Gap	71.2	0.353325	2	0.2	1	0
2854810	11	Rolling	86.8	0.353325	2	0.2	1	0
2854910	12	Gap	33.5	0.353325	2	0.2	1	0
2854910	13	Rolling	65	0.353325	2	0.2	1	0
2855010	14	Gap	40.6	0.353325	2	0.2	1	0
2855010	15	Rolling	46	0.353325	2	0.2	1	0
2855110	16	Gap	67.8	0.353325	2	0.2	1	0
2855110	17	Rolling	74.9	0.353325	2	0.2	1	0

Figure A-9: The Data Editor form

The third option in the "Data" menu is the "Data Editor" form. This is an improved version of the "Data Viewer", mostly identical save for the ability to add and delete rows, providing more editing flexibility. Clicking on the scale at the left edge of the control will highlight a row, allowing the user to delete it by pressing the "DEL" or "delete" key. Right-clicking on a cell will bring up a context menu, allowing the user to insert a row below the selected cell. Any selected cell can be edited by typing in a new value, the "Save Data" button saving any changes to a new file or overwriting the original.

Figure A-10: The Bulk data editor form

The next item in the "Data" menu is the bulk data editor, represented in Figure A-10. The form is designed for changing values en-masse and so does not boast the same expansive data display. The "Open" button allows the user to select the data file containing the data which needs to be edited. This file will not be affected by the process. Once the file is open, the user can choose to adjust the entire data set, or just those rows of the data set which correspond to mill faults, or the empty time between strips being rolled (a rolling "Gap"), etc. The "File

length" value shows how many rows there are in the data set. Typical options are shown in Figure A-11.

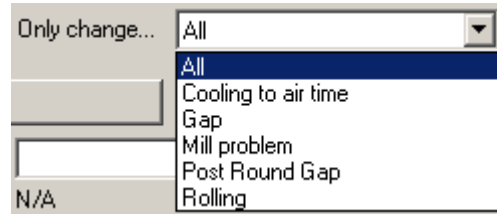


Figure A-11: Update every value, or just those rows representing a specific kind of stand behaviour

The column parameter defines which column of the data set should be adjusted, so in Figure A-10, the "Tstrip" parameter for every "Rolling" data row will be adjusted. The "Typical value" figure shows the average of (in this case) every "Tstrip" parameter belonging to a "Rolling" data row. A new value can be entered below and, by clicking the "Replace" button, applied. The "Save" button brings up a file dialog, allowing the user to save the adjusted data set to a new file or overwrite the original file.

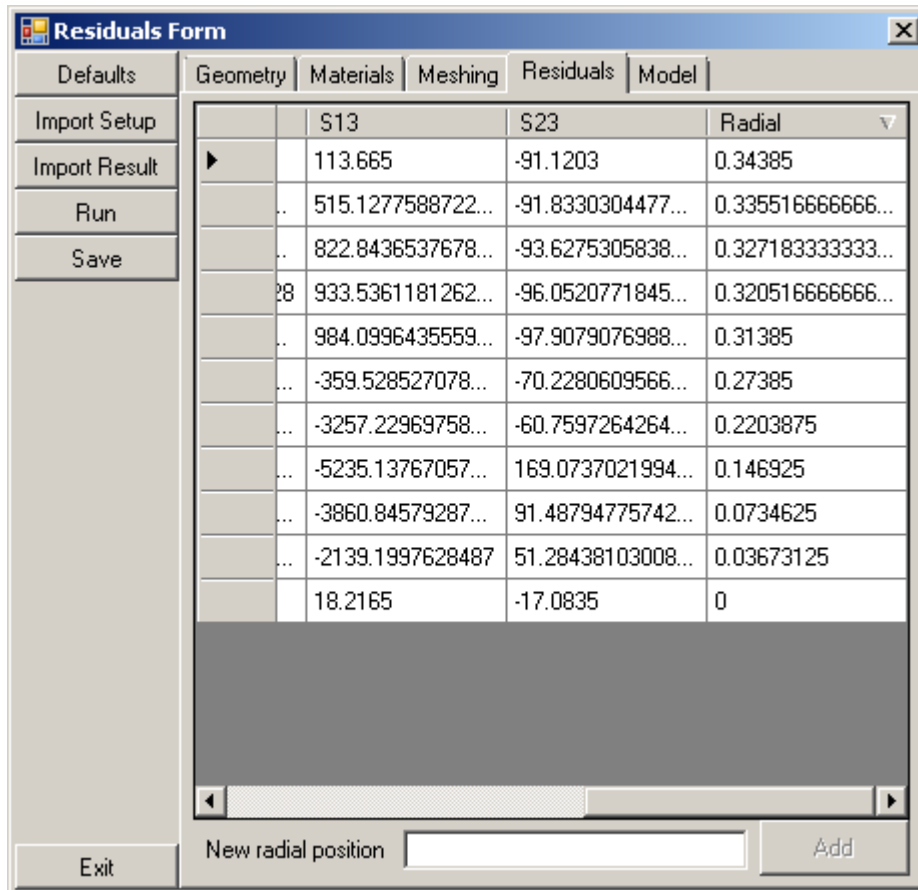


Figure A-12: The residuals form

The final item in the "Data" menu is the residuals form, as shown in Figure A-12. This form comes pre-loaded with default settings, but can be adjusted manually or updated from any

saved properties file. The displayed tab shows a portion of a residual stress profile which has already been balanced. The far right column gives the radial coordinate, with S11, S22, S33, S12, S13 and S23 referring to radial, hoop and axial stresses and the three shear components, respectively. Each stress value can be manually set to any value, by typing the new value into the appropriate box. Any selected row can be deleted by pressing "DEL" and new rows can be added by adding a new radial coordinate into the "New radial position" text box, then clicking the "Add" button.

The purpose of the form is to "fill in the blanks" of a stress profile which is incomplete, i.e. missing any measurement for a number of the stress components. As such, any cell in the Residuals tab which is left at 0 (zero) will be assumed to be unknown and allowed to vary (excepting the "0" radial coordinate). Pressing the "Run" button begins the stress balancing process (this process requires all of the software associated with the stress analysis to be installed, as discussed in section A.1) and automatically switches to the "Model" tab.

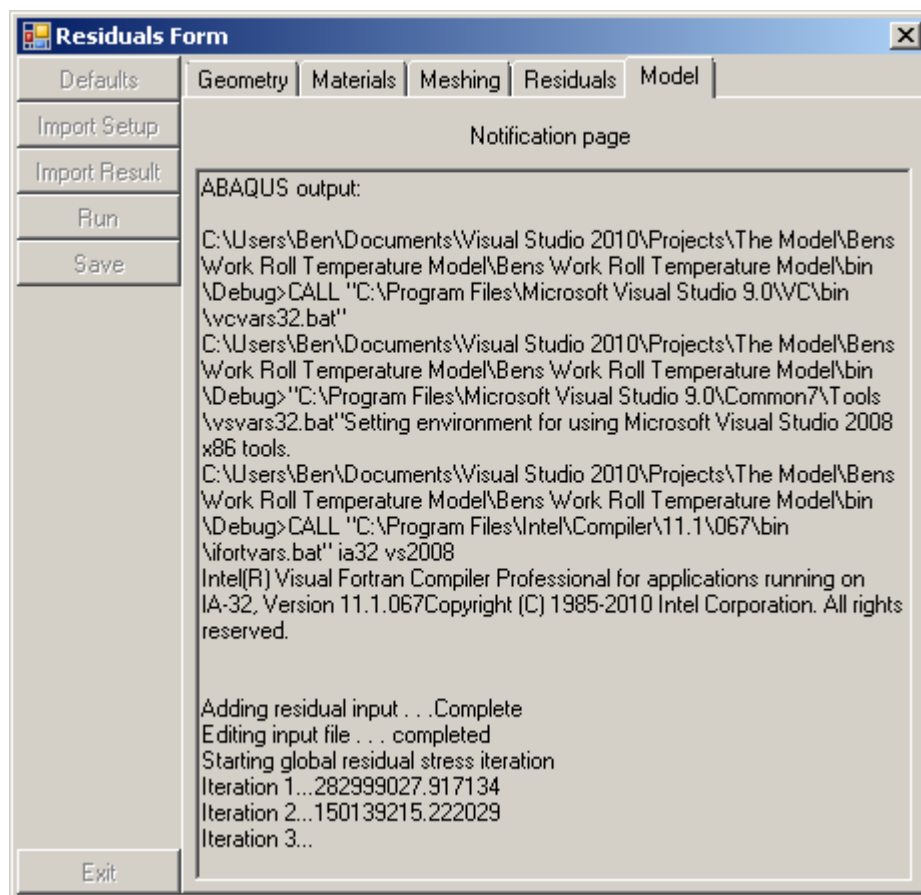


Figure A-13: The "Model" tab on the residuals form

Figure A-13 shows the view presented to the user on pressing the "Run" button. Most of the buttons are disabled to prevent the user from changing anything while the analysis is being performed. The "Model" tab is mostly taken up with a large log box, which contains output

diverted from ABAQUS, allowing the user to track the progress of the model. This usually takes quite a long time. The process is iterative, repeating until the change in the predicted outcome has become relatively small. A measure of convergence of this loop is provided next to the iteration count, in the log box, as the current error per iteration divided by the target error per iteration. The process is converged when the measure reaches 1.

Once convergence has been attained the "Residuals" tab is updated with the new results. The "Save" button creates a file from the residuals data, ready to be applied to any stress analysis (providing the correct radial scale has been used).

## A.5. Thermal models

---

The thermal models are run by clicking on the "play" icon on the right side of the menu bar. This will open up the properties window, shown in Figure A-14.

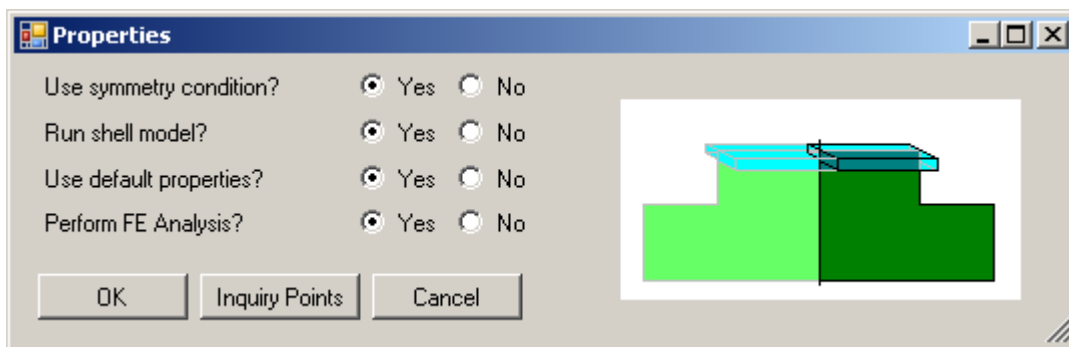


Figure A-14: The properties form

This basic setup menu allows the user to choose the components of the analysis which should be run, whether the models should be symmetrically reduced (generally advisable) and whether default properties should be used. Generally speaking a completely full analysis will not be required, so the shell model and FE analysis can be disabled. These analyses can still be performed at a later date, based on the core model results, without the overhead of running them for every rolling stage in the data set.

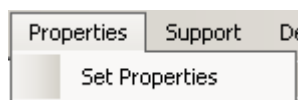
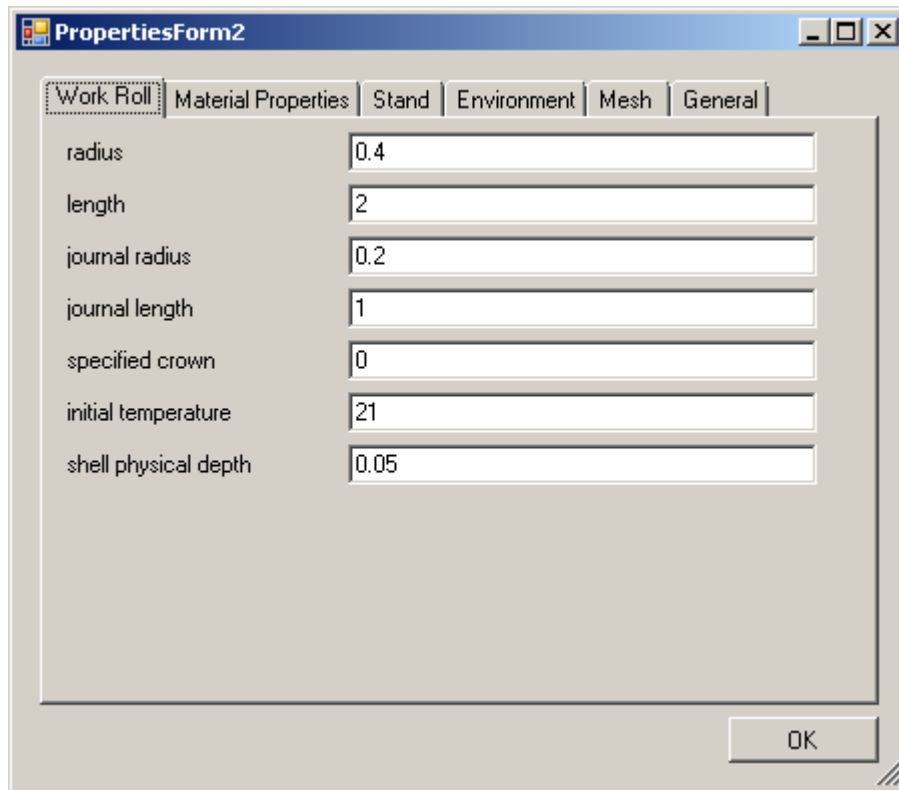


Figure A-15: Editing the properties without running a simulation

The properties form can also be accessed without running a simulation, through the properties menu item shown in Figure A-15. The steps in Figure A-14 will still need to be

performed when the "play" icon is clicked, but the default properties can be changed by selecting "no" for "use default properties?".



The screenshot shows a window titled "PropertiesForm2" with a tabbed interface. The "Work Roll" tab is selected. The form contains several input fields with the following values:

Parameter	Value
radius	0.4
length	2
journal radius	0.2
journal length	1
specified crown	0
initial temperature	21
shell physical depth	0.05

An "OK" button is located at the bottom right of the form.

Figure A-16: Default properties form

Figure A-16 shows the default properties editing form. Many items, such as the work roll radius ("radius") will be overwritten by data retrieved from the hot mill database. Where process information is not available, properties information is used instead, so changing such properties as the "shell physical depth", which dictates the thickness of the work roll shell material, can have a significant impact on results. A default set of values is provided with the model in "properties.csv", which is the file used by this form and should be backed up.

Most values can be adjusted as the user sees fit, although parameters in the "mesh" tab should be treated with care. Some parameters control the Abaqus model building process, so changes can cause unforeseen complications. The "general" tab contains two radio controls, for "debug mode" and "animation". When "debug mode" is enabled the core and shell models will save a number of extra files, such as solver coefficient and heat transfer coefficient arrays. The "animation" control prompts the core model to save the required data to compile the solution animation described previously.

When the model properties have been chosen the user will be prompted to select an .xml data file. Any compatible .xml file will work, such as files created by the SQLQuery interface form. The core model will then work through the data file row by row, running a simulation

for each stage, saving the results into a dedicated folder and updating the log box as it goes. The results can then be accessed graphically using the "Draw Graphs" functions, or through the raw data files which can be found in "C:\Bens Work Roll Temperature Model\Output\" or through the "Open folder" item in the context menu.

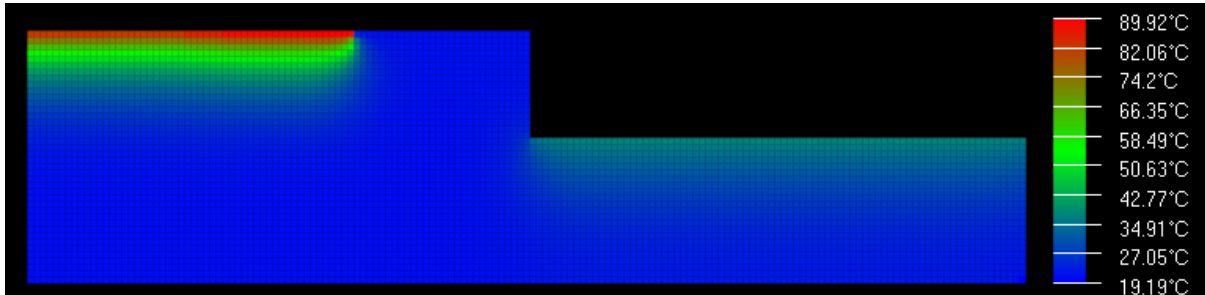


Figure A-17: Example false-colour graph

Figure A-17 shows an example temperature map, as might be produced following a symmetrically reduced core model analysis. The outline of the roll is represented and labelled in Figure A-18.

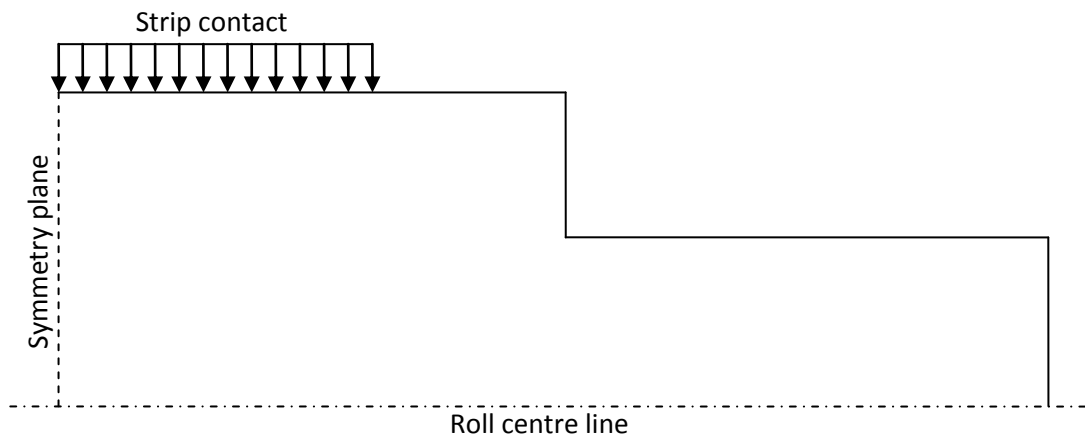


Figure A-18: false-colour graph outline

## A.6. Stress models

---

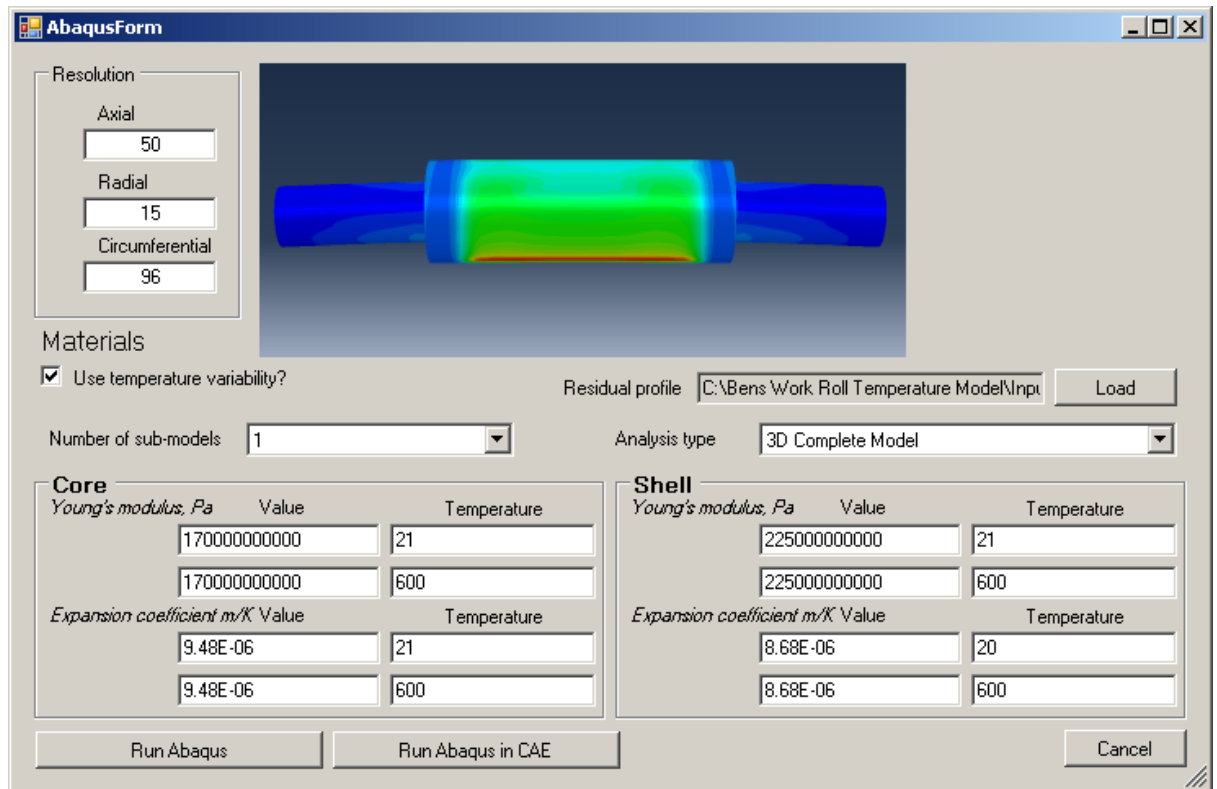


Figure A-19: Abaqus forms interface

The "Run Abaqus..." option is presented to the user if they right-click on a results item in the main menu and the item contains shell model data. Selecting the "Run Abaqus..." option opens the form shown in Figure A-19. The form allows the user to run a stress analysis of a work roll, using the temperature data developed by the thermal models. A number of customization options are available.

- Resolution – The radial, axial and circumferential resolutions can be adjusted. The mesh for the stress models is not perfectly regular, concentrating mesh density near the model surface, so the values are indicative rather than prescriptive. Larger numbers will result in a finer mesh, though care should be taken to vary the values evenly to reduce the generation of distortions.
- Material Properties – The temperature variability term allows the user to turn on or off temperature variability. If temperature variability is enabled then the user can input a value for the young's modulus and expansion coefficient for two different temperatures, for both shell and core materials. In the subsequent stress analysis, Abaqus will implement a linear temperature dependency of the two properties.
- Number of sub-models – The software makes three stress models available. The global model simulates the whole volume of the work roll, the surface model only simulates the surface, and the bite model simulates the area

immediately surrounding the roll bite. Each model uses the results of the previous models to ensure that a realistic boundary condition is applied at surfaces which are inside the work roll geometry. The control next to the "number of sub models" label allows the user to choose how many of these models should be used. 0 – Global model, 1 – Global and surface models, 2 – Global, surface and bite models.

- Analysis type – Allows the user to choose which combination of thermal, mechanical and residual stresses should be applied.
- Residual profile – the process of generating a residual stress profile is time-consuming, so residual stress profiles are generated in an independent menu. The "Load" button allows the user to select a saved profile, so that its effects can be combined with other stress generating effects.

The "Cancel" button will exit the form without initiating an analysis. "Run Abaqus in CAE" is a debugging option, opening the user interface of ABAQUS and loading the scripts used to build the model. This option will not result in any actual analysis being performed but will build all of the selected models so that the meshing quality, application of boundary conditions, etc can be inspected. Any errors in the build process will result in an error message box, providing a more information than when the user interface is suppressed.

The "Run Abaqus" button will run ABAQUS with the user interface suppressed. Standard outputs are redirected to the same log box used by the thermal models to relay basic information to the user while the simulations are underway. As a note of caution, ABAQUS can be very resource-intensive, so other programs may run considerably slower while a stress analysis is being performed. At the end of the analysis the results are output to a text file which is then converted into an \*.sdf file, the format for an SQLCE database. The original \*.odb files, ABAQUS's own results format, are left in place so that advantage can be taken of ABAQUS Viewer's visualisation tools. The "Report reader" software, discussed later, is available to the user for the purposes of inspecting the data, extracting any sections of interest and saving results to \*.csv files or copying in a format suitable for pasting directly into excel.

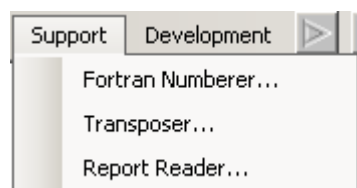


Figure A-20: "Support" main menu item



A few other options are available through the "Support" tab of the menu bar. "Fortran Numberer" is a small program which can be to automatically modify line numbers in fortran code. Simply paste the code which needs to be re-numbered into the text box of the "fortran numberer" form and press the "Re-Number" button. The "copy" button will then copy the contents of the text box to the clipboard, so it can be pasted back into the source code file.

The "Transposer" is another small software package which transposes (i.e. rows become columns, columns become rows) data presented in comma-separated-variable form. The form function is similar to the "fortran numberer".

"Report Reader" is a more complicated program which allows users to interact with the "\*.sdf" output files created after running an ABAQUS job.

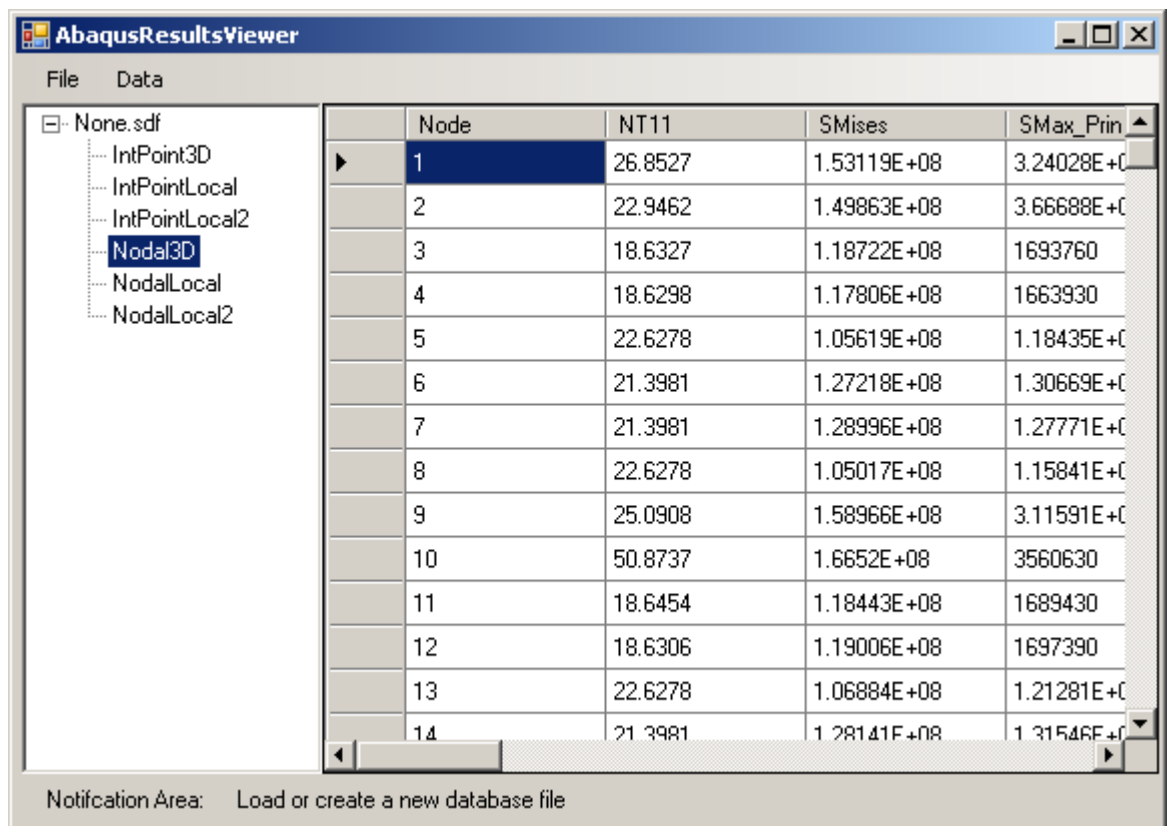
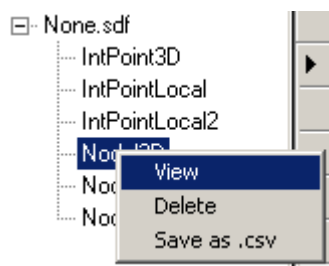


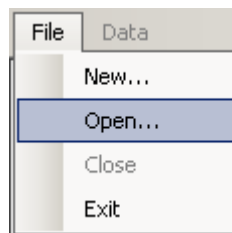
Figure A-21: Report Reader main menu

Figure A-21 shows the main menu of the "Report Reader". The form has a menu bar, similar to the main menu, and a solution explorer on the left hand side. However, in this case, each item in the solution explorer represents a different results file. For a full Abaqus analysis, 6 results files are created, two for each model, with "3D" referring to the global model, "Local" referring to the surface model and "Local2" referring to the bite model. Each model produces two results files, one containing data taken from node points, the other containing data taken from integration points. For the element types used by the current models, node points correspond to the corners of each element, integration points correspond to the centroid.

