MIXED CONVECTION IN VERTICAL PARALLEL CHANNELS CONNECTED

AT UPPER AND LOWER PLENA

Vol. 1

by

VICTOR IANNELLO

S.M., S.B., Massachusetts Institute of Technology
(1984)

SUBMITTED TO THE DEPARTMENT OF
NUCLEAR ENGINEERING
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

DOCTOR OF SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
February 1986

© Massachusetts Institute of Technology, 1986

Signature of Author

Department of Nuclear Engineering
February 18, 1986

Certified by

Neil E. Todreas
Thesis Supervisor

Accepted by

Allan F. Henry
Chairman, Departmental Committee on Graduate Students

AUG 20 1986

ARCHIVES
MIXED CONVECTION IN VERTICAL PARALLEL CHANNELS CONNECTED
AT UPPER AND LOWER PLENA

by

Victor Iannello

Submitted to the Department of Nuclear Engineering on the 18th of
February, 1986, in partial fulfillment of the requirements for the
Degree of Doctor of Science in Nuclear Engineering

ABSTRACT

Analytical, numerical, and experimental studies were made
regarding the effect of buoyancy on the thermal hydraulic behavior of
a parallel channel system. For such conditions, the velocity and
temperature fields are distorted, rendering forced convection analyses
inaccurate. The buoyancy will cause a redistribution of flow between
and within channels; multi-dimensional recirculating flows may also
develop.

A one-dimensional time independent model was developed to predict
the buoyancy-induced flow redistribution between channels and was
incorporated into a small code. The code was used to predict the
behavior of a Liquid Metal Fast Breeder Reactor (LMFBR) core during
natural circulation of the primary sodium loop. The analysis predicts
that the temperature rise in the hot channel is reduced by 31% due to
the flow redistribution.

Mixed convection friction factors and Nusselt numbers are
presented for use in lumped parameter codes. A fully-developed flow
analysis was applied to geometries with azimuthal symmetry, and the
ENERGY-IV subchannel code was used to successfully predict the
friction factor data for wire-wrapped rod bundles.

A time-dependent one-dimensional code was used to simulate a
parallel channel test. The prediction was adequate for channels for
which the flow was one-dimensional. The onset of a recirculating flow
at the exit of a heated channel was related to the channel exit-upper
plenum temperature difference and channel flowrate using an annular
channel test section. In addition, a qualitative study was made of
the manner in which a heated channel reverses from upflow to downflow
using injected dye to visualize the flow field.

Thesis Supervisor: Dr. Neil E. Todreas
Title: Head, Department of Nuclear Engineering

Thesis Reader: Dr. Muqted S. Kazimi
Title: Professor of Nuclear Engineering
ACKNOWLEDGMENTS

The author wishes to thank Profs. M. Kazimi and A. Schor for their helpful suggestions made during the course of this work. Prof. Kazimi also served as the thesis reader. Thanks are also extended to Drs. P. Magee and A. Hunsbedt of General Electric for the transmission of valuable technical data.

The test section design and construction could not have been accomplished without the expert advice, as well as machining, electrical, and plumbing work provided by Mr. Joseph (Tiny) Caloggero of the Mechanical Engineering Experimental Projects Laboratory. Mr. John DiFrancesco, Mr. Fred Jonson, and Mr. Donald Wassmouth also assisted in providing technical assistance for the maintenance of the test section and loop.

Special thanks go to Mr. Paul Ulbrich, who spent many hours expertly typing this manuscript.

Other students who have provided valuable technical critiques of this work include Tsing-Tung Huang, Kune-Yull Suh, Tae-Sun Ro, and Geun-Sun Ueo.

Professor N. Todreas, the thesis supervisor, provided unlimited technical support and encouragement in all phases of this work. The author greatly appreciates and respects the concern that he has for his students.

The financial and academic support of General Electric is gratefully acknowledged.
DEDICATION

To Chris, who through her constant love, respect, and confidence in me, gave me the encouragement that I needed to complete this work.
TABLE OF CONTENTS

Abstract................................................................. 2
Acknowledgments....................................................... 3
List of Figures.......................................................... 12
List of Tables............................................................ 18

Chapter One - Introduction............................. 20

Chapter Two - Steady One-Dimensional Mixed Convection in Parallel Channels
2.1 Introduction ....................................................... 30
2.2 Formulation of the Conservation Equations............... 31
  2.2.1 Momentum Equation ........................................... 31
  2.2.2 Energy Equation ............................................. 36
  2.2.3 Dimensionless Momentum Energy Equation............. 36
  2.2.4 Heat Flux Shape Factor S .................................. 37
  2.2.5 Dimensionless Continuity Equation ..................... 40
2.3 One-Dimensional Mixed Convection Channel Performance ... 41
  2.3.1 Onset of Mixed Convection ............................... 41
  2.3.2 Adiabatic Channel Reversal from Upflow to Downflow.. 42
  2.3.3 Stability of Cooled Upflow ............................. 43
  2.3.4 Stability of Heated Downflow ........................... 44
  2.3.5 Preference for Upflow ................................... 47
2.4 Solution of the Coupled Momentum-Energy Equations .... 49
  2.4.1 General Solution Procedure ............................. 49
  2.4.2 The Computer Code CHANSOL ............................ 53
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5</td>
<td>Application of CHANSOL to Model the General Electric PRISM Reactor Core During Natural Circulation of the Primary Sodium Loop</td>
<td>55</td>
</tr>
<tr>
<td>2.5.1</td>
<td>PRISM Description</td>
<td>55</td>
</tr>
<tr>
<td>2.5.2</td>
<td>Primary Sodium Loop Natural Circulation Modeling</td>
<td>63</td>
</tr>
<tr>
<td>2.5.3</td>
<td>Wire-Wrapped Bundle Friction Factors</td>
<td>66</td>
</tr>
<tr>
<td>2.5.4</td>
<td>Local Form Loss Factor Due to Orificing</td>
<td>68</td>
</tr>
<tr>
<td>2.5.5</td>
<td>Core Decay Power</td>
<td>68</td>
</tr>
<tr>
<td>2.5.6</td>
<td>Flow Distribution Results</td>
<td>71</td>
</tr>
<tr>
<td>2.5.7</td>
<td>Calculation of the Upflow Preference for the PRISM Assemblies</td>
<td>76</td>
</tr>
</tbody>
</table>

Chapter Three - Laminar Mixed Convection Flow in a Duct

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Introduction</td>
<td>80</td>
</tr>
<tr>
<td>3.2</td>
<td>Non-Dimensional Equations and Parameters for Developing Laminar Mixed Convection Flow in a Heated Vertical Duct</td>
<td>83</td>
</tr>
<tr>
<td>3.3</td>
<td>Non-Dimensional Equations and Parameters for Fully-Developed Laminar Mixed Convection Flow in a Heated Vertical Duct</td>
<td>86</td>
</tr>
<tr>
<td>3.3.1</td>
<td>Velocity Profile</td>
<td>86</td>
</tr>
<tr>
<td>3.3.2</td>
<td>Friction Factor</td>
<td>90</td>
</tr>
<tr>
<td>3.3.3</td>
<td>Temperature Field</td>
<td>93</td>
</tr>
<tr>
<td>3.3.4</td>
<td>Modified Friction Factor</td>
<td>95</td>
</tr>
<tr>
<td>3.3.5</td>
<td>Nusselt Number</td>
<td>96</td>
</tr>
<tr>
<td>3.4</td>
<td>Solution for Fully-Developed Laminar Mixed Convection Flow in a Heated Vertical Duct for Geometries Exhibiting Azimuthal Symmetry</td>
<td>97</td>
</tr>
<tr>
<td>3.4.1</td>
<td>Velocity Profile</td>
<td>97</td>
</tr>
<tr>
<td>3.4.2</td>
<td>Friction Factor</td>
<td>110</td>
</tr>
<tr>
<td>3.4.3</td>
<td>Modified Friction Factor</td>
<td>119</td>
</tr>
<tr>
<td>3.4.4</td>
<td>Nusselt Number</td>
<td>132</td>
</tr>
</tbody>
</table>
Table of Contents (continued)

3.5 Comparison of Equivalent Annulus and Infinite Rod Array Analytical Results
..................................................139

3.6 Comparison of Analytical Results with Experimental Data..151
3.6.1 Friction Factor..............................................151
3.6.2 Nusselt Number..............................................160

3.7 Developing Flow Mixed Convection Pressure Drop in Finite Rod Bundles.................................164
3.7.1 Experimental Data.............................................164
3.7.2 Description of Flow Redistribution Effects Within the Bundle........................................166
3.7.3 ENERGY-IV Code..............................................168
3.7.4 ENERGY-IV Code Prediction.................................170

Chapter Four - Transient One-Dimensional Mixed Convection
in Parallel Channels

4.1 Introduction....................................................171

4.2 One Dimensional Conservation Equations and Numerical Methods..............................................172
4.2.1 One Dimensional Continuity Equation...............172
4.2.2 One Dimensional Momentum Equation .................173
  4.2.2.1 Spatial Averaging........................................173
  4.2.2.2 Axial Integration (Momentum Integral).....174
  4.2.2.3 Temporal Discretization..............................175
  4.2.2.4 Parallel Channel Equations.........................177
  4.2.2.5 Solution Procedure....................................178
4.2.3 One Dimensional Energy Equation......................179
  4.2.3.1 Spatial Averaging......................................179
  4.2.3.2 Spatial and Temporal Discretization..............180
4.2.4 Plena Energy Balance.................................181
4.2.5 Coupled Conservation Equation Solution Procedure..182
4.3 The Code MICON ........................................... 183
4.4 Numerical Prediction of the Shutdown Core Coolability
Model Test ....................................................... 184
  4.4.1 Description of Experimental Apparatus ............... 184
  4.4.2 Description of Experimental Conditions ............... 188
  4.4.3 Experimental Results .................................. 190
  4.4.4 MICON Model ........................................... 195
    4.4.4.1 Noding Scheme ....................................... 195
    4.4.4.2 Forced Convection Flow Resistance
             Modeling ............................................... 198
    4.4.4.3 Mixed Convection Friction Factor
             Modeling ............................................... 202
  4.4.5 MICON Results and Discussion ......................... 204
    4.4.5.1 Steady State Results ............................... 204
    4.4.5.2 Transient Results ................................... 207

Chapter Five - The Experimental Apparatus

5.1 Introduction ............................................... 217
5.2 Test Section Design and Construction ...................... 217
  5.2.1 The Channels .......................................... 217
  5.2.2 The Heated Rods ....................................... 224
  5.2.3 The Upper and Lower Plena ............................ 226
5.3 The Flow Loop, Instrumentation, and Experimental
    Operation .................................................... 229
  5.3.1 Flow Loop .............................................. 229
  5.3.2 Flowmeters ............................................. 229
  5.3.3 Thermocouples ......................................... 229
  5.3.4 Heated Rod Electrical Power ............................ 232
  5.3.5 Heat Loss Measurement .................................. 235
  5.3.6 Experimental Operation ................................ 236
<table>
<thead>
<tr>
<th>Chapter Six - Channel Backflow from the Upper Plenum into the Channel (Channel-Plenum Interaction)</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1 Introduction.................................................................................................................. 237</td>
</tr>
<tr>
<td>6.2 Description of the MIT Onset of Channel Backflow Experiments........................................ 239</td>
</tr>
<tr>
<td>6.2.1 Instrumentation........................................................................................................ 239</td>
</tr>
<tr>
<td>6.2.2 Dye Injection System................................................................................................. 240</td>
</tr>
<tr>
<td>6.2.3 Experimental Procedure and Qualitative Results....................................................... 240</td>
</tr>
<tr>
<td>6.3 Onset of Backflow Model Development............................................................................. 242</td>
</tr>
<tr>
<td>6.4 Experimental Results and Comparison with Model.......................................................... 249</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter Seven - Buoyancy-Induced Flow Redistribution in Two Parallel Channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1 Introduction........................................................................................................... 252</td>
</tr>
<tr>
<td>7.2 Instrumentation........................................................................................................ 252</td>
</tr>
<tr>
<td>7.3 Procedure to Calculate the Flow Redistribution and Channel Power.............. 253</td>
</tr>
<tr>
<td>7.4 Error Analysis.......................................................................................................... 254</td>
</tr>
<tr>
<td>7.5 Results and Discussion............................................................................................ 259</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 8 - A Visual Study of the Reversal of a Heated Channel from Upflow to Downflow</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.1 Introduction.............................................................................................................. 269</td>
</tr>
<tr>
<td>8.2 Experimental Procedures.......................................................................................... 270</td>
</tr>
<tr>
<td>8.3 Results and Discussion............................................................................................ 272</td>
</tr>
</tbody>
</table>
Table of Contents (continued)

Chapter 9 - Conclusions and Recommendations for Future Work

9.1 Introduction.........................................................276
9.2 Conclusions..........................................................276
  9.2.1 Steady, One-Dimensional Parallel Channel Flow...... 276
  9.2.2 Mixed Convection Flow in a Duct.........................277
  9.2.3 Transient One-Dimensional Mixed Convection Flow....281
  9.2.4 Multi-Dimensional Channel-Plenum Interaction......282
  9.2.5 Experimentally Observed Buoyancy-Induced Flow
       Redistribution.....................................................
  9.2.6 Qualitative Observation of a Heated Channel Flow
       Reversal From Upflow to Downflow............................284

9.3 Recommendations for Future Work..........................284

References...............................................................287

Appendix A - CHANSOL Input Description and Listing..........293

Appendix B - ANNO and ANNA Codes..............................304

Appendix C - FIT Code.................................................317

Appendix D - MICON Code
  D.1 Introduction....................................................323
  D.2 Mathematical Model for the Fluid..........................324
    D.2.1 Fluid Momentum-Integral Equation.....................324
    D.2.1.1 Formulation............................................324
    D.2.1.2 Integration Methods..................................328
    D.2.2 Fluid Energy Equation Formulations....................329
    D.2.3 Ambient (Environmental) Temperature Calculation....334
    D.2.4 Plena Temperature Calculation..........................334
Table of Contents (continued)

D.3 Convective Heat Transfer Constitutive Equations............... 335
D.4 Equation of State and Liquid Thermophysical Properties.... 336
  D.4.1 Introduction............................................. 336
  D.4.2 Polynomial Fits.......................................... 338
D.5 MICON Code Description....................................... 344
D.6 Input Description............................................. 348
D.7 Code Listing.................................................. 357
LIST OF FIGURES

1.1 Comparison of natural and mixed convection..................21
1.2 Mixed convection in parallel channels connected at plena.................................23
1.3 Important core phenomena associated with natural circulation heat removal in an LMFBR.................................25
2.1 Arrangement of parallel channels connected only at upper and lower plena.................................32
2.2 Drawing of a channel with upper and lower unheated lengths.................................39
2.3 Cooled channel flow reversal from upflow to downflow........45
2.4 Heated channel flow reversal from downflow to upflow........46
2.5 General solution procedure for solving the coupled equations for parallel channels connected only at plena...50
2.6 CHANSOL subroutine flowchart.................................54
2.7 Qualitative drawing of the PRISM reactor primary loop....56
2.8 PRISM reactor core showing assembly type distribution and a 30° sector.................................58
2.9 Dimensions of the fuel pins for the PRISM reactor core...61
2.10 PRISM primary loop friction pressure drop versus flowrate.................................65
2.11 Dimensionless temperature rise for the individual assemblies calculated by CHANSOL for the PRISM reactor during natural circulation of the primary loop........73
2.12 Mixed convection flow redistribution for the individual assemblies calculated by CHANSOL for the PRISM reactor during natural circulation of the primary loop........75
2.13 Dimensionless pressure drop versus flowrate curves for the PRISM core channels with highest and lowest upflow preferences.................................79
3.1 Definition of the normal derivative........................91
3.2 Geometries with azimuthal symmetry........................99
3.3 Velocity profile for the inner radius heated annulus for aiding flow.................................102
List of Figures (continued)

3.4 Velocity profile for the outer radius heated annulus for aiding flow .................................................. 103
3.5 Velocity profile for the heated circular tube for aiding flow ................................................................. 104
3.6 Velocity profile for the equivalent annulus for aiding flow ................................................................. 105
3.7 Velocity profile for the inner radius heated annulus for opposing flow .................................................. 106
3.8 Velocity profile for the outer radius heated annulus for opposing flow .................................................. 107
3.9 Velocity profile for the heated circular tube for opposing flow ............................................................... 108
3.10 Velocity profile for the equivalent annulus for opposing flow .............................................................. 109
3.11 Mixed convection friction factor ratio for the inner radius heated annulus for aiding flow ..................... 113
3.12 Mixed convection friction factor ratio for the outer radius heated annulus for aiding flow .................... 114
3.13 Mixed convection friction factor ratio for the equivalent annulus for aiding flow ................................. 115
3.14 Mixed convection friction factor ratio for the inner radius heated annulus for opposing flow ................. 116
3.15 Mixed convection friction factor ratio for the outer radius heated annulus for opposing flow ................. 117
3.16 Mixed convection friction factor ratio for the equivalent annulus for opposing flow ............................ 118
3.17 Comparison of modified and conventional friction factor ratios for the inner radius heated annulus for aiding flow ............................................................................................................. 126
3.18 Comparison of modified and conventional friction factor ratios for the outer radius heated annulus for aiding flow ............................................................................................................. 127
3.19 Comparison of modified and conventional friction factor ratios for the equivalent annulus for aiding flow .......................... 128
3.20 Comparison of modified and conventional friction factor ratios for the inner radius heated annulus for opposing flow.............................. 129

3.21 Comparison of modified and conventional friction factor ratios for the outer radius heated annulus for opposing flow.............................. 130

3.22 Comparison of modified and conventional friction factor ratios for the equivalent annulus for opposing flow ..... 131

3.23 Mixed convection Nusselt number for the inner radius heated annulus for aiding flow................................. 140

3.24 Mixed convection Nusselt number for the outer radius heated annulus for aiding flow................................. 141

3.25 Mixed convection Nusselt number for the equivalent annulus for aiding flow ......................................142

3.26 Mixed convection Nusselt number for the inner radius heated annulus for opposing flow................................. 143

3.27 Mixed convection Nusselt number for the outer radius heated annulus for opposing flow................................. 144

3.28 Mixed convection Nusselt number for the equivalent annulus for opposing flow ......................................145

3.29 Angular dependence of the normalized shear stress for exterior cells in a (a) square (b) triangular array for forced convection................................. 148

3.30 Comparison of interior cell and equivalent annulus approaches................................................................. 149

3.31 Angular dependence of the normalized shear stress for interior cells in a triangular array for aiding mixed convection flow ......................................150

3.32 Comparison of the mixed convection friction factor ratio for interior channels in a triangular array with the equivalent annulus prediction (aiding flow)............................152

3.33 Comparison of the mixed convection friction factor ratio for interior cells in a square array with the equivalent annulus prediction (aiding flow)............................ 153
3.34 Comparison of the mixed convection friction factor ratio for interior channels in a triangular array with the equivalent annulus prediction (opposing flow)........... 154

3.35 Comparison of the mixed convection friction factor ratio for interior channels in a square array with the equivalent annulus prediction (opposing flow)........... 155

3.36 Experimental data compared with the analytical prediction for the mixed convection friction factor ratio for the heated circular tube for aiding flow................. 158

3.37 Experimental data compared with the analytical prediction for the mixed convection friction factor ratio for the MIT 4×4 bare rod bundle........................................ 159

3.38 Experimental data compared with the analytical prediction for the mixed convection Nusselt number for the heated circular tube for aiding flow............................ 161

3.39 Experimental data compared with the analytical prediction for the mixed convection Nusselt number for the heated circular tube for opposing flow........................ 162

3.40 Experimental data compared with the analytical prediction for the mixed convection Nusselt number for the inner radius heated annulus for aiding flow................. 163

3.41 Experimental data compared with the analytical prediction for the mixed convection Nusselt number for the MIT 19-pin wire-wrapped bundle................................. 165

3.42 ENERGY-IV prediction of the mixed convection friction factor ratio for 19- and 91-pin wire-wrapped rod bundles..167

4.1 The GE Shutdown Core Coolability Model Test (SCCMT) experiment with dimensions.......................... 186

4.2 Location of the channels for the SCCMT................. 187

4.3 The SCCMT loop diagram .................................. 189

4.4 Main loop coastdown curve for the SCCMT............... 192

4.5 Experimental flow in the F1, F2, IB, and bypass channels for the SCCMT................................. 193

4.6 Experimental flows in the RB1, RB2, and RS channels for the SCCMT..................................... 194
## List of Figures (continued)

4.7 Experimental peak coolant temperatures of the heated channels for the SCCMT ........................................... 196

4.8 MICON noding scheme for the SCCMT ................................. 197

4.9 Total pressure drop across each channel versus volumetric flow for the SCCMT ........................................... 200

4.10 Friction factor ratio data for different L/D_e's .......................... 205

4.11 Predicted and experimental flow in the main loop for the SCCMT ........................................... 208

4.12 Predicted and experimental flow in the F1 channel for the SCCMT ........................................... 209

4.13 Predicted and experimental flow in the F2 channel for the SCCMT ........................................... 210

4.14 Predicted and experimental flow in the IB channel for the SCCMT ........................................... 211

4.15 Predicted and experimental peak fluid temperatures in the F1 channel for the SCCMT ........................................... 213

4.16 Predicted and experimental peak fluid temperatures in the F2 channel for the SCCMT ........................................... 214

4.17 Predicted and experimental peak fluid temperatures in the IB channel for the SCCMT ........................................... 215

5.1 Schematic drawing of the parallel channel test section ............... 218

5.2 Screw positioners for the annular channels .............................. 219

5.3 Mechanism to block individual channels for the parallel channel test section ........................................... 224

5.4 Heated rods used for the parallel channel test section ............... 225

5.5 Upper and lower plena for the parallel channel test section ........................................... 227

5.6 Channel penetration into the plena for the parallel channel test section ........................................... 228

5.7 Flow loop for the parallel channel test section ......................... 230

5.8 Location of the thermocouples for the parallel channel test section ........................................... 231

5.9 Power supply circuit for the parallel channel test section ............. 233
List of Figures (continued)

6.1 The WANACO experiment ........................................ 238
6.2 The dye injection system ........................................ 241
6.3 Backflow of fluid from the upper plenum into the channel ... 243
6.4 Idealized model for the backflow into the channel ........... 244
6.5 Force balance for the idealized model for the backflow ....... 245
6.6 Onset of channel backflow map ................................... 250
7.1 Effect of eccentricity on channel flowrate for annular channels ......................................................... 256
7.2 Buoyancy-induced flow redistribution for a power ratio of 4.259 ................................................................. 261
7.3 Buoyancy-induced flow redistribution for a power ratio of 2.536 ................................................................. 262
7.4 Buoyancy-induced flow redistribution for a power ratio of 1.719 ................................................................. 263
7.5 Buoyancy-induced flow redistribution for a power ratio of 1.186 ................................................................. 264
7.6 Onset of backflow map for a two-channel system of different powers ......................................................... 266
8.1 Core velocities measured in an isolated PWR scale model core showing recirculation from the upper plenum ........ 271
8.2 Flow pattern of backflow from the upper plenum in the upper part of CH2 as it reversed from heated upflow to heated downflow ........................................... 273
8.3 Dimensionless pressure drop curve for a heated channel including multi-dimensional flow field effects ........... 275
D.1 Block diagram of MICON ........................................... 345
LIST OF TABLES

2.1 PRISM assembly type and CHANSOL numbering ................. 59
2.2 PRISM assembly geometry ..................................... 60
2.3 PRISM assembly flow orificing by location .................... 69
2.4 PRISM assembly power by location at beginning of cycle ....... 70
2.5 PRISM decay power and primary loop natural circulation flowrate used for CHANSOL Input .................................... 72
2.6 Channel upflow preference number for the PRISM reactor core by location ......................................................... 78
3.1 Boundary conditions for geometries with azimuthal symmetry ................................................................. 100
3.2 Coefficients for Eq. (3.79a) for the inner radius heated annulus for aiding flow ............................................. 120
3.3 Coefficients for Eq. (3.79a) for the outer radius heated annulus for aiding flow ............................................. 121
3.4 Coefficients for Eq. (3.79a) for the equivalent annulus for aiding flow ......................................................... 122
3.5 Coefficients for Eq. (3.79a) for the inner radius heated annulus for opposing flow ........................................... 123
3.6 Coefficients for Eq. (3.79a) for the outer radius heated annulus for opposing flow ........................................... 124
3.7 Coefficients for Eq. (3.79a) for the equivalent annulus for opposing flow ..................................................... 125
3.8 Coefficients for Eq. (3.79b) for the inner radius heated annulus for aiding flow .............................................. 133
3.9 Coefficients for Eq. (3.79b) for the outer radius heated annulus for aiding flow .............................................. 134
3.10 Coefficients for Eq. (3.79b) for the equivalent annulus for aiding flow ....................................................... 135
3.11 Coefficients for Eq. (3.79b) for the inner radius heated annulus for opposing flow ......................................... 136
3.12 Coefficients for Eq. (3.79b) for the outer radius heated annulus for opposing flow ......................................... 137
List of Tables (continued)

3.13 Coefficients for Eq. (3.79b) for the equivalent annulus for opposing flow........................................... 138
3.14 Forced convection Nusselt numbers for the annulus........... 146
3.15 Range of applicability of the equivalent annulus model..... 156
4.1 SCCMT Channel and Plena Dimensions................................. 185
4.2 SCCMT Steady State Conditions........................................... 191
4.3 Forced Convection Friction Factor Correlations Coefficients Used as MICON Inputs for the SCCMT............. 199
4.4 Least Squares Power Fit for Channel Valve and Flowmeter Pressure Drop for the SCCMT................................. 201
4.5 Channel Valve and Flowmeter Pressure Drop Input Parameters for MICON for the SCCMT.............................. 203
4.6 Steady State Flowrates Predicted by MICON for the SCCMT... 206
5.1 Channel dimensions for the parallel channel test section... 222
Chapter 1

INTRODUCTION

The mixed convection phenomenon (also referred to as "combined free and forced convection") can occur in thermal hydraulic components when the flow is small enough so that the local buoyancy force is not negligible compared to the local pressure gradient, inertial, and viscous forces. Examples of components that might operate in the mixed convection regime include nuclear reactor systems, heat exchangers, and solar systems. Under such conditions, the local buoyancy effect can severely alter both the quantitative and qualitative nature of the velocity and temperature fields, rendering forced convection analyses inaccurate. As a result, engineering parameters derived from the forced convection analyses for use in the forced convection regime (such as friction factors and Nusselt numbers) are also inaccurate when used for a component in the mixed convection regime.

The distinction between natural (free) convection and mixed convection is shown in Figure 1.1. In a qualitative sense, a flow field can be thought of as composed of a forced and natural convection component. The test for the presence of the forced convection component is to observe if a flow develops in the absence of heat transfer to and from the fluid. Figure 1.1a shows a tube in a pool of water. If the tube is adiabatic and no internal heat generation is present in the fluid, there there will be no flow through the tube. As heat is added to the tube, the fluid is warmed, and rises out of the tube. The hot fluid exiting the tube is replaced by flow into the
(a) Natural Convection  
(b) Mixed Convection

Figure 1.1 Comparison of natural and mixed convection.  
(Taken from Ref. H-1)
lower inlet to the tube. The flow field is thus pure natural convection since it is totally bouyancy-induced.

In Figure 1.1b, the tube is now a component in a loop that also includes a cooler and pump. If the pump is turned on, a flow will occur even if the tube heating is zero. There is therefore a forced convection component in the flow. Suppose the heating of the tube is now increased. The point at which the tube heating causes a significant altering of the flow field is the onset of mixed convection. At this point, both the forced and the natural convection components of the flow are significant.

The problem becomes more complicated if there is more than one tube connected to common inlet and outlet plena, as shown in Figure 1.2. The inlet flow to the system of parallel channels may or may not be zero. If the inlet flow is zero and a flow field exists in the system, the flow regime is a pure natural convection. If however, the inlet flow is non-zero, and heating or cooling the fluid in the system significantly alters the flow field, then the flow is mixed convection.

The analysis of mixed convection in parallel channels connected only at plena can be directly applied to the thermal hydraulic behavior of a Liquid Metal Fast Breeder Reactor (LMFBR) being passively cooled by natural circulation of the primary sodium loop. For such conditions, the flow through the core is expected to be low (Re<1000), and thus local bouyancy forces may alter the temperature and flow fields from that at high flow conditions.
Figure 1.2 Mixed convection in parallel channels connected only at plena.
Although the primary loop as a whole constitutes a system in natural circulation since the flows are exclusively buoyancy-induced, typically detailed analyses of the core are performed assuming the core can be decoupled from the remaining loop. The reason for this decoupling is due to economy. To analyze the entire primary loop using a numerical code that is capable of detailed modeling of the core is prohibitively expensive. Instead, typically what is done is to use a coarse noding scheme to model the primary loop in order to obtain the core boundary conditions such as the loop flowrate and core inlet temperature. Once these parameters are obtained from the coarse noding scheme, they are used as input to a code employing a fine noding scheme to model the core region. When the core is decoupled from the loop in this manner, a mixed convection analysis is performed in the core since the inlet flowrate is assumed to be independent of the details of the core flow field. Since the individual assemblies in an LMFBR are connected only at their inlets and outlets, an LMFBR core can be modeled as a system of parallel connected only at upper and lower plena with an inlet flowrate prescribed.

Figure 1.3 illustrates the important thermal hydraulic phenomena that need to be addressed for mixed convection flow through the core. Under high flow conditions, an LMFBR core is designed, using flow orifices, to create a certain flow distribution among the individual core assemblies. For low flow conditions, buoyancy forces cause this redistribution to be modified. This is due to several effects. The first is that the fluid in the hotter channels is less dense, which causes increased flow in these channels. The second effect is that the friction factors for the assemblies will be altered under mixed
Figure 1.3  Important core phenomena associated with natural circulation heat removal in an LMFBR.
convection conditions. Finally, the loss coefficients for the flow orifices that were designed for high flowrates might not be identical at low flowrates. All these effects will cause a buoyancy-induced flow redistribution of flow between assemblies in the core, called inter-assembly flow redistribution.

Flow redistribution is also expected within the individual assemblies, and is referred to as intra-assembly flow redistribution. The hotter subchannels will tend to draw more flow, altering the flow and temperature fields within the assembly. It is this flow redistribution within the assembly that caused the assembly friction factor to be altered.

For assemblies with low power, there is concern regarding the stability of the flow and the possibility of flow reversal in these assemblies. In particular, heated channels with weak upflow might interact with the upper plenum fluid to produce a complicated multi-dimensional flow pattern. For this regime, the channel and plenum interact so that they can not be decoupled and studied separately.

In the upper plenum, the low flow is expected to cause flow stratification. This might cause temperature gradient-induced stresses on component surfaces. In addition, buoyant plumes rising from the assemblies and recirculating flow patterns add to the complexity of the upper plenum analysis.

In this work, the phenomena associated with mixed convection flow through the core are examined. This is accomplished through the following steps:
(a) The formulation of the parameters and equations that characterize steady mixed convection flow in vertical parallel channels

(b) The analysis of buoyancy-Induced flow redistribution within the heated duct

(c) The formulation of numerical methods to study the transient behavior of mixed convection flow in vertical parallel channels

(d) The experimental investigation of the channel-plenum interaction

(e) The experimental investigation of inter-channel buoyancy induced flow redistribution

(f) A qualitative experimental investigation of the reversal of a heated channel from upflow to downflow.

In Chapter 2 of this work, the coupled equations that describe one-dimensional steady mixed convection flow in parallel channels are non-dimensionalized, and the resulting dimensionless parameters are identified. The non-dimensional equations are then used to derive the stability criteria for heated and cooled channels in mixed convection. A numerical method is then presented to solve the coupled equations, and is incorporated in a small code called CHANSOL. CHANSOL is then applied to the General Electric PRISM reactor core to predict the buoyancy-induced flow redistribution within the core for natural circulation of the primay sodium loop.

The laminar analysis of mixed convection flow in a duct is presented in Chapter 3. This analysis first develops general equations for the velocity and temperature fields across the duct for
fully-developed mixed convection flow. From these general expressions, new expressions for the mixed convection friction factor and Nusselt number are obtained. A modified friction factor is defined to include the effect of using the bulk instead of spatially-averaged density to calculate the gravity pressure gradient. This modified friction factor should be used in lumped parameter codes instead of the convectional friction factor. The fully-developed analysis developed is then applied to geometries that exhibit azimuthal symmetry.

Experimental mixed convection friction factor data from vertical rod bundles is also analyzed in Chapter 3. The data is predicted using a numerical subchannel code which can model buoyancy-induced flow redistribution in the developing region.

The numerical methods necessary to model transient one-dimensional mixed convection in parallel channels are presented in Chapter 4. The numerical analysis is incorporated into a code called MICON. MICON was then used to analyze the data collected from the GE Shutdwon Core Coolability Model Test (SCCMT) experiments.

The details of the design and construction of a parallel channel test section can be found in Chapter 5. The test section was then used to experimentally investigate the channel-plenum interaction, which consisted of backflow from the upper plenum into a channel with weak heated upflow, as presented in Chapter 6. The parameters that can be used to identify the onset of the phenomenon are identified and a simple correlation is devised from data collected in this work.

In Chapter 7, the CHANSOL code is used to predict buoyancy-induced flow redistribution in the parallel channel test section. The
data shows the limitation of using a one-dimensional analysis for prediction of a multi-dimensional flow field. The onset of the deviation of the data from the one-dimensional theory corresponds to the onset channel backflow analyzed in Chapter 6.

The behavior of a channel reversing from heated upflow to downflow is studied in Chapter 8. One-dimensional theory will not allow a heated channel to reverse from upflow to downflow with a pressure drop prescribed boundary condition. The multi-dimensional flow pattern that occurs during this transition is investigated using dye injection to visualize the flow fields in a qualitative manner.
CHAPTER TWO

STEADY ONE-DIMENSIONAL MIXED CONVECTION IN PARALLEL CHANNELS

2.1 Introduction

In this chapter an analytical analysis will be made of the steady behavior of multiple parallel channels connected only at a lower and upper plenum for a single phase fluid. Such an arrangement is shown in Figure 2.1 where a net flow through the system, \( \dot{m}_0 \), is allowed. Although there are a number of one dimensional codes (SCC [A-1,G-1], BIFR [S-1], MICON) that can solve the system of parallel channels for transients and steady conditions, the codes require that the channels be divided into an appropriate number of connected nodes. Once the conservation equations that describe the channels are discretized to the noding scheme employed, parameters that describe the thermal hydraulic behavior of the channel in an integral sense are lost. The motivation of this chapter is to describe the steady behavior of a system of parallel channels under mixed convection conditions in terms of integral channel parameters.

Chato [C-1] considered mixed convection in a system of three parallel channels each with uniform heat flux along its length. The inlet flowrate to the system, \( \dot{m}_0 \), was assumed to be zero. Viscosity variations along the channels were included in the analysis. Chato studied, both analytically and experimentally, patterns that occurred when one channel is heated, one channel is adiabatic, and the other channel is either heated or cooled. For this arrangement he found that a metastable regime exists in which two metastable flowrates can occur for a given set of channel heating rates. The pattern which actually occurs depends on the history of the system. An unstable heated downflow flow pattern also was predicted and as a result could not be observed experimentally.
Yahalam and Bein [Y-1] studied mixed convection in a multiple channel system and included two-phase flow effects. A solution procedure was presented which allowed the determination of a unique flow distribution among the channels when all the channels are originally in downflow. This procedure is based on defining a dimensionless preference for upflow number, which was derived neglecting viscosity variations along the channel lengths.

In this chapter, Chato's analysis is extended to include a non-zero inlet flowrate. The steady-state form of conservation equations are first solved for a given channel in an integral sense. Based on this formulation, a single dimensionless momentum-energy equation for each channel is presented. As in Chato's analysis, viscosity variations are included. Chato's analysis is further extended to include arbitrarily prescribed axial heat fluxes, which are accounted for by a heat flux shape factor. The momentum-energy equation is then utilized to predict the onset of mixed convection, the stability of heat flows, and flow reversal criteria.

A general solution procedure is presented to solve the coupled momentum-energy equations for the system of parallel channels. The solution procedure is incorporated into the computer code CHANSOL. The code is then used to model the reactor core of the General Electric PRISM reactor for the primary loop flowrate and decay power level expected during natural circulation of the primary sodium loop.