To the right of the solution explorer is the data grid view. When a results file in the solution explorer is right-clicked a context menu is displayed, as shown in Figure A-22. The "view" option displays the results in the data grid view, which can be used to directly inspect the data. The data can be order in ascending or descending numerical or alphabetical order, depending on the data type of the column selected, by clicking one of the column titles. The "Delete" option deletes the selected results (the user will be prompted to confirm), while "Save as .csv" brings up the standard windows "Save file dialog" and then writes the data to file at the desired location, in comma-separated-variable format. NB:- an option to "Copy to clipboard" has been added, which copies a tab-delimited version of the results to the clipboard, allowing it to be pasted directly into excel.

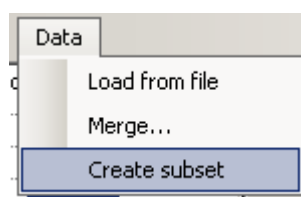


**Figure A-22: The solution explorer's context menu**



**Figure A-23: The file menu**

The "File" menu of the menu bar contains commands for opening and closing files. The "New" command creates an empty data set and generally has little practical value, but has been left in for posterity. The "Open" command uses the windows standard "Open file dialog" to allow the user to select a file to open. The "Close" command closes the connection to the current database, allowing a new one to be opened. "Exit" closes the Results Reader form.



**Figure A-24: The data menu**

The "Data" menu allows the user to select from a number of options for importing and editing data files. The "Load from file" option allows the user to select a "\*.rpt" file, a text file which can be output by ABAQUS containing a selection of data. The data will be loaded into the current database as another results item in the solution explorer. The "Merge" option requests that the user select two abaqus "\*.rpt" files, one containing coordinate data, the other containing the rest of the data. This had to be implemented to circumvent an error in ABAQUS which replaces the coordinate data with random noise when a coordinate transformation is performed. The "Merge" option reads in the coordinate data and performs the coordinate transformation manually (from rectangular to cylindrical-polar coordinates). The rest of the data is then loaded from the second file and the two data sets are combined. This command is used by the main program software automatically to create the "\*.sdf" file which results from each stress analysis.

The final command, "Create subset", allows the user to isolate a part of a results file using a simple rule-based system.

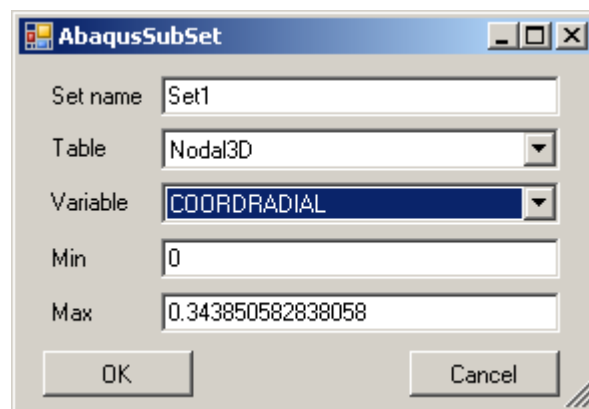
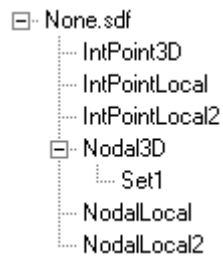


Figure A-25: Subset form

Figure A-25 shows the "create subset" form. The set name is self explanatory. The "Table" picker allows the user to select from the available results sets. The "Variable" picker allows the user to choose which variable will be used to identify which entries should be included in the subset. When a "Variable" is selected, the minimum and maximum values in that column are automatically calculated. Consider a situation where only the results from the surface of the roll is required, say from a depth of 0.3m and greater. Changing the minimum value to 0.3 and pressing "OK" will create a new set of data by copying only those entries of the "Nodal3D" table whose "COORDRADIAL" value is between 0.3 and the calculated maximum



**Figure A-26: A new subset**

Figure A-26 shows the new subset in the solution explorer. Subsets appear as children of the data set from which they were created. The subset can be used in the same way as any of the other results files, viewed in the data grid view, saved to file, deleted or copied to the clipboard. If more refinement is needed, maybe with reference to the values of a different variable, a new subset can be made using a subset as its parent. The resulting "Set2" will still be a child of the "Nodal3D" data set, but will not include any values which were excluded from "Set1".

## Appendix B. C# Code

---

This appendix describes the main features of the C# code developed in this project. A level of familiarity with the C# programming language is assumed.

### B.1. Available classes

---

A large number of classes are included in the program developed during this project. The list below briefly introduces those that are of use to the finished software package.

#### Interface Items

- Main.cs – acts as the entry point for the program.
- MainMenu.cs – The main user interface which provides access to all of the project's functionality.
- DataEditor.cs – A user interface to allow easy access and visualisation of process data.
- ImageGenerator.cs – Draws the 2D and 2.5D core and shell false-colour temperature maps.
- ImageViewer.cs – Provides a control for users to look at, copy and save the false-colour temperature maps.
- Graphs.cs – Used to draw the temperature history graph (Makes use of the ZedGraph library, an open source graphing library available (as of 18/11/2011) from the URL: <<http://sourceforge.net/projects/zedgraph/>>)
- SQLInterface.cs – A user interface which allows users to automatically extract and process all of the required data from the hot mill database.
- SQLQuery.cs – A programming object which makes a connection with an SQL database, passes a query and returns a result.
- PropertiesForm.cs – A user interface which lets the user choose between basic model setup types.
- PropertiesForm2.cs – A user interface which allows users a more comprehensive property editing facility.
- BulkDataEditor.exe – A separate project, allows bulk editing of large process data files.
- AbaqusResultsViewer.cs – A form which allows easier manipulation and exporting of stress results data to SQLCE database files and Microsoft Excel-friendly formats.

## Thermal classes

- CoreModelTDP.cs – the core model, rewritten to have temperature dependent properties.
- ExpandedCoreModelTDP.cs – A class inheriting from the core model and overriding methods to expand the symmetry condition.
- ShellModelTDP.cs – The shell model, rewritten to have temperature dependent properties.
- ExpandedShellModelTDP.cs – A class inheriting from the shell model and overriding methods to expand the symmetry condition.
- ConservativeCoreModel.cs – The core model, rewritten to implement temperature dependent properties while properly conserving energy.
- ConservativeShellModel.cs – The shell model, rewritten to implement temperature dependent properties while properly conserving energy.
- RollModelWithBearing.cs – The model which organises the core and shell models.
- ExpandedRollModel.cs – A class inheriting from the roll model and overriding variable declarations to use the expanded versions of the models.
- StandModel.cs – The model which breaks the input data files into stages of analysis and sets the time step lengths.
- ExpandedStandModel.cs – A class inheriting from the stand model and overriding variable declarations to use the expanded version of the roll, shell and core models
- CoolingDesigner.cs – Uses process data and the provided cooling map to generate an array of heat transfer coefficients for the core and shell cooling spray boundary conditions.
- FrictionType.cs – An enumeration declaration used to choose between different types of roll bite behaviour.
- HeatPenetration.cs – Contains all the methods necessary to calculate the heat penetration depth, the depth at which the shell model will be based.
- Maths.cs – Contains definitions for mathematical functions which are not included in the default Math library.
- TemperatureHistory.cs – A class definition for an object which stores the change of temperature with time in the core model. Used to help draw the temperature history graphs on the main menu.

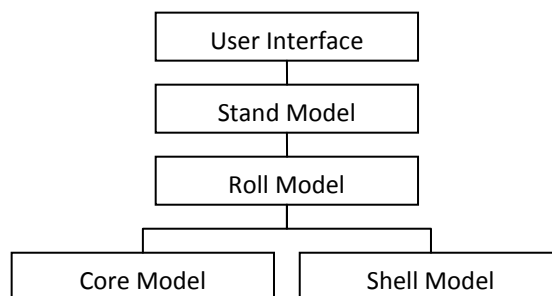
## Stress classes

- AbaqusBuilder2.cs – Responsible for writing all of the file required by ABAQUS to run.
- AbaqusProps.cs – An object which encapsulates the settings which should be used to build the ABAQUS model.
- AbaqusType.cs – An enumeration of the different types of ABAQUS analysis that can be performed.
- Contact.cs – Contains methods to calculate contact sizes and pressures.
- ResidualsForm.cs – A user interface object which sets up the residual stress profile for the work roll.
- AbaqusForm.cs – A user interface to allow the user to set up a stress analysis
- CoilPicker.cs – A small user interface which presents the user with a selection of coil numbers. Used in conjunction with other classes.
- VaryingDouble.cs – A simple data structure to carry temperature varying property data.
- OutputReader.cs – Translates ABAQUS report files into SQLCE database files, for easier and more practical manipulation.
- FileOperations.cs – Organises the writing and reading of files
- ModelProperties.cs – Encapsulates all of the material, geometric and process data required by the thermal and stress models.

## B.2. Thermal models

---

This section will focus on how the classes interact, rather than on how individual functions work. The mathematics behind all of the activity is described in the body of the thesis and any precise information can be found in the code files on the attached CD.



**Figure B-1: Model hierarchy**

## B.2.1. Core-Roll interaction

The roll model is responsible for creating the core model and passing it the data it needs to function.

```
public ConservativeCoreModelTDP cmodel3;
```

**Listing B-1**

Listing B-1 shows the variable declaration, which states that the roll model contains an instance of the ConservativeCoreModelTDP class, under the name cmodel3.

```
cmodel3 = new ConservativeCoreModelTDP(mp);
```

**Listing B-2**

Listing B-2 instantiates the core model from inside the constructor of the roll model. This means that any instance of a roll model will always contain an instance of a core model, and that this will always be the same instance of the core model, ensuring that data is carried between rolling stages. The "mp" variable is a "modelproperties" object, containing almost all of the data needed to construct the core model.

```
cmodel3.CompletedEvent +=  
    new  
ConservativeCoreModelTDP.CompletedEventHandler(cmodel_CompletedEvent);
```

**Listing B-3**

Listing B-3 occurs immediately after Listing B-2 and ties the event, which is triggered by the core model finishing, to a method in the roll model. The method does nothing in this instance, but could be made to pass messages on to the stand model very easily if the need ever arises.

```
cmodel3.CurrentDirectory = CurrentDirectory + FolderName +  
"\\Roll";  
cmodel3.AnimationDirectory = AnimationDirectory;
```

**Listing B-4**

At the beginning of the "RunModel" method, Listing B-4 is used to set the output directories of the core model. The "CurrentDirectory" is the default directory for file output, while the "AnimationDirectory" is a single directory used to collect core model data from different stages, to combine into a single animation.

```
cmodel3.InterfaceDepth = dmesh;
```



#### Listing B-5

Listing B-5 then sets the distance below the surface of the interface between the core and shell models, in mesh points. This is calculated from the heat penetration depth, as discussed in the White model<sup>36</sup>.

```
cmodel3.RecalculateCooling();  
cmodel3.Reload(properties);  
cmodel3.Build();
```

#### Listing B-6

Listing B-6 then rebuilds the core model. If the core model has not been used before then the first cooling map is generated, the property data are loaded and the core model is constructed from the combined data. If the core model has been used before then the process is basically the same, but some values from the previous run are overwritten and boundary conditions are recalculated based on the new values.

```
cmodel3.RunModel(executeshellmodel || LastTimeStep);
```

#### Listing B-7

Finally, Listing B-7 runs the model, passing "True" as a parameter if the shell model is due to run after the core model is finished or if this particular model run concludes with the last time step for the current data set. In either case the  $T'$  term, representing the change of temperature with time for each point in the core model, is calculated and saved at the end of the run, for later use or to aid debugging efforts.

### B.2.2. Shell-Roll interaction

```
smodel3 = new ConservativeShellModelTDP(properties,  
    cmodel3.ReturnTemperatureSlice(hpshelldepth), cmodel3);
```

#### Listing B-8

If the shell model is due to be run, Listing B-8 is used to instantiate it. The constructor requires that the model properties are supplied, along with a one dimensional temperature array which will act as the boundary condition on the lower surface, and the core model itself.

```
smodel3.WriteTextToLogBox += new  
    ConservativeShellModelTDP.WriteTextToLogBoxEvent (smodel_WriteTextToLogBox);
```

#### Listing B-9

Listing B-9 adds one of the roll model's methods to the shell model's event handler, which in this case does pass the message through to the stand model.

```
smodel3.CurrentDirectory = CurrentDirectory + FolderName +  
"\\Roll";  
smodel3.Build();
```

#### Listing B-10

Listing B-10 sets the output directory for the shell model, then runs the "Build" method to construct the model, setting up arrays and calculating boundary conditions.

```
smodel3.RunModel();
```

#### Listing B-11

Finally, Listing B-11 runs the shell model.

### B.2.3. Stand-Roll interaction

```
workroll = new RollModelWithBearing(mp);  
workroll.WriteToLogBox += new  
RollModelWithBearing.WriteToLogBoxEvent(workroll_WriteToLogBox);
```

#### Listing B-12

As with previous models, the first step in using the roll model is to create a new instance of the model and add a method to its event handlers, as shown in Listing B-12.

```
workroll.CurrentDirectory = CurrentDirectory+FolderName;  
workroll.AnimationDirectory = CurrentDirectory + FolderName +  
"\\Animation\\";
```

#### Listing B-13

Next, Listing B-13 passes the roll model the base directory of the whole solution and the directory for animation data to be stored. After these basic settings have been applied, the stand model begins to pick through the process data row by row.

```
workroll.FolderName =  
"\\C"+Convert.ToString(datarow["CoilNumber"])+"  
"+Convert.ToString(datarow["Comment"]);
```

#### Listing B-14

For each row, the roll model is given a new folder name, made out of a combination of the current coil number and the comment attached to that row, i.e. "Rolling", "Gap", "Mill stop", etc.

```
workroll.RunModel (LastTimeStep);
```

**Listing B-15**

By adjusting the time step length and run time it is possible to use Listing B-15 multiple times to slowly increase the time step length. This allows the use of long time steps later on in a stage, when temperatures are not changing very much, while keeping time steps short at the beginning when numerical instability is more of a concern. When the optimum time step length has been reached the "LastTimeStep" value can be set to "True", to inform the roll model that the current call to the "RunModel" method is the last for that stage.

```
tH.AddData (workroll.cmodel3.tH);  
for (int i = 0; i < workroll.cmodel3.tH.GetTime (); i++)  
{  
    t = t + workroll.cmodel3.deltat;  
    tH.AddTime (t);  
}  
workroll.cmodel3.tH.Clear ();
```

**Listing B-16**

The "tH" objects in Listing B-16 are "temperatureHistory" objects. These record the change of temperature with time and are used to draw the temperature change graphs for the main menu. Temperature data and time data are not stored together in the core model, so Listing B-16 takes temperature data from the core model then adds the time data on based on the time step length. Finally the listing clears the core models temperature history object read for the next stage of the analysis.

#### B.2.4. Interface-Stand interaction

```
smodel = new StandModel (convertdatatable, properties);
```

**Listing B-17**

The stand model is instantiated in the same way as the other models, but in this case it is passed a "DataTable" object, called "convertdatatable" here. This data table contains all of the process data for the current analysis. The properties data is added separately.

```

    smodel.CompletedEvent += new
StandModel.CompletedEventhandler (smodel_CompletedEvent);
    smodel.WriteToLogBox += new
StandModel.WriteToLogBoxEvent (smodel_WriteToLogBox);
    smodel.UpdateProgress += new
StandModel.UpdatedProgressEventHandler (smodel_UpdateProgress);
    smodel.NewResultsEvent += new
StandModel.NewResultsEventHandler (smodel_NewResultsEvent);

```

**Listing B-18**

Listing B-18 subscribes to all the stand model events, so that the updates can be made to the log box display and the graphs which are drawn as the model progresses.

```

smodel.Start(true);

```

**Listing B-19**

Finally, the stand model is "Started". The word "Start" is used instead of "Run" because the stand model will be opened in a new thread, allowing the main menu to respond to the user while the thermal models are working. The "true" value passed to the "Start" method indicates that a normal analysis should be run. The alternative is a deprecated "radial expansion model", which is no longer functional.

### B.3. Stress models

---

To run an ABAQUS model requires that a number of files are created, edited and executed in a particular order. The process starts with the "Start" method.

```

public void Start(string filename, bool OpenAbaqus, string
RootDirectory, AbaqusType AnalysisType,
    bool Expanded, int SubModels)
{
    _filename = filename;
    _OpenAbaqus = OpenAbaqus;
    _RootDirectory = RootDirectory.Substring(2, RootDirectory.Length
- 2) + "\\Roll\\ABAQUS\\";
    _Directory = RootDirectory;
    this.AnalysisType = AnalysisType;
    this.Expanded = Expanded;
    subModelCount = SubModels;

    modelthread = new Thread(new ThreadStart(this.RunModel));
    modelthread.Start();
}

```

**Listing B-20**

The method takes a number of input parameters, which correspond with the options in the "AbaqusForm" for customising stress analyses. The file name is the name to be used for

the Python script which will define the construction of the stress models (see appendix C). The "OpenAbaqus" Boolean term defines whether the user interface of ABAQUS should be opened (true) or suppressed (false). The "RootDirectory" is the folder which should be used to store the output files. The "AnalysisType" defines which kinds of loading should be applied to the roll. The "Expanded" Boolean term indicates whether the accompanying temperature data uses a symmetry condition at the roll centre plane. The "SubModels" values indicate how many sub-models should be used, varying from 0 (just the global model) to 2 (the global, surface and bite models). The input parameters are transferred to internal variables and the "RunModel" method is launched in a new thread, for reasons which will become clear.

Listing B-21 presents a portion of the contents of the "RunModel" method. The switch statement launches a different kind of stress analysis for each value of the "AnalysisType" variable. The case presented in Listing B-21 is the "SubModel" value, which includes all of the available loads by passing three "true" values as parameters to one of the "BuildScript\_SubModel" methods. Each "true" value corresponds to thermal, mechanical and residual loading respectively. The first parameter "\_Directory" gives the location of the folder used for storing output data. The "subModelCount" gives the number of sub-models to be used, as described above.

```

switch (AnalysisType)
{
    case AbaqusType.SubModel:
        {
            if (Expanded) BuildScript_SubModel(_Directory, true, true,
            true, subModelCount);
            else BuildScript_SubModelSym(_Directory, true, true, true,
            subModelCount);

            RunScript(_filename, _OpenAbaqus, _RootDirectory,
            subModelCount, true);
        }
    break;
}

```

**Listing B-21**

The "BuildScript\_SubModel" methods produce the python script which defines how the stress model is constructed, along with the temperature data files and the fortran user subroutine files. Once all the files have been created, the "RunScript" method begins the process of using them all.

```

CreateBatchFile(filename, OpenAbaqus, RootDirectory);

```

**Listing B-22**

The first line of the "RunScript" method, given in Listing B-22, creates a batch file. This file allows multiple commands to be pre-loaded and supplied to the command prompt at once.

```
RunProcess = new Process();
RunProcess.StartInfo.FileName = "cmd";
RunProcess.StartInfo.Arguments = "/c \"C:\" + RootDirectory +
"CmdCommands.bat\"";
```

**Listing B-23**

Opening an external program from within a C# program requires the use of "Processes", as shown in Listing B-23. The "StartInfo" represents all of the data which the process will need to start up the desired program. The "FileName" is the name of the program that needs to be run. The simplest solution would be to use the "abaqus" file name to run ABAQUS directly. Unfortunately all programs open through the command prompt, which needs to be initialised to use the C++ and Visual Fortran compilers for the user subroutines to work. The "cmd" file name opens the command prompt directly, while the "Arguments" field specifies that the command prompt should open the supplied file path on opening, i.e. the batch file. The batch file sets up the command prompt window with the required compilers enabled and launches ABAQUS with a reference to the Python script.

```
RunProcess.StartInfo.UseShellExecute = false;
RunProcess.StartInfo.CreateNoWindow = true;
RunProcess.StartInfo.ErrorDialog = false;
RunProcess.StartInfo.RedirectStandardError = true;
RunProcess.StartInfo.RedirectStandardOutput = true;
RunProcess.Start();
```

**Listing B-24**

Listing B-24 finishes setting up the "StartInfo", suppressing any windows from showing (UseShellExecute and CreateNoWindow), prevents any message boxes from popping up (ErrorDialog) and specifying that the standard output and error streams are going to be redirected (RedirectStandardError and RedirectStandardOutput). Finally the process is started.

```
StreamReader outputReader = RunProcess.StandardOutput;
StreamReader errorReader = RunProcess.StandardError;
```

**Listing B-25**

Listing B-25 creates two stream reader objects from the standard output and standard error streams, essentially renaming them for easier referral in later code.

```
if (!outputReader.EndOfStream)
{
    WriteToLogbox("\n" + outputReader.ReadLine());
}
```

```
}
```

**Listing B-26**

While the process is running, the "RunScript" method sits in a loop which will only exit when the process has finished. This is why the "RunScript" needed to be launched in a separate thread, since a loop of this kind would prevent a single-thread application from being able to update the user interface. Listing B-26 sits inside the loop, continually checking that no new messages have been added to the output stream. If any messages are found they are immediately passed to the "WriteToLogBox" event, which passes the message straight through to the main menu where they can appear in the log box used by the thermal models. This keeps the user informed of how the analysis is progressing.

```
FileOperations.WriteToErrorLog(errorReader.ReadToEnd());
```

**Listing B-27**

If an error occurs then the process will finish and the loop can be exited. Listing B-27 then checks the error stream. If there was an error then the "ReadToEnd" method will return any information which ABAQUS makes available, which will be written to the standard error log.

```
if (!File.Exists("C:" + RootDirectory +  
"3DbJob.odb"))  
{  
    WriteToLogbox("Running global model...");  
    EditInputFile("C:" + RootDirectory +  
"3DbJob.inp");  
    if (Residual) DevelopResiduals("C:" +  
RootDirectory + "3DbJob.inp");  
    RunFromInputFile(RootDirectory + "3DbJob.inp",  
"usersubroutine3dG.for");  
    WriteToLogbox("OK\n");  
}
```

**Listing B-28**

For each of the models that have been requested, a variation of Listing B-28 is performed. The opening if-statement checks that a results file for that stage of the analysis does not already exist. This allows a user to delete and re-simulate or add an extra sub-model to existing analyses without having to re-run earlier models.

The "EditInputFile" method opens the input file created by ABAQUS from the instructions in the python script and adds the "COORD" variable to the integration point part of the list of output variables required in the results. This is required to allow integration point data to be used at all, and cannot be added from within ABAQUS CAE. The "DevelopResiduals" method

also edits the input file, adding a line which specifies an initial stress state defined by user-subroutine. This is required to make ABAQUS use the SIGINI subroutine to apply residual stresses and, again, cannot be added from within ABAQUS CAE.

Finally, "RunFromInputFile" runs ABAQUS Standard from the command prompt, specifying the input file, user subroutine file and, where relevant, the global model results file. Once the analysis is finished the method runs another python script which loads the output data file and writes the report files, outputting all of the requested model data in a text file format.

```
OutputReader reader = new OutputReader(RootDirectory +
"Results.sdf");
reader.MergeFiles(RootDirectory + coordpath[i] + ".rpt",
RootDirectory + datapath[i] + ".rpt", names[i]);
```

**Listing B-29**

When all of the required models have been run, an instance of the OutputReader class is used to convert the text file outputs into a .sdf database file, which can then be accessed easily from the "ReportReader" menu item in the main menu.



## Appendix C. ABAQUS Code

---

The scripts which control ABAQUS are presented, along with comments and analysis. The first part of the appendix considers the Python scripts, which are provided to the scripting interface to control ABAQUS CAE. The rest of the appendix considers each user-subroutine in turn, with each user-subroutine being responsible for a different model boundary condition.

All of the listings given in this chapter are extracted from the `AbaqusBuilder2.cs` C# class file.

### C.1. Python Script

---

#### C.1.1. Global model

```
from part import *
from material import *
from section import *
from assembly import *
from step import *
from interaction import *
from load import *
from mesh import *
from job import *
from sketch import *
from visualization import *
import regionToolset
```

Listing C-1

Listing C-1 shows the header of the “script.py” file. These first few lines are instructions to ABAQUS CAE to load the elements of various modules. For example, “from part” tells ABAQUS to look in the “part” module, while the asterisk in “import \*” means “everything”. The final line does not contain a “from” statement as the “regionToolset” item is used by all the modules. This toolset deals with collections of geometric entities and is used to specify points, areas or volumes of application.

```
os.chdir(r'path')
```

Listing C-2

The “chdir” function in Listing C-2 changes the working directory, allowing the user to specify which folder all of the output files should be saved to. The word in bold – “path” –

represents the file path to the current working directory as created and used by the thermal models. The following listing shows how this line looks in the C# program.

```
sw.WriteLine(@"os.chdir(r' " + path + "')");
```

**Listing C-3**

In Listing C-3 “sw” is a StreamWriter object, a C# object used to write information to a file. The WriteLine function saves the following string (i.e. text) to the file. The “@” symbol declares the following string to be literal, in other words special characters that normally define line breaks or the end of a string, such as an apostrophe, are considered part of the string. The function argument contains three strings which are added together, two of which are constant and one of which is a variable. If the “path” variable held the value “this is a file path”, the WriteLine function would add the line “os.chdir(r’t his is a file path’)” to the file (without the quote marks). The path variable will hold the current working directory for the thermal model, ensuring that all related results files are saved to the same folder.

For the rest of this chapter, bold text in a listing will always refer to this kind of value substitution.