2.2 Formulation of the Conservation Equations

2.2.1 Momentum Equation

It is desired to derive the conservation equations that describe the system of N parallel channels depicted in Figure 2.1. Assuming that no radial (transverse) pressure gradients occur in the upper and lower plena, the pressure drops for all the channels are the same.
Figure 2.1 Arrangement of parallel channels connected only at lower and upper plena.
This pressure drop is defined by

$$\Delta p = p_1 - p_u$$  \hspace{1cm} (2.1)

where $p_1$ and $p_u$ are the pressures at the lower and upper extremes of the channels ($z=0$ and $z=L$), respectively.

The steady-state momentum equation for a channel at an axial location $z$ can be written as

$$(- \frac{dp}{dz}) = \rho g + \frac{f}{D_{en}} \left[ \frac{G_n}{2} \frac{C_n}{2} \right] + K \frac{G_n}{2} \frac{C_n}{2} \frac{1}{dz}$$  \hspace{1cm} (2.2)

For upflow, $G_n$ is positive and is negative for downflow. $K$ represents the local form loss factor at an axial position $z$, and can be used to model entrance and exit losses at the appropriate axial level. The $(1/dz)$ factor is a device that says that local form losses are at discrete points and are not continuous. This factor disappears upon integrating along the channel length.

The friction factor $f$ can be expressed as

$$f = \frac{c_n}{b}$$  \hspace{1cm} (2.3)

where the local Reynolds number is

$$Re = \frac{G_n}{D_{en}} \frac{b}{\mu}$$  \hspace{1cm} (2.4)

Substituting Eqs. (2.3) and (2.4) into (2.2) and integrating from the bottom of the channel ($z=0$) to the top of channel ($z=L$) yields

$$\Delta p = g \int_0^L \rho dz + \frac{c_n}{2} \frac{G_n}{D_{en}} \left[ \frac{1}{b+1} \right] \int_0^L \frac{b}{\mu} dz + \frac{G_n}{2} \frac{C_n}{2} \sum_{i=1}^n \frac{K_{in}}{\rho_{in}}$$  \hspace{1cm} (2.5)

The next goal is to express the integral expressions in Eq. (2.5) in terms of average fluid properties in the channel.

For a single component fluid, fluid properties such as the density and viscosity are a function of temperature and pressure. It is assumed here that the fluid is incompressible, and thus fluid properties are assumed to be pressure independent. As a first approximation, fluid properties and combinations of fluid properties
can be expressed as linear functions of temperature. Therefore, we make the following approximations:

\[ \rho = \rho_o (1 - \beta (t - t_o)) \]  \hspace{1cm} (2.6)

and

\[ \frac{\mu^b}{\rho} = \frac{\mu_o^b}{\rho_o} (1 - \Phi (t - t_o)) \]  \hspace{1cm} (2.7)

The reference density \( \rho_o \) and viscosity \( \mu_o \) are evaluated at the reference temperature \( t_o \).

It will be assumed for now that the average temperature in a channel can be expressed as

\[ \bar{t}_n = \frac{1}{L} \int_0^L t_n \, dz = \left[ t_{1n} + S_n (t_{un} - t_{1n}) \right] \] \hspace{1cm} ; \ \text{upflow} \hspace{1cm} (2.8a)

\[ \bar{t}_n = \frac{1}{L} \int_0^L t_n \, dz = \left[ t_{un} + (1 - S_n) (t_{1n} - t_{un}) \right] \] \hspace{1cm} ; \ \text{downflow} \hspace{1cm} (2.8b)

\( S_n \) represents the axial shape of the heat flux and is used to relate the average channel temperature to the temperatures at the channel upper and lower extremes. For a linear temperature rise (uniform axial heat flux) the average temperature is the arithmetic average of the temperatures at the channel extremes, and thus the \( S_n \) in Eq. (2.8) equals 0.5. Equations for \( S_n \) for other axial heat flux shapes will be derived in Section 0.2.4.

The integral in the first term on the right hand side of Eq. (2.5) can be combined with (2.6), and rearranged to

\[ \int_0^L \rho \, dz = \rho_o L \left[ 1 - \beta \left( 1 - \frac{1}{L} \int_0^L t \, dz - t_o \right) \right] \]  \hspace{1cm} (2.9)

Substituting Eq. (2.8) into (2.9) yields

\[ \int_0^L \rho \, dz = \rho_o L \left[ 1 - \beta (t_{1n} - t_o + S_n (t_{un} - t_{1n})) \right] \] \hspace{1cm} ; \ \text{upflow} \hspace{1cm} (2.10a)

\[ \int_0^L \rho \, dz = \rho_o L \left[ 1 - \beta (t_{un} - t_o + (1 - S_n) (t_{1n} - t_{un})) \right] \] \hspace{1cm} ; \ \text{downflow} \hspace{1cm} (2.10b)
In a similar manner, the integral in the second term on the RHS of Eq. (2.5) can be combined with Eqs. (2.7) and (2.8) to yield

\[
\int_0^L \frac{\mu^b}{\rho} \, dz = \frac{\mu^b}{\rho_o} L \left[ 1 - \phi(t_{un} - t_o + S_n(t_{un} - t_{ln})) \right] ; \text{ upflow} \tag{2.11a}
\]

\[
\int_0^L \frac{\mu^b}{\rho} \, dz = \frac{\mu^b}{\rho_o} L \left[ 1 - \phi(t_{ln} - t_o + (1 - S_n)(t_{ln} - t_{un})) \right] ; \text{ downflow} \tag{2.11b}
\]

Finally, the last term of Eq. (2.5) can be simplified by assuming that all local losses will be evaluated by using the reference density, \( \rho_o \). Thus,

\[
\frac{G_n}{2} \sum_{1 \rightarrow n} \frac{K_{in}}{\rho_{in}} = \frac{G_n}{2 \rho_o} \sum_{1 \rightarrow n} K_{in} \tag{2.12}
\]

Equation (2.5) can now be rewritten without integral expressions. Substituting Eqs. (2.10) through (2.12) and rearranging yields

\[
\frac{\Delta p}{\rho_o q L} - 1 = -\beta(t_{ln} - t_o + S_n(t_{un} - t_{ln}))
\begin{align*}
&+ \frac{|G_n|^{2-b}}{2D_{en} \rho_o^2 q} \frac{\mu^b}{\rho_o} \left[ 1 - \phi(t_{un} - t_o + S_n(t_{un} - t_{ln})) \right] \\
&+ \frac{|G_n|^2}{2 \rho_o^2 q L} \sum_{1 \rightarrow n} K_{in} ; \text{ upflow} \tag{2.13a}
\end{align*}
\]

\[
\frac{\Delta p}{\rho_o q L} - 1 = -\beta(t_{un} - t_o + (1 - S_n)(t_{ln} - t_{un}))
\begin{align*}
&- \frac{|G_n|^{2-b}}{2D_{en} \rho_o^2 q} \frac{\mu^b}{\rho_o} \left[ 1 - \phi(t_{ln} - t_o + (1 - S_n)(t_{ln} - t_{un})) \right] \\
&- \frac{|G_n|^2}{2 \rho_o^2 q L} \sum_{1 \rightarrow n} K_{in} ; \text{ downflow} \tag{2.13b}
\end{align*}
\]

Equation (2.13) is the integrated momentum equation for our system.
2.2.2 Energy Equation

The steady-state momentum equation for a channel \( n \) at an axial location \( z \) can be written as

\[
G_n A_n c_p \frac{dt}{dz} = q_n' \tag{2.14}
\]

Assuming a constant specific heat, Eq. (2.14) can be integrated along the channel yielding, after rearranging,

\[
t_{un} - t_{ln} = \frac{q_n}{G_n A_n c_p} \tag{2.15}
\]

which can be rewritten as

\[t_{un} - t_{ln} = \frac{q_n}{|G_n A_n c_p|}, \quad \text{upflow} \tag{2.16a}\]

\[t_{ln} - t_{un} = \frac{q_n}{|G_n A_n c_p|}, \quad \text{downflow} \tag{2.16b}\]

Equation (2.16) is the integrated energy equation.

2.2.3 Dimensionless Momentum-Energy Equation

It is now desired to non-dimensionalize the momentum and energy equations and combine them into a single momentum-energy equation.

Defining

\[
Re_n = \frac{|G_n| D_{en}}{\mu_o} \tag{2.17}
\]

\[
\Delta p^* = \frac{\Delta p}{\rho_o q L} - 1 \tag{2.18}
\]

\[
X_{un} = \beta(t_{un} - t_o) \tag{2.19}
\]

\[
X_{ln} = \beta(t_{ln} - t_o) \tag{2.20}
\]

\[
\theta_n = \frac{q_n \beta D_{en}}{A_n c_p \mu_o} \tag{2.21}
\]

\[
\delta_n = \frac{c_n^2}{2D_{en} \rho_o^2 g} \tag{2.22}
\]
\[
\gamma_n = \frac{\sum K_{in} \mu^2}{20 \phi^2 \rho_o q_n} \quad \text{(2.23)}
\]
\[
\xi = \frac{\phi}{\beta} \quad \text{(2.24)}
\]

we can combine Eqs. (2.13) and (2.16) to obtain

\[
\Delta p^+ = -X_{in} - \frac{\theta n}{n Re_n} + \delta Re^{2-b} (1 - \xi X_{in} - \xi S \frac{\theta n}{n Re_n}) + \gamma Re^2, \quad \text{upflow} \quad \text{(2.25a)}
\]

\[
\begin{align*}
(1) & \quad (2) & \quad (3a) & \quad (3b) & \quad (4) \\
\Delta p^+ = -X_{in} - (1 - S) \frac{n}{n Re_n} - \delta Re^{2-b} (1 - \xi X_{in} - \xi (1 - S) \frac{\theta n}{n Re_n}) - \gamma Re^2 \\
& \quad \text{downflow} \quad \text{(2.25b)}
\end{align*}
\]

Term (1) in the above equation accounts for differences between the reference density \(\rho_o\) and the density of the fluid entering the channel from the plenum. If the inlet temperatures of the channels match the reference temperature, the first term vanishes. Term (2) accounts for the density change caused by heating (or cooling) the fluid. It is proportional to \(q_n\) (through \(\theta_n\)) and inversely proportional to \(G_n\) (through \(Re_n\)). The next term can be divided into two factors, (3a) and (3b). Factor (3a) represents the frictional pressure loss in the channel. For laminar flow, \(b=1\), and this factor is proportional to channel flowrate. Factor (3b) is a correction to account for variations of \(\mu^b/\rho\) along the channel. If variations in \(\mu^b/\rho\) can be neglected, \(\xi = \phi = 0\) and the term reduces to one. Term (4) represents the sum of the local form losses in the channel. This term is proportional to the flowrate squared.

2.2.4 Heat Flux Shape Factor \(S\)

The heat flux shape factor \(S\) was introduced as a way to express the average temperature along a channel in terms of the inlet and
outlet temperatures of the channel. This device allowed us to replace integrals involving fluid properties with simpler arithmetic functions of inlet and outlet temperatures. It is now desired to derive an expression for \( S_n \) in terms of the axial heat flux in the channel.

To begin, Eq. (2.8a) is solved for \( S_n \) yielding

\[
S_n = \frac{1}{t_{un} - t_{ln}} \int_0^L t_n(z) dz - t_{ln} \quad ; \text{upflow}
\]  

(2.26)

Next, an expression for \( t_n(z) \) is desired. Integrating Eq. (2.14) from \( z' = 0 \) to \( z' = z \) results in

\[
t_n(z) = \frac{z}{G_n A_{fn} c_p} + \frac{z}{c_{ln}} \quad ; \text{upflow}
\]  

(2.27)

Combining Eqs. (2.26), (2.27) and (2.16a), one obtains

\[
S_n = \frac{1}{q_n L} \int_0^L dz \int_0^z q_n(z') dz' 
\]  

(2.28)

An identical equation results when the downflow forms of Eqs. (2.8), (2.27), and (2.16) are used to derive \( S_n \).

Suppose a channel has a lower length \( L_1 \) adiabatic, a middle length \( L_h \) uniformly heated, and an upper length \( L_u \) also adiabatic, as shown in Figure 2.2. It is desired to find an expression for \( S_n \). Rewriting Eq. (2.28),

\[
S_n = \frac{1}{q_n L} \left[ \int_0^{L_1} dz \int_0^z q_n(z') dz' + \int_{L_1}^{L_1+L_h} dz \int_0^z q_n(z') dz' + \int_{L_1+L_h}^L dz \int_0^z q_n(z') dz' \right] 
\]  

(2.29)
Figure 2.2 Drawing of a channel with upper and lower unheated lengths.
\[ S_n = \frac{0.5 L_h + L_1}{L} \]  

(2.31)

For the case of a uniformly heated channel, \( L_{a1} = 0 \), \( L_h = L \), and \( S_n = 0.5 \).

2.2.5 Dimensionless Continuity Equation

Typically in a system with a large number of parallel channels, there will be groups of channels with identical channel heat fluxes and geometries. For each channel type \( n \), it will be assumed that there will be \( y_n \) identical channels. If it can be assumed that each channel in the set of channels of type \( n \) behave identically, then the system mass balance is

\[ \sum_{n=1}^{N} y_n g_n A_{f_n} = \dot{m}_0 \]  

(2.32)

The subscript \( n \) refers to a channel type and not an individual channel so there are assumed to be \( N \) sets of channels. The total number of channels is given by

\[ N_c = \sum_{n=1}^{N} y_n \]  

(2.33)

In order to non-dimensionalize Eq. (2.32), it is combined with (2.17), resulting in

\[ \sum_{n} y_n g_n Re_n - \sum_{n} y_n g_n Re_n = Re_0 \]  

upflow \hspace{1cm} \text{downflow} \hspace{1cm} (2.34)

where

\[ g_n = \frac{A_{f_n} D_{en}}{A_{fo} D_{en}} \]  

(2.35)
\[ \text{Re}_o = \frac{m \text{De}_o}{\mu_o A_{fo}} \]  

(2.36)

and

\[ A_{fo} = \sum_{i=1}^{N} \frac{n_{i}}{A_{fn}} \]  

(2.37)

\[ D_{eo} = \frac{4 \sum_{i=1}^{N} \frac{n_i}{A_{fn}}}{\sum_{n=1}^{N} \frac{n_i}{P}} \]  

(2.38)

The parameters \( A_{fo} \) and \( D_{eo} \) represent the flow area and hydraulic diameter of the system if the system of channels were treated as a single channel.

Once the channel heat fluxes are specified, the unknowns for the system are \( \text{Re}_n \) and \( \Delta p^+ \), which total to \((N+1)\) unknowns. The equations necessary for closure are Eqs. (2.25) (N momentum-energy equations) and (2.34) (one mass balance equation).

2.3 One-Dimensional Channel Performance

2.3.1 Onset of Mixed Convection

Now that the conservation equations have been formulated and non-dimensionalized for the parallel channel system, it is desired to apply the results to predict the behavior of a parallel channel arrangement. For forced convection flow through the channels, the analysis is less complex because the momentum and energy equations are not coupled since the buoyancy-induced density variations in the channel have a minimal effect on the channel pressure drop. It is thus desirable to know when a mixed convection analysis is warranted. The point at which the momentum and energy equations should no longer be solved independently is called the "onset of mixed convection."
Buoyancy effects are significant if the density change caused by heating the fluid alters the channel pressure drop an amount comparable to the resistive pressure drop (frictional and form losses). When buoyancy effects are 10% of resistive effects, one can write, from Eq. (2.25), for upflow,

\[ S_{n} \frac{\theta_{n}}{Re_{n}} = 0.10 \left[ \delta \frac{Re_{n}^{2-b}}{n^{n}} \left( 1 - \chi_{ln} - 5S_{n} \frac{\theta_{n}}{Re_{n}} \right) + \gamma \frac{Re_{n}^{2}}{n^{n}} \right] \] ; upflow \hspace{1cm} (2.39)

Equation (2.39) can be solved iteratively for \( Re_{n} \) if \( \theta_{n} \) is known. Once \( Re_{n} \) is found, it can be substituted into Eq. (2.25) to obtain \( \Delta p^{+} \).

For the special case of laminar flow \((b = 1)\), no form losses \((\gamma_{n} = 0)\) and negligible viscosity variation \((\xi = 0)\), Eq. (2.39) reduces to

\[ S_{n} \frac{\theta_{n}}{Re_{n}} = 0.10 \delta \frac{Re_{n}}{n} \] \hspace{1cm} (2.40)

which can be solved for \( Re_{n} \) yielding

\[ Re_{n} = \left[ \frac{10 \delta \frac{S_{n}}{\theta_{n}}}{\delta n} \right]^{0.5} \] \hspace{1cm} (2.41)

The corresponding dimensionless pressure drop is

\[ \Delta p^{+} = -\chi_{ln} - \theta \left[ \frac{\delta S_{n}}{10 \delta n} \right]^{0.5} + \left[ 10 \delta \frac{S_{n}}{\theta_{n}} \right]^{0.5} \] \hspace{1cm} (2.42)

2.3.2 Adiabatic Channel Reversal From Upflow to Downflow

For a system of parallel channels of which some are adiabatic and some are heated, there will be a certain inlet flowrate below which the flow in one or more of the adiabatic channels will reverse from upflow to downflow. Writing down Eq. (2.25) for \( \theta_{n} = 0 \), one obtains.
\[ \Delta p^+ = -X_{ln} + \delta_n \Re^{2-b} (1 - \xi X_{ln}) + \gamma \Re^2 \quad \text{upflow} \quad (2.43) \]

At the point of flow stagnation, all terms vanish from this equation but the first, leaving

\[ \Delta p^+ = -X_{ln} \quad (2.44) \]

If all the channels have the same inlet temperature, all the adiabatic channels will reverse simultaneously. The inlet flowrate at which this occurs can be calculated by iteratively solving Eq. (2.25) for \( \Re_n \) for the heated channels. The inlet flowrate can then be calculated from the system mass balance, Eq. (2.34). For the special case of laminar flow (\( b = 1 \)), no form losses (\( \gamma_n = 0 \)), and negligible viscosity variation (\( \xi = 0 \)), we can solve Eqs. (2.44) and (2.25) for \( \Re_n \), yielding

\[ \Re_n = \left[ \frac{s \theta_n}{n} \right]^{0.5} \quad (2.45) \]

Substituting Eq. (2.45) into Eq. (2.25) yields

\[ \Re_0 = \sum_{n=1}^{N} \left[ \frac{s \theta_n}{n} \right]^{0.5} \quad (2.46) \]

where \( \theta_n = 0 \) for the adiabatic channels. Equation (2.46) thus can be used to calculate the inlet flowrate at which the adiabatic channels reverse from upflow to downflow.