### Create the first definitions

---

```
model=mdb.Model(name='SymmetricRollModel')  
model.Material(name='Steel')  
model.Material(name='SteelShell')
```

**Listing C-4**

In Listing C-4 the model definition is created and named “SymmetricRollModel” and given two material definitions.

```
model.materials['Steel'].Density(table=((rho, ), ))  
model.materials['Steel'].Elastic(table=((coreyoungsmodulus,  
poissonsRatio, ), ))  
model.materials['Steel'].Expansion(table=((coreexpansion, ), ))  
model.materials['Steel'].Conductivity(table=((k, ), ))  
model.materials['Steel'].SpecificHeat(table=((Cp, ), ))  
model.materials['SteelShell'].Density(table=((rho_shell, ), ))  
model.materials['SteelShell'].Elastic(table=((youngsmodulus_shell,  
poissonsRatio_shell, ), ))  
model.materials['SteelShell'].Expansion(table=((expansioncoefficient_shell, ), ))  
model.materials['SteelShell'].Conductivity(table=((km_shell, ), ))  
model.materials['SteelShell'].SpecificHeat(table=((Cp_shell, ), ))
```

**Listing C-5**

Listing C-5 shows the lines required to fully define the materials used by the model. All data input must be provided in a table format, with brackets around each row and values separated by commas. For example;

1.5	1.0	10
0.1	0.05	50

...would be represented by...

```
table=((1.5, 1.0, 10), (0.1, 0.05, 50), )
```

**Listing C-6**

With only a single row of values in any of the definitions in Listing C-5, the final column value can be left blank but must be included to meet the input requirements. The value in this final column would normally be a temperature value, allowing temperature variability for any material property. Exact values are not mentioned here as they will vary depending on what values the user specifies through the user interface of the C# software.

```
model.HomogeneousSolidSection(material='Steel',  
name='SteelSection', thickness=1.0)  
model.HomogeneousSolidSection(material='SteelShell',  
name='SteelShellSection', thickness=1.0)
```

**Listing C-7**

The two lines in Listing C-7 create the section definitions which are used to apply the material properties to homogeneous solid blocks of material. The thickness parameter is a default value which must be specified but is never used if the section is applied to a three dimensional solid.

## *Model Geometry*

---

With the material and section definitions fully prescribed it is time to create the model geometry. There are numerous ways of creating solid parts in ABAQUS CAE, all of which require a two-dimensional sketch of a shape which will be used as a template for forming the part.

```
mSketch = model.ConstrainedSketch(name='RollSketch', sheetSize =  
4.0)
```

### Listing C-8

In Listing C-8 a sketch is created with a maximum dimension of 4 meters, enough to cover most sizes of work roll. This specification controls the size of the snap-to grid in the graphical user interface so need not be exact. The part will be created by revolving this sketch around a centre line.

```
mSketch.ConstructionLine(point1=(0.0, 2.0), point2=(0.0, 0.0))
```

### Listing C-9

Listing C-9 creates a construction line so that ABAQUS knows which axis to rotate the sketch around. If the first coordinate is *x* and the second is *y* then this sets up the centre line of the roll as being along the *y* axis.

```
mSketch.Line(point1=(0.0, 0.0), point2=(radius, 0.0))
mSketch.Line(point1=(radius, 0.0), point2=(radius, length / 2))
mSketch.Line(point1=(radius, length / 2), point2=(bradius, length / 2))
mSketch.Line(point1=(bradius, length / 2), point2=(bradius, blength + length / 2))
mSketch.Line(point1=(bradius, blength + length / 2), point2=(0.0, blength + length / 2))
mSketch.Line(point1=(0.0, blength + length / 2), point2=(0.0, 0.0))
```

### Listing C-10

In Listing C-10 a series of commands are used to create lines between pairs of points. The specific value for each point is taken from the geometric data from the thermal model, *radius* is the work roll radius, *length* is the barrel length, *blength* is the bearing length and *bradius* is the bearing radius.

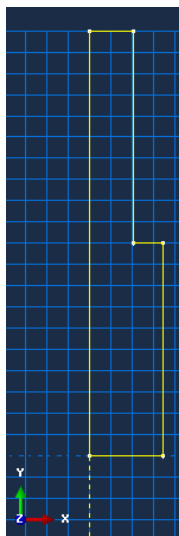


Figure C-1: Outline of the work roll model

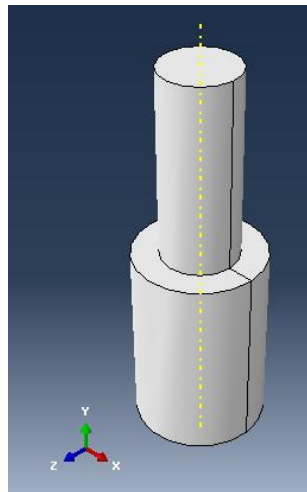
Figure C-1 shows the outline resulting from Listing C-10. This profile can be rotated to generate the work roll volume.

```
mPart = model.Part(name='Roll', dimensionality=THREE_D,
type=DEFORMABLE_BODY)
mPart.BaseSolidRevolve(sketch=mSketch, angle=360.0,
flipRevolveDirection=OFF)
mSketch.unsetPrimaryObject()
```

**Listing C-11**

The first line of Listing C-11 creates the part definition, complete with name, dimensionality and the kind of behaviour it is to be used to simulate. The second line defines the revolve process, specifying the sketch to be used and the angle through which it should be rotated. The revolve direction is a required parameter but doesn't really matter given that a full rotation would be achieved regardless of which direction was chosen. Figure C-2 shows the result of this action. The final line in Listing C-11 is required by ABAQUS to complete the part creation process, as a way of taking focus from the sketch object.

The outline of the work roll model has been created but various modifications must be made so that the meshing process can be conducted efficiently and the material properties can be applied to the right locations.



**Figure C-2: The solid part**

```
model.ConstrainedSketch(
    gridSpacing=0.07,
    name='__profile__',
    sheetSize=2.81,
    transform=model.parts['Roll'].MakeSketchTransform(sketchPlane=mo
del.parts['Roll'].faces[2],
    sketchPlaneSide=SIDE1,
    sketchUpEdge=model.parts['Roll'].edges[3],
    sketchOrientation=RIGHT,
```

```
origin=(0.0, 1.0, 0.0)))
```

**Listing C-12**

Listing C-12 is a single line which has been split across multiple lines for easier reading, as the function takes a lot of parameters. The transform parameter moves the sketch to the right global coordinate to coincide with the supplied face (a face is any unbroken surface, in this case the annular face in Figure C-2). Each face has two sides, the sketch plane side is set as SIDE1 to specify the top of the face as viewed in Figure C-2. An edge code is specified for the sketchUpEdge, found by reference to the journal file when a suitable edge (edge number 3) was selected through the graphical user interface. The up edge defines the direction perpendicular to the sketch y axis. The sketch orientation works with the up edge to define the orientation of the sketch. Finally, the origin gives the initial location of the sketch origin in global coordinates. The end result of this code is a sketch that lines up with the barrel shoulder, so that the single face in Figure C-2 can be split into two faces.

```
model.parts['Roll'].projectReferencesOntoSketch(filter=COPLANAR_EDGE,  
    sketch=model.sketches['__profile__'])  
model.sketches['__profile__'].CircleByCenterPerimeter(center=(0.0,  
0.0), point1=(0.0, (coreradius +  
bradius * 2) / 3))  
model.sketches['__profile__'].CoincidentConstraint(  
    entity1=model.sketches['__profile__'].vertices[3],  
    entity2=model.sketches['__profile__'].geometry[2])
```

**Listing C-13**

The first line of Listing C-13, split across two lines, incorporates the part geometry into the sketch. This allows the sketch items and the part items to interact with each other. The second line, also split across two lines, draws a circle by specifying the centre point and a point on the radius. This point is a third of the distance between the bearing radius and the radius of the interface between the hardened shell and softer core materials. The final line forces the centre point of the circle and the centre line of the part geometry to be coincident, effectively lining them up.

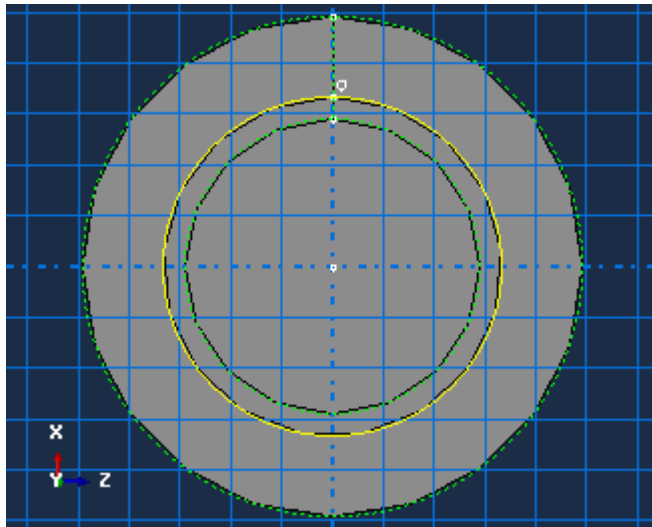


Figure C-3: Circle sketched onto projected geometry

Figure C-3 shows the resulting sketch, where the yellow circle is the newly created line.

```

model.parts['Roll'].PartitionFaceBySketch(
    faces=model.parts['Roll'].faces[2],
    sketch=model.sketches['__profile__'],
    sketchUpEdge=model.parts['Roll'].edges[3])
del model.sketches['__profile__']

```

Listing C-14

With all the requisite items created the task of partitioning the face is accomplished in Listing C-14 by specifying the face and sketch involved and the reference edge on the part geometry that was used to orient the sketch. The final line deletes the sketch as it is no longer needed. Figure C-4 shows the newly partitioned face in ABAQUS CAE.

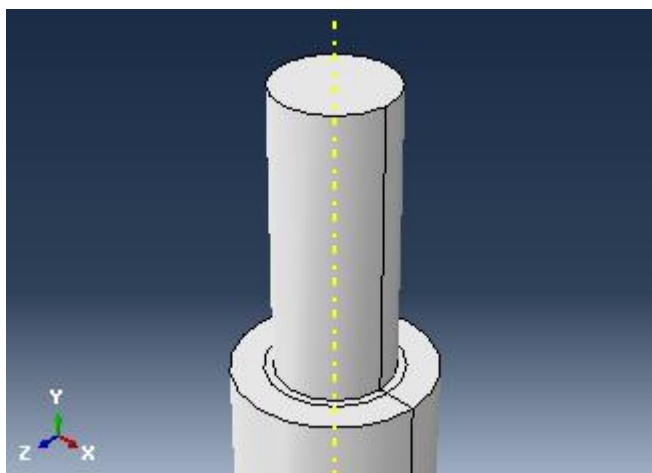
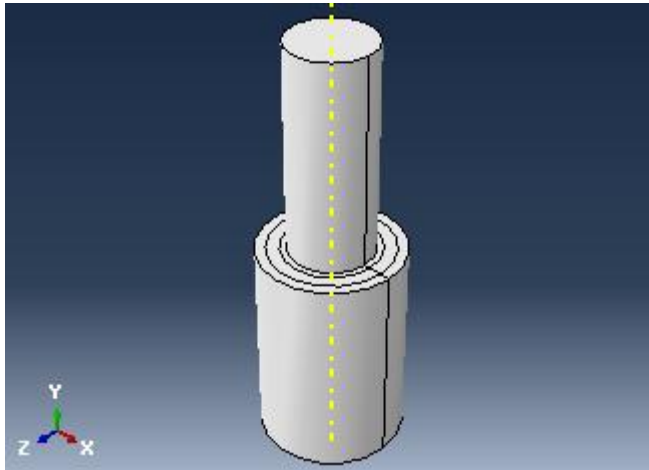


Figure C-4: The partitioned shoulder face

The previous process is repeated to create a face partition at the material interface. For brevity the listings will not be repeated as the only differences are updates to geometry codes.



**Figure C-5: The fully partitioned shoulder face**

Although all of the faces have been adequately partitioned the underlying volume is still one whole piece. A part must represent a single continuous solid but can be divided into sections, with each section being called a “cell”. Each cell can have a different material specification but the whole still counts as a continuous part, making cell partitions very useful for solid parts with more than one material definition. As there is only one section to the part there is currently only one cell.

```
model.parts['Roll'].PartitionCellBySweepEdge (
    cells=model.parts['Roll'].cells[0],
    edges=(model.parts['Roll'].edges[0], ),
    sweepPath=model.parts['Roll'].edges[9])
```

**Listing C-15**

The function used in Listing C-15 creates a partition by sweeping a specified edge (edge code = 0) through a specified cell (cell code = 0) along a path given by another specified edge (edge code = 9). The inner radial line is chosen as the sweeping line, to carry the outer radius of the bearing through the barrel so that there are no sudden changes in radius for the meshing tools to deal with. Having created one partition there is more than one cell available to choose from so the correct cell cannot be assumed to have an index of zero.

```
model.parts['Roll'].PartitionCellByExtrudeEdge (
    cells=model.parts['Roll'].cells.getSequenceFromMask(['#2 '],
    ), ),
    edges=(model.parts['Roll'].edges[6], ),
    line=model.parts['Roll'].datums[1], sense=FORWARD)

model.parts['Roll'].PartitionCellByExtrudeEdge (
    cells=model.parts['Roll'].cells.getSequenceFromMask(['#4 '], ),
),
    edges=(model.parts['Roll'].edges[10], ),
    line=model.parts['Roll'].datums[1], sense=FORWARD)
```



#### Listing C-16

Listing C-16 introduces masks as a method of selecting geometric entities. The mask code is a hex code or series of hex codes which pick out a collection of objects from a large number of potential objects very efficiently. While not strictly necessary they also drastically reduce the amount of code required later on in the analysis, when there are greater than two hundred edges to choose from, for instance. The two remaining partitions are created by extruding a selected edge through a selected cell in a direction given by the datum line which runs down the work roll centre line. When extruding a partition the edges are swept in the indicated direction until they have completely left the given volume. Figure C-6 shows a wireframe representation of the work roll, illustrating the segmentation of the internal volume.

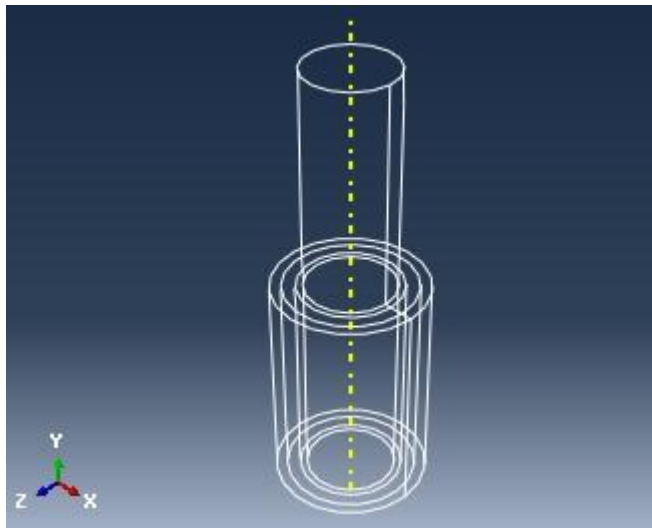


Figure C-6: Wireframe view of partitioned cells

While there are now enough partitions to address the requirements for material specification and allow a good level of radial mesh control, the automatic meshing tools will still struggle with this geometry. Abaqus often has trouble meshing volumes with a circular or annular cross-section. To solve this problem the volume can also be split into quarters.

```
model.parts['Roll'].DatumPlaneByPrincipalPlane(offset=0.0,  
principalPlane=XYPLANE)  
model.parts['Roll'].DatumPlaneByPrincipalPlane(offset=0.0,  
principalPlane=YZPLANE)
```

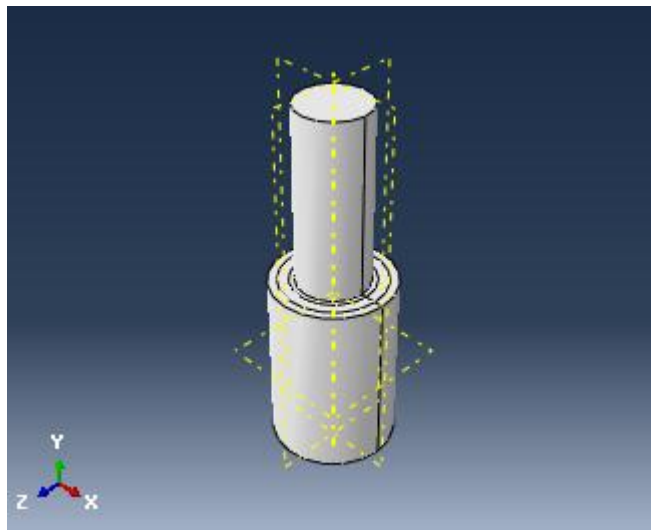
#### Listing C-17

Listing C-17 creates two datum planes, infinite surfaces which do not exist as part of the geometry but can be used as a reference for various functions. These two planes are created parallel to the y-axis and at right angles to each other.

```
model.parts['Roll'].DatumPlaneByPrincipalPlane(offset=offset,
principalPlane=XZPLANE)
```

**Listing C-18**

While partitions are being made it is useful to be able to control the mesh density in the axial direction, concentrating density where the strip will make contact and reducing density along the unloaded surface area. Listing C-18 uses the variable value “offset”, which is linked to the strip width recorded by the hot mill database, to create a partition perpendicular to the y-axis and along the strip edge. Figure C-7 illustrates the three datum planes.



**Figure C-7: Datum planes**

The three partitions are created by the following lines;

```
model.parts['Roll'].PartitionCellByDatumPlane(
    cells=model.parts['Roll'].cells,
    datumPlane=model.parts['Roll'].datums[7])

model.parts['Roll'].PartitionCellByDatumPlane(
    cells=model.parts['Roll'].cells,
    datumPlane=model.parts['Roll'].datums[8])

model.parts['Roll'].PartitionCellByDatumPlane(
    cells=model.parts['Roll'].cells,
    datumPlane=model.parts['Roll'].datums[9])
```

**Listing C-19**

Each of the three functions in Listing C-19 uses all the cells in the model and the datum plane which will define the partition as parameters. This will cut the part into quarters and also isolate the area of the barrel which makes contact with the strip, making the meshing process much easier. However, the roll journal is still attached to the top half of the barrel and needs to be separated.

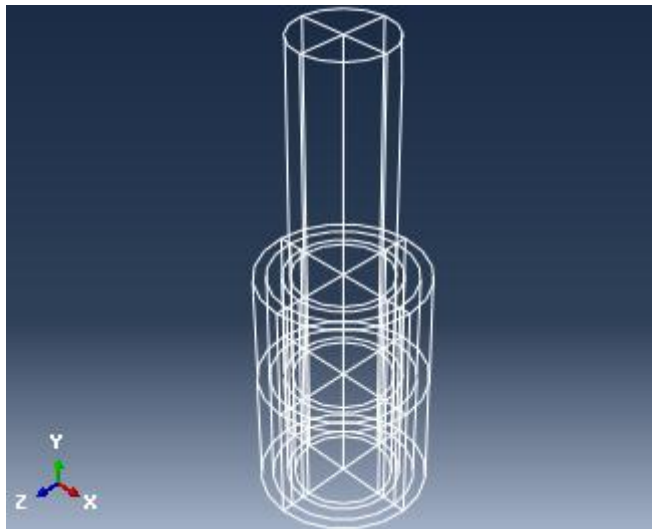
```

model.parts['Roll'].PartitionCellByPlaneThreePoints(
    cells=model.parts['Roll'].cells.getSequenceFromMask(['#1818'],
), ),
    point1=model.parts['Roll'].vertices[36],
    point2=model.parts['Roll'].vertices[42],
    point3=model.parts['Roll'].vertices[37])

```

**Listing C-20**

A datum plane has not been created at this level, however there are already various geometrical features which can be used to define a plane to partition the necessary cells. Listing C-20 picks three points from the barrel shoulder surface to create the partition.



**Figure C-8: Fully partitioned part**

Figure C-8 gives a wireframe illustration of the fully partitioned work roll model. There are now enough partitions to accurately specify the required mesh resolution throughout the model and each individual volume can be represented by extruding a single face in a straight line, making the meshing process easier.

### *Properties and instances*

---

The geometry exists but is not yet associated with the material definitions.

```

p = model.parts['Roll']
c = p.cells

p.SectionAssignment(
    offset=0.0,
    offsetField='',
    offsetType=MIDDLE_SURFACE,

```

```

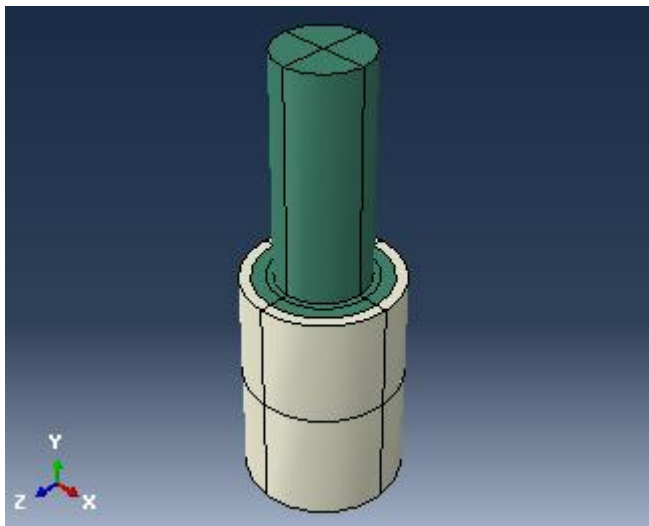
        region=Region(cells=c.getSequenceFromMask(mask=('[#24242420
#4 ]', ), ), ), ),
        sectionName='SteelShellSection',
        thicknessAssignment=FROM_SECTION)

p.SectionAssignment(
    offset=0.0,
    offsetField='',
    offsetType=MIDDLE_SURFACE,
    region=Region(cells=c.getSequenceFromMask(mask=('[#dbdbdbdf #b
]', ), ), ), ),
    sectionName='SteelSection',
    thicknessAssignment=FROM_SECTION)

```

**Listing C-21**

The functions following the two variable declarations are responsible for assigning material properties to the part, with “SteelShellSection” being applied to the shell material and “SteelSection” being applied to the core. The offset parameters are left at their default values and masks are used to select the cells rather than specifying the 36 individual cell codes directly.



**Figure C-9: The roll part with materials assigned**

Figure C-9 shows the roll part with the different material sections highlighted. The part is now complete but is still only a part definition. In order to be represented in a model an “instance” of this part must be made.

```

a = model.rootAssembly
a.DatumCsysByDefault (CARTESIAN)

```

**Listing C-22**

The first line of Listing C-22 sets up the variable “a” to be a shortcut to the root assembly object. It is the instances in the root assembly which are actually analysed by ABAQUS

Standard. The second line sets the coordinate frame used by the assembly. Ideally the coordinate system used should be a cylindrical polar system as the work roll is cylindrical and the stress measures are most useful when considered in cylindrical components, i.e. hoop stress, radial stress, etc. Unfortunately the underlying ABAQUS code will still function in rectangular (Cartesian) coordinates and the user subroutines which communicate with ABAQUS Standard will always use rectangular coordinates. To avoid unnecessary confusion it is worth keeping all of ABAQUS's modelling tasks in the same reference frame.

```
a.Instance(name='Roll-1', part=p, dependent=OFF)
i = a.instances['Roll-1']
```

Listing C-23

Listing C-23 creates a new instance and sets up “i” as the shortcut to it. The dependent property controls whether the meshing task should be performed on the part or on instances of the part. Technically this allows the user to choose whether individual instances should be meshed depending on their own circumstance or whether one uniform mesh can be applied to all of them. As this model contains only one part the decision is arbitrary, the mesh will be constructed on the instance.

## Seeding

---

Before the meshing process can begin the model edges need to be seeded. This involves creating a series of nodes along every edge in the model which the mesh generation tools will use to define how the internal volume should be broken up. A very fine seeding will put a lot of nodes on each edge, which will result in there being a lot of elements around that area of the model. A very coarse seeding will have the opposite effect, resulting in very few elements being created. The task of choosing the right seeding requires balancing the need for sufficient resolution against the capabilities of the machine performing the analysis.

```
a.seedEdgeByNumber(
    constraint=FIXED,
    edges=i.edges.getSequenceFromMask(('[#40000000 #5096724 #40061
#b0ac0000 #1000 ]', ), ),
    number=kSize / 4)
```

Listing C-24

Listing C-24 gives the command which seeds the first set of edges, the circumferential edges at and just below the work roll surface. The “constraint” parameter dictates whether ABAQUS is allowed to change the seeding of any edges when it is trying to create a mesh. If

allowed to do so, ABAQUS will resolve issues with the meshing process by changing edge seeds. This can (and often does) result in meshes which do not reflect the initially specified seeding and therefore should be disabled by setting the constraint parameter to “FIXED”, i.e. a fixed number of nodes. The edges are selected by mask for efficiency. The seeding number specifies how many nodes the edge should be broken into and is set from a user-supplied variable from the C# user interface. The value is divided by four to reflect the four partitions around the circumference.

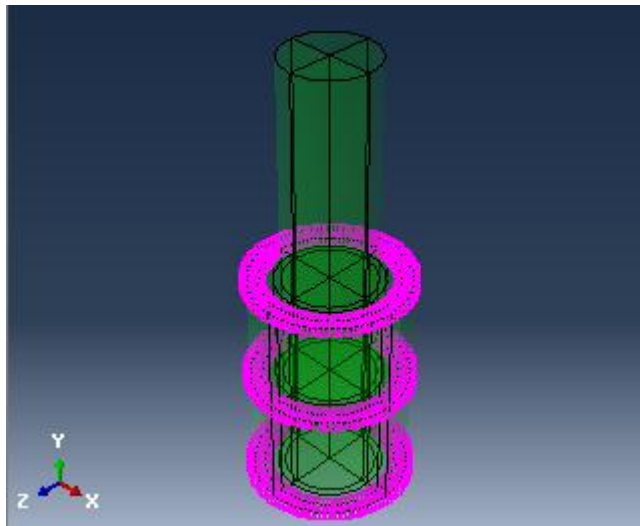


Figure C-10: Outer circumferential seeding

Figure C-10 highlights the nodes created in Listing C-24. The surface edges are seeded with the same number of nodes as those edges just below the surface to ensure a good uniformity of mesh near the surface, where precision is most important. The results will be rough as the resolution of this model is low but any errors due to poor element shape will have the most impact near the surface where the boundary conditions vary the most intensely.

```
a.seedEdgeByNumber(constraint=FIXED,
    edges=i.edges.getSequenceFromMask(['[#4200000 #3 #1100c00
#6000000 #a ]', ), ), ),
    number=kSize / 8)
```

Listing C-25

Listing C-25 is responsible for seeding the intermediary circumferential edges using the same function as the previous listing. This time half the element density is requested. This will force ABAQUS to reduce the mesh density below the work roll surface, saving processing time by reducing the resolution where conditions are relatively calm. As long as **kSize / 4** is an even number then the progression from one density to another should be achievable in a structured

manner, giving the best chance for a mesh of well shaped elements. The location of the seeded edges is shown in Figure C-11.