2.3.3 Stability of Cooled Upflow

For stable operation of a channel with a prescribed pressure drop, the channel pressure drop must increase as the flowrate is increased. An equivalent requirement is that the channel must operate under conditions such that \( \partial \Re / \partial \Delta p^+ > 0 \). A channel in an array of many parallel channels experiences a prescribed pressure drop since a change in its conditions will have a small effect on the other
channels in the system. Thus, the $\frac{3\text{Re}}{\Delta p^+} > 0$ stability requirement must be maintained to avoid a flow excursion.

Figure 2.3 is a plot of $\Delta p^+$ versus $\text{Re}_n$ for a cooled channel. In the figure, a negative $\text{Re}_n$ indicates downflow. For a $\Delta p^+$ greater than $\Delta p^+_c$, three channel flow rates are possible, two in upflow and one in downflow. Using our stability requirement, we see that only two of these flow rates are stable. For $\text{Re}_n < \text{Re}_c$, upflow is unstable. Thus, a channel originally in cooled upflow will reverse to downflow as $\text{Re}_n$ decreases to $\text{Re}_c$, as shown by the dashed line. Notice that stable downflow can exist for any $\Delta p^+$ so that once the channel reverses to downflow, it may not reverse again to upflow.

Re$_c$ can be found by differentiating Eq. (2.25) with respect to Re, setting the result to zero, and solving for Re$_c$. For the special case of laminar flow ($b = 1$), no form losses ($\gamma_n = 0$), and negligible $\mu_b/\rho$ variation, this procedure produces

$$\text{Re}_c = \left[ \frac{S \theta}{\delta_n} \right]^{0.5}$$  \hspace{1cm} (2.47)

Substituting this result into Eq. (2.25) with the simplifying assumptions results in

$$\Delta p^+_c = - \chi_{\text{in}} + 2 \left[ -\frac{\theta \delta_n S_n}{\theta_n} \right]^{0.5}$$  \hspace{1cm} (2.48)

Thus, when the system pressure is less than that given by the above equation, cooled upflow channels will reverse to downflow.

2.3.4 Stability of Heated Downflow

Figure 2.4 is a plot of $\Delta p^+$ versus $\text{Re}_n$ for a heated channel. In the figure, a negative $\text{Re}_n$ indicates downflow. For a $\Delta p^+$ less than $\Delta p^+_c$, three channel flow rates are possible, two in downflow and
Figure 2.3 Cooled channel flow reversal from upflow to downflow.
Figure 2.4 Heated channel flow reversal from downflow to upflow.
one in upflow. Using our stability requirement we see that only two of these flowrates are stable. For \( Re_n < Re_c \), downflow is unstable. Thus, a channel originally in heated downflow will reverse to upflow as \( Re_n \) approaches \( Re_c \), as shown by the dashed line.

Notice that stable upflow can exist for any \( \Delta p^+ \) so that once the channel reverses to upflow, it may not reverse again to downflow. \( Re_c \) can be found by differentiating Eq. (2.25) with respect to \( Re \), setting the result to zero, and solving for \( Re_c \), as was done for cooled upflow. For the special case of laminar flow (\( b=1 \)), no form losses (\( \gamma_n = 0 \)), and negligible \( \mu^b / \rho \) variation, this procedure produces

\[
Re_c = \left[ \frac{S n \delta}{\delta_n} \right]^{0.5} \tag{2.49}
\]

Substituting this result into Eq. (2.25) with the simplifying assumptions results in

\[
\Delta p_c^+ = -X_{ln} - 2 \left[ \theta n \delta_n S_n \right]^{0.5} \tag{2.50}
\]

Thus, when the system pressure is greater than that given by the above equation, heated downflow will reverse to upflow.

2.3.5 Preference for Upflow

Suppose a system of heated and adiabatic parallel channels are all in downflow. As the pressure drop is increased by increasing the inlet flowrate, the channels will in turn reverse to upflow. It can be said that the channel that reverses first has the strongest "preference for upflow." The \( \Delta p^+ \) at which a heated channel reverses from downflow to upflow is given by Eq. (2.50), and thus this \( \Delta p_c^+ \) is a measure of the upflow preference. Since a low \( \Delta p^+ \) indicates a
high upflow preference, an upflow preference function must decrease with increasing $\Delta p_+$. The upflow preference number chosen here is defined by

$$ H_n = \frac{1}{2} \left[ \chi_{un} - \Delta p_+ \right] $$

(2.51)

where $\Delta p_+$ is given by Eq. (2.50). Substituting for $\Delta p_+$ yields

$$ H_n = \left[ \theta_n \delta_n S_n \right]^{0.5} $$

(2.52)

By definition,

$$ \theta_n \alpha \frac{q_n D_{en}}{A_{fn}} $$

(2.53)

and

$$ \delta_n \alpha \frac{1}{D_{en}^3} $$

(2.54)

so that

$$ H_n \alpha \frac{q_n^{0.5}}{A_{fn}^{0.5} D_{en}^{0.5} S_n^{0.5}} $$

(2.55)

Yahalom and Rein [Y-1] suggested that preference for downflow in a channel can be determined by the dimensionless number $Y_n$ defined by

$$ Y_n \equiv \frac{\left[ \frac{D_{en}}{D_{e1}} \right]^{a_1} \left[ \frac{A_{fn}}{A_{f1}} \right]^{a_2} \left[ \frac{q_n}{q_1} \right]^{a_3}}{q_n} $$

(2.56)

where channel 1 is that channel with the greatest value of

$$ \frac{q_n^{a_3}}{D_{en}^{a_1} A_{fn}^{a_2}} $$

(2.57)
Yahalam and Bein concluded that \( a_1 = 1 \), \( a_2 = 0.5 \), and \( a_3 = 0.5 \) are good approximations for obtaining the \( Y \) number. This agrees with the expression for \( H_n \) derived here with the assumptions of laminar flow, no form losses, and negligible variation of \( \mu^b/\rho \).

**2.4 Solution of the Coupled Momentum-Energy Equations**

**2.4.1 General Solution Procedure**

In this section a solution procedure is described to solve the system of non-dimensional equations for \( R_e_n \) and \( \Delta p^+ \). The equations necessary for closure are rewritten below:

\[
\Delta p^+ = -X_{\ln} - S_{n} \frac{\theta}{n R_e_n} + \delta \frac{R_e^2}{n} (1 - \xi X_{\ln} - \xi S_{n} \frac{\theta}{n R_e_n}) + \gamma \frac{R_e^2}{n} \quad \text{upflow}
\]

(2.25a)

\[
\Delta p^+ = -X_{un} - (1 - S_{n}) \frac{\theta}{n R_e_n} - \delta \frac{R_e^2}{n} (1 - \xi X_{un} - \xi (1 - S_{n}) \frac{\theta}{n R_e_n}) + \gamma \frac{R_e^2}{n} \quad \text{downflow}
\]

(2.25b)

\[
\sum y_{n} q_{n} R_e_n - \sum y_{n} q_{n} R_e_n = R_e_0 \quad \text{upflow}
\]

(2.34)

\[
\sum y_{n} q_{n} R_e_n - \sum y_{n} q_{n} R_e_n = R_e_0 \quad \text{downflow}
\]

It will be assumed that all channel heat fluxes are prescribed so that all \( S_{n} \) and \( \theta_{n} \) are known.

The general solution procedure is shown in Figure 2.5. First an initial estimate of \( \Delta p^+ \) is chosen. Once this has been obtained, \( R_e_n \) is determined for each channel type based on this \( \Delta p^+ \). Each value of \( R_e_n \) is determined iteratively solving the momentum-energy equation for each channel type \( n \). The Newton-Raphson method is used, which finds the roots of a function \( h(x) \) by iteratively approximating the roots from the equation.
Figure 2.5 General solution procedure for solving the coupled equations for parallel channels connected only at plena.
\[ x_{i+1} = x_i - \frac{h(x_i)}{(\frac{dh}{dx})_{x_i}} \]  

(2.58)

where the subscripts \( i \) and \( i+1 \) refer to the current and new estimates respectively. When solving the momentum-energy equation, we seek a solution of the equation

\[
h_n = -X_{in} - S_n \frac{\theta_n}{Re_n} + \delta_n Re_n^{2-b}(1 - \xi X_{in} - \xi S_n \frac{\theta_n}{Re_n}) + \gamma_n Re_n^2 - \Delta p^+ = 0
\]

; upflow  \hspace{1cm} (2.59a)

\[
h_n = -X_{un} - (1 - S_n) \frac{\theta_n}{Re_n} - \delta_n Re_n^{2-b}(1 - \xi X_{un} - \xi (1 - S_n) \frac{\theta_n}{Re_n})
- \gamma_n Re_n^2 - \Delta p^+ = 0
\]

; downflow  \hspace{1cm} (2.59b)

Successive approximation of \( Re_n \) can be obtained from

\[
(Re_n)_{i+1} = (Re_n)_i - \frac{h_n((Re_n)_i)}{(\frac{dh}{d(Re_n)})(Re_n)_i}
\]

(2.60)

where

\[
\frac{dh_n}{d(Re_n)} = S_n \frac{\theta_n}{Re_n^2} + (2 - b)(\delta_n Re_n^{1-b})(1 - \xi X_{in} - \xi S_n \frac{\theta_n}{Re_n})
+ \delta_n Re_n^{2-b} \xi S_n \frac{\theta_n}{Re_n^2} + 2\gamma_n Re_n \hspace{1cm} \text{; upflow}
\]

(2.61a)

\[
\frac{dh_n}{d(Re_n)} = (1 - S_n) \frac{\theta_n}{Re_n^2} - (2 - b)(\delta_n Re_n^{1-b})(1 - \xi X_{un} - \xi (1 - S_n) \frac{\theta_n}{Re_n})
- \delta_n Re_n^{2-b} \xi (1 - S_n) \frac{\theta_n}{Re_n^2} + 2\gamma_n Re_n \hspace{1cm} \text{; downflow}
\]

(2.61b)

After Eqs. (2.59) through (2.61) are used to solve for \( Re_n \) for each channel, the system mass balance (Eq. (2.34)) must be chosen. If this equation is not satisfied within a given tolerance, a new value
of $\Delta p^+$ is chosen, and the procedure repeated. The mass balance 
constant is exactly satisfied when,
\[ h_{n+1} = \sum_{\text{upflow}} y_n g_n R_{en} - \sum_{\text{downflow}} y_n g_n R_{en} - R_e = 0 \] (2.62)

If this equation is not satisfied within the allowed tolerance, a new 
estimate of $\Delta p^+$ is given by
\[ \Delta p_{n+1}^+ = \Delta p_n^+ - \frac{h_{n+1}(\Delta p^+)}{\frac{dh_{n+1}}{d\Delta p^+}} \] (2.63)

It might seem odd that a mass balance equation is being used to 
find successive approximations to the pressure drop $\Delta p^+$, especially 
since $\Delta p^+$ does not explicitly appear in the Eq. (2.62). In fact, since 
each $R_{en}$ is a function of $\Delta p^+$, one can write
\[ h_{n+1}(\Delta p^+) = h_{n+1}(R_{e1}, R_{e2}, R_{e3}, \ldots, R_{eN}) \] (2.64)

and
\[ \frac{dh_{n+1}}{d\Delta p^+} = \frac{dh_{n+1}}{dR_{e1}} \frac{dR_{e1}}{d\Delta p^+} + \frac{dh_{n+1}}{dR_{e2}} \frac{dR_{e2}}{d\Delta p^+} + \frac{dh_{n+1}}{dR_{e3}} \frac{dR_{e3}}{d\Delta p^+} + \ldots, \frac{dh_{n+1}}{dR_{eN}} \frac{dR_{eN}}{d\Delta p^+} \] (2.65)

The derivatives of Eq. (2.65) are given by
\[ \frac{dh_{n+1}}{dR_{en}} = y_n g_n \text{; upflow} \] (2.66a)
\[ \frac{dh_{n+1}}{dR_{en}} = -y_n g_n \text{; downflow} \] (2.66b)

and
\[ \frac{dR_{en}}{d\Delta p^+} = \left[ \frac{dR_{en}}{d\Delta p^+} \right] = \left[ \frac{dh_{n+1}}{dR_{en}} \right] \] (2.67)
Substituting Eqs. (2.64) through (2.67) into (2.63) yields

\[
\Delta p_{i+1} = \Delta p_i - \frac{h_{n+1}(Re_1, Re_2, Re_3, \ldots, Re_N)}{\sum_{n} y_n g_n \left[ \frac{dh}{dRe_n}(Re_i) \right] \text{upflow} - \sum_{n} y_n g_n \left[ \frac{dh}{dRe_n}(Re_i) \right] \text{downflow}}
\]

Equation (2.68) illustrates explicitly that \( \Delta p^+ \) is a function of the flowrate in each channel and hence it is appropriate to use the mass balance equation to find successive approximations to \( \Delta p^+ \). The values of \( Re_n \) are then solved for each channel type \( n \) and the process is repeated until convergence is achieved.

2.4.2 The Computer Code CHANSOL

The computer code CHANSOL was developed as a tool to rapidly solve the \((N+1)\) simultaneous equations formulated above. The code was written to be executed on an IBM PC or a compatible microcomputer so that many runs could be made with little computer expense. The code is relatively small (about 250 FORTRAN statements) and so has modest memory requirements.

Figure 2.6 is a subroutine flowchart of the code. Subroutine READIT first reads in the data for the channels in dimensional form. The input data is then non-dimensionalized so that the non-dimensionalized form of the equations can be solved.

After the data is non-dimensionalized, subroutine SOLVER solves the \((N+1)\) coupled equations. The solution method is similar to that described in the above section. The exception is that Eq. (2.58) is modified to achieve a faster convergence. When a successive approximation to either \( Re_n \) or \( \Delta p^+ \) is sought, the following equation is used:
Figure 2.6 CHANSOL subroutine flowchart.
\[ x_{i+1} = x_i - \frac{h(x_i)}{(dh/dx)_x} F_{OR} \]  

(2.69)

where \( x_i \) is either \( (R_{en}) \) or \( \Delta p_i \). \( F_{OR} \) is an over-relaxation factor used to increase the stability of the convergence process. For the momentum-energy equation iteration used to find \( R_{en} \), a value of \( F_{OR} = 0.6 \) was found to enhance the convergence. For the equation used to update \( \Delta p^+ \) (Eq. (2.68)), a value of \( F_{OR} = 0.9 \) was found to stabilize the convergence with few iterations.

After the \( (N+1) \) simultaneous equations are solved in SOLVER, the results are dimensionalized and output in subroutine RESULTS. The FORTRAN listing and input description for CHANSOL is in Appendix A.

2.5 Application of CHANSOL to Model the General Electric PRISM Reactor Core During Natural Circulation of the Primary Sodium Loop

2.5.1 Prism Description

The PRISM (Power Reactor Inherently Safe Module) is a small, modular liquid metal reactor [H-3]. The reactor has a pool-type primary heat transport loop. Figure 2.7 is a qualitative drawing of the PRISM primary sodium loop. Coolant flows from the lower plenum and up through the assemblies of the core. The coolant then mixes in the upper plenum pool and flows through the intermediate heat exchanger (IHX) where, under normal conditions, heat is transferred to the intermediate sodium loop. The sodium then flows through the electromagnetic (EM) pump and back to the lower plenum. The sodium in the primary loop is totally contained in the reactor vessel.
Figure 2.7 Qualitative drawing of the PRISM reactor primary loop.
The reactor core is composed of 211 assemblies arranged in a hexagonal pattern with all six corner assemblies removed, as shown in Figure 2.8. The interior region of the core is primarily comprised of fuel assemblies and the exterior region of blanket assemblies. The extreme peripheral assemblies are radial shield assemblies. The individual assemblies will be identified here by their geographic location. Each assembly can be associated with a unique ring and radial row. The center assembly of the core is Ring 1 and the outer ring of radial shield assemblies is Ring 9. The radial rows are shown in Figure 2.8. Only five rows are identified because symmetry requires only a thirty degree sector model of the core. Thus, each assembly within the thirty degree sector will be treated as a unique channel type. The total number of channels within the core for a specific channel type can be calculated as follows: For those channel types completely within the sector, there are twelve total channels within the core. For those channel types half within the sector, there are a total of six channels within the core. For the center assembly, there is obviously one channel of that type.

At full power conditions, 95.82% of the core inlet flowrate flows through the fuel and radial blanket assemblies. Of the remaining flow fraction, the radial shield assemblies account for 0.36% of the flow, control assemblies account for 1.60%, and 2.22% is leakage, which is unheated [M-1]. Based on this flow distribution between assembly types, only the fuel and radial blanket assemblies will be modeled. It will be assumed that 95.82% of the inlet flowrate flows through the fuel and radial blanket assemblies regardless of the inlet flowrate and core power conditions.
Figure 2.8  PRISM reactor core showing assembly type distribution and the radial rows for the 30° sector. (Adapted from Ref. M-1)
Table 2.1

PRISM Assembly Type and CHANSOL Numbering*

<table>
<thead>
<tr>
<th>Ring Number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>F(1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>F(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>F(3)</td>
<td>F(4)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>C(NM)</td>
<td>F(5)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>F(6)</td>
<td>F(7)</td>
<td>F(8)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>RB(9)</td>
<td>F(10)</td>
<td>F(11)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>RB(12)</td>
<td>RB(13)</td>
<td>RB(14)</td>
<td>RB(15)</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>RS(NM)</td>
<td>RB(16)</td>
<td>RB(17)</td>
<td>RB(18)</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td>RS(NM)</td>
<td>RS(NM)</td>
<td>RS(NM)</td>
<td>RS(NM)</td>
</tr>
</tbody>
</table>

Key: F Fuel Assembly  
C Control Assembly  
RB Radial Blanket Assembly  
RS Radial Shield Assembly  
NM Not modeled in CHANSOL simulation

* CHANSOL numbering in parenthesis
### Table 2.2

**PRISM Assembly Geometry**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>F</th>
<th>RB</th>
<th>RS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pins per Assembly</td>
<td>217</td>
<td>127</td>
<td>61</td>
</tr>
<tr>
<td>Spacer Type</td>
<td>Wire</td>
<td>Wire</td>
<td>-</td>
</tr>
<tr>
<td>Flat-to-Flat Distance (cm)</td>
<td>13.48</td>
<td>13.48</td>
<td>13.48</td>
</tr>
<tr>
<td>Pin Outer Diameter (cm)</td>
<td>0.663</td>
<td>0.986</td>
<td>1.529</td>
</tr>
<tr>
<td>Wire Diameter (cm)</td>
<td>0.046</td>
<td>0.036</td>
<td>-</td>
</tr>
<tr>
<td>Helical Pitch (cm)</td>
<td>22.35</td>
<td>10.16</td>
<td>-</td>
</tr>
<tr>
<td>Rod Pitch-to-Diameter Ratio</td>
<td>1.069</td>
<td>1.036</td>
<td>1.004</td>
</tr>
<tr>
<td>Helical Pitch-to-Diameter Ratio</td>
<td>33.72</td>
<td>10.31</td>
<td>-</td>
</tr>
</tbody>
</table>

**Key:**
- **F** Fuel
- **RB** Radial Blanket
- **RS** Radial Shield
Figure 2.9  Dimensions of the fuel pins for the PRISM reactor core.  
(From Ref. M-1)
Table 2.1 shows the CHANSOL numbering scheme used to model the PRISM core assemblies. There are eighteen channel types, each corresponding to one of the assemblies in the 30° sector of the core shown in Figure 2.8.

In the individual assemblies, rods are arranged in a triangular array surrounded by a hexagonal housing (sometimes called a "can"). Each assembly has an inner flat-to-flat distance of 13.47 cm (5.305 in) with a wall thickness of 0.348 cm (0.137 in) so that the assembly pitch is 13.82 cm (5.442 in). Because the housing isolates each assembly, the individual assemblies within the core can be modeled as parallel channels connected only at upper and lower plena. Heat conduction between assemblies through the assembly housing will be neglected. This is a conservative assumption since the heat conduction would tend to decrease the temperatures in the hotter assemblies.

As stated above, there are primarily three types of assemblies within the core, each with its own rod diameter and spacing. In the fuel assemblies, 217 wire-wrapped rods with diameter 0.663 cm (0.261 in) are arranged with \( P/D = 1.069 \). The blanket assemblies are composed of 127 wire-wrapped rods with a diameter of 0.986 cm (0.388 in). These assemblies have \( P/D = 1.036 \). The radial shield assemblies are composed of 61 bare rods with diameter 1.529 cm (0.602 in) and \( P/D = 1.004 \). The geometrical parameters for the assemblies are listed in Table 2.2.

Each pin shown in Figure 2.9 has a lower length of 176.5 cm (69.5 in) consisting primarily of shielding, gas plenum, and a lower axial blanket. The active core is above this length and is 101.6 cm (40 in) in height. Above this is an upper length of 177.8 cm (70 in) that
consists primarily of an upper axial blanket, gas plenum, and shielding. The total fuel pin length is 455.93 cm (179.5 in).

2.5.2 Primary Sodium Loop Natural Circulation Modeling

It is desired to evaluate the performance of the core under high power-to-flow conditions. Such conditions would occur if the primary sodium loop EM pump were not operating, due to a pump trip or due to a component failure. Under these conditions, the flow of sodium in the primary loop is controlled by the balance of the thermal head that develops and the resistive forces in the loop.

If CHANSOL is to be used to analyze the core performance, an inlet flowrate boundary condition is needed. Typically, the procedure to find the core inlet flowrate boundary condition is to run a system code that models the components in the primary, using course noding for the core components. Once the gross parameters that describe the thermal hydraulic behavior of the primary loop have been obtained (such as the loop flowrate, core-averaged outlet temperature, and core inlet temperature), the boundary conditions necessary for the analysis of the core are known. The next step would be to run a core code that could calculate individual assembly performance, or if necessary and economically feasible, the individual subchannel performance.

For the case considered here, the procedure for the determination of the core inlet flowrate is simplified. The primary loop friction pressure drop as a function of the primary sodium flow has already been calculated [H-3]. From these calculated values, a friction pressure drop versus flowrate curve was obtained here by least square fitting these calculated values, yielding
\[ \Delta p_f \text{(Pa)} = 2.026 \dot{m}_o \text{(kg/s)}^{1.557} \]  
\[ (2.70) \]

Figure 2.10 compares the friction pressure drop calculated in Ref. H-3 with Eq. (2.47). The agreement is excellent.

In order to calculate the primary loop flowrate, it is necessary to calculate the buoyancy head as a function of the primary loop flowrate. For quasi-steady flow in the loop, the loop flowrate can then be obtained by equating the loop friction pressure drop with the thermal head. For a loop with the heat sink thermal center a height \( L_T \) above the heat source thermal center, as such as in Figure 2.7, the thermal head developed is

\[ \Delta p_b = (\rho_c - \rho_h)qL_T \]  
\[ (2.71) \]

where the subscripts \( c \) and \( h \) refer to the fluid leaving the heat sink (the IHX) and the heat source (the core), respectively. Using the linear temperature-density relationship (Eq. (2.6)), Eq. (2.71) is rewritten as

\[ \Delta p_b = \rho_o \beta (t_c - t_h)qL_T \]  
\[ (2.72) \]

From an overall energy balance of the core, the temperature rise is

\[ t_h - t_c = \frac{q_T}{m_o c_p} \]  
\[ (2.73) \]

where

\[ q_T = \sum_{i=1}^{N} y_n q_n \]  
\[ (2.74) \]

so that Eq. (2.72) can be written as

\[ \Delta p_b = \frac{\rho_o \beta q T q_T}{m_o c_p} \]  
\[ (2.75) \]
Figure 2.10  PRISM primary loop friction pressure drop versus flowrate.

\[ \Delta p_f (\text{Pa}) = 2.026 m_0 (\text{kg/s})^{1.557} \]
For sodium between 316 and 649°C (600 and 1200°F), \( \beta = 2.773 \times 10^{-4}/^\circ C \) with \( t_o = 316^\circ C \) and \( \rho_o = 876.4 \text{ kg/m}^3 \), and \( c_p = 1270 \text{ J/kg} \cdot ^\circ C \). The PRISM reactor is arranged so that the thermal center of the IHX is 2.273 m above the thermal center of the core. Substituting these values in to Eq. (2.75) yields

\[
\Delta p_b (\text{Pa}) = \frac{4263 \ q_T (\text{MW})}{m_0 (\text{kg/s})}
\]  

(2.76)

For the primary sodium loop in quasi-steady natural circulation, the thermal head balances the friction pressure drop around the loop. Equating Eqs. (2.70) and (2.76) and solving for \( \dot{m}_o \), the natural circulation loop flowrate as a function of the decay power is found to be

\[
\dot{m}_o (\text{kg/s}) = 19.94 \ [q_T (\text{MW})]^{0.3911}
\]  

(2.77)

Substituting Eq. (2.77) into (2.73), the associated core temperature rise is found to be

\[
(t_h - t_c)(^\circ C) = 39.49 \ (q_T (\text{MW}))^{0.6089}
\]  

(2.78)

If the decay power curve is known, Eqs. (2.77) and (2.78) can be used to calculate the primary sodium flowrate and temperature rise, respectively, as a function of time during quasi-steady natural circulation of the primary loop.

2.5.3 Wire-Wrapped Bundle Friction Factors

The friction factors for each channel type modeled must be obtained in order to model the core fluid resistive forces. In the CHANSOL simulation of the PRISM core, channels are comprised of single assemblies. Therefore, bundle friction factors must be calculated for each channel type.
Cheng and Todreas [C-2] have proposed friction factor empirical equations for wire-wrapped bundles in the range $1.025 \leq P/D \leq 1.42$ and $8.0 \leq H/D \leq 50.0$ based on available data. In the laminar regime,

$$f = \frac{c_L}{Re} \quad ; \text{laminar}$$

(2.79)

In the turbulent regime, $Re > Re_T$, the friction factor is

$$f = \frac{c_T}{Re^{0.18}} \quad ; \text{turbulent}$$

(2.80)

In the transition regime, $Re_L < Re < Re_T$, the friction factor is

$$f = \left( \frac{c_T}{Re^{0.18}} \right)^{1/3} + \left( \frac{c_L}{Re} \right) (1 - \psi)^{1/3} \quad ; \text{transition}$$

(2.81)

where the intermittency factor $\psi$ is calculated from

$$\psi = \log_{10}(Re/Re_L)/(\log_{10}(Re_T/Re_L))$$

(2.82)

which is empirically found to be

$$\psi = (\log_{10}(Re) - (1.7 P/D + 0.78))/(2.52 - P/D)$$

(2.83)

The laminar constant $c_L$ is obtained from

$$c_L = (-974.6 + 1612.0(P/D) - 598.5(P/D)^2) \times (H/D)^{0.06-0.085(P/D)}$$

(2.84)

while the turbulent constant $c_T$ is calculated from

$$c_T = (P/D)^{9.7}(H/D)^{1.78-2.0(P/D)^2} \times \left[ 0.8063 - 0.9022 \log_{10}(H/D) + 0.3526(\log_{10}(H/D))^2 \right]$$

(2.85)

The upper end of the fully laminar regime is calculated from

$$\log_{10}(Re_L) = 1.7(P/D) + 0.78$$

(2.86)

and the lower end of the fully turbulent regime is calculated from

$$\log_{10}(Re_T) = 0.7(P/D) + 3.3$$

(2.87)
CHANSOL requires that the friction factor be of the form of Eq. (2.3), that is,
\[ f = \frac{c_n}{Re^b} \]  
(2.3)
It will be assumed (and later checked) that for primary loop natural circulation, the flow through the assemblies will be in the laminar regime for the fuel and blanket assemblies, and thus \( b = 1 \).

For the fuel assemblies, the resulting equation for the friction factor for flow in the laminar regime is
\[ f = \frac{58.07}{Re} \quad \text{for} \quad Re \leq 396 \]  
(2.88)
For the radial blanket assemblies, the corresponding equation for the laminar regime is
\[ f = \frac{49.70}{Re} \quad \text{for} \quad Re \leq 348 \]  
(2.89)

2.5.4 Local Form Loss Factor Due to Orificing

In liquid metal reactors, assemblies or groups of assemblies are orificed. The purpose of the orificing is to distribute the flow through the core to minimize outlet temperature differences between assemblies. The orifices are represented as local form losses in the CHANSOL model for the PRISM reactor. Table 2.3 shows the value of \( K \) used for each assembly. These values were calculated here using the flowrate and pressure drop data from Ref. M-1 for the orifice region of each assembly for full power operation using
\[ K = 2pA_pG^2 \]  
(2.90)

2.5.5 Core Decay Power

It will be assumed that the radial and axial power shape in the core after reactor shut down is the same as that before shutdown. Table 2.4 lists the assembly powers assumed before shutdown, from
Table 2.3

PRISM Assembly Flow Orificing by Location*

**Local Loss Coefficient K**

<table>
<thead>
<tr>
<th>Ring Number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>93.22(1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>93.22(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>94.85(3)</td>
<td>94.85(4)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>(NM)</td>
<td>94.85(5)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>93.06(6)</td>
<td>96.48(7)</td>
<td>96.48(8)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>209.6(9)</td>
<td>93.22(10)</td>
<td>93.22(11)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>259.3(12)</td>
<td>227.1(13)</td>
<td>216.6(14)</td>
<td>209.6(15)</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>(NM)</td>
<td>259.3(16)</td>
<td>259.3(17)</td>
<td>259.3(18)</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>-</td>
<td>(NM)</td>
<td>(NM)</td>
<td>(NM)</td>
<td>(NM)</td>
</tr>
</tbody>
</table>

Key: NM Not modeled in CHANSOL simulation

* CHANSOL numbering in parenthesis
### Table 2.4

*PRISM Assembly Power by Location* at Beginning of Cycle

<table>
<thead>
<tr>
<th>Ring Number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.642(1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>5.497(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>4.991(3)</td>
<td>5.208(4)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2.170(NM)</td>
<td>4.629(5)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>5.425(6)</td>
<td>4.051(7)</td>
<td>4.268(8)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.847(9)</td>
<td>4.774(10)</td>
<td>5.136(11)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.339(12)</td>
<td>0.550(13)</td>
<td>0.720(14)</td>
<td>0.762(15)</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0.020(NM)</td>
<td>0.212(16)</td>
<td>0.296(17)</td>
<td>0.339(18)</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>-</td>
<td>0.020(NM)</td>
<td>0.020(NM)</td>
<td>0.020(NM)</td>
<td>0.020(NM)</td>
</tr>
</tbody>
</table>

**Key:** NM  Not modeled in CHANSOL simulation

* CHANSOL numbering in parenthesis
Ref. M-1. Once the initial power shape is known, only the decay power fraction is then needed to calculate the power in each assembly in the core. The core decay power fraction was also obtained from Ref. M-1.

2.5.6 Flow Distribution Results

Table 2.5 lists the case studied here. Since the decay power fraction as a function of time was known, Eq. (2.54) could be used to calculate the primary loop flowrate for quasi-steady natural circulation. Since only fuel and radial blanket flow is considered for the core modeling, this flow is assumed to total 95.82% of the total loop flow, as discussed previously.

The channel dimensionless temperature rise is defined as the channel temperature rise divided by the core-averaged temperature rise. This temperature rise is plotted in Figure 2.11 for each channel for the mixed convection case, where density and viscosity variations are included, and the forced convection case, which neglects property variations. Both a high and low power-to-flow ratio case, corresponding to the 0 and $1 \times 10^6$ seconds cases from Table 2.5, are considered.

For the forced convection case, the higher power (fuel) assemblies have significantly higher outlet temperatures than the lower power (radial blanket) assemblies. When property variations are included, significant temperature-flattening results. The dimensionless temperature rise in the hottest assembly (number 1) is reduced from 1.6 to 1.1. The dimensionless temperature rise in the coldest assembly (number 12) is increased from 0.2 to 0.65. This is due to two effects. The first is that the hotter assemblies have a lower density and thus lower gravity pressure drop, and as a result
<table>
<thead>
<tr>
<th>Time(s)</th>
<th>Power Fraction</th>
<th>Flow Fraction*</th>
<th>Power-to-Flow Ratio**</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>6.87E-2</td>
<td>3.315E-2</td>
<td>2.07</td>
</tr>
<tr>
<td>1</td>
<td>5.95E-2</td>
<td>3.133E-2</td>
<td>1.90</td>
</tr>
<tr>
<td>10</td>
<td>4.76E-2</td>
<td>2.872E-2</td>
<td>1.66</td>
</tr>
<tr>
<td>1E2</td>
<td>3.34E-2</td>
<td>2.500E-2</td>
<td>1.34</td>
</tr>
<tr>
<td>1E3</td>
<td>2.08E-2</td>
<td>2.077E-2</td>
<td>1.00</td>
</tr>
<tr>
<td>1E4</td>
<td>9.60E-3</td>
<td>1.535E-2</td>
<td>0.63</td>
</tr>
<tr>
<td>1E5</td>
<td>5.43E-3</td>
<td>1.229E-2</td>
<td>0.44</td>
</tr>
<tr>
<td>1E6</td>
<td>2.56E-3</td>
<td>9.155E-3</td>
<td>0.28</td>
</tr>
</tbody>
</table>

* Based on an initial power of 425 MW and initial primary sodium loop flowrate of 2251 kg/s

** Based on an initial power-to-flow ratio of unity
Figure 2.11 Dimensionless temperature rise for the individual assemblies calculated by CHANSOL for the PRISM reactor during natural circulation of the primary loop.
draw more flow. This is referred to as buoyancy-induced flow redistribution. The second effect is that the hotter assemblies have lower viscosity since for ordinary liquids, viscosity decreases with temperature. This effect will also cause flow redistribution towards the hotter assemblies.

A channel mixed-to-forced convection flow ratio was defined as the channel flow rate when property variations are included in the analysis divided by the channel flow rate when constant fluid properties are used. If this ratio is greater than one for an assembly, the mixed convection flow redistribution increases the flow for that assembly. As can be seen by comparing Figures 2.11 and 2.12, the dimensionless temperature rise curves have the same qualitative shape as the mixed convection flow ratio curve that includes density variations. This signifies that the hottest channels in the core experience the largest flow increase due to mixed convection effects. The hottest channel (number 1) experiences more than a 40% increase in flow, while the coldest channel (number 12) has its flow reduced by 75%.

In Figures 2.11 and 2.12, little differences can be observed in the curves for the high and low power-to-flow cases. This indicates that for the core conditions expected during natural circulation of the primary loop, the dimensionless temperature rise for the channels is relatively insensitive to the inlet flow rate and the power-to-flow ratio.

It was explained above that both density and viscosity variations caused the mixed-convection flow redistribution. In order to estimate the significance of the viscosity variation, the cases were run using
Figure 2.12 Mixed convection flow redistribution for the individual assemblies calculated by CHANSOL for the PRISM reactor during natural circulation of the primary loop.
a very small value of $\beta$, thereby effectively neglecting density variations. Figure 2.11 shows that the mixed-to-forced convection flow ratio for the high and low power-to-flow cases when only viscosity variations are included is essentially unity for all channels. This indicates that viscosity variations account little for the flow redistribution and thus the redistribution is predominantly buoyancy-induced.

2.5.7 Calculation of the Upflow Preference for the PRISM Assemblies

In Section 2.3.5, a procedure was presented by which the upflow preference for each channel in a parallel system could be determined. The results of that section are now applied to the channels of the PRISM reactor.