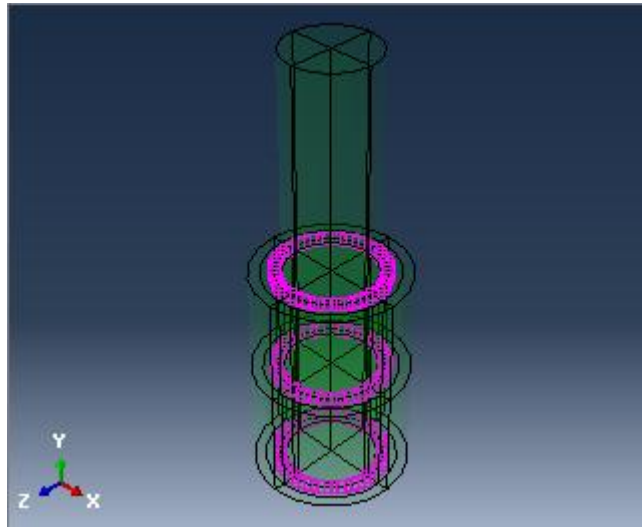


Figure C-11: Intermediary circumferential seeding

The final set of circumferential edges are seeded in Listing C-26.

```
a.seedEdgeByNumber(  
    constraint=FIXED,  
    edges=i.edges.getSequenceFromMask(('[#31400d4 #0 #20000000 #21  
#6241 ]', ), ), ),  
    number=kSize / 8)
```

Listing C-26

This seeding mimics the spacing of the previous listing to give an even and regular mesh in the space between the two sets of edges. This speeds up the meshing process by allowing the volume to be meshed using a simple structured pattern which requires very little thought from ABAQUS.

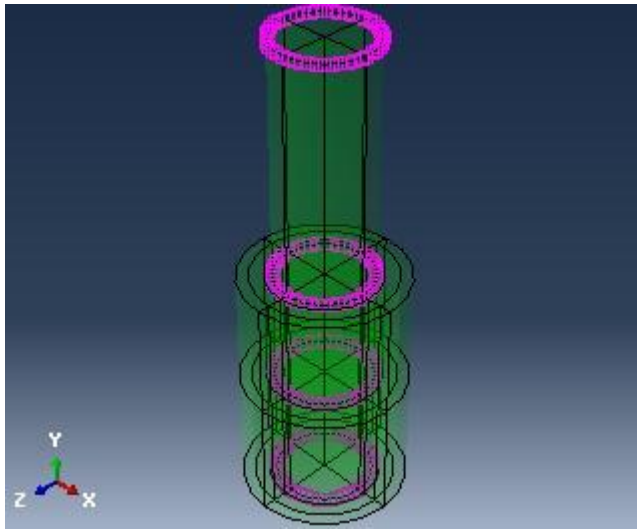


Figure C-12: Inner circumferential edge seeding

Figure C-12 illustrates the inner circumferential edges by highlighting the new seeds.

```

a.seedEdgeByNumber (
  constraint=FIXED,
  edges=i.edges.getSequenceFromMask(['#2800 #0:2 #84 #400 '], ),
),
  number=iSize / 4)

```

Listing C-27

Listing C-27 adds seeding to the axial edges of the work roll journal. The number of seeds specified is taken from the “iSize” variable which is used to define the number of layers of elements required in the barrel section of the work roll. This value is divided by four to reduce the density along the work roll journal, which is not of principle concern for this model and so can be sacrificed to allow higher resolution at the work roll surface.



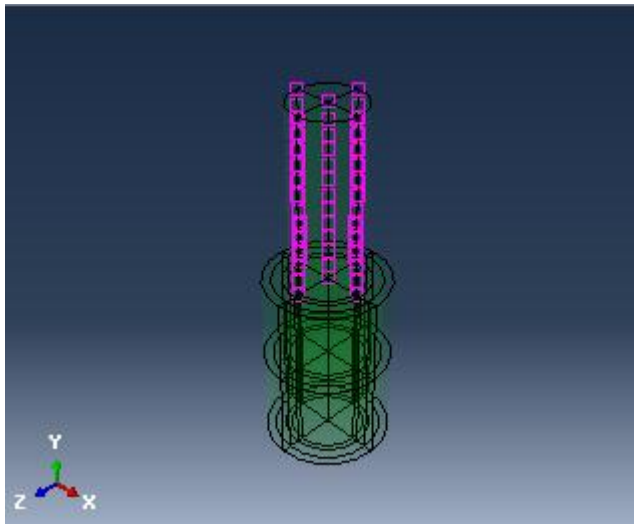


Figure C-13: Journal axial edge seeding

Figure C-13 illustrates the lower resolution along the length of the roll journal.

```

a.seedEdgeByNumber(
    constraint=FIXED,
    edges=i.edges.getSequenceFromMask(['[#14500 #8340000 #4082909a
]', ), ),
    number=iSize * ratio)
a.seedEdgeByNumber(
    constraint=FIXED,
    edges=i.edges.getSequenceFromMask(['[#0 #a2828000 #94002200
#9101408 ]', ), ),
    number=iSize * ratio)

```

Listing C-28

In Listing C-28 the axial seeding of the barrel is applied. Since the barrel is split into more than one section, the “iSize” parameter must be multiplied by a value which reflects the relative size of each section. This is performed in the C# code where the required values (barrel length and strip width) are readily available.

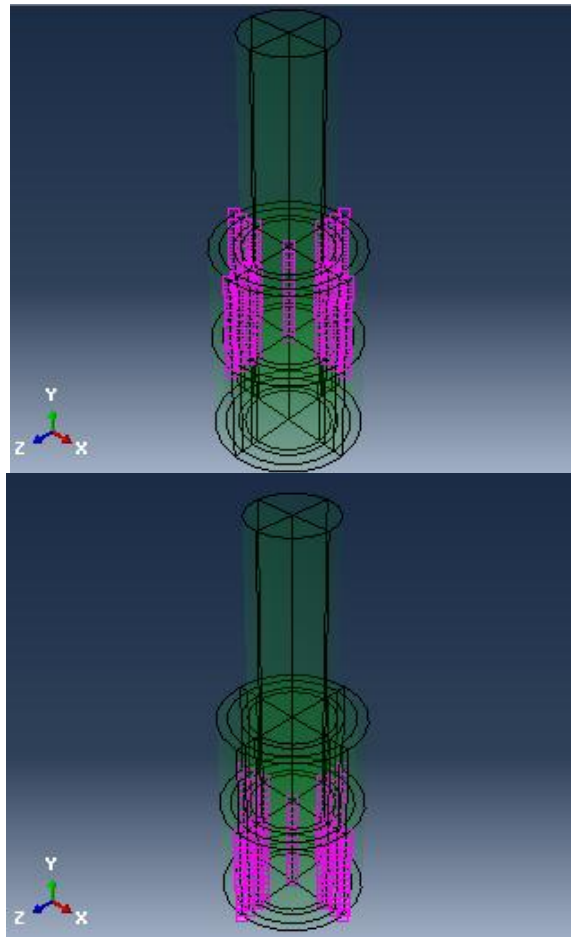


Figure C-14: (left) outer barrel seeding and (right) inner barrel seeding

Figure C-14 shows the two seeding conditions applied by Listing C-28, leaving all axial model edges fully defined.

```

a.seedEdgeByNumber (
  constraint=FIXED,
  edges=i.edges.getSequenceFromMask(['#a922b #0 #8000000 #4142
#880 ]', ), ),
  number=jSize / 4)

a.seedEdgeByNumber (
  constraint=FIXED,
  edges=i.edges.getSequenceFromMask(['#88c00000 #0 #604000 #a010
#120 ]', ), ),
  number=jSize / 8)

a.seedEdgeByNumber (
  constraint=FIXED,
  edges=i.edges.getSequenceFromMask(['#3000000 #40000088 #2010100
#10800 #14 ]', ), ),
  number=jSize / 4)

```

Listing C-29

Listing C-29 treats the radial edges of the model using a variable called “jSize”, with various divisors to create the right balance of relative densities so that the resolution is coarse near the roll centre and fine approaching the roll surface.

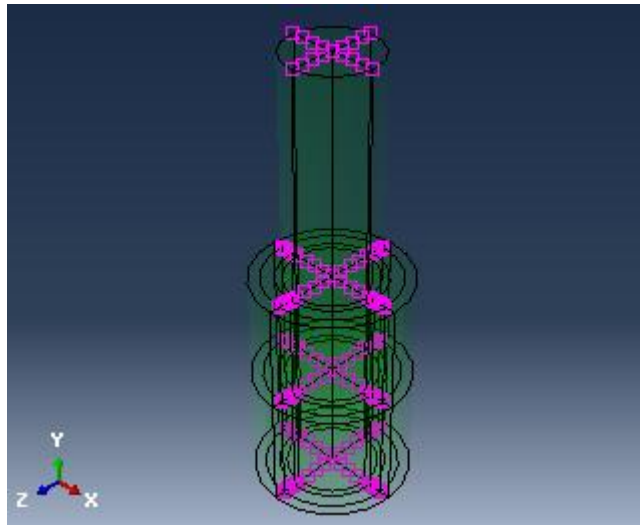


Figure C-15: Inner and intermediate radial edge seeding

Figure C-15 shows those edges affected by Listing C-29.

```

a.seedEdgeByBias(
    biasMethod=SINGLE,
    constraint=FIXED,
    end1Edges=i.edges.getSequenceFromMask(['[#0 #1040 #0 #40020200
] ', ), ),
    end2Edges=i.edges.getSequenceFromMask(['[#0 #10400810 #80004
#400000 ] ', ), ),
    number=jSize / 4,
    ratio=5.0)

```

Listing C-30

Very near the surface of the roll, resolution is of utmost import to obtaining valid results. To ensure the best mesh density possible where the temperature results are at their most extreme, a bias is applied in Listing C-30. The additional terms in the listing control the bias. The bias method determines whether the bias should be single or double, i.e. denser at one end or denser/coarser in the middle. The end1 and end2 edges allow ABAQUS to choose which end of an edge should be densely or coarsely seeded, while the ratio parameter gives the relationship between the size of an element and the element next to it. The resulting seeding is illustrated in Figure C-16.

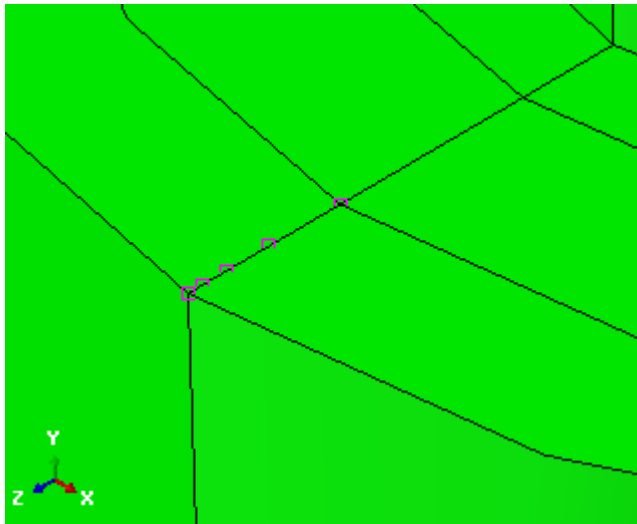


Figure C-16: Seeding bias

## Meshing

---

With the model fully seeded the type of meshing scheme must be chosen. The default scheme is the “structured” scheme, which creates a regular, ordered mesh but can only be used when the geometry is simple and opposite edges are seeded identically. The structured scheme is fine for the surface and the inner intermediary volume, however the outer intermediary volume provides a change in circumferential mesh density and the cross-sections of the central volumes are quarter circles. For these oddly shaped or oddly seeded volumes the bottom-up scheme is the only scheme which will provide any solution at all. In the bottom-up scheme a face is selected as a generating face and meshed. The volume mesh is then created by “sweeping” this face from one side of the volume to the other, generating a new face mesh for each node passed in the sweeping direction. All of the face meshes are connected to create the final volume mesh.

```
a.setMeshControls(regions=i.cells.getSequenceFromMask(['#c3c3c3cf  
#3 ]', ), ),  
technique=BOTTOM_UP)
```

Listing C-31

Listing C-31 selects the intermediary and central regions of the roll and changes the meshing scheme to the bottom-up technique. The areas affected by this are shown in Figure C-17.

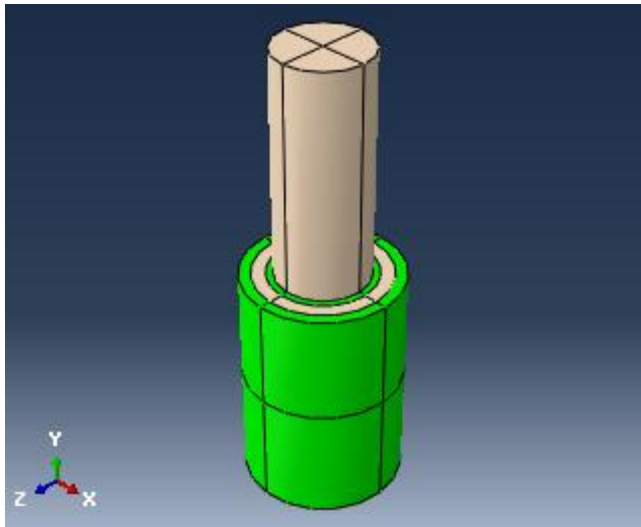


Figure C-17: Colour coded meshing schemes, green for structured meshing, grey for bottom-up meshing

The last consideration before meshing any volumes is the choice of element type.

```

a.setElementType(
  elemTypes=(
    ElemType(
      elemCode=C3D8R,
      elemLibrary=STANDARD,
      secondOrderAccuracy=OFF,
      kinematicSplit=AVERAGE_STRAIN,
      hourglassControl=ENHANCED,
      distortionControl=DEFAULT),
    ElemType(elemCode=C3D6, elemLibrary=STANDARD),
    ElemType(elemCode=C3D4, elemLibrary=STANDARD)),
  regions=(i.cells.getSequenceFromMask(['#ffffffff #ffffff'], ),
), ))

```

Listing C-32

Listing C-32 introduces the code which selects the element types to be used by a certain region of the model, in this case the entirety of it. The listing is staggered to isolate certain elements of the command but is essentially a single line. The function detailed contains two main parameters, an “elemTypes” parameter and the regions to which those element types should be assigned. Three “ElemType” definitions are created to fill the “elemTypes” parameter, as for a three-dimensional mesh ABAQUS requires a choice of hexahedral (8 node), wedge (6 node) and tetrahedral (4 node) element shapes. With these choices available ABAQUS will try to use the highest order (most nodes) element for any given situation but will reduce the order of element to reduce mesh warping caused by seeding and/or geometrical constraints where appropriate and if the meshing scheme permits.

The first parameter required to create an element definition is the code of the element to be used. Considering the code used by the most important element, “C3D8R”, each character has a meaning which hints at the form and function of the element. The “C” character stands

for “continuum” and means that each individual element represents a pure block of matter whose properties are constant throughout. “3D” gives the dimensionality of the element. The “8” describes the number of nodes in the element, marking this as a hexahedral or “brick” element. Finally the “R” indicates that the selected element is a reduced integration element.

Reduced integration refers to the integration scheme used to define the stress and strain behaviour of the element. A fully integrated element uses the Gauss Integration scheme which gives completely accurate representation of the material behaviour of a perfectly shaped element (parallel sides, mid-side nodes opposite each other for quadratic elements). A reduced integration element uses a scheme one order lower than a fully integrated element. A useful upside to this decision is that the number of integration points (internal points the element uses as a centre of integration in calculating stress and strain values) is reduced from eight to one. This reduction in complexity gives a big reduction in processing cost, to less than 30% according to the ABAQUS literature<sup>37</sup>. An additional benefit is that the stress values in reduced integration elements are calculated at the positions of optimal accuracy, so point-wise results can be more accurate than the reduced resolution would suggest.

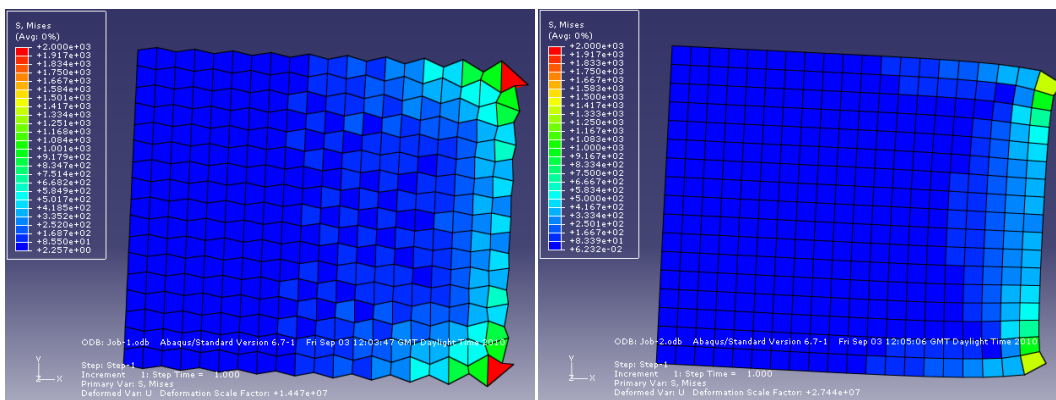


Figure C-18: a) standard reduced integration element and b) with enhanced hourglass control

Figure C-18a illustrates the main weakness of reduced integration elements, the “zero energy modes”. Figure C-19 gives an example of the “hourglassing” zero energy mode. In this example the element is subject to a bending moment, the result of which the measure of stress generation (the change in the length of and angle between the imaginary lines connecting the centre point of opposite faces) cannot detect. If the element cannot detect the deformation then it cannot resist it, leading to a “zero stiffness” kind of behaviour. This form of deformation tends to propagate through a model either in bulk bending situations or under concentrated point loads, drastically reducing accuracy.

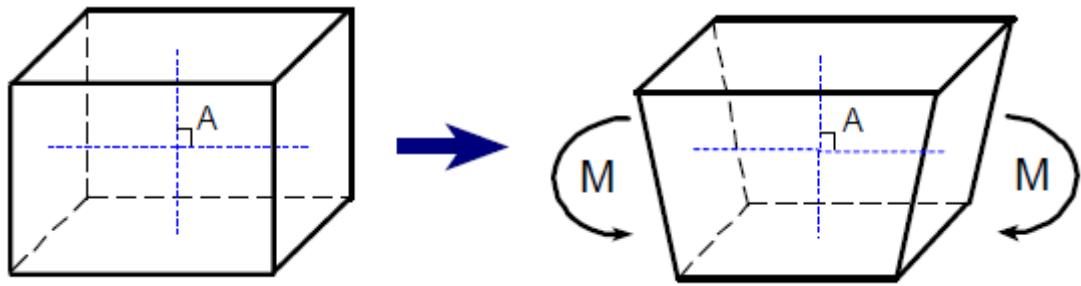


Figure C-19: Zero energy deformation<sup>38</sup>.

Given the drastic variations in the thermal profile there is a distinct possibility of highly concentrated load sources, even if the mechanical loading is not being considered. Listing C-32 therefore sets the hourglass control mode to “enhanced”, introducing an artificial stiffness term into the element formulation to provide a realistic level of resistance to bending moments. The effects of enhanced hourglass control can be observed in Figure C-18b.

```

a.generateBottomUpSweptMesh(
    cell=i.cells[3],
    geometryConnectingSides=Region(
        faces=i.faces.getSequenceFromMask(mask=('[#20 #400000
#4000000 ]', ), ), ),
    geometrySourceSide=Region(
        faces=i.faces.getSequenceFromMask(mask=('[#0:2 #1000000
] ', ), ), ),
    targetSide=i.faces[0])

```

Listing C-33

Listing C-33 is the first step in meshing the work roll. The first parameter of the “generateBottomUpSweptMesh” function is the cell which is being meshed. The second and third parameters specify which faces perform which functions in the meshing process. Considering the volume shown in Figure 4-18, the three light grey sides that connect the two triangles are the connecting sides defined in the second parameter. These faces define a sweep path connecting the two faces at either end. The white face is the source side in parameter three, this face will be broken into two dimensional pieces of as near to perfectly square (or triangular for tetrahedral / wedge shaped elements) shape as possible. The dark face is the target side in parameter three, this face gives the target size, shape and orientation for the sweeping process to finish at.

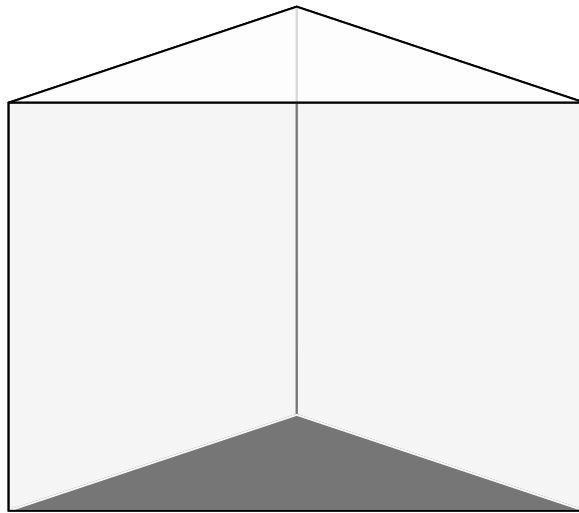


Figure C-20: Selecting sweeping faces

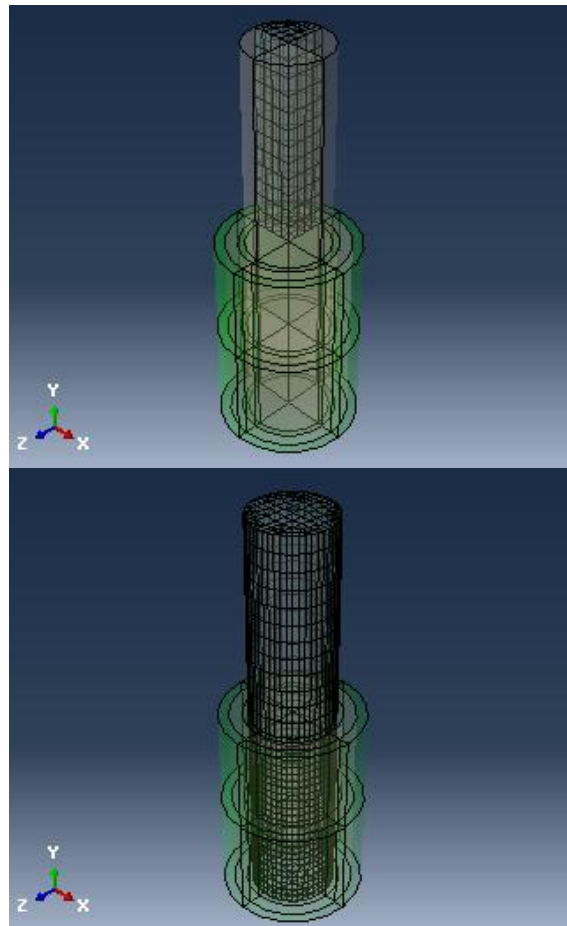


Figure C-21: (left) A meshed cell and (right) a mesh for every cell within the journal radius

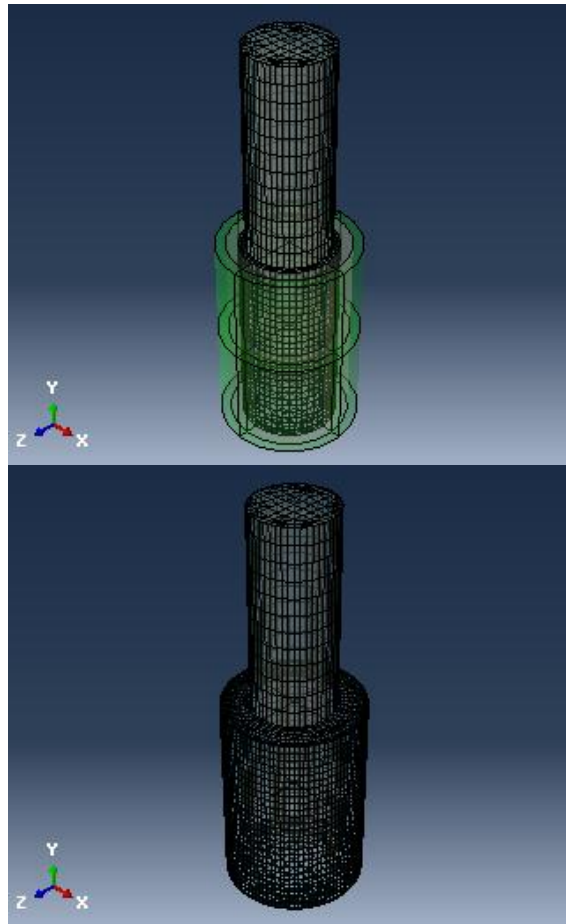
Figure C-21 (left) shows the mesh created by Listing C-33. By repeating the process using different cell and associated face mask codes (see source code "AbaqusBuilder2.BuildScript\_SubModelSym()"), all of the cells within the bearing radius can be meshed.



```
a.generateMesh(regions=i.cells.getSequenceFromMask(['#18181810 #8]'), ))
```

**Listing C-34**

Listing C-34 generates the mesh for the cells which immediately surround the meshed region in Figure C-21 (right). This volume is too small a space to easily vary the mesh density, so a regularly structured mesh is the best option. This allows the use of the automated meshing tools, hence the reduced amount of code required to carry out the meshing process.



**Figure C-22: (left) structured intermediary mesh and (right) complete mesh**

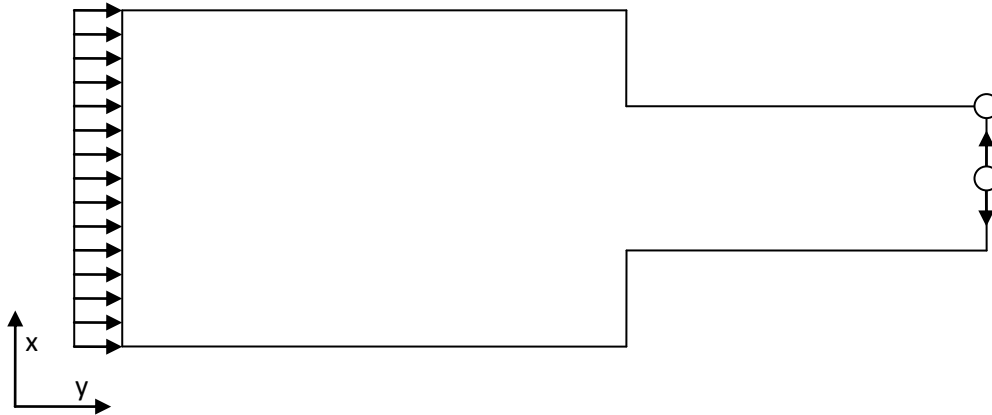
Figure C-22 shows the updated mesh on the left, and also the complete part mesh on the right. The rest of the meshing process follows the same pattern as the previous two steps, with an unstructured intermediary mesh surrounding the previously generated mesh, and finally a structured mesh at the surface.

### *Boundary conditions*

---

The most basic boundary condition requirement of any finite element model is a restraint boundary condition. Without at least one point having a fixed location the whole model would

be free to move, which could lead to very unpredictable results. In order to avoid generating any stresses accidentally, the number and complexity of restraint conditions should be kept to a minimum, only free-body rotation and translation should be prevented.



**Figure C-23: Section of the work roll stress model, showing restraint conditions**

In this instance there is a symmetry condition at the centre plane which will provide complete restraint against any y-deflection, as shown in Figure C-23. Listing C-35 applies the symmetry condition.

```
model.YsymmBC (
  createStepName='Initial',
  name='Symmetry Condition',
  region=Region (faces=i.faces.getSequenceFromMask (mask= (' [#0:2
#294624 #8152b00 ]', ), ))
```

**Listing C-35**

Two further restraint conditions are required to prevent free body rotation. The first lies on the centre line of the roll and at the very end of the journal, preventing the point from moving away from the centre line. y-direction deflection is allowed, to allow the roll to expand axially, but x- and z-directions (as indicated in Figure C-23 with the z-axis being into the page) are restrained. This restraint works with the symmetry restraint to prevent rotation around the x or z axes and also prevents translation in the x or z directions. The first restraint is applied in Listing C-36, where the directions 1, 2 and 3 refer to the dimensions x, y and z in Figure C-23.

```
model.DisplacementBC (
  amplitude=UNSET,
  createStepName='Initial',
  distributionType=UNIFORM,
  fieldName='',
  localCsys=None,
  name='Bottom Restraint',
```

```

        region=Region(edges=i.edges.getSequenceFromMask(mask=(' [#0:4
#400 ]', ), ), ), ),
        u1=SET,
        u2=UNSET,
        u3=SET,
        ur1=UNSET,
        ur2=UNSET,
        ur3=UNSET)

```

**Listing C-36**

The second restraint concerns only a single point, also at the end of the work roll journal but on the edge (i.e. the most extreme x-coordinate). The point is restrained from moving in the z-direction, working together with the previous restraint to prevent rotation around the y-axis. The second restraint is applied using Listing C-37.

```

model.DisplacementBC(
    amplitude=UNSET,
    createStepName='Initial',
    distributionType=UNIFORM,
    fieldName='',
    localCsys=None,
    name='Rotation Restraint',
    region=Region(edges=i.edges.getSequenceFromMask(mask=(' [#0:3 #80
]', ), ), ), ),
    u1=UNSET,
    u2=UNSET,
    u3=SET,
    ur1=UNSET,
    ur2=UNSET,
    ur3=UNSET)

```

**Listing C-37**

In order for the thermal expansion to be calculated an initial temperature is required.

```

model.Temperature(
    createStepName='Initial',
    crossSectionDistribution=CONSTANT_THROUGH_THICKNESS,
    distributionType=UNIFORM,
    magnitudes=(21.0, ),
    name='InitialTemp',
    region=Region(
        cells=model.rootAssembly.instances['Roll-1'].cells,
        faces=model.rootAssembly.instances['Roll-1'].faces,
        edges=model.rootAssembly.instances['Roll-1'].edges,
        vertices=model.rootAssembly.instances['Roll-1'].vertices)
)

```

**Listing C-38**

Listing C-38 creates a temperature distribution as a pre-defined field, with a constant magnitude of 21 degrees centigrade. Any temperature change will be taken as relative to this

base temperature. The "Step" of the analysis in which the initial temperature is defined is called the "Initial" step, which is used to define all the initial conditions. To apply any loads to the model and create a deflected shape, a new analysis step must be created.

```
model.StaticStep(  
    name='ApplyTemperatures',  
    previous='Initial',  
    timeIncrementationMethod=FIXED,  
    amplitude=STEP,  
    noStop=OFF)
```

**Listing C-39**

Listing C-39 creates a step called "ApplyTemperatures", which will be used to apply the temperatures. Since the analysis is static, the time-related parameters can be left at their default values.

```
model.Temperature(  
    createStepName='ApplyTemperatures',  
    distributionType=USER_DEFINED,  
    name='TemperatureFile',  
    region=Region(  
        cells=model.rootAssembly.instances['Roll-1'].cells,  
        faces=model.rootAssembly.instances['Roll-1'].faces,  
        edges=model.rootAssembly.instances['Roll-1'].edges,  
        vertices=model.rootAssembly.instances['Roll-1'].vertices))
```

**Listing C-40**

Listing C-40 defines the temperature profile for the work roll model. The region parameter is given a region made of every geometric element of the model assembly, meaning that the temperature profile will be applied to every part of the work roll model. The actual temperature data will be written to a file, so the "distributionType" must be set to "USER\_DEFINED". This tells ABAQUS to use its user-subroutine, UTEMP, which has been written to open the temperature data file and import the results.

```
model.Pressure(  
    amplitude=UNSET,  
    createStepName='ApplyTemperatures',  
    distributionType=USER_DEFINED,  
    field='',  
    magnitude=1.0,  
    name='Pressure',  
    region=Region(sidelFaces=  
        i.faces.getSequenceFromMask(mask=('[#83000000 #2 #102  
        #4000004 ]', ), ))))
```

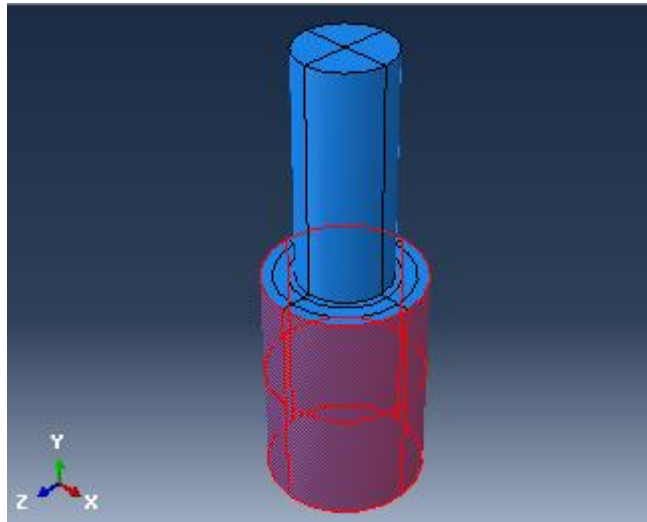
**Listing C-41**

Listing C-41 applies the pressures loads (i.e. the normal and bending loads) to the surface of the work roll. Since only the surface is affected, the "region" parameter is provided with the mask code for the work roll barrel faces. Again, the profile will be applied using a user-subroutine, so the "distributionType" parameter must be given the value "USER\_DEFINED".

```
model.SurfaceTraction(
  createStepName='ApplyTemperatures',
  directionVector=((0.0, 0.0, 0.0), (0.0, 0.0, 1.0)),
  distributionType=USER_DEFINED,
  field='',
  localCsys=None,
  magnitude=1.0,
  name='Shear',
  region=Region(side1Faces=
    i.faces.getSequenceFromMask(mask=('[#83000000 #2 #102
    #4000004 ]', ), ), ),
  resultant=ON)
```

**Listing C-42**

Listing C-42 applies the shear loading as a surface traction in much the same way as the previous two loads. The additional parameter "directionVector" defines the sense of the traction force and is arbitrarily chosen. The region of application of the mechanical loads is highlighted in Figure C-24.



**Figure C-24: Work roll surface faces**

### *Create job*

```
model.FieldOutputRequest(
  name='FieldOutputs',
  createStepName='ApplyTemperatures',
  variables=('S', 'NT', 'COORD'))
```

#### Listing C-43

Listing C-43 defines which outputs should be saved to file, with "S" representing stress data, "NT" representing temperature data and "COORD" representing coordinate data. Specifying the output data types allows ABAQUS to throw away unneeded data, saving large amounts of file space.

```
mdb.Job (
    atTime=None,
        contactPrint=OFF,
        description='',
        echoPrint=OFF,
        explicitPrecision=SINGLE,
        getMemoryFromAnalysis=True,
        historyPrint=OFF,
        memory=75,
        memoryUnits=PERCENTAGE,
        model='SymmetricRollModel',
        modelPrint=OFF,
        multiprocessingMode=DEFAULT,
        name='3DbJob',
        nodalOutputPrecision=SINGLE,
        numCpus=1,
        numDomains=1,
        queue=None,
        type=ANALYSIS,
        userSubroutine=' + path + usersubroutine3dG.for',
        waitHours=0,
        waitMinutes=0)
```

#### Listing C-44

Listing C-44 creates the actual job itself. Most of the parameters are left at their default values, the important differences are the "Memory" parameter is set as "75" instead of "50", allowing ABAQUS to use more memory, "Print" parameters are set "OFF" to reduce the amount of saved data output, and "userSubroutine" is provided with the file path to the Fortran file containing all the user subroutines.

```
mdb.jobs['3DbJob'].writeInput()
```

#### Listing C-45

Listing C-45 tells ABAQUS to write an input file. This file contains all the information which the ABAQUS Standard component needs in order to perform the simulation. The input file can then be edited to add the functionality which ABAQUS CAE does not support, before being provided to ABAQUS Standard, a process described in Appendix B (Page 308).

### C.1.1. Surface Model

```
local = mdb.Model(name='LocalModel',  
objectToCopy=mdb.models['SymmetricRollModel'])
```

**Listing C-46**

Rather than creating a completely new model, Listing C-46 copies the existing model, allowing the previous model setup to be carried across.

```
local.setValues(globalJob='3DbJob.odb')
```

**Listing C-47**

Listing C-47 defines the output of the global model as the global job. This parameter works with the sub-model boundary condition, specifying which file should be used to provide the results along the interface between the global and local models.

```
del local.rootAssembly.instances['Roll-1']  
del local.parts['Roll']  
del local.boundaryConditions['Bottom Restraint']  
del local.boundaryConditions['Rotation Restraint']
```

**Listing C-48**

By copying the previous model, the local model has inherited a number of objects which it will not need. Listing C-48 deletes those components.

```
lsketch = local.ConstrainedSketch(name='sketch', sheetSize=4.0)  
lsketch.ConstructionLine(point1=(0.0, 0.0), point2=(0.0, 2.0))  
lsketch.FixedConstraint(entity=lsketch.geometry[2])  
lsketch.rectangle(point1=(radius, length / 2), point2=(coreRadius,  
0.0))
```

**Listing C-49**

Listing C-49 creates the two dimensional sketch which will be revolved to create the surface model part. The surface model covers the entire length of the work roll barrel, but only from the outer radius down to the material interface.

```
lPart = local.Part(dimensionality=THREE_D, name='Part-1',  
type=DEFORMABLE_BODY)  
lPart.BaseSolidRevolve(angle=360.0, flipRevolveDirection=OFF,  
sketch=lsketch)  
del lsketch
```

**Listing C-50**

Listing C-50 revolves the sketch around the construction line created in Listing C-49 to create the part. With the part created the sketch is no longer needed, so can be deleted.

```
lPart.DatumPlaneByPrincipalPlane(offset=0.0,  
principalPlane=XYPLANE)  
lPart.DatumPlaneByPrincipalPlane(offset=0.0,  
principalPlane=YZPLANE)  
lPart.DatumPlaneByPrincipalPlane(offset=stripEdge,  
principalPlane=XZPLANE)
```

**Listing C-51**

Listing C-51 creates a series of planes which will be used to break the surface model into smaller pieces. As with the global model, the surface model geometry is much easier to mesh when it is split up in this manner.

```
lPart.PartitionCellByDatumPlane(  
cells=lPart.cells.getSequenceFromMask(['#1'], ), ),  
datumPlane=lPart.datums[3])  
lPart.PartitionCellByDatumPlane(  
cells=lPart.cells.getSequenceFromMask(['#3'], ), ),  
datumPlane=lPart.datums[2])  
lPart.PartitionCellByDatumPlane(  
cells=lPart.cells.getSequenceFromMask(['#f'], ), ),  
datumPlane=lPart.datums[6])
```

**Listing C-52**

Listing C-52 uses the planes to partition the surface model cells, in an identical process to that performed on the global model in Listing C-19. The two planes running parallel to the roll axis break the annulus of the model into four segments, which can be meshed using the automatic meshing tools. The perpendicular plane separates the portion of the barrel which is within the strip width and the portion of the barrel which is not.

```
lPart.SectionAssignment(  
offset=0.0,  
region=Region(cells=lPart.cells),  
sectionName='SteelShellSection')
```

**Listing C-53**

Listing C-53 applies the material properties to the model. The actual material data was copied from the global model, so doesn't need to be re-entered.

```
lInstance = local.rootAssembly.Instance(dependent=OFF, name='Part-  
1-1', part=lPart)  
a = local.rootAssembly
```



```
i = a.instances['Part-1-1']
```

**Listing C-54**

Listing C-54 creates the instance object and instantiates the surface model part. The listing also sets up the two variables, 'a' and 'i', to act as shortcuts, so that the path to the instance and assembly objects does not need to be repeated each time.

```
//Roll Bite - Circum
a.seedEdgeByNumber(
  constraint=FIXED,
  edges=i.edges.getSequenceFromMask(['[#4000c00 #2804 ]', ), ),
  number=kSize / 2)
```

**Listing C-55**

Listing C-55 applies a seed to the circumferential edges within the ninety degree segment of the surface model which contains the roll bite. Since the mesh of the surface model is not tied to a low resolution internal mesh, the circumferential resolution can be varied without warping element shapes. The resolution within this segment is increased to take advantage of this fact.

```
//90 degrees from roll bite - Circum
a.seedEdgeByBias(
  biasMethod=SINGLE,
  constraint=FIXED,
  end1Edges=i.edges.getSequenceFromMask(['[#240108 #5000 ]', ), ),
  end2Edges=i.edges.getSequenceFromMask(['[#40000202 #a0010 ]', ),
),
  number=kSize / 4,
  ratio=3.0)
```

**Listing C-56**

The segments either side of the roll bite segment can have their resolutions reduced, since there will be less rapid change of temperature in these areas. The single edge bias in Listing C-56 means that elements will be wider at one end of the edge than the other. This allows for a gradual expansion in element size moving away from the roll bite, softening the transition from fine to coarse resolution and therefore reducing any associated errors.

```
//Opposite the roll bite - Circum
a.seedEdgeByNumber(
  constraint=FIXED,
  edges=i.edges.getSequenceFromMask(['[#810060 #50000 ]', ), ),
  number=kSize / 6)
```

**Listing C-57**

Since the segments adjacent to the roll bite have gradually stepped down the resolution, the segment opposite the roll bite can carry that low resolution across the gap. A double end bias would allow the expansion to continue, but would be risking inaccuracy through having over-large element sizes. Furthermore, the backup roll contact could exacerbate any errors introduced through varying element sizes. A regular, low resolution mesh gives the best compromise between stability and efficiency.

```
a.seedEdgeByNumber (
    constraint=FIXED,
    edges=i.edges.getSequenceFromMask(['#14a0000 #14a ]', ), ),
    number=iSize)
a.seedEdgeByBias (
    biasMethod=SINGLE,
    constraint=FIXED,
    end1Edges=i.edges.getSequenceFromMask(['#28009000 ]', ), ),
    end2Edges=i.edges.getSequenceFromMask(['#82104000 ]', ), ),
    number=iSize / 4,
    ratio=4.0)
```

**Listing C-58**

The axial seeding is provided by Listing C-58. A high-resolution, regular mesh is used inside the strip width, whereas a lower resolution is used outside the strip width, using a single edge bias to gradually increase element sizes.

```
//Radial edges - Radial
a.seedEdgeByBias (
    biasMethod=SINGLE,
    constraint=FIXED,
    end1Edges=i.edges.getSequenceFromMask(['#2084 #2a0 ]', ), ),
    end2Edges=i.edges.getSequenceFromMask(['#10000011 #8401 ]', ),
),
    number=2/3 * jSize,
    ratio=5.0)
```

**Listing C-59**

Finally, the radial edges are seeded using Listing C-59, with a single edge bias to concentrate the elements at the surface.

```
mdb.models['LocalModel'].rootAssembly.setElementType (
    elemTypes=(
        ElemType (
            elemCode=C3D8R,
            elemLibrary=STANDARD,
            secondOrderAccuracy=OFF,
            kinematicSplit=AVERAGE_STRAIN,
            hourglassControl=ENHANCED,
            distortionControl=DEFAULT),
        ElemType (
            elemCode=C3D6,
            elemLibrary=STANDARD),
```

```

ElemType (
    elemCode=C3D4,
    elemLibrary=STANDARD)),
regions=(mdb.models['LocalModel'].rootAssembly.instances['Part-
1-1'].cells, ))
a.generateMesh(regions=(i, ))

```

**Listing C-60**

Listing C-60 sets the element types for the surface model and generates the mesh. As with the global model, enhanced hourglass control is used to mitigate against hourglass buckling. The rest of the options can be left at their default values. The final line in the listing is all that is required to generate the mesh, since the automatic meshing tools can perform the task without any more help.

```

local.predefinedFields['InitialTemp'].setValues (
    region=Region(
        cells=i.cells,
        faces=i.faces,
        edges=i.edges,
        vertices=i.vertices))
local.predefinedFields['TemperatureFile'].setValues (
    region=Region(
        cells=i.cells,
        faces=i.faces,
        edges=i.edges,
        vertices=i.vertices))
local.boundaryConditions['Symmetry Condition'].setValues (
    region=Region(faces=i.faces.getSequenceFromMask(mask=(' [#3020000
0 #2 ]', ), ), )))

```

**Listing C-61**

Listing C-61 adjusts the "region" property of the temperature fields and the symmetry condition, overriding the references to the global model geometry.

```

local.SubmodelBC (
    absoluteExteriorTolerance=0.0,
    createStepName='ApplyTemperatures',
    dof=(1, 2, 3),
    exteriorTolerance=0.05,
    globalDrivingRegion='',
    globalIncrement=0,
    globalStep='1',
    name='SubModelBC',
    region=Region(faces=i.faces.getSequenceFromMask(mask=(' [#082860
#4 ]', ), ), ),
    timeScale=OFF)

```

**Listing C-62**

Listing C-62 applies the sub-model boundary condition. It is this boundary condition which is responsible for enforcing the deflections recorded in the global model results file at the

interface between the two models. The "dof" field allows the developer to specify which degrees of freedom should be driven by the global model results. The "exteriorTolerance" parameter is set arbitrarily small and specifies how far a point must be from the sub-model boundary condition to not be included in it. The global increment and step define at what stage of the global model analysis the results are to be taken from, 0 and 1 respectively for a single-step, static model. The global driving region allows the results of a certain part of the global model to be used, but is redundant in this case, since the whole global model is a single part.

```

local.Pressure(
    amplitude=UNSET,
    createStepName='ApplyTemperatures',
    distributionType=USER_DEFINED,
    field='',
    magnitude=1.0,
    name='Pressure',
    region=Region(sidelFaces=i.faces.getSequenceFromMask(mask=('[#c0
104380 #1 ]', ), ))))
local.SurfaceTraction(
    createStepName='ApplyTemperatures',
    directionVector=((0.0, 0.0, 0.0), (0.0, 0.0, 1.0)),
    distributionType=USER_DEFINED,
    field='',
    localCsys=None,
    magnitude=1.0,
    name='Shear',
    region=Region(sidelFaces=i.faces.getSequenceFromMask(mask=('[#c0
104380 #1 ]', ), ))))

```

**Listing C-63**

Listing C-63 applies the normal and shear forces generated by the strip and backup roll contact, overwriting the similarly named boundary conditions in the global model.

```

mdb.Job(
    atTime=None,
        contactPrint=OFF,
        description='LocalModel',
        echoPrint=OFF,
        explicitPrecision=SINGLE,
        getMemoryFromAnalysis=True,
        historyPrint=OFF,
        memory=75,
        memoryUnits=PERCENTAGE,
        model='LocalModel',
        modelPrint=OFF,
        multiprocessingMode=DEFAULT,
        name='LocalJob',
        nodalOutputPrecision=SINGLE,
        numCpus=1,
        numDomains=1,
        queue=None,
        type=ANALYSIS,

```

```

        userSubroutine="" + connected +
"usersubroutine3dG.for",
        waitHours=0,
        waitMinutes=0)
mdb.jobs['LocalJob'].writeInput()

```

**Listing C-64**

Finally, the job is created and the input file written using Listing C-64. The job parameters are identical to the global model, except for the job name. The task of integrating the global and local models is left to input file editing, since ABAQUS CAE will be bypassed for the actual simulation.

### C.1.2. Bite Model

```

local = mdb.Model(name='LocalModel2',
objectToCopy=mdb.models['LocalModel'])
local.setValues(globalJob='LocalJob.odb')

del local.rootAssembly.instances['Part-1-1']
del local.parts['Part-1']
del local.boundaryConditions['SubModelBC']

```

**Listing C-65**

Listing C-65 creates the bite model as a copy of the surface model, similarly to the creation of the surface model from the global model. The listing also links the bite model to the results file for the surface model, and finally deletes all of the elements of the surface model which are not required.

```

lsketch = local.ConstrainedSketch(name='sketch', sheetSize=2.0)
lsketch.ConstructionLine(point1=(0.0, 0.0), point2=(0.0, 2.0))
lsketch.FixedConstraint(entity=lsketch.geometry[2])
lsketch.rectangle(point1=(radius, stripEdge), point2=(hpDepth,
0.0))

```

**Listing C-66**

Listing C-66 creates the sketch which will be used to create the bite model part. The volume used by the bite model begins at the outer radius of the work roll, at a position just beyond the edge of the strip. The slight offset allows room for the stresses to settle, since placing the sub-model interface in an area of highly fluctuating stresses has been shown to have negative effects on model accuracy (see Chapter 5). The inner corner of the bite model lies on the symmetry boundary and at "hpDepth", the heat penetration depth, i.e. the depth at which circumferential temperature fluctuations are assumed to be insignificant.

```

lPart = local.Part(dimensionality=THREE_D, name='Part-1',
type=DEFORMABLE_BODY)
lPart.BaseSolidRevolve(angle=90, flipRevolveDirection=OFF,
sketch=lsketch)
del lsketch

```

**Listing C-67**

Listing C-67 creates the bite model part. To allow the maximum resolution possible, only the area directly around the roll bite is modelled. This is achieved by only sweeping the sketch through ninety degrees, creating a greatly reduced volume, as shown in **Error! Reference source not found.**

```

lPart.SectionAssignment(
    offset=0.0,
    region=Region(cells=lPart.cells),
    sectionName='SteelShellSection')
a = local.rootAssembly
i = a.Instance(dependent=OFF, name='Part-1-1', part=lPart)

```

**Listing C-68**

Listing C-68 assigns the material properties to the part and creates the instance of the part, along with a couple of shortcuts for ease of use.

```

a.seedEdgeByNumber(
    constraint=FIXED,
    edges=i.edges.getSequenceFromMask(['#12a'], ), ),
    number=kSize)
a.seedEdgeByNumber(
    constraint=FIXED,
    edges=i.edges.getSequenceFromMask(['#285'], ), ),
    number=iSize * 2)

```

**Listing C-69**

Listing C-69 seeds the circumferential and axial edges, using a constant mesh spacing. This ensures that the meshed area is smooth as well as being high density.

```

a.seedEdgeByBias(
    biasMethod=SINGLE,
    constraint=FIXED,
    end1Edges=i.edges.getSequenceFromMask(['#50'], ), ),
    end2Edges=i.edges.getSequenceFromMask(['#c00'], ), ),
    number=2/3 * jSize,
    ratio=5.0)

```

**Listing C-70**

Listing C-70 seeds the radial edges, applying a single edge bias to concentrate the mesh density near the surface.

```

a.rotate(
  angle=90.0,
  axisDirection=(0.0, -1.0, 0.0),
  axisPoint=(0.0, 0.0, 0.0),
  instanceList=('Part-1-1', ))

```

**Listing C-71**

Only a quarter of the rotation of the work roll is modelled by the bite model, so the angular location of the model needs to be chosen to coincide with the roll bite. The default starting location for the solid revolve process (the process used to generate the bite model part from the outline sketch) is ninety degrees too far from the roll bite. Listing C-71 is therefore required to rotate the part around to the appropriate angle.

```

local.predefinedFields['InitialTemp'].setValues(
  region=Region(cells=i.cells, faces=i.faces, edges=i.edges,
  vertices=i.vertices))
local.predefinedFields['TemperatureFile'].setValues(
  region=Region(cells=i.cells, faces=i.faces, edges=i.edges,
  vertices=i.vertices))

```

**Listing C-72**

Listing C-72 resets the regions for the temperature fields.

```

local.SubmodelBC(
  absoluteExteriorTolerance=0.0,
  createStepName='ApplyTemperatures',
  dof=(1, 2, 3),
  exteriorTolerance=0.05,
  globalDrivingRegion='',
  globalIncrement=0,
  globalStep='1',
  name='SubModelBC',
  region=Region(faces=i.faces.getSequenceFromMask(mask=['#3c'],
), ), ),
  timeScale=OFF)
local.boundaryConditions['Symmetry Condition'].setValues(
  region=Region(faces=i.faces.getSequenceFromMask(mask=['#2'],
), ), ))
local.loads['Pressure'].setValues(
  region=Region(sidelFaces=i.faces.getSequenceFromMask(mask=['#1
]', ), ), ))
local.loads['Shear'].setValues(
  directionVector=((0.0, 0.0, 0.0), (1, 0.0, 1)),
  region=Region(sidelFaces=i.faces.getSequenceFromMask(mask=['#1
]', ), ), ))

```

**Listing C-73**

Listing C-73 sets the regions for the sub-model and symmetry boundary conditions, as well as the rolling loads. The geometry is much simpler in the bite model, so the mask codes are much smaller.

```
a.generateMesh(regions=(i, ))
```

**Listing C-74**

Listing C-74 generates the mesh for the entire part. Again, the simplified geometry makes the process much simpler by enabled the automatic meshing tools to mesh the whole component without any additional direction.

```
mdb.Job (
    atTime=None,
        contactPrint=OFF,
        description='LocalModel2',
        echoPrint=OFF,
        explicitPrecision=SINGLE,
        getMemoryFromAnalysis=True,
        historyPrint=OFF,
        memory=75,
        memoryUnits=PERCENTAGE,
        model='LocalModel2',
        modelPrint=OFF,
        multiprocessingMode=DEFAULT,
        name='LocalJob2',
        nodalOutputPrecision=SINGLE,
        numCpus=1,
        numDomains=1,
        queue=None,
        type=ANALYSIS,
        userSubroutine="" + connected +
"usersubroutine3dL.for',
        waitHours=0,
        waitMinutes=0)
mdb.jobs['LocalJob2'].writeInput()
```

**Listing C-75**

Finally, Listing C-75 creates the job and writes the input file, ready for editing and feeding to ABAQUS Standard.

### C.1.3. Residual Model

The residual model is almost identical to the global model, only differing in not including any of the loading conditions. The only differences lie in the use of the SIGINI subroutine, which can only be implemented by editing the input file. As such, the python script will not be considered here, since every element of the script has already been described in the "local model" section.



## C.2. UTEMP

---

ABAQUS applies temperature to a finite element model one node at a time, passing through all the nodes in the model until every node has been assigned a temperature. This means that the UTEMP subroutine will be called multiple times to supply individual temperatures at specific points in space.

```
SUBROUTINE UTEMP(TEMP, NSECPT, KSTEP, KINC, TIME, NODE, COORDS)
INCLUDE 'ABA_PARAM.INC'

RETURN
END
```

**Listing C-76**

Listing C-76 shows the FORTRAN code required in the UTEMP subroutine. A selection of parameters are passed in to the subroutine from ABAQUS. The important ones are TEMP, which must be assigned a temperature value before the end of the subroutine, and COORDS, an array containing the coordinate location of the current point in space. The “INCLUDE” command specifies a file containing FORTRAN code, essentially substituting that code in where the command is made. The following space is where the user can insert code to provide the subroutine functionality, before “RETURN” passes the parameters back to ABAQUS and END signifies the end of the subroutine.

```
SAVE
```

**Listing C-77**

The first line added into the subroutine is given in Listing C-77 and is simply the command to SAVE. This causes ABAQUS to make a record of the state of every variable in the subroutine when the subroutine ends. The next time the subroutine is activated the parameters are reset to their previous values. This means that the data files only need to be loaded once, rather than reloaded for every node in the model, reducing the processing time.