For stable operation of a heated channel in downflow, the channel Reynolds number must exceed the Reynolds number $Re_c$ at which $\partial \Delta p^+/\partial Re = 0$. In Section 2.3.4, this point was found for channels in which form loss and $u_b/\rho$ variations are neglected. For the PRISM reactor, form loss cannot be neglected since the orificing region contributes significantly to the total pressure drop. The variations in viscosity will be neglected, though, since it was shown in Section 2.5.6 that it has negligible effect on the thermal hydraulic behavior of the system. For the PRISM core, Eq. (2.25b) therefore reduces to

$$\Delta p^+ = -X_{un} - (1 - S_n) \frac{\theta_{n}}{Re_n} - \delta_{n} Re - \gamma_n Re_n^2$$

(2.71)

which, when differentiating with respect to $Re_n$ yields

$$\frac{\partial \Delta p^+}{\partial Re} = (1 - S_n) \frac{\theta_{n}}{Re_n^2} - \delta_{n} - 2\gamma_n Re_n$$

(2.72)
The root of this equation is the critical Reynolds number \( \text{Re}_C \) below which heated downflow is unstable. Once \( \text{Re}_C \) is obtained, Eq. (2.71) is used to find \( \Delta p^+ \) and this result is then substituted in the equation for the upflow preference number, Eq. (2.51):

\[
H_n = \frac{1}{2} [-X_{un} - \Delta p^+]
\]  

(2.51)

Table 2.6 lists the upflow preference numbers calculated for the fuel and blanket channels in the PRISM core for 6.87% decay power. Channel number 1 has the strongest preference for upflow, while number 16 has the least. This analysis thus predicts channel number 16 as the one with the highest tendency to reverse to downflow. Figure 2.13 plots \( \Delta p^+ \) versus \( \text{Re}_n \) for these two channels. The mechanism by which a channel "jumps" from the upflow portion of the curve to the downflow portion will be qualitatively described in Chapter 8.
Table 2.6

Channel Upflow Preference Number for the PRISM Reactor Core by Location

\[ H_n = \frac{1}{2} \left[ -X_{\text{un}} - \Delta p^+ \right] \]

<table>
<thead>
<tr>
<th>Row Number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.510E-2(1)\textsuperscript{a}</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3.453E-2(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3.264E-2(3)</td>
<td>3.352E-2(4)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>-</td>
<td>(NM)</td>
<td>3.113E-2(5)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>3.422E-2(6)</td>
<td>2.876E-2(7)</td>
<td></td>
<td>2.972E-2(8)</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1.623E-2(9)</td>
<td>3.160E-2(10)</td>
<td></td>
<td>3.308E-2(11)</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>9.674E-3(12)</td>
<td>1.265E-2(13)</td>
<td></td>
<td>1.478E-2(14)</td>
<td>1.519E-2(15)</td>
</tr>
<tr>
<td>8</td>
<td>(NM)</td>
<td>7.428E-3(16)\textsuperscript{b}</td>
<td></td>
<td>8.898E-3(17)</td>
<td>9.674E-3(18)</td>
</tr>
<tr>
<td>9</td>
<td>-</td>
<td>-</td>
<td>(NM)</td>
<td>-</td>
<td>(NM)</td>
</tr>
</tbody>
</table>

Key: NM Not modeled in CHANSOL simulation

* CHANSOL numbering in parenthesis

\textsuperscript{a} Channel with highest upflow preference

\textsuperscript{b} Channel with lowest upflow preference
Figure 2.13 Dimensionless pressure drop versus flowrate curves for the PRISM core channels with highest and lowest upflow preference.
CHAPTER THREE
LAMINAR MIXED CONVECTION FLOW IN A DUCT

3.1 Introduction

In order to accurately predict the behavior of parallel channels in mixed convection, it is desirable to first study the behavior of single, isolated channels in the mixed convection regime. The motivation behind the work in this chapter is to obtain the engineering parameters, i.e., friction factors and heat transfer coefficients, that describe the thermal hydraulic behavior of the individual channels in a system of parallel channels. A distributed parameter analysis must be performed in order to calculate these parameters. Once the parameters are obtained, they can be input into computer codes that model the channels using one-dimensional lumped parameter methods (such as MICON).

A number of workers have solved for the laminar fully-developed velocity profile of mixed convection flow in a vertical tube [B-1, H-2, H-4, M-2, O-1]. Hallman [H-1] obtained extensive heat transfer data for water in a heated vertical tube for both upflow and downflow in the developing and fully-developed regimes. Kemeny and Somers [K-1] also obtained heat transfer data in a heated vertical tube using water and oil as the test fluids. In addition, they obtained pressure drop data which indicated that as the power-to-flow ratio increased, the pressure drop increased, and accounted for this increase by defining a mixed convection friction factor ratio, \( f/f_0 \), which increased with \( Grq/Re \). Bishop, Willis, and Markley [B-1] obtained an analytical expression for \( f/f_0 \) for fully-developed laminar mixed convection flow in a circular tube for heated upflow. Heated downflow was not included.
Sherwin [S-2] has similarly presented an analysis of fully-developed laminar mixed convection flow through an annular duct with its inner radius heated. Velocity profiles, temperature profiles, and Nusselt numbers were presented for an outer radius to inner radius ratio of three. The limiting Nusselt number obtained as \(Gr_v/Re\) approached zero was consistent with the forced convection analysis of Lundberg, McCuen, and Reynolds [L-1] for this radius ratio. Sherwin also obtained heat transfer data for downflow in the heated annulus, which did not match the trends of the analysis due to buoyancy-induced turbulence. Maitra and Sabba Raju [M-3] extended Sherwin's analysis to higher \(Gr_v/Re\) values, although the value of their analytical effort is questionable since for this range flow reversal in some regions of the channel is predicted, which violates the assumptions of the original analysis. They do however, present data for heated upflow through an annular channel, and the data does follow the trend of the analysis, that being an increase in the Nusselt number as heating is increased. An analytical prediction of the friction factor ratio was not made.

predicted the mixed convection pressure drop in the wire-wrapped bundles and concluded that the increase in pressure drop with $Gr_q/Re$ was due to inter-subchannel flow redistribution within the bundle.

In this chapter, the conservation equations for developing mixed convection flow, and the corresponding governing parameters, are formulated. The developing flow equations are then simplified to their fully-developed form. General expressions are presented for the velocity profile, temperature profile, friction factor, and Nusselt number. Two types of friction factors are presented. The first type is based on the standard definition of friction factor, and requires that the spatially-averaged density be known if it is to be used in lumped parameter analyses to calculate the axial pressure gradient. The second type of friction factor that is presented is modified to allow the calculation of the axial pressure gradient based on the bulk (or mixing-cup) temperature, which is the only temperature available in lumped parameter analyses. It is referred to as the modified friction factor.

The conservation equations are first solved for geometries that exhibit azimuthal symmetry. These include the circular tube and the annulus with its inner radius heated, which have already been presented in the literature. The new results are for the annulus with outer radius heated, and the equivalent annulus. The equivalent annulus in forced convection has been solved by Sparrow, Loeffler and Hubbard [5-3,5-4] because it can be used to approximate infinite rod arrays for large $P/D$ ratios. The circular tube is presented as a limiting case of the annulus with its outer radius heated. Friction factors, modified friction factors, and Nusselt
numbers are presented for all the geometries. The analytical results are then compared with experimental data, where available. Mixed convection pressure drop data in finite rod bundles is also analyzed. Data from two separate experiments utilizing water coolant in wire-wrapped triangular array bundles show poor agreement with the equivalent annulus model. The data was predicted, however, by the ENERGY-IV code, which simulated global flow redistribution within the bundle.

3.2 Non-Dimensional Equations and Parameters for Developing Laminar Mixed Convection Flow in a Heated Vertical Duct

In this section the governing non-dimensional equations and parameters for steady, developing mixed convection flow in a heated vertical duct will be derived, based on the general differential conservation equations of mass, momentum, and energy. It is assumed that the flow is laminar with no internal heat generation. Also, since the flows considered are predominantly axial, radial pressure drops will be neglected. Finally, we assume that the fluid specific heat, viscosity, and thermal conductivity do not vary. Variations of the density will also be ignored, except for the gravity body force term of the momentum equation, where the density is assumed to be of the form

\[ \rho = \rho_0 (1 - \beta (t - t_0)) \]  

(3.1)

where \( \rho_0 \) and \( t_0 \) are the reference density and temperature, respectively. This treatment of fluid properties is often referred to as the Boussinesq approximation often made in the analysis of buoyancy-induced phenomena.
Momentum Equation

The axial momentum equation can be written as

\[ \rho_o \frac{Du_z}{Dt} = - \frac{dp}{dz} \pm \rho_o \beta(t - t_o) + \nu \nabla^2 u_z \quad (3.2) \]

where

\[ \hat{p} = p + \rho_o g z \quad (3.3) \]

where the material derivative is defined by

\[ \frac{D}{Dt} = \frac{\partial}{\partial t} + (u \cdot \nabla) \quad (3.4) \]

In Eq. (3.2), the \((\pm)\) operator denotes the \((+)\) operator aiding flow and the \((-)\) operator for opposing flow. This convention will be followed for the remainder of this work. Similarly, the \((\mp)\) operator will denote the \((-)\) and \((+)\) operators for aiding and opposing flows, respectively.

Using the steady flow assumption and Eq. (3.4), Eq. (3.2) can be rewritten as

\[ \rho_o \hat{u} \cdot \hat{\nabla} u_z = - \frac{dp}{dz} \pm \rho_o \beta(t - t_o) + \nu \nabla^2 u_z \quad (3.5) \]

In order to non-dimensionalize this equation, the following parameters are introduced:

\[ u_o = \frac{u}{\rho_o A_f} \quad (3.6) \]

\[ \hat{U} = \frac{\hat{u}}{u_o} \quad (3.7) \]

\[ \hat{p} = \frac{p}{\rho_o u_o^2} \quad (3.8) \]

\[ \hat{q}'' = \frac{q}{\rho_o L} \quad (3.9) \]

\[ T = \frac{(t - t_o)k}{\hat{q}'' D_e} \quad (3.10) \]
\[ \nabla^* = D_e \nabla \]  
(3.11)

The length scales are non-dimensionalized by \( D_e \), that is

\[ Z = \frac{z}{D_e} \]  
(3.12a)

\[ R = \frac{r}{D_e} \]  
(3.12b)

Substituting these groups into Eq. (3.5) yields

\[ \text{Re}(\hat{U} \cdot \hat{V}^* U_z) = -\text{Re} \frac{dP}{dz} + \frac{Gr^2}{\text{Re} T} + \nabla^2 U_z \]  
(3.13)

where the following non-dimensional groups have been introduced:

Reynolds number: \( \text{Re} = \frac{\rho_o u_o D_e}{\mu} = \frac{\text{Inertial force}}{\text{viscous force}} \)  
(3.14)

Grashof-q number: \( Gr_q = \frac{\rho_o^2 q^4 D_e^4 q^n}{k u'^2} = \frac{(\text{buoyancy force})(\text{inertial force})}{(\text{viscous force})^2} \)  
(3.15)

Prandtl number: \( \text{Pr} = \frac{c_p u}{k} = \frac{\text{(rate of momentum diffusion)}}{\text{(rate of heat diffusion)}} \)  
(3.16)

**Energy Equation**

For the assumptions listed above, one can write the energy equation as

\[ \rho_o c_p \frac{D\theta}{Dt} = k \nabla^2 \theta + \mu \phi \]  
(3.17)

where \( \mu \phi \) represents viscous dissipation in the fluid. Again using the steady flow assumption and Eq. (3.4), we can write Eq. (3.17) as

\[ \rho_o c_p (U \cdot \nabla \theta) = k \nabla^2 \theta \]  
(3.18)

where viscous dissipation has been neglected. Combining Eq. (3.18) with (3.6) through (3.12) and (3.14) through (3.16) produces

\[ \text{RePr}(\hat{U} \cdot \nabla^* \theta) = \nabla^2 \theta \]  
(3.19)
Equations (3.13) and (3.19) are the dimensionless forms of the momentum and energy equations. This analysis indicates that for incompressible developing flow in a duct in the mixed convection regime, the dominant dimensionless groups that must be matched for similarity are the following:

Re, Grq/Re, Pe, and boundary conditions

where Pe is defined by

Peclet number: \( Pe = \frac{RePr}{k} \) \( \frac{\rho_o u_o D e p}{k} \) \hspace{1cm} (3.20)

Since the flow is developing, engineering parameters such as the friction factor and heat transfer coefficient are also a function of Z.

The continuity equation need not be considered when deriving the governing dimensionless parameters since in its steady, differential form for an incompressible fluid,

\[ \nabla \cdot \mathbf{u} = 0 \] \hspace{1cm} (3.21)

and thus it is invariant to non-dimensionalizing, i.e., its non-dimensional form is

\[ \hat{\nabla}^* \hat{\mathbf{u}} = 0 \] \hspace{1cm} (3.22)

No new non-dimensional parameters are present.

3.3 Non-Dimensional Equations and Parameters for Fully-Developed Laminar Mixed Convection Flow in a Heated Vertical Duct

3.3.1 Velocity Profile

In the last section, the general conservation equations for developing mixed convection flow in a vertical duct were presented. In general, these equations are difficult to solve analytically due to the non-linear nature of the terms that contain the convective derivative \( \mathbf{u} \cdot \hat{\nabla}^* \). We instead seek a solution to the fully-developed flow form of these equations, in which case the
terms containing the non-linear convective terms vanish since $\partial U_z / \partial Z = U_r = U_\theta = 0$. The fully-developed forms of the non-dimensional momentum and energy equations are written as

$$0 = -\text{Re} \frac{dP}{dZ} \pm \frac{Gr}{\text{Re}} q T + \nabla^* U_z$$  \hspace{1cm} (3.23)

and

$$\text{RePr} U_z \frac{\partial T}{\partial Z} = \nabla^* T$$  \hspace{1cm} (3.24)

The left hand side of Equation (3-24) might look non-linear because a velocity ($U_z$) is multiplied by the axial temperature gradient ($\partial T / \partial Z$). Actually, for fully-developed flow, the axial gradient of the temperature field ($\partial T / \partial Z$) must be constant across the cross section of the duct. It is possible to obtain an expression for this axial temperature gradient in terms of the average heat flux, $\bar{q}''$. First, an overall-energy balance shows that

$$\frac{1}{mc} \frac{\partial \bar{q}''}{\partial Z} = \bar{q}'' P_h$$  \hspace{1cm} (3.25)

After non-dimensionalizing, one can solve for $\partial T / \partial Z$, yielding

$$\frac{\partial T}{\partial Z} = \frac{4}{\text{RePr}} \frac{P_h}{P_w}$$  \hspace{1cm} (3.26)

Substituting Eq. (3-26) into (3-24) yields

$$4U_z \frac{P_h}{P_w} = \nabla^* T$$  \hspace{1cm} (3.27)

For the rest of this chapter, the subscript $z$ will be implied when referring to the axial velocity since for fully-developed flow, this is the only velocity. In addition, the gradient operator symbol $\nabla$ will be used to denote the non-dimensional operator $\nabla^*$ when operating on dimensionless variables.
The dimensionless Laplacian operator \( \nabla^2 \) can now be applied to Eq. (3.23), which eliminates the first term on the right hand side of the equation. Combining this result with Eq. (3.27), \( \zeta \) can be eliminated, resulting in

\[
0 = \pm \frac{\text{Gr}_q}{\text{Re}} \frac{P_h}{P_w} U + \nabla^4 U \tag{3.28}
\]

This result indicates that for incompressible fully-developed mixed convection flow, similarity is achieved when the \( \text{Gr}_q/\text{Re} \) parameter and boundary conditions are matched. The dimensionless velocity \( U \) is thus a function of these parameters.

The general solution (in polar co-ordinates) to Eq. (3.28) for aiding flow can be written as

\[
U = \sum_{\nu=0}^{\infty} \left[ a_{\nu} \text{ber}_\nu(\eta R) + b_{\nu} \text{bei}_\nu(\eta R) + c_{\nu} \text{ker}_\nu(\eta R) + d_{\nu} \text{kei}_\nu(\eta R) \right] \left[ \cos(\nu \theta) + f \sin(\nu \theta) \right]
\tag{3.29}
\]

and for opposing flow as

\[
U = \sum_{\nu=0}^{\infty} \left[ a_{\nu} J_\nu(\eta R) + b_{\nu} Y_\nu(\eta R) + c_{\nu} I_\nu(\eta R) + d_{\nu} K_\nu(\eta R) \right] \left[ \cos(\nu \theta) + f \sin(\nu \theta) \right]
\tag{3.30}
\]

where the parameter \( \eta \) is defined by

\[
\eta = \frac{\text{Gr}_q}{\text{Re}} \frac{P_h}{P_w} \tag{3.31}
\]

It will be assumed that the boundaries \( \theta=0 \) and \( \theta=\pi/s \) are lines of symmetry for the geometry of interest. It can therefore be said that

\[
\frac{\partial U}{\partial \theta} \bigg|_{\theta=0} = \frac{\partial U}{\partial \theta} \bigg|_{\theta=\pi/s} = 0 \tag{3.32}
\]
Applying Eq. (3.32) to (3.29) and (3.30) yields $f=0$ and specifies possible values of $\nu$ as

$$\nu = n_s \quad n=0,1,2...$$  \hspace{1cm} (3.33)

so that Eqs. (3.29) and (3.30) can be written as

$$U = \sum_{n=0}^{\infty} \left[ a_{ns} \text{ber}(n_R) + b_{ns} \text{hel}(n_R) + c_{ns} \text{ker}(n_R) + d_{ns} \text{kel}(n_R) \right] \cos(n_s \theta) \right]$$  \hspace{1cm} (3.34)

for aiding flow, and

$$U = \sum_{n=0}^{\infty} \left[ a_{ns} J(n_R) + b_{ns} Y(n_R) + c_{ns} I(n_R) + d_{ns} K(n_R) \right] \cos(n_s \theta) \right]$$  \hspace{1cm} (3.35)

for opposing flow. (Note that the constant $e$ has been incorporated into the constants $a_{ns}$, $b_{ns}$, $c_{ns}$, and $d_{ns}$.)

The four sets of constants are evaluated from the velocity and heat flux boundary conditions. The heat flux boundary conditions are specified by prescribing the heat flux shape along the boundary.

The dimensionless heat flux $Q$ is related to the dimensionless temperature by

$$Q = -\frac{k\nabla T}{\dot{q}''} = -\nabla T$$  \hspace{1cm} (3.36)

From Eq. (3.36), it can be seen that the dimensionless heat flux $Q$ is the ratio of the local heat flux to the average heat flux along the heated perimeter. By applying the gradient operator to Eq. (3.23) and rearranging, one obtains

$$\pm \frac{\nabla^3 U}{Gr q / Re} = -\nabla T$$  \hspace{1cm} (3.37)

since the axial pressure gradient is constant across the duct, and thus

$$Q = \pm \frac{\nabla^3 U}{Gr q / Re}$$  \hspace{1cm} (3.38)
In order to solve for the velocity profile, the velocity and heat flux boundary conditions must be specified. The velocity boundary conditions are satisfied by specifying either the velocity or its gradient along the perimeter. The heat flux boundary condition is satisfied by specifying $Q$ along the perimeter of the region. Equation (3.38) relates $Q$ to $\nabla^3 U$. The velocity and heat flux boundary conditions thus reduce to specifying either $U$ or $\nabla U$, and $\nabla^3 U$ along the boundary of the region.

3.3.2 Friction Factor Calculation

In the last section it was shown that the mixed convection velocity profile in the duct is a function of the $Gr_q/Re$ parameter, boundary conditions, and geometry. Since the friction factor is directly related to the shear at the fluid-solid interface which results from a velocity gradient normal to the surface, one would expect mixed convection effects to alter the friction factor from its forced convection value. In this section it will be shown how the friction factor is calculated once the velocity profile is known.