### C.2.1. General Case

The general case of the UTEMP subroutine deals with a three dimensional stress models, using both the shell and core temperature results.

Before any functional code can be written the variables used by the subroutine must be declared.

```

REAL, ALLOCATABLE, DIMENSION (:,:) :: ARRAY2D
REAL, ALLOCATABLE, DIMENSION (:,:,) :: ARRAY3D

INTEGER :: ISIZE
INTEGER :: BSIZE

```

**Listing C-78**

For brevity Listing C-78 only gives a brief example of variable declaration, the complete code can be found in the source code. Further variables will be introduced as they are used. The variable types used in the subroutine are;

- Integer – A whole number, integers cannot be used to store fractions or decimals
- Real – A floating point number, allows decimal places and very large or small numbers. s  
Stored in the format  $\pm 1.XXXXXXXXXXX \times 10^{\pm XXX}$
- Array – A collection of integer or real values. Dimensionality is assigned when an array is initialized, e.g. DIMENSION (:) specifies a one dimensional array, DIMENSION(:,) specifies a two dimensional array, etc.

```

IF (LOAD==0) THEN
! OPEN FILES

```

**Listing C-79**

In Listing C-79 the LOAD variable is tested to see whether its value is still the initial value of zero. If it is then the following steps are performed to read in the temperature data, if not then Listing C-80 to Listing C-88 can be skipped as the data has already been loaded. The ‘!’ symbol indicates that the rest of the line is a comment, and is ignored by FORTRAN.

```

OPEN(UNIT=10, STATUS = "OLD", FILE = "C:"
& // "\filepath\WRTMtemp2d.dat")
OPEN(UNIT=11, STATUS = "OLD", FILE = "C:"
& // "\filepath\WRTMtemp3d.dat")

```

**Listing C-80**

The code in Listing C-80 opens the file streams which will be used to read from the data files. The ‘&’ symbol indicates that the following line is a continuation of the previous line and is used to prevent a single line from becoming longer than the 72 character limit. The ‘//’ symbols are the operator for concatenation, meaning that the text before and after the ‘//’ symbol are added together (“Hel” // “lo” is the same as “Hello”).

```

! READ IN 2D DATA
READ(10,*) LENGTH, RADIUS, ISIZE, JSIZE, DELTAZ, DELTAR,
& DELTAR2, INTERPOLATE, INITTEMP

```

**Listing C-81**

The first line of the core temperature data file is read in Listing C-81. Recalling the structure of the core temperature data file the first line contains a list of variable values that relate to the work roll dimensions and other parameters. When the READ function is used in this context FORTRAN sets the value of the first supplied variable equal to the first value in the line, then sets the value of the second variable equal to the second value in the line, etc. until all the variable values have been supplied.

```

ALLOCATE (ARRAY2D ( ISIZE, JSIZE ) )

```

**Listing C-82**

Listing C-82 uses two of the recently set variables (the mesh dimensions for the core model) to allocate the right amount of space in memory for the incoming data. The ISIZE variable gives the number of nodes in the horizontal direction, i.e. along the roll axis, while JSIZE gives the number of nodes in the vertical direction, i.e. in the radial direction.

```

DO J=1, JSIZE
  READ (10,*) ( ARRAY2D(I,J), I = 1, ISIZE )
ENDDO

```

**Listing C-83**

Listing C-83 reads the rest of the temperature file into the ARRAY2D array. This is achieved through two nested loops, the inner of which is contained in the READ function itself. The READ function initially reads a single line from the data file and separates the line into individual values. Then starting at 1 (as specified by I = 1) the function counts up to (the value of) ISIZE, at each step copying a value into the ARRAY2D array, using the values of I and J to specify where in the array the value should be placed. When ISIZE is reached and the process is complete the value of J is incremented and the process restarted for the next line of the data file, until JSIZE is reached. This process is displayed in flow-chart form in Figure C-25 below.

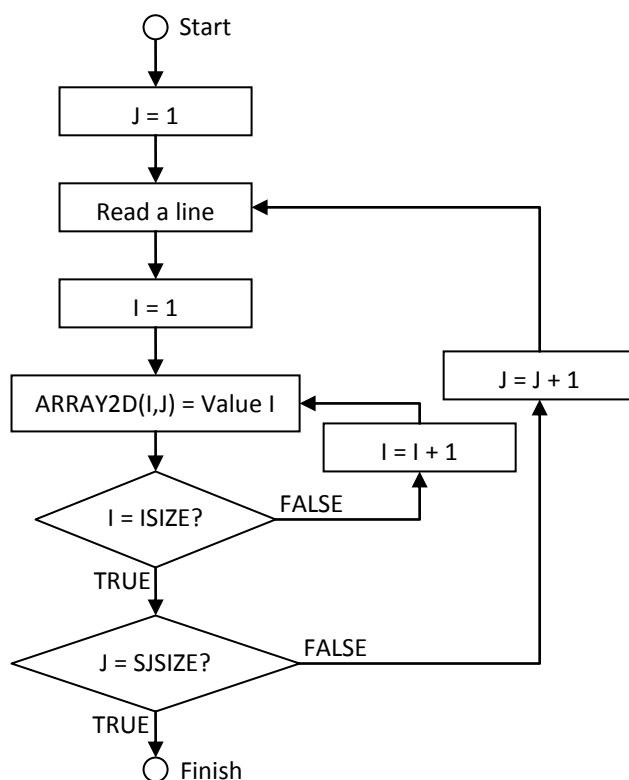


Figure C-25: Process flow for reading in two dimensional temperature data

```

DO J=1, JSIZE
  DO I=1, ISIZE
    ARRAY2D(I, J) = ARRAY2D(I, J) + INITTEMP
  ENDDO
ENDDO
  
```

Listing C-84

Listing C-84 uses the same kind of nested loops as shown in Figure C-25 to iterate through every member of the ARRAY2D array, incrementing every temperature by the initial temperature. This is a feature of the original program, to account for issues in transferring the temperatures without references to an initial temperature.

```

! READ IN 3D DATA
READ (11,*) SISIZE, SJSIZE, KSIZE, DELTATHETA
ALLOCATE (ARRAY3D(SISIZE, SJSIZE, 0:KSIZE + 1))
  
```

Listing C-85

Listing C-85 begins the process of reading in the three dimensional data by reading the first line of variables from the data file and allocating the space required to store the data. The first two dimensions will be accessed using an index which varies from one to the value of SISIZE or SJSIZE respectively. The third dimension varies from index zero to KSIZE+1, giving an extra space at either end to allow for wrap-around, i.e. to simulate the fact that the maximum and minimum angular positions actually refer to the same location.

```

DO K=1, KSIZE
  DO J=1, SJSIZE
    READ(11,*) ( ARRAY3D(I,J,K), I=1, SISIZE )
  ENDDO
  READ(11,*)
ENDDO

```

Listing C-86

Listing C-86 uses another set of nested loops to read in the three dimensional data a line at a time. The flow of the process is outlined in Figure C-26.

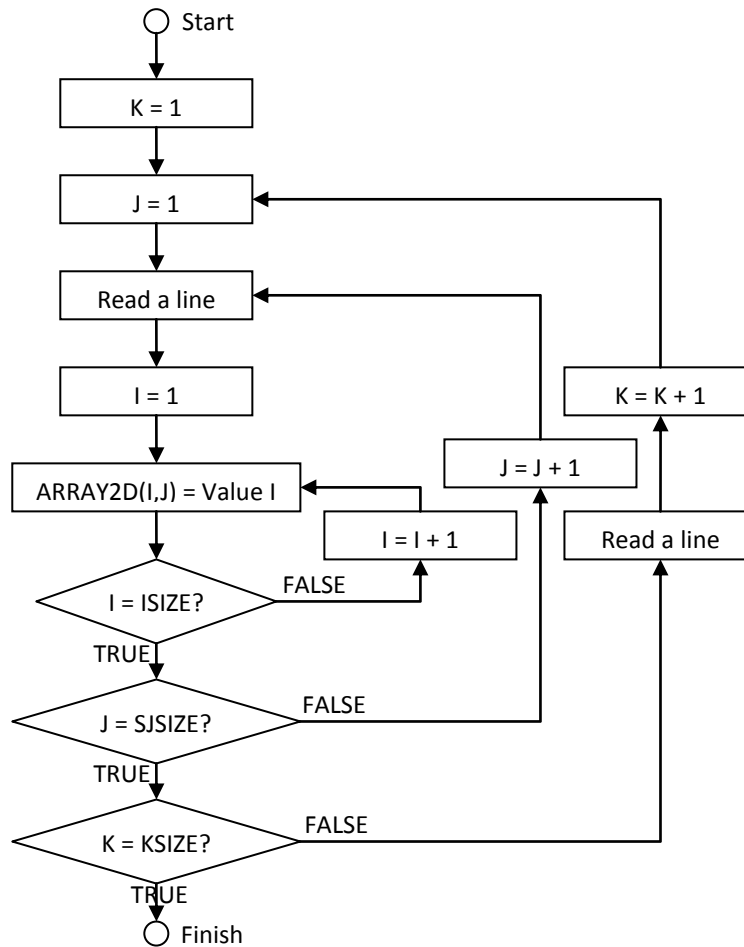


Figure C-26: Process flow for reading in three dimensional temperature data

```

! DO WRAP AROUND IN THETA DIRECTION
DO J=1, SJSIZE
  DO I=1, SISIZE
    ARRAY3D(I,J,0) = ARRAY3D(I,J,KSIZE)
    ARRAY3D(I,J,KSIZE+1) = ARRAY3D(I,J,1)
  ENDDO
ENDDO

```

Listing C-87

Listing C-87 iterates through the range of axial and radial indices, copying the "wrap-around" temperatures mentioned previously to provide a seamless "join" in the data.

```
! CLOSE FILES
CLOSE (UNIT=10)
CLOSE (UNIT=11)
LOAD = 1
ENDIF
```

**Listing C-88**

Listing C-88 closes the file streams and sets the value of the LOAD variable to 1. With the SAVE command making all variable changes persistent, the new value of LOAD means that the subroutine will not attempt to read the data in again, saving time.

```
! GET CARTESIAN COORDINATES
XCOORD = COORDS (2)
YCOORD = COORDS (1)
ZCOORD = COORDS (3)
```

**Listing C-89**

Listing C-89 extracts the coordinate data from the supplied COORDS array, mostly for the purpose of renaming them to make later code easier to understand, but also so that the index referring to a certain direction is only used once. If any changes occur in either the python files or the Fortran files which necessitate a change in dimension indices, the update will only need to be applied once in the Fortran file.

```
! CALCULATE CYLINDRICAL COORDS FROM CARTESIAN COORDINATES
RC = SQRT (ABS (ZCOORD * ZCOORD) + ABS (YCOORD * YCOORD))
TC = 0
ZC = XCOORD + 2 OR ZC = XCOORD
```

**Listing C-90**

Listing C-90 converts the coordinates to a cylindrical reference frame. The term in bold refers to an if-statement in the C# code which writes the user subroutine. If the stress model is not symmetrically collapsed then the axial dimension of the model will stretch from -2m to 2m, whereas the temperature data is stored using an axial dimension of 0m to 4m. The left term accounts for this difference. If the model is symmetrically collapsed then the thermal and stress data use the same axial scale, so no adaptation is required and the right term can be used. A value of zero is given to the angular component, TC, for reasons which will become clear in the following listings.

```
IF (RC/=0) THEN
  IF (YCOORD>0) THEN
```

```

      TC = ACOS (ZCOORD/RC)
    ELSE
      TC = ASIN (ZCOORD/RC) + (3.142 + (3.142/2.0))
    ENDIF

```

**Listing C-91**

Listing C-91 calculates the angular coordinate, a process which requires logical statements to identify which quadrant of the prospective 360 degree space the current point lies in. Two statements combine the 180 degree range of the two trigonometric functions to give coverage of the full range of possible coordinate values. The opening if-statement leaves the angular coordinate as zero if the radial coordinate is zero, avoiding divide-by-zero errors.

```

!DOCTOR TC SO THAT THE HEATING IS IN THE RIGHT AREA
TC = TC + (3.14159265359 / 2) - 0.23
IF (TC > (2 * 3.14159265359)) THEN
  TC = TC - (2 * 3.14159265359)
ENDIF
ENDIF

```

**Listing C-92**

Listing C-92 has the effect of rotating the entire data set through a fixed angle. This rotation moves the roll bite area away from the partitions which break up the work roll stress model geometry, ensuring that the most severe thermal conditions occur where the mesh is likely to be the least warped.

```

IF (INTERPOLATE == 1) THEN

```

**Listing C-93**

Listing C-93 provides a simple and accessible way for a developer to disable the interpolation routine, for error checking or some unpredicted purpose. The variable value is never presented to the user but can be altered in the C# code without needing to delve into the Fortran code.

```

! DO 2D PART
ZTRUNC = INT (ZC/DELTAZ)
RTRUNC = INT (RC/DELTAR)
Y1 = ARRAY2D (ZTRUNC+1, RTRUNC+1)
Y2 = ARRAY2D (ZTRUNC+2, RTRUNC+1)
Y3 = ARRAY2D (ZTRUNC+2, RTRUNC+2)
Y4 = ARRAY2D (ZTRUNC+1, RTRUNC+2)

```

**Listing C-94**

Listing C-94 defines the index of the current position relative to the index system of the temperature data. Y1 through Y4 are then assigned the temperatures of the surrounding

points as indicated in Chapter 4 (Page 188) (the mathematical approach to linear interpolation is described in the thesis so shall not be repeated). The index is offset by 1 and 2, rather than 0 and 1, to allow for the fact that the temperature data is unity indexed (indexes start at 1) rather than zero indexed (indexes start at 0).

```
T= (ZC- (ZTRUNC*DELTAZ)) / (( (ZTRUNC+1)*DELTAZ) - (ZTRUNC*DELTAZ))
U= (RC- (RTRUNC*DELTAR)) / (( (RTRUNC+1)*DELTAR) - (RTRUNC*DELTAR))
TEMP (1) = ((1-T)*(1-U)*Y1) + (T*(1-U)*Y2) + (T*U*Y3) + ((1-T)*U*Y4)
```

**Listing C-95**

Listing C-95 performs the actual interpolation, based on the difference between the whole-number indices and the floating-point coordinates. The TEMP variable is actually an array, although it only takes one value whose index is automatically 1, since all arrays are unity indexed unless otherwise stated in Fortran.

```
! DO 3D PART
ZC = XCOORD + 1 OR ZC = XCOORD
ZTRUNC = INT(ZC/DELTAZ) + 1
IF (RC > RADIUS - ((SJSIZE-1) * DELTAR2)) THEN
  IF (YCOORD > 0) THEN
    TTRUNC = INT(TC / DELTATHETA) + 1
  ELSE
    TTRUNC = INT(TC / DELTATHETA) - 1
  ENDIF
RTRUNC = SJSIZE - INT((RADIUS - RC) / DELTAR2) - 2
```

**Listing C-96**

Listing C-96 recalculates the index variables to suit the shell model data. The axial direction needs to be adjusted again, since the first axial index of the temperature data will either be at the centre line (position X = 0) or at the barrel edge (position X = -1m) depending on which stress model is being used. Again, the C# code can decide which line is suited to dealing with the situation. The other coordinate which really needs close attention is the radial coordinate, since the first radial index of the temperature data is at the heat penetration depth, not zero, and the radial mesh spacing might be completely different to the core model. Knowing the shell thickness in nodes (SJSIZE) and the radial mesh spacing (DELTAR2) allows these values to be calculated.

```
Y1 = ARRAY3D(ZTRUNC+1, RTRUNC+1, TTRUNC+1)
Y2 = ARRAY3D(ZTRUNC+2, RTRUNC+1, TTRUNC+1)
Y3 = ARRAY3D(ZTRUNC+2, RTRUNC+2, TTRUNC+1)
Y4 = ARRAY3D(ZTRUNC+1, RTRUNC+2, TTRUNC+1)
Y5 = ARRAY3D(ZTRUNC+1, RTRUNC+1, TTRUNC+2)
Y6 = ARRAY3D(ZTRUNC+2, RTRUNC+1, TTRUNC+2)
Y7 = ARRAY3D(ZTRUNC+2, RTRUNC+2, TTRUNC+2)
Y8 = ARRAY3D(ZTRUNC+1, RTRUNC+2, TTRUNC+2)
```



#### Listing C-97

Listing C-97 assigns the temperatures of the eight nodes surrounding the current coordinate to the Y variables.

```
T = (ZC / DELTAZ) - INT(ZC / DELTAZ)
U = (RC / DELTAR2) - INT(RC / DELTAR2)
V = (TC / DELTATHETA) - INT(TC / DELTATHETA)
```

#### Listing C-98

Listing C-98 then calculates the weighting variables.

```
PLANE1 = ((1-T)*(1-U)*Y1) + (T*(1-U)*Y2) + (T*U*Y3) + ((1-
T)*U*Y4)
PLANE2 = ((1-T)*(1-U)*Y5) + (T*(1-U)*Y6) + (T*U*Y7) + ((1-
T)*U*Y8)
TEMP(1) = ((1-V)*PLANE1) + (V*PLANE2)
ENDIF
```

#### Listing C-99

Finally, Listing C-99 performs the interpolation.

```
ELSE
! NON INTERPOLATED VERSION
ENDIF
```

#### Listing C-100

Listing C-100 would show the non-interpolation version of the subroutine, which has been omitted since it is only ever used for debugging purposes and bears no real relevance. This marks the end of the user subroutine. With no more lines of code to process, control would return to ABAQUS along with the newly calculated temperature value.

### C.2.2. 2D verification

```
REAL, ALLOCATABLE, DIMENSION (:) :: ARRAY2D
REAL, ALLOCATABLE, DIMENSION (:,:) :: TEMP2D
REAL, ALLOCATABLE, DIMENSION (:,:) :: ARRAY3D
REAL, ALLOCATABLE, DIMENSION (:,:,) :: TEMP3D
```

#### Listing C-101

The first main difference between the 2D verification model UTEMP subroutine and the main model UTEMP subroutine can be found in the array declarations. The ARRAY2D and

ARRAY3D arrays are reduced dimension versions of the full scale data arrays. Because the 2D model only uses a thin slice of the temperature data it is easier to isolate that slice at the beginning of the process.

```

IF (LOAD==0) THEN
! OPEN THE 2D FILE
OPEN(UNIT=11, STATUS = "OLD", FILE = "C:\filepath\WRTMtemp2d.dat")
! READ IN 2D DATA
READ(11,*) LENGTH, RADIUS, ISIZE, JSIZE, DELTAZ, DELTAR,
& DELTAR2, INTERPOLATE, INITTEMP

ALLOCATE (ARRAY2D(JSIZE))
ALLOCATE (TEMP2D(ISIZE,JSIZE))

DO J=1, JSIZE
  READ (11,*) ( TEMP2D(I, J), I = 1, ISIZE )
ENDDO

```

**Listing C-102**

Listing C-102 opens the 2D data file, loads the first line of variables, allocates the array space and reads in the data. The process is mostly identical to the process used in the main model.

```

CENTRE = ISIZE / 2 OR CENTRE = 1

DO J=1, JSIZE
  ARRAY2D(J) = TEMP2D(CENTRE, J) + INITTEMP
ENDDO

CLOSE (UNIT=11)
LOAD=1

```

**Listing C-103**

The first line of Listing C-103 defines the centre plane location in terms of its axial index. This index will either be half the length of the data set or the first element in the data set, for non-symmetrically-expanded and symmetrically expanded models respectively. The C# code picks the relevant statement.

```

! OPEN THE 3D FILE
OPEN(UNIT=11, STATUS = "OLD", FILE = "C:\filepath\WRTMtemp3d.dat")

! READ IN 3D DATA
READ (11,*) SISIZE, SJSIZE, KSIZE, DELTATHETA
ALLOCATE (TEMP3D(SISIZE, SJSIZE, 0:KSIZE + 1))
ALLOCATE (ARRAY3D(SJSIZE, 0: KSIZE + 1))
DO K=1, KSIZE
  DO J=1, SJSIZE
    READ(11,*) ( TEMP3D(I,J,K), I=1, SISIZE )
  ENDDO
  READ(11,*)
ENDDO

```

```

CLOSE (UNIT=11)

! DO WRAP AROUND IN THETA DIRECTION
DO J=1, SJSIZE
  DO I=1, SISIZE
    TEMP3D(I,J,0) = TEMP3D(I,J,KSIZE)
    TEMP3D(I,J,KSIZE+1) = TEMP3D(I,J,1)
  ENDDO
ENDDO

```

**Listing C-104**

While reading in the thermal data, Listing C-104 does not present many differences from the main UTEMP definition described previously.

```

CENTRE = SISIZE / 2 OR CENTRE = 1

DO J=1, SJSIZE
  DO K=0, KSIZE + 1
    ARRAY3D(J,K) = TEMP3D(CENTRE,J,K)
  ENDDO
ENDDO
ENDIF

```

**Listing C-105**

As with the two dimensional data set, Listing C-105 calculates the centre plane axial coordinate and uses it to reduce the dimensionality of the three dimensional data set.

```

! CONVERT COORDINATES
XCOORD = COORDS(2)
YCOORD = COORDS(1)
ZCOORD = COORDS(3)

RCOORD = SQRT(ABS(XCOORD * XCOORD) + ABS(YCOORD * YCOORD))
THETAC = 0
IF (RCOORD/=0) THEN
  IF (YCOORD>0) THEN
    THETAC = ACOS(XCOORD/RCOORD)
  ELSE
    THETAC = ASIN(XCOORD/RCOORD) + (3.14159265359 +
(3.14159265359/2.0))
  ENDF
ENDIF

! DOCTOR THETAC TO MARRY WITH 3D MODEL
THETAC = THETAC + (3.14159265359 / 2) - 0.23
IF (THETAC > (2 * 3.14159265359)) THEN
  THETAC = THETAC - (2 * 3.14159265359) "
ENDIF
ENDIF

RC = INT(RCOORD / DELTAR) + 1
TC = INT(THETAC / DELTATHETA) + 1

```

**Listing C-106**

Listing C-106 is the same as the equivalent lines of the main model UTEMP subroutine in converting the coordinates from rectangular to cylindrical.

```

! CALCULATE VARIABLES IF IN 3D RESULTS AREA
IF (RC > (RADIUS - ((SJSIZE-1) * DELTAR2)) / DELTAR) THEN
  RCS = INT((RCOORD - (RADIUS - ((SJSIZE - 1) * DELTAR2))) /
DELTAR2) + 1

IF (TC == 257) THEN
  TC = 1
  THETAC = 0
ENDIF

```

**Listing C-107**

Listing C-107 opens an if-statement that identifies whether the radial coordinate puts the current point within the shell material. If it does, the listing calculates the relevant shell index and adjusts the theta coordinate to avoid interpolation issues on the edge of the temperature data set.

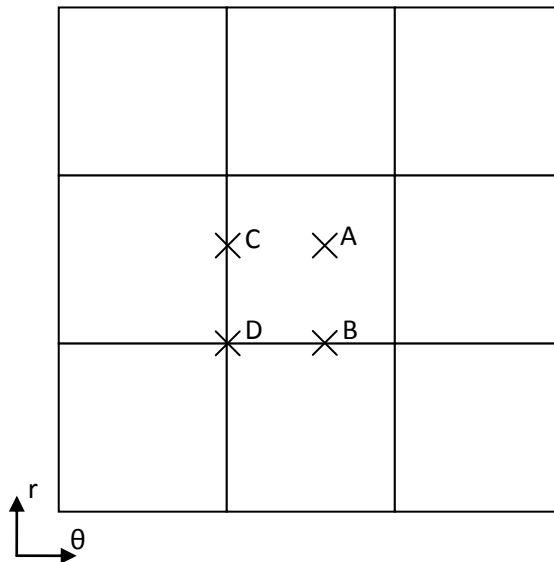
```

! DO 3D INTERPOLATION
IF ((TC-1) * DELTATHETA < THETAC) THEN
  IF (RCS < ((RCOORD-RADIUS)/DELTAR2)+SJSIZE) THEN
    U = ((RCOORD-RADIUS)/DELTAR2)+SJSIZE-RCS
    V = (THETAC / DELTATHETA) - (TC - 1)
    TEMP(1) = (1-U) * (1-V) * ARRAY3D(RCS, TC) + U * (1-V) * ARRAY3D(RCS+1
& , TC) + (1-U) * V * ARRAY3D(RCS, TC+1) + U * V * ARRAY3D(RCS+1, TC+1)
  ELSE
    V = (THETAC / DELTATHETA) - (TC - 1)
    TEMP(1) = (1-V) * ARRAY3D(RCS, TC) + V * ARRAY3D(RCS, TC+1)
  ENDIF
ELSE IF (RCS < ((RCOORD-RADIUS)/DELTAR2)+SJSIZE) THEN
  U = ((RCOORD-RADIUS)/DELTAR2)+SJSIZE-RCS
  TEMP(1) = (1-U) * ARRAY3D(RCS, TC) + U * ARRAY3D(RCS+1, TC)
ELSE
  TEMP(1) = ARRAY3D(RCS, TC)
ENDIF
ENDIF

```

**Listing C-108**

Listing C-108 contains some fairly dense code which can be simplified by considering the end of each logical branch. The first if-statement isolates those points whose angular coordinate does not lie exactly on top of the angular coordinate of one of the temperature data points, i.e. a point which needs interpolation in the angular direction. The second if-statement isolates those points which are in the first category but also do not lie directly on top of the radial coordinate of one of the temperature data points. This scenario corresponds to case A in Figure C-27 below, where the cross marks the location of the FE node coordinate and the corner of each square marks the location of a point in the temperature data file.



**Figure C-27: potential coordinate locations when comparing a FE node position to the temperature data set.**

The first "ELSE" statement will be called only if the first statement is true and the second is false, so the point needs angular interpolation but not radial interpolation. This corresponds to case B in the figure. If the point does not need angular interpolation then a second set of if statements are required to discern between those points which need radial interpolation and those which do not, represented by cases C and D respectively. These four cases will catch any possible location that a coordinate can be and provide the required level of linear interpolation.

```

! DO 2D INTERPOLATION
ELSE
IF ((RC-1) * DELTAR < RCOORD) THEN
! INTERPOLATION REQUIRED
U = (RCOORD / DELTAR) - (RC - 1)
TEMP(1) = (1-U) * ARRAY2D(RC) + U * ARRAY2D(RC+1)
ELSE
! EXACT MATCH FOUND
TEMP(1) = ARRAY2D(RC)
ENDIF
ENDIF
ENDIF

```

**Listing C-109**

If the FE node coordinate is not within the range of the shell model then only the core model results are needed. Listing C-109 performs the interpolation, with an if-statement to check whether any interpolation is needed at all, based on the node coordinate.

```

CLOSE (UNIT=10)
RETURN
END

```

### Listing C-110

Finally, Listing C-110 closes the subroutine, returning the calculated temperature value.

## C.3. UTRAC

---

The UTRAC subroutine is responsible for applying shear forces to surfaces, in this case to the surface of the work roll.

```
SUBROUTINE UTRACLOAD (ALPHA, T_USER, KSTEP, KINC, TIME, NOEL, NPT,
& COORDS, DIRCOS, JLTYP, SNAME)

INCLUDE 'ABA_PARAM.INC'

DIMENSION T_USER(3), TIME(2), COORDS(3), DIRCOS(3,3)
CHARACTER*80 SNAME

RETURN
END
```

### Listing C-111

Listing C-111 shows the required code for the subroutine, including the subroutine definition, include parameters and variable definitions. Any additional code must be supplied between the variable definitions and the RETURN and END statements. The ALPHA variable represents the shear stress at the current position and has a default value of zero. The coordinate data is loaded into discrete variables and converted to a cylindrical reference frame in the usual, so that section of the code will not be repeated.

```
!CALCULATE TRACTION VALUE
if (Expanded)
IF (ZC < 2 + (StripWidth / 2)) THEN
  IF (ZC > 2 - (StripWidth / 2)) THEN
else
IF (ZC < StripWidth / 2) THEN
```

### Listing C-112

Listing C-112 isolates the area within the strip with, since there will only be a shear force where the strip is in contact with the roll. The C# code chooses which of the sets of statements to write to the subroutine file, based on the type of model results being used. The other terms in bold can also be written directly into the subroutine file, since the C# code will have access to all of the process data.

```
IF (TC < BiteAngle) THEN
```

```

X = BiteAngle / 2 - TC
IF (X > 0) THEN
  ALPHA = - alpha
ELSE
  ALPHA = alpha
ENDIF
ENDIF

```

**Listing C-113**

In Listing C-113 the ALPHA variable is given its magnitude, in this case based on which half of the roll bite the point lies in. The values in bold are all calculated in the C# code and written directly into the subroutine file. The directionality of the shear stress is defined by the sense of alpha in combination with the direction vector defined in Listing C-42.

```

if (Expanded)
  ENDF
ENDIF
else
  ENDF

```

**Listing C-114**

Listing C-114 ends the subroutine by closing the if-statements that were opened in Listing C-112.

## C.4. DLOAD

---

The DLOAD subroutine applies pressures normal to a surface.

```

SUBROUTINE DLOAD (F, KSTEP, KINC, TIME, NOEL, NPT, LAYER, KSPT,
1 COORDS, JLTYP, SNAME)

INCLUDE 'ABA_PARAM.INC'

DIMENSION TIME (2), COORDS (3)
CHARACTER*80 SNAME

RETURN
END

```

**Listing C-115**

Listing C-115 shows the required code for the DLOAD subroutine. The F variable represents the pressure normal to the surface at the given location and has a default value of zero. The section of code which imports the current coordinates and converts them to a cylindrical reference frame has been withheld for brevity.

```

if (!global)
! Backup Position
IF (TC < 3.14159265359 + BiteAngleBackup * 2) THEN
IF (TC > 3.14159265359) THEN
X = TC - 3.14159265359
X = X - BiteAngleBackup
IF (X < 0) THEN
X = -X
ENDIF
F = P0Backup * (1 - ((X / BiteAngleBackup) * (X /
BiteAngleBackup)))
ENDIF
ENDIF

```

Listing C-116

Listing C-116 applies the normal pressure between the work roll and backup roll, using a Hertzian profile as described in Chapter 4 (Page 198). The terms in bold are calculated in the C# code. The whole of the listing is contained in a C# if-statement, conditional on the current model being of higher resolution than the global model.

```

else
! Backup Position
IF (TC < 3.14159265359 + BiteAngle) THEN
IF (TC > 3.14159265359) THEN
X = TC - 3.14159265359
X = X - BiteAngle / 2
IF (X < 0) THEN
X = -X
ENDIF
F = P0Backup * (1 - ((X / BiteAngle / 2) * (X / BiteAngle / 2)))
X = X / BiteAngle / 2
ENDIF
ENDIF

```

Listing C-117

If the current model is the global model then the resolution will be too low to use the actual backup contact size. The contact would be too small and it would be impossible to properly balance the backup and strip pressures to prevent undesirable rotation. Listing C-117 solves this problem with a C# else clause, an alternative definition which uses the bite width as the contact size, rather than the smaller backup contact size. This approach is justified by ensuring that the same overall force is applied in both scenarios. The global model will be inaccurate at the surface, but if the overall applied force is the same then the response at a distance from the surface will be the same. The surface model can then provide accurate results near the surface which will take precedent over the inaccurate global model results, concealing any errors.

```

if (Expanded)
! Roll Bite Position
IF (ZC < 2 + (StripWidth / 2)) THEN

```



```

IF (ZC > 2 - (StripWidth / 2)) THEN

else
! Roll Bite Position
IF (ZC < StripWidth / 2) THEN

```

Listing C-118

In Listing C-118 the C# code chooses between the two sets of statements. If the model is not symmetrically reduced (i.e. if the model has been 'expanded' from the original, reduced version) then the normal pressure will lie between the two strip edges. Otherwise the axial coordinate will be less than half the strip width.

```

IF (TC < BiteAngle) THEN
X = TC - BiteAngle / 2
IF (X < 0) THEN
X = -X
ENDIF
F = P0 * (1 - ((X / BiteAngle / 2) * (X / BiteAngle / 2)))
ENDIF

```

Listing C-119

Listing C-119 applies the pressure profile for any point within the roll bite, based on a Hertzian pressure profile, as described previously.

```

if (Expanded)
ENDIF
ENDIF

else
ENDIF

```

Listing C-120

Listing C-120 closes the subroutine, which takes one or two ENDIF statements, depending on the model type.

## C.5. SIGINI

---

```

SUBROUTINE SIGINI (SIGMA, COORDS, NTENS, NCRDS, NOEL, NPT,
LAYER, KSPT, LREBAR, NAMES)

INCLUDE 'ABA_PARAM.INC'
SAVE

DIMENSION SIGMA (NTENS), COORDS (NCRDS)
CHARACTER NAMES (2) *80

RETURN
END

```

#### Listing C-121

Listing C-121 gives the required code for the SIGINI subroutine. In this subroutine SIGMA is the variable which must be defined, an array which takes six values, one for each component of the stress tensor.

```
REAL, ALLOCATABLE, DIMENSION (:,:) :: SIG
REAL, ALLOCATABLE, DIMENSION (:) :: RAD
```

#### Listing C-122

Listing C-122 declares the arrays which will be used in the subroutine. The SIG routine takes six values in its first dimension, each representing a component of the stress tensor. Each index in the second dimension represents a data point in the residual data file. The RAD array is a single dimension and gives the radial coordinate of each data point in the residual data file.

```
! Load files if they haven't been loaded already
IF (LOAD == 0) THEN
OPEN(UNIT=10, STATUS = "OLD", FILE = "C:\filepath\Residuals.dat")
```

#### Listing C-123

As with the UTEMP subroutine, there is no point loading the residual data multiple times. The if-statement in Listing C-123 uses the LOAD variable to discern whether the data has already been loaded. If not, then a stream to the data file is opened.

```
READ (10, *) MAXINDEX
ALLOCATE (SIG(6, MAXINDEX))
ALLOCATE (RAD(MAXINDEX))
DO I=1, MAXINDEX
READ (10, *)
SIG(1,I),SIG(2,I),SIG(3,I),SIG(4,I),SIG(5,I),SIG(6,I),RAD(I)
ENDDO
LOAD = 1
CLOSE (UNIT=10)
ENDIF
```

#### Listing C-124

Listing C-124 allocates the array sizes and loads the residual stress data.

```
! Calculate the stress at this point
INDEX = 0
RADIAL = SQRT((COORDS(1)*COORDS(1))+(COORDS(3)*COORDS(3)))
DO I=1, MAXINDEX-1
IF (RAD(I)>RADIAL.AND.RADIAL>RAD(I+1)) THEN
INDEX = I
ENDIF
```

```
ENDDO
```

#### Listing C-125

Listing C-125 calculates the radial coordinate of the current point, then compares it to each of the data file's radial coordinates in turn, saving the index position for the nearest entry to the current coordinate.

```
IF (INDEX == 0) THEN
IF (RADIAL>=RAD(1)) THEN
INDEX = 1
ELSEIF (RADIAL <= RAD(MAXINDEX)) THEN
INDEX = MAXINDEX
ELSE
INDEX = 1
ENDIF
ENDIF
```

#### Listing C-126

Listing C-126 ensures that a legal option is picked, correcting for any issues relating to points being fractionally outside of the coordinates in the data file. If the index is zero then it hasn't been updated from its initial value in Listing C-125. This could either be because the radial coordinate is greater than the maximum coordinate in the file, or less than the minimum. The first two if-statements catch any instances of the coordinate being out of bounds. The final case is a fail-safe, for unforeseen circumstances.

```
IF (INDEX < MAXINDEX) THEN
U = (RADIAL-RAD(INDEX)) / (RAD(INDEX+1) -RAD(INDEX))
ELSE
U = 0
ENDIF
```

#### Listing C-127

Listing C-127 sets the variable value used for linear interpolation, forcing it to be zero if the current index is the maximum index. This prevents a small error causing Fortran to attempt to access the data point beyond the final index, which by definition does not exist, for the sake of interpolation.

```
IF (U==0.OR.INDEX==MAXINDEX) THEN
SIG1 = SIG(1, INDEX)
SIG2 = SIG(2, INDEX)
SIG3 = SIG(3, INDEX)
SIG4 = SIG(4, INDEX)
SIG5 = SIG(5, INDEX)
SIG6 = SIG(6, INDEX)
ELSE
SIG1 = (1-U)*SIG(1, INDEX)+U*SIG(1, INDEX+1)
SIG2 = (1-U)*SIG(2, INDEX)+U*SIG(2, INDEX+1)
SIG3 = (1-U)*SIG(3, INDEX)+U*SIG(3, INDEX+1)
```

```

SIG4 = (1-U)*SIG(4, INDEX)+U*SIG(4, INDEX+1)
SIG5 = (1-U)*SIG(5, INDEX)+U*SIG(5, INDEX+1)
SIG6 = (1-U)*SIG(6, INDEX)+U*SIG(6, INDEX+1)
ENDIF

```

**Listing C-128**

Listing C-128 performs a linear interpolation between the data points which surround the current coordinate. If U is equal to zero then no interpolation is required, so the relevant value can be assigned directly.

```

! Transform from cylindrical to rectangular coordinates
R11 = COORDS(1)/SQRT((COORDS(1)*COORDS(1)+(COORDS(3)*COORDS(3)))
R31 = COORDS(3)/SQRT((COORDS(1)*COORDS(1)+(COORDS(3)*COORDS(3)))
R13 = -R31
R33 = R11

```

**Listing C-129**

The previous listing calculated the radial, hoop and axial components of the stress tensor, along with the associated shear components. Unfortunately, ABAQUS always works to a rectangular coordinate frame, meaning that the stress tensor will need to be rotated depending on the angular coordinate of the current point. Listing C-129 calculates the terms of the rotation matrix required to perform the coordinate rotation. Further information on the theory of this process can be found in Chapter 4 (Page 204).

```

SIGMA(1) = R11*R11*SIG1 + R13*R13*SIG3 + 2*R11*R13*SIG5
SIGMA(2) = SIG2
SIGMA(3) = R31*R31*SIG1 + R33*R33*SIG3 + 2*R31*R33*SIG5
SIGMA(4) = R11*SIG4 + R13*SIG6
SIGMA(5) = R11*R31*SIG1 + R13*R33*SIG3 + (R11*R33 + R13*R31)*SIG5
SIGMA(6) = R31*SIG4 + R33*SIG6

```

**Listing C-130**

Listing C-130 performs the expanded matrix multiplication to give the component values in their newly rotated reference frame.

```

IF (COORDS(2) > (rollLength * 3) / 8) THEN
X = 1 - ((COORDS(2) - 3 * rollLength / 8) / rollLength / 8)
IF (X < 0) THEN
X = 0
ENDIF
SIGMA(1) = X * SIGMA(1)
SIGMA(2) = X * SIGMA(2)
SIGMA(3) = X * SIGMA(3)
SIGMA(4) = X * SIGMA(4)
SIGMA(5) = X * SIGMA(5)
SIGMA(6) = X * SIGMA(6)
ENDIF

```

**Listing C-131**

Finally, Listing C-131 tapers off the residual stresses towards the end of the work roll. This allows for the fact that the axial stress, a large component of the residual stress, has to become zero at the end of the work roll.

## Appendix D. Data

---

### D.1. Debug data set

---

Comment	Gap	Rolling
Time	120	64.5
WR_radius	0.351425	0.351425
WR_length	2	2
WR_journalradius	0.2	0.2
WR_journallength	1	1
WR_inittemp	21	21
WR_core_conductivity	22	22
WR_core_density	7150	7150
WR_core_specificeat	460	460
WR_core_youngsmodulus	1.70E+11	1.70E+11
WR_core_poissonsratio	0.28	0.28
WR_core_expansioncoefficient	9.48E-06	9.48E-06
WR_shell_conductivity	25	25
WR_shell_density	7.80E+03	7.80E+03
WR_shell_specificeat	410	410
WR_shell_youngsmodulus	2.25E+11	2.25E+11
WR_shell_poissonsratio	0.31	0.31
WR_shell_expansioncoefficient	8.68E-06	8.68E-06
WR_shelldepth	0.1	0.1
Pstand	0.00E+00	4.31E+06
beta	0	0.3
bitelength	0	0.157544
angularvelocity	12.43962	12.43962
stripwidth	0	1.038839
Tstrip	0	1197.875
Tambientair	20	20
Twatert	21.4	21.4
hbarreledge	4000	4000
hjournaledge	50	50
hstriptoroll	0	24000
dissipationportion	0	0.3
hbearing	1000	1000
Tbearinghousing	40	40
hwiper	400	400
hbackup	10000	10000
hsurfacewaterpeak	4000	4000
hsurfacewaterlow	4000	4000

deltat	1	1
relaxation	1	1
targeterror	1.00E-10	1.00E-10
cyclelimit	750000	750000
exitcentreflowrate	2.63E+03	2.63E+03
exitedgeflowrate	718	718
entrycentreflowrate	2971	2971
entryedgeflowrate	1087	1087

## D.2. Extended debug data set

CoilNumber	7496110	7496110	7496110	0
Stage	0	1	2	3
Comment	Gap	Rolling	Post Round Gap	Cooling to air time
Time	120	64.5	30	2700
WR_radius	0.351425	0.351425	0.351425	0.351425
WR_length	2	2	2	2
WR_journalradius	0.2	0.2	0.2	0.2
WR_journallength	1	1	1	1
WR_inittemp	21	21	21	21
WR_core_conductivity	22	22	22	22
WR_core_density	7150	7150	7150	7150
WR_core_specifichat	460	460	460	460
WR_core_youngsmodulus	1.70E+11	1.70E+11	1.70E+11	1.70E+11
WR_core_poissonsratio	0.28	0.28	0.28	0.28
WR_core_expansioncoefficient	9.48E-06	9.48E-06	9.48E-06	9.48E-06
WR_shell_conductivity	25	25	25	25
WR_shell_density	7800	7800	7800	7800
WR_shell_specifichat	410	410	410	410
WR_shell_youngsmodulus	2.25E+11	2.25E+11	2.25E+11	2.25E+11
WR_shell_poissonsratio	0.31	0.31	0.31	0.31
WR_shell_expansioncoefficient	8.68E-06	8.68E-06	8.68E-06	8.68E-06
WR_shelldepth	0.1	0.1	0.1	0.1
Pstand	0.00E+00	4.31E+06	0	0
beta	0	0.3	0	0
bitlength	0.00E+00	1.58E-01	0	6.283185
angularvelocity	12.43962	12.43962	12.43962	0
stripwidth	0.00E+00	1.04E+00	0	2
Tstrip	0	1197.875	0	20
Tambientair	2.00E+01	2.00E+01	20	20
Twatert	21.4	21.4	21.4	20
hbarreledge	4000	4000	4000	1.6
hjournaledge	50	50	50	1.6
hstriptoroll	0	24000	0	1.6

dissipationportion	0	0.3	0	0
hbearing	1000	1000	1000	1.6
Tbearinghousing	40	40	40	20
hwiper	400	400	400	1.6
hbackup	10000	10000	10000	1.6
hsurfacewaterpeak	4000	4000	4000	1.6
hsurfacewaterlow	4000	4000	4000	1.6
deltat	1	1	1	10
relaxation	1	1	1	1
targeterror	1.00E-10	1.00E-10	1.00E-10	1.00E-10
cyclelimit	750000	750000	750000	750000
exitcentreflowrate	2630	2630	2630	0
exitedgeflowrate	718	718	718	0
entrycentreflowrate	2971	2971	2971	0
entryedgeflowrate	1087	1087	1087	0

### D.3. Heavy duty test data set

---

#### D.3.1. Geometry and materials

---

WR_radius	0.353325	WR_shell_poissonsratio	0.31
WR_length	2	WR_shell_expansioncoefficient	8.68E-06
WR_journalradius	0.2	WR_shelldepth	0.05
WR_journallength	1	Tambientair	20
WR_inittemp	21	Twatert	21.4
WR_core_conductivity	22	hbarreledge	50
WR_core_density	7150	hjournaledge	50
WR_core_specificeat	460	hstriptoroll	24000
WR_core_youngsmodulus	1.70E+11	dissipationportion	0.065
WR_core_poissonsratio	0.28	hbearing	1000
WR_core_expansioncoefficient	9.48E-06	Tbearinghousing	40
WR_shell_conductivity	25	deltat	1
WR_shell_density	7800	relaxation	1
WR_shell_specificeat	410	targeterror	1.00E-10
WR_shell_youngsmodulus	2.25E+11	cyclelimit	750000



### D.3.2. Process data

CoilNumber	Stage	Comment	Time	Pstand	StripForce	BendingForce	bitlength	angularvelocity	stripwidth	Tstrip	exitcentreflowrate	exitedgeflowrate	entrycentreflowrate	entryedgeflowrate
2854420	0	Gap	190.6	0	0	0	0	10.21479	0	0	1497	0	5265	39
2854420	1	Rolling	72.6	2350844	12301090	784800	0.136856	10.21479	1.296774	1199.098	1497	0	5265	39
2854510	2	Gap	26.8	0	0	0	0	11.66607	0	0	1477	0	5267	37
2854510	3	Rolling	60.9	3751852	15974120	1863900	0.140371	11.66607	1.291533	1184.32	1477	0	5267	37
2854520	4	Gap	43	0	0	0	0	11.94898	0	0	1475	0	5250	39
2854520	5	Rolling	64.4	3605797	15679690	1359738	0.144558	11.94898	1.293812	1191.298	1475	0	5250	39
2854610	6	Gap	48.9	0	0	0	0	12.18151	0	0	1487	0	5275	39
2854610	7	Rolling	68.4	3723959	16148650	196200.6	0.146683	12.18151	1.287819	1192.556	1487	0	5275	39
2854710	8	Gap	34.3	0	0	0	0	11.77593	0	0	1480	0	5250	43
2854710	9	Rolling	68.1	3485503	15480150	642662.6	0.142219	11.77593	1.296316	1198.766	1480	0	5250	43
2854810	10	Gap	71.2	0	0	0	0	8.387558	0	0	1486	0	5292	41
2854810	11	Rolling	86.8	2567971	15752730	1863899	0.147563	8.387558	1.321511	1219.124	1486	0	5292	41
2854910	12	Gap	33.5	0	0	0	0	11.49231	0	0	1496	0	5302	44
2854910	13	Rolling	65	3796011	15524430	196199.9	0.136413	11.49231	1.489582	1181.586	1496	0	5302	44
2855010	14	Gap	40.6	0	0	0	0	12.76747	0	0	1496	0	5302	44
2855010	15	Rolling	46	4136879	14689480	196200.4	0.137502	12.76747	1.517628	1214.037	1496	0	5302	44
2855110	16	Gap	67.8	0	0	0	0	11.14138	0	0	1495	0	5277	42
2855110	17	Rolling	74.9	4137039	19533940	1517914	0.134535	11.14138	1.531147	1197.478	1495	0	5277	42
2855210	18	Gap	66	0	0	0	0	12.03947	0	0	1491	0	5307	42
2855210	19	Rolling	74.1	4275810	18675110	196199.6	0.133778	12.03947	1.547163	1174.22	1491	0	5307	42
2855310	20	Gap	60.8	0	0	0	0	12.58344	0	0	1495	0	5285	42
2855310	21	Rolling	72.6	4404079	17184580	196199.8	0.138405	12.58344	1.550476	1197.128	1495	0	5285	42

CoilNumber	Stage	Comment	Time	Pstand	StripForce	BendingForce	bitelength	angularvelocity	stripwidth	Tstrip	exitcentreflowrate	exitedgeflowrate	entrycentreflowrate	entryedgeflowrate
2855410	22	Gap	52.2	0	0	0	0	12.82809	0	0	1495	0	5285	42
2855410	23	Rolling	55.7	4422084	16662900	196199.3	0.136093	12.82809	1.587448	1216.432	1495	0	5285	42
2855510	24	Gap	90	0	0	0	0	8.886371	0	0	1477	0	5260	38
2855510	25	Rolling	73	3387638	20168220	1863900	0.134213	8.886371	1.630628	1237.077	1477	0	5260	38
2855610	26	Gap	76	0	0	0	0	14.4622	0	0	1477	0	5260	38
2855610	27	Rolling	56.9	5178853	17946020	196199.9	0.130112	14.4622	1.734273	1212.946	1477	0	5260	38
2855710	28	Gap	102.9	0	0	0	0	13.87833	0	0	1491	0	5282	40
2855710	29	Rolling	52.7	4553782	16646660	196200.2	0.124905	13.87833	1.73883	1224.08	1491	0	5282	40
2855810	30	Gap	117.6	0	0	0	0	14.65506	0	0	1506	0	5290	36
2855810	31	Rolling	65.5	5117491	18759760	431398.5	0.128272	14.65506	1.756473	1210.341	1506	0	5290	36
2855910	32	Gap	82.3	0	0	0	0	13.8009	0	0	1487	0	5272	37
2855910	33	Rolling	58.8	5247841	18959940	196199.9	0.133762	13.8009	1.840028	1222.671	1487	0	5272	37
2856010	34	Gap	121.9	0	0	0	0	12.92301	0	0	1493	0	5305	40
2856010	35	Rolling	66.9	6468554	22490060	196199.8	0.131884	12.92301	1.843032	1198.363	1493	0	5305	40
2856020	36	Gap	106.7	0	0	0	0	12.5348	0	0	1475	0	5270	43
2856020	37	Rolling	70	6126052	21769140	196199.5	0.13242	12.5348	1.842628	1202.926	1475	0	5270	43
2856110	38	Gap	106	0	0	0	0	9.753949	0	0	1492	0	5302	40
2856110	39	Rolling	85.5	4029659	22847310	980463.3	0.128298	9.753949	1.851992	1190.839	1492	0	5302	40
2856120	40	Gap	92.8	0	0	0	0	10.87053	0	0	1490	0	5280	40
2856120	41	Rolling	79.4	4424720	22404520	196200	0.130444	10.87053	1.850752	1200.32	1490	0	5280	40
2856210	42	Gap	89.1	0	0	0	0	11.25475	0	0	1488	0	5280	40
2856210	43	Rolling	74.1	4482863	21341800	196200	0.130285	11.25475	1.848858	1197.219	1488	0	5280	40
2856310	44	Gap	90.7	0	0	0	0	10.81004	0	0	1483	0	5312	43
2856310	45	Rolling	77.8	4314858	20822050	196200	0.129874	10.81004	1.844719	1206.61	1483	0	5312	43
2856410	46	Gap	96.9	0	0	0	0	11.04098	0	0	1491	0	5302	40
2856410	47	Rolling	75.4	4434345	21174630	196200.3	0.129927	11.04098	1.846936	1201.451	1491	0	5302	40

CoilNumber	Stage	Comment	Time	Pstand	StripForce	BendingForce	bitelength	angularvelocity	stripwidth	Tstrip	exitcentreflowrate	exitedgeflowrate	entrycentreflowrate	entryedgeflowrate
2856510	48	Gap	97.9	0	0	0	0	11.05002	0	0	1498	0	5302	36
2856510	49	Rolling	81.9	4604075	21718240	196200.1	0.130816	11.05002	1.845688	1206.219	1498	0	5302	36
2856520	50	Gap	93.7	0	0	0	0	11.28709	0	0	1475	0	5235	38
2856520	51	Rolling	77.9	4926569	22297450	196199.9	0.131361	11.28709	1.845215	1199.081	1475	0	5235	38
2856530	52	Gap	111.9	0	0	0	0	10.94871	0	0	1488	0	5237	39
2856530	53	Rolling	82.1	4666729	21715420	196199.7	0.130657	10.94871	1.843268	1207.409	1488	0	5237	39
2856540	54	Gap	101.9	0	0	0	0	11.12053	0	0	1480	0	5255	40
2856540	55	Rolling	80.5	4801486	21980860	196199.8	0.129985	11.12053	1.845681	1197.939	1480	0	5255	40
2856550	56	Gap	101.4	0	0	0	0	10.84249	0	0	1493	0	5272	37
2856550	57	Rolling	82.7	4737192	21843890	196199.5	0.130714	10.84249	1.844562	1210.512	1493	0	5272	37
2856610	58	Gap	104.2	0	0	0	0	11.14905	0	0	1475	0	5285	39
2856610	59	Rolling	80.4	4820998	21959510	430590.8	0.131481	11.14905	1.845069	1197.718	1475	0	5285	39
2856620	60	Gap	99.5	0	0	0	0	10.86141	0	0	1495	0	5272	38
2856620	61	Rolling	83	4640203	21522530	630793.1	0.130095	10.86141	1.843531	1211.283	1495	0	5272	38
2856630	62	Gap	115.6	0	0	0	0	11.11418	0	0	1505	0	5257	37
2856630	63	Rolling	79.8	4846323	21933540	383105.6	0.130282	11.11418	1.845155	1199.452	1505	0	5257	37
2856710	64	Gap	126.1	0	0	0	0	10.83456	0	0	1465	0	5275	41
2856710	65	Rolling	82.5	4680787	21710550	652441.6	0.129455	10.83456	1.846811	1210.672	1465	0	5275	41
2856720	66	Gap	116.2	0	0	0	0	10.91658	0	0	1491	0	5265	39
2856720	67	Rolling	81.6	4763048	21791800	634260.5	0.129609	10.91658	1.84753	1204.736	1491	0	5265	39
2856730	68	Gap	123.2	0	0	0	0	10.94464	0	0	1495	0	5307	41
2856730	69	Rolling	82.1	4718872	21656220	911136.1	0.129392	10.94464	1.848125	1207.091	1495	0	5307	41
2856740	70	Gap	114.4	0	0	0	0	10.89671	0	0	1483	0	5302	38
2856740	71	Rolling	81.7	4707638	21688160	993789.5	0.129687	10.89671	1.847375	1205.844	1483	0	5302	38
2856750	72	Gap	109.8	0	0	0	0	10.90236	0	0	1498	0	5317	40
2856750	73	Rolling	83.7	4528768	21347670	1655378	0.129408	10.90236	1.846519	1207.517	1498	0	5317	40