A simple force balance shows that

$$\left[-\frac{dp}{dz}\right]_f = \frac{1}{A_f} \int_P \tau_w dp_w$$

(3.39)

where $\left[-\frac{dp}{dz}\right]_f$ is the pressure gradient due to skin friction, and $\tau_w$ is the fluid shear at the duct wall. For a Newtonian fluid

$$\tau_w = \mu \frac{\partial u}{\partial n}$$

(3.40)

where the normal derivative is defined by

$$\frac{\partial}{\partial n} = \cos\phi \frac{\partial}{\partial r} - \sin\phi \frac{1}{r} \frac{\partial}{\partial \theta}$$

(3.41)

and $\phi$ is the angle between the surface normal and the $\theta$-direction, as shown in Figure 3.1. (For a radial surface, $\phi = 0$.)
\[ \frac{\partial}{\partial n} = \cos \phi \frac{\partial}{\partial r} - \sin \phi \frac{1}{r} \frac{\partial}{\partial \theta} \]

Figure 3.1 Definition of the normal derivative.
Substituting Eq. (3.40) into Eq. (3.39) yields

\[
[- \frac{dp}{dz}]_{f} = \frac{\mu}{A_f} \int \frac{\partial u}{\partial n} \, dp_w
\]

(3.42)

The Darcy friction factor is defined by

\[
[- \frac{dp}{dz}]_{f} = \frac{f}{D_e} \frac{\rho_u u_0^2}{2}
\]

(3.43)

Equating Eqs. (3.42) and (3.43), non-dimensionalizing, and solving for \(fRe\) yields

\[
fRe = \frac{8}{\rho_w} \int \frac{\partial U}{\partial n} \, dp_w
\]

(3.44)

where the non-dimensional normal derivative is defined by

\[
\frac{\partial}{\partial n} = \cos \phi \frac{\partial}{\partial R} - \sin \phi \frac{1}{R} \frac{\partial}{\partial \theta}
\]

(3.45)

From Eq. (3.44) it can be seen that the \(fRe\) parameter is a function of the velocity profile, which is itself a function of the \(Grq/Re\) parameter, geometry, and boundary conditions. For mixed convection flow, it is convenient to define a mixed convection friction factor ratio, \((f/f_0)\), where \(f_0\) is the forced convection friction factor for a given Reynolds number. For fully-developed laminar flow,

\[
f_0 Re = \text{constant}
\]

(3.46)

so that the friction factor ratio can be evaluated from

\[
(f/f_0) = \frac{fRe}{f_0 Re}
\]

(3.47)

As the \(Grq/Re\) parameter approaches zero, \((f/f_0)\) approaches one.
3.3.3 Temperature Field

The expressions needed to calculate the mixed convection friction factor were obtained without knowledge of the temperature field. This is because friction pressure drop is caused by fluid shear at the wall. Once the velocity profile in the channel is known, we can solve for the temperature field by solving Eq. (3.23) for \( T \). The one problem is that the \(- \frac{dp}{dz}\) Re term in Eq. (3.23) is still an undetermined constant. This constant will be evaluated by relating it to the \( f \) Re parameter.

First, \(- \frac{dp}{dz}\) is expanded into

\[
- \frac{dp}{dz} = \frac{D}{\rho_o u_0} \left[ \frac{dp}{dz} \right]_q + \rho_0 q
\]

by using Eqs. (3.3), (3.8), and (3.12).

It can be easily shown that if \( \rho_0 \) is assumed to be the spatially-averaged density in the cross-section of the duct, then \( \rho_0 q \) is the axial pressure gradient due to the gravity body force. The expression in the brackets must therefore be the axial pressure gradient due to friction, \( \left[ - \frac{dp}{dz} \right]_f \), since

\[
\left[ - \frac{dp}{dz} \right] = \left[ - \frac{dp}{dz} \right]_q + \left[ - \frac{dp}{dz} \right]_f
\]

Equation (3.48) can be written as

\[
- \frac{dp}{dz} = \frac{D}{\rho_o u_0} \left[ - \frac{dp}{dz} \right]_f
\]

Solving Eq. (3.43) for \( \left[ - \frac{dp}{dz} \right]_f \) and then substituting that expression into Eq. (3.50), it is found that

\[
- \frac{dp}{dz} \text{ Re} = \frac{f \text{ Re}}{2}
\]
Substituting the previously derived equation for fRe (Eq. (3.44)) into Eq. (3.51) results in

$$- \frac{dP}{dz} \text{Re} = \frac{4}{P} \int \frac{\partial U}{\partial N} \frac{dP}{P} w w$$

(3.52)

The temperature field is then given by

$$T = \frac{-dP}{dz} \text{Re} + \varphi^2 U \frac{Gr_q}{Re}$$

(3.53)

Eq. (3.51) is valid only if \( \rho_0 \) (and thus \( t_0 \)) is assumed to be a spatially-averaged fluid property. If another reference density and temperature are chosen, Eq. (3.51) will give erroneous results. Also, since local variations in density and temperature from the reference (spatially-averaged) values must cancel when integrated over the cross-section of the duct, one easily proves that

$$\iint_T dA_f = 0$$

(3.54)

when Eqs. (3.51) and (3.53) are used to calculate \( T \).

If one is to calculate the axial pressure gradient for a particular application, it is necessary to calculate the correct pressure gradient due to gravity, especially if the operating flow regime is mixed convection. Typically the gravity pressure gradient is calculated assuming the bulk (or mixing-cup) temperature and density are equal to the spatially-averaged temperature and density, since the bulk temperature can be calculated from lumped parameter analysis without knowledge of the details of the temperature field. As the \( Gr_q/Re \) parameter increases, the error associated with this approximation also increases.
The bulk temperature for axial flow is defined as

\[ t_b = \frac{1}{u_0 A_f} \int \int u r dr d\theta \]  

Substituting \( U, T \) and \( R \) for \( u, t \) and \( r \) results in

\[ T_b = \frac{e}{A_f} \int \int UTR dr d\theta \]  \hspace{1cm} (3.56)

Once \( U \) and \( T \) are known, Eq. (3.56) can be evaluated numerically.

3.3.4 Modified Friction Factor

Now that an expression for the dimensionless bulk temperature, \( T_b \), has been obtained, it is desirable to find an easy way to

"correct" for using the bulk density to compute the gravity term in

the momentum equation. With a little hindsight, the gravity term in

Eq. (3.49) is expanded so that

\[ \left[ - \frac{dp}{dz} \right] = \pm \rho_b q \pm (\rho_o - \rho_b)g + \left[ - \frac{dp}{dz} \right]_f \]  \hspace{1cm} (3.57)

and a solution for \( F \) is sought such that

\[ \left[ - \frac{dp}{dz} \right] = \pm \rho_b q + \left[ - \frac{dp}{dz} \right]_f (1 + F) \]  \hspace{1cm} (3.58a)

where

\[ F = \pm \frac{(\rho_o - \rho_b)q}{\left[ - \frac{dp}{dz} \right]_f} \]  \hspace{1cm} (3.58b)

\( F \) is a correction to the momentum equation that accounts for using

\( \rho_b \) instead of \( \rho_o \) to calculate the gravity term. It would seem

more natural to correct the gravity term instead of the friction term

since it is actually the gravity term that is computed incorrectly.

The following mathematics, however, dictate the correction as

presented in Eq. (3.58). By equating Eqs. (3.57) and (3.58a) and

solving the resulting equation simultaneously with Eqs. (3.1) and

(3.43), one obtains
\[ F = \pm \frac{Gr q \eta_T}{Re \frac{T_b}{fRe}} \]  

(3.59)

Thus \( F \) can easily be calculated from previously determined non-dimensional parameters when defined by Eq. (3.58b).

It was shown previously that the factor \( (f/f_0) \) corrects the friction factor, and hence the friction pressure gradient through Eq. (3.43), for mixed convection effects. If the gravity pressure gradient was calculated using the bulk density (as is typically done in lumped parameter codes), the friction pressure gradient needs to be further corrected by a factor of \((1+F)\). The quantity \((1+F)(f/f_0)\) therefore represents a mixed convection modified friction factor ratio to be used for these cases.

3.3.5 Nusselt Number

Once the temperature distribution is known, the Nusselt number can easily be calculated. By definition,

\[ Nu = \frac{-q''}{\bar{T}_w - T_b} \frac{D_e}{k} \]  

(3.60)

where \( \bar{T}_w \) is the average temperature of the heated surface, and can be calculated from

\[ \bar{T}_w = \frac{1}{P_h} \int_{P_h}^{P_h} T \, d\theta \]  

(3.61)

Equations (3.60) and (3.61) can then be non-dimensionalized to

\[ Nu = \frac{1}{\bar{T}_w - T_b} \]  

(3.62)

and

\[ \bar{T}_w = \frac{D_e}{P_h} \int_{P_h}^{P_h} T \, d\theta \]  

(3.63)
In geometries where more than one wall is heated, one can use Eqs. (3.62) and (3.63) to define a different Nusselt number for each wall. It must be noted, however, that in situations where more than one wall is heated, the Nusselt number obtained for each heated surface will depend upon the heat fluxes on the other surfaces, too. This phenomenon is not unique to mixed convection, but occurs also for forced convection [L-1].

Now that the generalized conservation equations have been presented, one can proceed to solve the equations for specific geometries. For those geometries considered in this chapter, azimuthal symmetry is assumed, thereby simplifying the analysis.

3.4 Solution for Fully-Developed Laminar Mixed Convection Flow in a Heated Vertical Duct for Geometries with Azimuthal Symmetry

3.4.1 Velocity Profile

Four geometries are investigated in this section and each geometry for aiding and opposing flow. For the first geometry, the duct is assumed to be concentrically annular in shape. The heat flux is imposed at the inner radius and an adiabatic outer radius is assumed. For the second geometry, the duct is annular with an adiabatic inner radius and heat flux imposed at the outer radius. The third geometry is the circular duct, with the heat flux imposed at the wall. The fourth geometry is called the equivalent annulus because like the annulus, it has an inner and outer radius. The difference is that instead of a no-slip velocity boundary condition imposed at the outer radius, a zero shear condition is assumed. The equivalent annulus can be used as an approximation to rod array geometries with large spacing, as will be discussed later. For each
case the parameter $\alpha$ is defined as the inner radius to outer radius ratio ($\alpha=0$ for the circular channel). These geometries are shown in Figure 3.2.

For geometries with azimuthal symmetry, Eq. (3.29) reduces to

$$U = a_0 \text{ber}_0(\eta R) + b_0 \text{bei}_0(\eta R) + c_0 \text{ker}_0(\eta R) + d_0 \text{kei}_0(\eta R)$$

(3.64)

(aiding flow)

and Eq. (3.30) reduces to

$$U = a_0 \text{J}_0(\eta R) + b_0 \text{Y}_0(\eta R) + c_0 \text{I}_0(\eta R) + d_0 \text{K}_0(\eta R)$$

(3.65)

(opposing flow)

Table 3.1 summarizes the boundary conditions for the four axially symmetric cases for aiding and opposing flow. The boundary conditions are imposed at $R_1$ and $R_0$, corresponding to the inner and outer radii of the geometries.

For the circular tube, $R_1 = 0$ and $R_0 = 0.5$. A problem arises at the inner radius (the center of the tube), however, because the functions $\text{ker}_0$, $\text{kei}_0$, $\text{Y}_0$, and $\text{K}_0$ are not defined at zero. To insure a velocity solution that is finite at the center of the circular tube, the constants $c_0$ and $d_0$ are set to zero for aiding flow, and $b_0$ and $d_0$ to zero for opposing flow. The boundary conditions are then specified only at the outer radius.

Differentiating (in polar coordinates) Eqs. (3.64) and (3.65), one obtains for aiding flow:

$$\nabla U = \nabla^2 U = \eta^2 [a_0 \text{ber}_0''(\eta R) + b_0 \text{bei}_0''(\eta R) + c_0 \text{ker}_0''(\eta R) + d_0 \text{kei}_0''(\eta R)]$$

(3.66)

$$\nabla^2 U = \eta^2 [-a_0 \text{ber}_0''(\eta R) + b_0 \text{ber}_0''(\eta R) - c_0 \text{kei}_0' (\eta R) + d_0 \text{kei}_0' (\eta R)]$$

(3.67)

$$\nabla^3 U = \eta^3 [-a_0 \text{ber}_0'''(\eta R) + b_0 \text{ber}_0'''(\eta R) - c_0 \text{kei}_0'' (\eta R) + d_0 \text{kei}_0'' (\eta R)]$$

(3.68)
Figure 3.2 Geometries with azimuthal symmetry.
Table 3.1

Boundary Conditions for Geometries with Azimuthal Symmetry

<table>
<thead>
<tr>
<th>Geometry</th>
<th>Aiding</th>
<th>Opposing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inner Radius</td>
<td>$U(R_1) = 0$</td>
<td>$U(R_1) = 0$</td>
</tr>
<tr>
<td></td>
<td>$U(R_0) = 0$</td>
<td>$U(R_0) = 0$</td>
</tr>
<tr>
<td>Heated Annulus</td>
<td>$\nabla^3 U(R_1) = Gr_q / Re$</td>
<td>$\nabla^3 U(R_1) = -Gr_q / Re$</td>
</tr>
<tr>
<td></td>
<td>$\nabla^3 U(R_0) = 0$</td>
<td>$\nabla^3 U(R_0) = 0$</td>
</tr>
<tr>
<td>Outer Radius</td>
<td>$U(R_1) = 0$</td>
<td>$U(R_1) = 0$</td>
</tr>
<tr>
<td></td>
<td>$U(R_0) = 0$</td>
<td>$U(R_0) = 0$</td>
</tr>
<tr>
<td>Heated Annulus</td>
<td>$\nabla^3 U(R_1) = 0$</td>
<td>$\nabla^3 U(R_1) = 0$</td>
</tr>
<tr>
<td></td>
<td>$\nabla^3 U(R_0) = -Gr_q / Re$</td>
<td>$\nabla^3 U(R_0) = Gr_q / Re$</td>
</tr>
<tr>
<td>Circle</td>
<td>$U(R_0) = 0$</td>
<td>$U(R_0) = 0$</td>
</tr>
<tr>
<td></td>
<td>$\nabla^3 U(R_0) = -Gr_q / Re$</td>
<td>$\nabla^3 U(R_0) = Gr_q / Re$</td>
</tr>
<tr>
<td>Equivalent</td>
<td>$U(R_1) = 0$</td>
<td>$U(R_1) = 0$</td>
</tr>
<tr>
<td></td>
<td>$\nabla U(R_0) = 0$</td>
<td>$\nabla U(R_0) = 0$</td>
</tr>
<tr>
<td>Annulus</td>
<td>$\nabla^3 U(R_1) = Gr_q / Re$</td>
<td>$\nabla^3 U(R_1) = -Gr_q / Re$</td>
</tr>
<tr>
<td></td>
<td>$\nabla^3 U(R_0) = 0$</td>
<td>$\nabla^3 U(R_0) = 0$</td>
</tr>
</tbody>
</table>
and for opposing flow:

\[
\nabla U = \eta [-a_0 J_1(\eta R) - b_0 Y_1(\eta R) + c_0 I_1(\eta R) - d_0 K_1(\eta R)]
\]

(3.69)

\[
\nabla^2 U = \eta^2 [-a_0 J_0(\eta R) - b_0 Y_0(\eta R) + c_0 I_0(\eta R) + d_0 K_0(\eta R)]
\]

(3.70)

\[
\nabla^3 U = \eta^3 [a_0 J_1(\eta R) + b_0 Y_1(\eta R) + c_0 I_1(\eta R) - d_0 K_1(\eta R)]
\]

(3.71)

The solutions for the velocity profiles are obtained by computing \(a_0, b_0, c_0,\) and \(d_0\) from the four boundary conditions listed in Table 3.1, for each case. These constants were computed here using the ANNA and ANNO codes described in Appendix B. In general, a closed form solution cannot easily be written. The solution for the circle can easily be written, though, since there are only two equations that need to be solved. For aiding flow, \(c_0 = d_0 = 0,\) and

\[
a_0 = \frac{\sqrt{2}}{4} \frac{Gr q}{Re} \frac{\text{bei}_0(\frac{\eta}{2})}{\text{ber}_0(\frac{\eta}{2})\text{ber}_0'(\frac{\eta}{2}) + \text{bei}_0(\frac{\eta}{2})\text{bei}_0'(\frac{\eta}{2})}
\]

(3.72)

\[
b_0 = -\frac{\sqrt{2}}{4} \frac{Gr q}{Re} \frac{-\text{ber}_0(\frac{\eta}{2})}{\text{ber}_0(\frac{\eta}{2})\text{ber}_0'(\frac{\eta}{2}) + \text{bei}_0(\frac{\eta}{2})\text{bei}_0'(\frac{\eta}{2})}
\]

(3.73)

and for opposing flow, \(b_0 = d_0 = 0,\) and

\[
a_0 = -\frac{\sqrt{2}}{4} \frac{Gr q}{Re} \frac{I_0(\frac{\eta}{2})}{J_0(\frac{\eta}{2})I_1(\frac{\eta}{2}) - I_0(\frac{\eta}{2})J_1(\frac{\eta}{2})}
\]

(3.74)

\[
c_0 = \frac{\sqrt{2}}{4} \frac{Gr q}{Re} \frac{J_0(\frac{\eta}{2})}{J_0(\frac{\eta}{2})I_1(\frac{\eta}{2}) - I_0(\frac{\eta}{2})J_1(\frac{\eta}{2})}
\]

(3.75)

As examples of the results, Figures 3.3 through 3.6 show the velocity profile for the four geometries (with \(\alpha = 0.6\) for the annular cases) for the case of aiding flow. The effect of increasing the Grq/Re parameter for aiding flow is to redistribute the flow towards the heated wall. Mass continuity requires a corresponding decrease of flow away from the adiabatic boundary. For the case of opposing flow, the effect is to redistribute the flow away from the heated wall. This can be seen in Figures 3.7 through 3.10 (with
Figure 3.3  Velocity profile for the inner radius heated annulus for aiding flow.
Figure 3.4  Velocity profile for the outer radius heated annulus for aiding flow.
Figure 3.5 Velocity profile for the heated circular tube for aiding flow.
Figure 3.6 Velocity profile for the equivalent annulus for aiding flow.
Figure 3.7 Velocity profile for the inner radius heated annulus for opposing flow.
Figure 3.8 Velocity profile for the outer radius heated annulus for opposing flow.
Figure 3.9  Velocity profile for the heated circular tube for opposing flow.
Figure 3.10 Velocity profile for the equivalent annulus for opposing flow.
\( \alpha = 0.6 \) for the annular cases). There is a corresponding increase of flow near the adiabatic boundary.