CoilNumber	Stage	Comment	Time	Pstand	StripForce	BendingForce	bitlength	angularvelocity	stripwidth	Tstrip	exitcentreflowrate	exitedgeflowrate	entrycentreflowrate	entryedgeflowrate
2856810	74	Gap	114.2	0	0	0	0	10.90354	0	0	1475	0	5222	40
2856810	75	Rolling	81.2	4796082	21758480	1707829	0.129482	10.90354	1.847282	1205.207	1475	0	5222	40
2856820	76	Gap	117.2	0	0	0	0	10.84261	0	0	1477	0	5265	39
2856820	77	Rolling	82.6	4707981	21511070	1691856	0.129893	10.84261	1.847456	1206.969	1477	0	5265	39
2856910	78	Gap	116.4	0	0	0	0	10.85103	0	0	1488	0	5240	39
2856910	79	Rolling	83.2	4735144	21475810	1204158	0.129715	10.85103	1.849349	1207.12	1488	0	5240	39
2857010	80	Gap	117.5	0	0	0	0	10.82573	0	0	1500	0	5300	42
2857010	81	Rolling	81.6	4817330	21443800	328454.8	0.130231	10.82573	1.851141	1206.952	1500	0	5300	42
2857020	82	Gap	120.9	0	0	0	0	10.65488	0	0	1497	0	5307	37
2857020	83	Rolling	82.8	4693703	21653930	566119.1	0.130414	10.65488	1.850404	1208.321	1497	0	5307	37
2857030	84	Gap	118.9	0	0	0	0	10.79681	0	0	1490	0	5282	39
2857030	85	Rolling	83.3	4639879	21647760	1401588	0.129411	10.79681	1.851199	1211.675	1490	0	5282	39
2857040	86	Gap	109.6	0	0	0	0	10.84152	0	0	1483	0	5265	42
2857040	87	Rolling	82.6	4698484	21657440	1704321	0.12955	10.84152	1.849814	1211.593	1483	0	5265	42
2857050	88	Gap	108.4	0	0	0	0	10.74163	0	0	1490	0	5320	40
2857050	89	Rolling	85.3	4411227	21125030	1637567	0.129755	10.74163	1.850232	1213.502	1490	0	5320	40
2857110	90	Gap	101	0	0	0	0	10.87514	0	0	1503	0	5322	40
2857110	91	Rolling	83.4	4568037	21439430	1524940	0.129641	10.87514	1.850576	1211.108	1503	0	5322	40
2857120	92	Gap	98.5	0	0	0	0	10.7719	0	0	1486	0	5305	40
2857120	93	Rolling	84.5	4527876	21397730	1615687	0.129657	10.7719	1.850084	1213.672	1486	0	5305	40
2857130	94	Gap	95.6	0	0	0	0	10.78753	0	0	1505	0	5335	40
2857130	95	Rolling	83.4	4583918	21441820	1668723	0.130411	10.78753	1.849964	1210.396	1505	0	5335	40
2857140	96	Gap	97.7	0	0	0	0	10.75287	0	0	1485	0	5307	42
2857140	97	Rolling	85.6	4399403	21072090	1400558	0.130399	10.75287	1.85099	1213.845	1485	0	5307	42
2857150	98	Gap	92	0	0	0	0	10.98704	0	0	1481	0	5252	42
2857150	99	Rolling	82.9	4622669	21421770	1656823	0.130048	10.98704	1.850703	1209.906	1481	0	5252	42

CoilNumber	Stage	Comment	Time	Pstand	StripForce	BendingForce	bitelength	angularvelocity	stripwidth	Tstrip	exitcentreflowrate	exitedgeflowrate	entrycentreflowrate	entryedgeflowrate
2857210	100	Gap	87.5	0	0	0	0	10.73912	0	0	1486	0	5265	40
2857210	101	Rolling	83.5	4546689	21133000	1645195	0.129744	10.73912	1.846296	1218.93	1486	0	5265	40
2857310	102	Gap	84.9	0	0	0	0	10.88973	0	0	1483	0	5245	43
2857310	103	Rolling	88	4729344	22075560	1765800	0.130569	10.88973	1.850415	1207.693	1483	0	5245	43
2857320	104	Gap	83	0	0	0	0	10.88312	0	0	1488	0	5277	39
2857320	105	Rolling	89.3	4620257	21859410	1765800	0.129478	10.88312	1.851307	1209.436	1488	0	5277	39
2857330	106	Gap	76.5	0	0	0	0	11.0697	0	0	1488	0	5262	42
2857330	107	Rolling	85.7	4873200	22314270	1765800	0.129954	11.0697	1.851459	1205.565	1488	0	5262	42
2857410	108	Gap	81.6	0	0	0	0	11.06958	0	0	1488	0	5320	41
2857410	109	Rolling	87.7	4847566	22233850	1765799	0.129646	11.06958	1.850306	1206.188	1488	0	5320	41
2857420	110	Gap	82.8	0	0	0	0	11.19526	0	0	1490	0	5280	41
2857420	111	Rolling	85.5	4932735	22287830	1765801	0.128731	11.19526	1.850459	1205.043	1490	0	5280	41
2857510	112	Gap	81	0	0	0	0	13.28525	0	0	1498	0	5305	41
2857510	113	Rolling	77.4	5002502	18365330	1765799	0.131122	13.28525	1.788504	1230.623	1498	0	5305	41
2857520	114	Gap	88.3	0	0	0	0	12.68386	0	0	1468	0	5252	43
2857520	115	Rolling	79.3	5167947	18432190	1765799	0.129312	12.68386	1.790384	1220.469	1468	0	5252	43
2857610	116	Gap	82.4	0	0	0	0	12.88477	0	0	1483	0	5297	41
2857610	117	Rolling	72.4	4919793	18093290	1765800	0.127124	12.88477	1.75805	1214.65	1483	0	5297	41
2857710	118	Gap	69.5	0	0	0	0	7.829578	0	0	1485	0	5300	42
2857710	119	Rolling	84.4	2914602	17985750	1765800	0.129587	7.829578	1.620337	1205.814	1485	0	5300	42
2857810	120	Gap	59.3	0	0	0	0	9.556245	0	0	1470	0	5285	42
2857810	121	Rolling	83.5	3544581	17616990	1765800	0.133388	9.556245	1.544697	1212.846	1470	0	5285	42
2857910	122	Gap	48.8	0	0	0	0	10.06016	0	0	1497	0	5322	40
2857910	123	Rolling	77.2	3808927	18427100	1765800	0.13611	10.06016	1.544431	1201.72	1497	0	5322	40
2858010	124	Gap	63.1	0	0	0	0	9.940017	0	0	1495	0	5292	42
2858010	125	Rolling	81.3	3804322	18863840	1863899	0.139663	9.940017	1.511226	1213.005	1495	0	5292	42

CoilNumber	Stage	Comment	Time	Pstand	StripForce	BendingForce	bitlength	angularvelocity	stripwidth	Tstrip	exitcentreflowrate	exitedgeflowrate	entrycentreflowrate	entryedgeflowrate
2858110	126	Gap	60.6	0	0	0	0	10.1259	0	0	1496	0	5297	41
2858110	127	Rolling	99.5	3935264	18477690	1863901	0.147928	10.1259	1.398119	1183.925	1496	0	5297	41
2858120	128	Gap	38.8	0	0	0	0	10.19029	0	0	1493	0	5312	39
2858120	129	Rolling	96.9	3782797	17963620	1863901	0.145354	10.19029	1.394097	1196.738	1493	0	5312	39
2858210	130	Gap	36.8	0	0	0	0	10.37778	0	0	1476	0	5240	42
2858210	131	Rolling	87.9	4360024	19050960	1863900	0.14879	10.37778	1.341692	1182.095	1476	0	5240	42
2858310	132	Gap	38.1	0	0	0	0	10.03328	0	0	1471	0	5270	40
2858310	133	Rolling	103.3	3534348	16581130	1863899	0.147488	10.03328	1.293505	1198.467	1471	0	5270	40
2858320	134	Gap	29.8	0	0	0	0	10.36115	0	0	1492	0	5270	41
2858320	135	Rolling	97.6	3840560	17471580	1863900	0.148922	10.36115	1.292983	1178.242	1492	0	5270	41
2858410	136	Gap	25.9	0	0	0	0	12.65175	0	0	1477	0	5260	43
2858410	137	Rolling	78.7	3936406	16627600	787508.1	0.144436	12.65175	1.290729	1198.815	1477	0	5260	43
2858420	138	Gap	26.5	0	0	0	0	12.17833	0	0	1500	0	5282	40
2858420	139	Rolling	76.8	4166902	17371570	196200.6	0.145708	12.17833	1.294419	1182.839	1500	0	5282	40
2858510	140	Gap	40.5	0	0	0	0	12.69485	0	0	1480	0	5265	38
2858510	141	Rolling	84	3882056	16847910	196199.7	0.143144	12.69485	1.275163	1218.103	1480	0	5265	38
2858520	142	Gap	23.2	0	0	0	0	11.8116	0	0	1495	0	5282	41
2858520	143	Rolling	80.5	3878223	17924000	196200.2	0.144011	11.8116	1.276047	1183.356	1495	0	5282	41
2858530	144	Gap	219.6	0	0	0	0	13.36147	0	0	1472	0	5252	37
2858530	145	Rolling	79.8	3956410	16919910	196198.8	0.14218	13.36147	1.272651	1217.719	1472	0	5252	37
2858610	146	Gap	29.9	0	0	0	0	12.70682	0	0	1505	0	5302	39
2858610	147	Rolling	74.2	4186093	16992010	709254.1	0.140378	12.70682	1.249183	1204.346	1505	0	5302	39
2858710	148	Gap	43.6	0	0	0	0	12.55734	0	0	1477	0	5295	42
2858710	149	Rolling	88.6	3480583	15662790	196200.1	0.140617	12.55734	1.238839	1220.675	1477	0	5295	42
2858720	150	Gap	22	0	0	0	0	12.24522	0	0	1495	0	5267	41
2858720	151	Rolling	81.4	3596373	17041520	196200.6	0.143111	12.24522	1.239483	1196.505	1495	0	5267	41

CoilNumber	Stage	Comment	Time	Pstand	StripForce	BendingForce	bitelength	angularvelocity	stripwidth	Tstrip	exitcentreflowrate	exitedgeflowrate	entrycentreflowrate	entryedgeflowrate
2858810	152	Gap	23.7	0	0	0	0	12.74119	0	0	1488	0	5282	40
2858810	153	Rolling	88.6	3682455	16008960	196199.1	0.14155	12.74119	1.236879	1218.139	1488	0	5282	40
2858910	154	Gap	23.4	0	0	0	0	11.98235	0	0	1491	0	5265	36
2858910	155	Rolling	81	3885897	17793650	196200.7	0.151293	11.98235	1.237301	1193.142	1491	0	5265	36
2859010	156	Gap	32.4	0	0	0	0	12.98719	0	0	1482	0	5292	42
2859010	157	Rolling	88.3	3759928	16245100	196199.5	0.148614	12.98719	1.225476	1203.864	1482	0	5292	42
2859110	158	Gap	23.2	0	0	0	0	12.32345	0	0	1478	0	5267	41
2859110	159	Rolling	87.3	3708350	16410740	196199.2	0.145096	12.32345	1.18723	1203.443	1478	0	5267	41
2859210	160	Gap	28.5	0	0	0	0	12.24918	0	0	1486	0	5270	38
2859210	161	Rolling	72.9	3330102	14345180	196199.3	0.150192	12.24918	1.05675	1211.886	1486	0	5270	38
2859220	162	Gap	130.4	0	0	0	0	10.70244	0	0	1505	0	5307	37
2859220	163	Rolling	69.9	3601115	17090220	196199.8	0.151405	10.70244	1.057449	1179.876	1505	0	5307	37
2859230	164	Gap	35.7	0	0	0	0	11.3754	0	0	1498	0	5272	42
2859230	165	Rolling	72.6	3848849	17201290	196199.5	0.151242	11.3754	1.057271	1166.996	1498	0	5272	42
2859240	166	Gap	92.8	0	0	0	0	12.71686	0	0	1482	0	5310	43
2859240	167	Rolling	73.2	3422245	14565310	196199.3	0.146901	12.71686	1.057587	1201.322	1482	0	5310	43
2859250	168	Gap	34.4	0	0	0	0	13.16457	0	0	1486	0	5282	39
2859250	169	Rolling	91	3331474	13790980	196199.9	0.147201	13.16457	1.057958	1202.421	1486	0	5282	39
2859310	170	Gap	20.3	0	0	0	0	12.15892	0	0	1456	0	5225	43
2859310	171	Rolling	89	3270734	14037650	196199.7	0.147763	12.15892	1.057432	1200.612	1456	0	5225	43
2859320	172	Gap	19.9	0	0	0	0	12.82543	0	0	1468	0	5225	39
2859320	173	Rolling	89	3497702	14311580	196199.8	0.148372	12.82543	1.057183	1197.061	1468	0	5225	39
2859330	174	Gap	20.2	0	0	0	0	12.63413	0	0	1480	0	5257	41
2859330	175	Rolling	82.1	3572372	14688450	196200.4	0.149532	12.63413	1.054357	1204.017	1480	0	5257	41
2859340	176	Gap	20.7	0	0	0	0	12.7488	0	0	1480	0	5255	41
2859340	177	Rolling	89.8	3490665	14455770	196199.6	0.148172	12.7488	1.057999	1200.562	1480	0	5255	41

CoilNumber	Stage	Comment	Time	Pstand	StripForce	BendingForce	bitelength	angularvelocity	stripwidth	Tstrip	exitcentreflowrate	exitedgeflowrate	entrycentreflowrate	entryedgeflowrate
2859350	178	Gap	20.6	0	0	0	0	12.31302	0	0	1478	0	5227	38
2859350	179	Rolling	84.5	3393954	14572240	196199.9	0.14844	12.31302	1.057111	1196.777	1478	0	5227	38
2859410	180	Gap	19.6	0	0	0	0	13.17062	0	0	1475	0	5290	39
2859410	181	Rolling	84.7	3508425	14167310	196200.5	0.146032	13.17062	1.056802	1200.811	1475	0	5290	39
2859420	182	Gap	20.2	0	0	0	0	12.44722	0	0	1483	0	5315	41
2859420	183	Rolling	89.3	3432835	14500930	196199.4	0.145974	12.44722	1.056998	1197.894	1483	0	5315	41
2859510	184	Gap	20	0	0	0	0	13.00041	0	0	1491	0	5282	42
2859510	185	Rolling	84.3	3021865	12698400	196200.4	0.145446	13.00041	0.992547	1210.416	1491	0	5282	42
2859520	186	Gap	20.1	0	0	0	0	12.8063	0	0	1456	0	5237	41
2859520	187	Rolling	81.2	3022867	12854960	196199.3	0.142907	12.8063	0.991857	1209.062	1456	0	5237	41
2859530	188	Gap	27.5	0	0	0	0	13.28322	0	0	1493	0	5257	40
2859530	189	Rolling	84.5	3103211	12615080	196200	0.143149	13.28322	0.992262	1212.732	1493	0	5257	40
2859610	190	Gap	118.5	0	0	0	0	12.11421	0	0	1486	0	5232	40
2859610	191	Rolling	69.9	2968876	11827410	196199.2	0.145286	12.11421	0.946211	1196.039	1486	0	5232	40
2859630	192	Mill problem	1682.5	0	0	0	0	0	0	0	-1	-1	-1	-1
2859630	193	Rolling	91.3	2914590	11835660	196200.6	0.144394	12.21618	0.946588	1198.916	1475	0	5255	38
2859620	194	Gap	82.7	0	0	0	0	12.50621	0	0	1493	0	5322	40
2859620	195	Rolling	91.4	2939231	11844010	196200.5	0.145487	12.50621	0.945735	1204.166	1493	0	5322	40
2859640	196	Gap	39.7	0	0	0	0	12.572	0	0	1485	0	5257	39
2859640	197	Rolling	89.8	2974431	11729980	196199.7	0.144667	12.572	0.946708	1204.738	1485	0	5257	39
2859650	198	Gap	21.9	0	0	0	0	12.07162	0	0	1490	0	5290	40
2859650	199	Rolling	89.6	2968380	11857670	196200.5	0.149461	12.07162	0.946666	1194.174	1490	0	5290	40
2859710	200	Gap	19.1	0	0	0	0	11.85215	0	0	1470	0	5270	39
2859710	201	Rolling	86.2	2679034	11546040	196200.5	0.146337	11.85215	0.946658	1193.902	1470	0	5270	39
2859720	202	Gap	20.1	0	0	0	0	11.77068	0	0	1491	0	5252	38
2859720	203	Rolling	90.3	2887968	12180360	196200.3	0.149085	11.77068	0.946494	1186.691	1491	0	5252	38



CoilNumber	Stage	Comment	Time	Pstand	StripForce	BendingForce	bitelength	angularvelocity	stripwidth	Tstrip	exitcentreflowrate	exitedgeflowrate	entrycentreflowrate	entryedgeflowrate
2859730	204	Gap	19.9	0	0	0	0	12.07045	0	0	1498	0	5282	39
2859730	205	Rolling	90.3	2850620	12002870	196198.9	0.147072	12.07045	0.946439	1198.13	1498	0	5282	39
2859740	206	Gap	20.5	0	0	0	0	11.44062	0	0	1492	0	5305	41
2859740	207	Rolling	90.6	2785829	12187440	196200	0.150439	11.44062	0.946712	1184.704	1492	0	5305	41
2859750	208	Gap	19.4	0	0	0	0	12.37404	0	0	1468	0	5270	41
2859750	209	Rolling	90.3	2895094	12403700	196200.2	0.147722	12.37404	0.946226	1190.584	1468	0	5270	41
2859810	210	Gap	20.2	0	0	0	0	11.67812	0	0	1466	0	5227	41
2859810	211	Rolling	88	2771739	12338930	196200.1	0.150721	11.67812	0.934282	1208.354	1466	0	5227	41
2859820	212	Gap	19.3	0	0	0	0	11.93876	0	0	1480	0	5250	38
2859820	213	Rolling	93.1	2741301	11814610	196199.8	0.149462	11.93876	0.933299	1210.942	1480	0	5250	38
2859830	214	Gap	19.9	0	0	0	0	11.76235	0	0	1461	0	5235	41
2859830	215	Rolling	88.3	2813931	12116320	196199.8	0.149798	11.76235	0.933915	1206.934	1461	0	5235	41
2859840	216	Gap	19.4	0	0	0	0	11.97757	0	0	1461	0	5252	39
2859840	217	Rolling	91.5	2718466	11682910	196199	0.149031	11.97757	0.932865	1221.307	1461	0	5252	39
2859910	218	Gap	21.8	0	0	0	0	11.29204	0	0	1491	0	5305	39
2859910	219	Rolling	106.6	2529677	11636970	196199.3	0.148661	11.29204	0.844176	1195.174	1491	0	5305	39
2859910	220	Post Round Gap	30	0	0	0	0	11.29204	0	0	1491	0	5305	39
0	221	Cooling to air time	2700	0	0	0	6.283185	0	2	20	0	0	0	0

### D.3.3. Replacement row for D.3.2 to give "Long Gap" condition

CoilNumber	Stage	Comment	Time	Pstand	StripForce	BendingForce	bitelength	angularvelocity	stripwidth	Tstrip	exitcentreflowrate	exitedgeflowrate	entrycentreflowrate	entryedgeflowrate
2859530	188	Gap	1200	0	0	0	0	13.28321901	0	0	1493	0	5257	40

#### D.3.4. Replacement row for D.3.2 to give "Mill Stop" condition

CoilNumber	Stage	Comment	Time	Pstand	StripForce	BendingForce	bitelength	angularvelocity	stripwidth	Tstrip	exitcentreflowrate	exitedgeflowrate	entrycentreflowrate	entryedgeflowrate
2859520	186	Mill problem	1200	0	0	0	0	0	0	0	-1	-1	-1	-1

## References

---

- <sup>1</sup> Corus, *Hot Rolling* [Presentation]
- <sup>2</sup> URL  
<[http://cspukonline.corusnet.corusgroup.com/company/cspuktoolkit.nsf/3ad83766708eb08080256ac40045ba60/245b936855bf565a80256ea90049df7a/\\$FILE/Hot Rolling.ppt](http://cspukonline.corusnet.corusgroup.com/company/cspuktoolkit.nsf/3ad83766708eb08080256ac40045ba60/245b936855bf565a80256ea90049df7a/$FILE/Hot%20Rolling.ppt)> accessed 13/07/2008
- <sup>3</sup> D. White, A hot strip mill work roll temperature model, Eng D Thesis submitted to Cardiff University, 2008
- <sup>4</sup> Personal communication, Harry Van Steden, Ijmuiden RD&T
- <sup>5</sup> R. E. Johnson, R. G. Keanini, An asymptotic model of work roll heat transfer in strip rolling, *International Journal of Heat and Mass Transfer* Vol. 41, Nos 6-7, 1995, pp. 871-879
- <sup>6</sup> C. F. Peck, J. M. Bonetti, F. T. Mavis, Temperature stresses in iron work rolls, *Iron Steel Engineering Year Book*, 1954, pp 389
- <sup>7</sup> R. Colás, Modelling heat transfer during hot rolling of steel strip, *Modelling simul. mater. sci. eng.* 3 (1995) 437-453
- <sup>8</sup> A.A.Tseng, S.Tong, F.H.Lin, Thermal stresses of rotating rolls in rolling processing, *Journal of Thermal Stresses* 12 (1989), 427-450
- <sup>9</sup> A.Ch.Yiannopoulos, N.K.Anifantis, A.D.Dimarogonas, Thermal stress optimization in metal rolling, *Journal of Thermal Stresses*, 20 (1997), 569-590
- <sup>10</sup> D. F. Chang, Thermal stresses in work rolls during the rolling of metal strip, *Journal of materials processing technology* 94 (1999), 45-51
- <sup>11</sup> A.M.Campos et al, A real time model for the intelligent control of cooling systems for hot strip mill work rolls, 26th Annual Conference of the IEEE, 2000
- <sup>12</sup> D. J. Lee, M. T. Manzari, Y-L. Shen, W. Zeng, A finite element approach to transient thermal analysis of work rolls in rolling process, *Journal of Manufacturing Science and Engineering* 122 (2000), 706-716
- <sup>13</sup> X. M. Zhang, Z. Y. Yiang, A. K. Tieu, X. H. Liu, G. D. Wang, Numerical modelling of the thermal deformation of CVC work rolls in hot strip rolling, *Journal of Materials Processing Technology* 130-131 (2002), 219-223
- <sup>14</sup> S. Serajzadeh, A. K. Taheri, F. Mucciardi, Unsteady state work-roll temperature distribution during continuous hot slab rolling, *International Journal of Mechanical Sciences* 44 (2002) 2447-2462
- <sup>15</sup> C. O. Hadly, J. K. Brimacombe, I. V. Samarasekara, E. B. Hawbolt, Heat transfer in the hot rolling of metal. *Metallurgical and Materials Transactions* 209 (1995) 1-11
- <sup>16</sup> S. M. Hwang, C. G. Sun, S. R. Ryoo, W. J. Kwak, An integrated FE process model for prevision analysis of thermo-mechanical behaviours of rolls and strip in hot strip rolling, *Computer methods in applied mechanics and engineering* 191 (2002) 4015-4033
- <sup>17</sup> R. L. Corral, R. Colás, A. Pérez, Modeling the thermal and thermoelastic responses of work rolls used for hot rolling steel strip, *Journal of Materials Processing Technology* 153-154 (2004) 886-893
- <sup>18</sup> L. Chang-sheng, Y. Hai-liang, D. Guan-yu, L. Xiang-hua, W. Guo-dong, Numerical simulation of temperature field and thermal stress field of work roll during hot strip rolling, *International journal of iron and steel research*, 2007, 14(5): 18-21
- <sup>19</sup> X. Wang, Q. Yang, A. He, R. Wang, Comprehensive contour prediction model of work rolls in hot wide strip mill, *Journal of University of Science and Technology Beijing* 14 (2007) 240-245
- <sup>20</sup> R. D. Mercado-Solis, J. Talamantes-Silva, J. H. Beynon, M. A. L. Hernandez-Rodriguez, Modelling surface thermal damage to hot mill rolls, *Wear* 263 (2007) 1560-1567
- <sup>21</sup> M. Abbaspour, A. Saboonchi, Work roll thermal expansion control in hot strip mill, *Applied Mathematical Modelling* (article in press)
- <sup>22</sup> D. Benasciutti, E. Brusa, G. Bazzaro, Finite elements prediction of thermal stresses in work roll of hot rolling mills, *Procedia Engineering* 2 (2010) 707-716
- <sup>23</sup> E. J. Patula, Steady-state temperature distribution in a rotating roll subject to surface heat fluxes and convective cooling, *Journal of Heat Transfer* 103 (1981) 36-41
- <sup>24</sup> P. J. Lee, Control of shape and profile in hot rolled steel strip, Engineering Doctorate Thesis, Corus Port Talbot
- <sup>25</sup> D. R. Pernas, Hot rolling mill thermal analysis, Division of Mechanical Engineering and Energy Studies, Cardiff University School of Engineering, 1998

---

<sup>26</sup> A. Clarke, K. J. Sharif, H. P. Evans, R. W. Snidle, Heat Partition in Rolling/Sliding Elastohydrodynamic Contacts, ASME Conf. Proc. 2004, 479 (2004), DOI:10.1115/TRIB2004-64250

<sup>27</sup> Personal communication; Dr Christian Onisa, Swinden Technology Centre.

<sup>28</sup> F. B. Hildebrand, 'Finite difference equations and simulations', Prentice-Hall, 1968, p200

<sup>29</sup> Personal communication – Jan De Roo, IJmuiden RD&T

<sup>30</sup> URL <[http://highered.mcgraw-hill.com/sites/dl/free/0073129305/314124/cen29305\\_ch04.pdf](http://highered.mcgraw-hill.com/sites/dl/free/0073129305/314124/cen29305_ch04.pdf)>  
accessed 15/12/2011

<sup>31</sup> Personal communication, Ron Wigfield, Port Talbot steelworks

<sup>32</sup> F. Hosseinzadeh, D.J. Smith, C.E. Truman, Through thickness residual stresses in large rolls and sleeves for metal working industry. Materials Science and Technology, 25(7), (2009), 862–873.

<sup>33</sup> J.M.Gere, W.Weaver Jr., Matrix Algebra for Engineers, D. Van Nostrand Company INC, Princeton, New Jersey, October 1965

<sup>34</sup> Personal communication – Terry O'Calaghan, Port Talbot steelworks

<sup>35</sup> Personal communication – Jan De Roo, IJmuiden RD&T

<sup>36</sup> The white model

<sup>37</sup> Abaqus Theory Manual, (6.10) Section 3.2.4

<sup>38</sup> <URL: [http://www.mscsoftware.com/events/vpd2006/na/presentations/tech\\_papers/27.pdf](http://www.mscsoftware.com/events/vpd2006/na/presentations/tech_papers/27.pdf)>

accessed 08/09/2010, Shear Locking and Hourglassing in MSC Nastran, ABAQUS and ANSYS, Eric Qiuli

Sun