As the \( \text{Gr}_q/\text{Re} \) number increases for a particular geometry, flow continues to redistribute. The analytical solution predicts that eventually there will be a local flow reversal caused by this redistribution. It will occur near the adiabatic boundary for aiding flow and near the diabatic boundary for opposing flow. When this occurs, the original assumption of fully-developed flow cannot be maintained. No analytical results will therefore be presented for values of \( \text{Gr}_q/\text{Re} \) greater than the value at which local flow reversal is predicted. Some workers assume that this value of \( \text{Gr}_q/\text{Re} \) is the highest at which fully-developed laminar flow can be maintained [B-1,H-4]. The range of results presented will therefore be for \( 0 < \text{Gr}_q/\text{Re} < (\text{Gr}_q/\text{Re})_{\text{max}} \), where \( (\text{Gr}_q/\text{Re})_{\text{max}} \) is the minimum of the \( \text{Gr}_q/\text{Re} \) number for which flow reversal is predicted, and \( \text{Gr}_q/\text{Re} = 10^4 \).

3.4.2 Friction Factor

In the last section it was shown that the velocity profile in a duct will be distorted as the \( \text{Gr}_q/\text{Re} \) parameter increases. As the velocity profile changes, the fluid shear at the no-slip surfaces must also change, since shear is caused by the velocity gradient. For aiding flow, the shear will increase at the heated wall and due to mass conservation, decrease at the adiabatic boundary. For opposing flow, the reverse will occur. These effects cause the friction factor to vary with the \( \text{Gr}_q/\text{Re} \) parameter.

The friction factor can be evaluated using previously derived Eqs. (3.44) and (3.55). For geometries with azimuthal symmetry, the surface normal must always be radial so that \( \phi = 0 \). Equations (3.44)
and (3.45) can therefore be combined as

\[ f_{Re} = \frac{8}{P_w} \int \frac{dU}{dR} \frac{dP}{d\alpha} \]  

The integral expression in Eq. (3.66) can easily be evaluated since azimuthal symmetry is assumed. The equation can be rewritten as

\[ f_{Re} = 8 \left[ \frac{dU}{dR} \right]_{R_1}^{\frac{\alpha}{1+\alpha}} + \frac{dU}{dR} \left|_{R_0}^{\frac{1}{1+\alpha}} \right. \]  

annuli,circle  \hspace{1cm} (3.77a)

\[ f_{Re} = 8 \left. \frac{dU}{dR} \right|_{R_1} \]  

equivalent annulus \hspace{1cm} (3.77b)

For the circle, the first term in the brackets of Eq. (3.77a) is equal to zero since \( \alpha = 0 \). For the equivalent annulus, only shear at the outer boundary need be considered since there is no velocity gradient at the outer boundary. The derivatives are evaluated using Eq. (3.66) for aiding flow and Eq. (3.69) for opposing flow.

The parameter \( f_0 \, Re \) can be evaluated from

\[ f_0 \, Re = 64 \]  
circle \hspace{1cm} (3.78a)

\[ f_0 \, Re = \frac{64(1-\alpha^2)(1-\alpha)^2}{1-\alpha^4(1-\alpha^2)^2} \]  
annulus \hspace{1cm} (3.78b)

\[ f_0 \, Re = \frac{64(1-\alpha^2)^3}{\alpha^2(-3+4\alpha^2-\alpha^4+4\ln(\alpha))} \]  
equivalent annulus \hspace{1cm} (3.78c)

which agree with the standard values that can be found in many textbooks. These results were obtained by solving Eq. (3.23) with \( Gr_q/Re = 0 \), applying the appropriate velocity boundary conditions, and substituting the expression for \( U \) into Eq. (3.77). Once the laminar mixed convection profiles are known, the friction factor ratio can be evaluated from Eq. (3.47), where the \( f_{Re} \) is calculated using Eq. (3.77) and the \( f_0 \, Re \) is from Eq. (3.78). For the forced convection limiting case, \( (f/f_0) \) approaches one as \( Gr_q/Re \) approaches zero.
The \((f/f_0)\) parameter is plotted against the \(Gr_q/Re\) parameter for the four geometries in Figures 3.11 through 3.14 for aiding flow. The circular duct is a limiting case of the outer radius heated annulus, and is presented as an outer radius heated annulus with the parameter \(\alpha\) equal to zero. It can be seen that in general the friction factor ratio increases as \(Gr_q/Re\) increases. This results from the buoyancy-induced flow redistribution that causes increased shear at the heated surface. It was noted earlier that mass continuity requires that there be a corresponding decrease of fluid velocity near the adiabatic boundary. This causes a decrease of shear at this boundary. For the inner radius heated annulus with small \(\alpha\), the decrease in shear at the adiabatic boundary more than compensates for the increase at the heated surface. As a result, increasing the \(Gr_q/Re\) parameter may cause a decrease in \((f/f_0)\), as can be seen in Figure 3.11 for \(\alpha=0.1\). The curve for the circular tube \((\alpha=0)\) matches that derived by Bishop, et al. [B-1].

In Figures 3.14 through 3.17, the \((f/f_0)\) parameter is plotted against the \(Gr_q/Re\) parameter for the four cases in opposing flow. For opposing flow, the buoyancy induced flow redistribution, in general, causes a decrease in friction factor ratio as \(Gr_q/Re\) increases. As with aiding flow, the exception is the inner radius heated annulus, which experiences an increase in \((f/f_0)\) as \(Gr_q/Re\) increases for small \(\alpha\), as can be seen in Figure 3.14 for \(\alpha<0.3\).

The above friction factor ratio results were fitted to the form

\[
\ln(f/f_0) = \sum_{l=0}^{4} q_i \lambda^l
\]  

(3.79a)
Figure 3.11 Mixed convection friction factor ratio for the inner radius heated annulus for aiding flow.
Figure 3.12 Mixed convection friction factor ratio for the outer radius heated annulus for aiding flow.
Figure 3.13  Mixed convection friction factor ratio for the equivalent annulus for aiding flow.
Figure 3.14 Mixed convection friction factor ratio for the inner radius heated annulus for opposing flow.
Figure 3.15 Mixed convection friction factor ratio for the outer radius heated annulus for opposing flow.
Figure 3.16 Mixed convection friction factor ratio for the equivalent annulus for opposing flow.
where
\[ \lambda = \ln(1+Cr_q/Re) \]  \hfill (3.80)

The results are presented in Tables 3.2 through 3.7. The values of \( q_1 \) were obtained by least squared error fitting, and less than 5% fitting error results when \( Cr_q/Re \) is within the range listed in the tables.

3.4.3 Modified Friction Factors

One can use Eqs. (3.51) and (3.53) to solve for the temperature field in the duct once an expression for \( fRe \) has been obtained. One can then use Eq. (3.56) to find the dimensionless bulk temperature, \( T_b \). For geometries with azimuthal symmetry, Eq. (3.56) simplifies to
\[ T_b = 2\pi \frac{D^2}{A_f} \int_{R_1}^{R_0} UTRdR \]  \hfill (3.81)
which can be integrated numerically. Once \( T_b \) is obtained, one can use Eq. (3.59) to solve for \( F \).

In Figures 3.17 through 3.22, the modified friction factor ratio \( (f/f_0)(1+F) \) is plotted against \( Cr_q/Re \), along with \( (f/f_0) \). It can be seen in Figure 3.17 that for the inner radius heated annulus in aiding flow, the modified friction factor ratio significantly exceeds the friction factor ratio (\( F \) is positive). For the outer radius heated annulus in aiding flow, \( F \) is negative for low values of \( Cr_q/Re \), but is positive for higher values, as shown in Figure 3.18. The result is a cross-over of modified and conventional friction factor ratio curves for this geometry. For the equivalent annulus in aiding flow, \( F \) is negative for \( Cr_q/Re < 1\times10^4 \), which is the range
Table 3.2

Coefficients for Eq. (3.79a) for the Inner-Radius Heated Annulus

for Aiding Flow

<table>
<thead>
<tr>
<th>α</th>
<th>g₀</th>
<th>g₁</th>
<th>g₂</th>
<th>g₃</th>
<th>g₄</th>
<th>((\text{Gr}<em>q / \text{Re})</em>{\text{max}})</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>6.973E-4</td>
<td>-1.053E-2</td>
<td>4.535E-3</td>
<td>-3.500E-4</td>
<td>-3.142E-5</td>
<td>1E4</td>
</tr>
<tr>
<td>0.2</td>
<td>2.668E-3</td>
<td>-8.523E-2</td>
<td>5.245E-2</td>
<td>-9.933E-3</td>
<td>5.804E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.3</td>
<td>1.892E-3</td>
<td>-8.203E-2</td>
<td>5.363E-2</td>
<td>-1.089E-2</td>
<td>6.997E-4</td>
<td>8E3</td>
</tr>
<tr>
<td>0.4</td>
<td>1.355E-3</td>
<td>-7.415E-2</td>
<td>5.047E-2</td>
<td>-1.075E-2</td>
<td>7.382E-4</td>
<td>7E3</td>
</tr>
<tr>
<td>0.5</td>
<td>8.873E-4</td>
<td>-6.171E-2</td>
<td>4.404E-2</td>
<td>-9.857E-3</td>
<td>7.214E-4</td>
<td>6E3</td>
</tr>
<tr>
<td>0.6</td>
<td>6.687E-4</td>
<td>-5.672E-2</td>
<td>4.149E-2</td>
<td>-9.549E-3</td>
<td>7.289E-4</td>
<td>6E3</td>
</tr>
<tr>
<td>0.7</td>
<td>4.044E-4</td>
<td>-4.792E-2</td>
<td>3.629E-2</td>
<td>-8.671E-3</td>
<td>6.967E-4</td>
<td>5E3</td>
</tr>
<tr>
<td>0.8</td>
<td>2.984E-4</td>
<td>-4.110E-2</td>
<td>3.227E-2</td>
<td>-7.987E-3</td>
<td>6.725E-4</td>
<td>5E3</td>
</tr>
<tr>
<td>0.9</td>
<td>1.614E-4</td>
<td>-3.554E-2</td>
<td>2.880E-2</td>
<td>-7.357E-3</td>
<td>6.472E-4</td>
<td>5E3</td>
</tr>
</tbody>
</table>
Table 3.3

Coefficients for Eq. (3.79a) for the Inner-Radius Heated Annulus for Opposing Flow

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$g_0$</th>
<th>$g_1$</th>
<th>$g_2$</th>
<th>$g_3$</th>
<th>$g_4$</th>
<th>$(Gr_q/Re)_{max}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>8.057E-4</td>
<td>-5.266E-2</td>
<td>3.662E-2</td>
<td>-7.983E-3</td>
<td>5.665E-4</td>
<td>5E3</td>
</tr>
<tr>
<td>0.2</td>
<td>4.104E-4</td>
<td>-4.572E-2</td>
<td>3.339E-2</td>
<td>-7.606E-3</td>
<td>5.591E-4</td>
<td>3E3</td>
</tr>
<tr>
<td>0.3</td>
<td>2.138E-4</td>
<td>-3.180E-2</td>
<td>2.367E-2</td>
<td>-5.476E-3</td>
<td>4.031E-4</td>
<td>2E3</td>
</tr>
<tr>
<td>0.4</td>
<td>2.204E-4</td>
<td>-3.002E-2</td>
<td>2.212E-2</td>
<td>-5.014E-3</td>
<td>3.508E-4</td>
<td>2E3</td>
</tr>
<tr>
<td>0.5</td>
<td>2.152E-4</td>
<td>-2.586E-2</td>
<td>1.808E-2</td>
<td>-3.852E-3</td>
<td>2.393E-4</td>
<td>2E3</td>
</tr>
<tr>
<td>0.6</td>
<td>1.634E-4</td>
<td>-1.220E-2</td>
<td>7.412E-3</td>
<td>-1.219E-3</td>
<td>2.207E-5</td>
<td>2E3</td>
</tr>
<tr>
<td>0.7</td>
<td>2.450E-5</td>
<td>1.147E-2</td>
<td>-1.052E-2</td>
<td>3.053E-3</td>
<td>-3.109E-4</td>
<td>2E3</td>
</tr>
<tr>
<td>0.8</td>
<td>-1.842E-4</td>
<td>4.566E-2</td>
<td>-3.685E-2</td>
<td>9.331E-3</td>
<td>-7.926E-4</td>
<td>2E3</td>
</tr>
<tr>
<td>0.9</td>
<td>-7.117E-5</td>
<td>4.296E-2</td>
<td>-3.590E-2</td>
<td>9.427E-3</td>
<td>-8.353E-4</td>
<td>1E3</td>
</tr>
</tbody>
</table>
Table 3.4

Coefficients for Eq. (3.79a) for the Outer-Radius Heated Annulus for Aiding Flow

<table>
<thead>
<tr>
<th>α</th>
<th>g₀</th>
<th>g₁</th>
<th>g₂</th>
<th>g₃</th>
<th>g₄</th>
<th>(Gr/q/Re)_{max}</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-2.443E-4</td>
<td>1.239E-1</td>
<td>-8.981E-1</td>
<td>1.976E-2</td>
<td>-9.683E-4</td>
<td>2.5E3</td>
</tr>
<tr>
<td>0.1</td>
<td>-5.241E-4</td>
<td>4.813E-2</td>
<td>-2.974E-2</td>
<td>5.138E-3</td>
<td>-3.049E-5</td>
<td>3E3</td>
</tr>
<tr>
<td>0.2</td>
<td>-4.183E-4</td>
<td>2.861E-2</td>
<td>-1.561E-2</td>
<td>1.956E-3</td>
<td>1.621E-4</td>
<td>3E3</td>
</tr>
<tr>
<td>0.3</td>
<td>-3.139E-4</td>
<td>1.436E-2</td>
<td>-5.332E-3</td>
<td>-3.246E-4</td>
<td>2.955E-4</td>
<td>3E3</td>
</tr>
<tr>
<td>0.4</td>
<td>-4.168E-4</td>
<td>1.307E-2</td>
<td>-4.015E-3</td>
<td>-6.953E-4</td>
<td>3.064E-4</td>
<td>4E3</td>
</tr>
<tr>
<td>0.5</td>
<td>-3.304E-4</td>
<td>1.949E-3</td>
<td>3.676E-3</td>
<td>-2.320E-3</td>
<td>3.975E-4</td>
<td>4E3</td>
</tr>
<tr>
<td>0.6</td>
<td>-2.170E-4</td>
<td>-6.905E-3</td>
<td>9.934E-3</td>
<td>-3.645E-3</td>
<td>4.706E-4</td>
<td>4E3</td>
</tr>
<tr>
<td>0.7</td>
<td>-1.181E-4</td>
<td>-1.547E-2</td>
<td>1.552E-2</td>
<td>-4.765E-3</td>
<td>5.290E-4</td>
<td>4E3</td>
</tr>
<tr>
<td>0.8</td>
<td>-2.665E-5</td>
<td>-2.169E-2</td>
<td>1.981E-2</td>
<td>-5.642E-3</td>
<td>5.736E-4</td>
<td>4E3</td>
</tr>
<tr>
<td>0.9</td>
<td>3.785E-5</td>
<td>-2.662E-2</td>
<td>2.323E-2</td>
<td>-6.334E-3</td>
<td>6.068E-4</td>
<td>4E3</td>
</tr>
</tbody>
</table>
Table 3.5
Coefficients for Eq. (3.79a) for the Outer-Radius Heated Annulus
for Opposing Flow

<table>
<thead>
<tr>
<th>α</th>
<th>g₀</th>
<th>g₁</th>
<th>g₂</th>
<th>g₃</th>
<th>g₄</th>
<th>((Gr_q/Re)_{max})</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-4.106E-4</td>
<td>7.297E0</td>
<td>-5.966E0</td>
<td>1.511E0</td>
<td>-1.221E-1</td>
<td>2E2</td>
</tr>
<tr>
<td>0.1</td>
<td>-1.446E-5</td>
<td>1.359E-1</td>
<td>-1.231E-1</td>
<td>3.543E-2</td>
<td>-3.483E-3</td>
<td>3E2</td>
</tr>
<tr>
<td>0.2</td>
<td>-4.182E-5</td>
<td>1.471E-1</td>
<td>-1.296E-1</td>
<td>3.614E-2</td>
<td>-3.402E-3</td>
<td>4E2</td>
</tr>
<tr>
<td>0.3</td>
<td>-7.192E-5</td>
<td>1.513E-1</td>
<td>-1.309E-1</td>
<td>3.573E-2</td>
<td>-3.262E-3</td>
<td>5E2</td>
</tr>
<tr>
<td>0.4</td>
<td>-4.812E-5</td>
<td>1.093E-1</td>
<td>-9.493E-2</td>
<td>2.602E-2</td>
<td>-2.403E-3</td>
<td>5E2</td>
</tr>
<tr>
<td>0.5</td>
<td>-8.292E-5</td>
<td>1.096E-1</td>
<td>-9.339E-2</td>
<td>2.511E-2</td>
<td>-2.264E-3</td>
<td>6E2</td>
</tr>
<tr>
<td>0.6</td>
<td>-1.075E-4</td>
<td>1.033E-1</td>
<td>-8.745E-2</td>
<td>2.328E-2</td>
<td>-2.068E-3</td>
<td>7E2</td>
</tr>
<tr>
<td>0.7</td>
<td>-1.185E-4</td>
<td>9.791E-2</td>
<td>-8.163E-2</td>
<td>2.140E-2</td>
<td>-1.869E-3</td>
<td>8E2</td>
</tr>
<tr>
<td>0.8</td>
<td>-1.421E-4</td>
<td>8.834E-2</td>
<td>-7.316E-2</td>
<td>1.901E-2</td>
<td>-1.644E-3</td>
<td>9E2</td>
</tr>
<tr>
<td>0.9</td>
<td>-1.573E-4</td>
<td>7.834E-2</td>
<td>-6.457E-2</td>
<td>1.666E-2</td>
<td>-1.431E-3</td>
<td>1E3</td>
</tr>
</tbody>
</table>
Table 3.6
Coefficients for Eq. (3.79a) for the Equivalent Annulus
for Aiding Flow

<table>
<thead>
<tr>
<th>( \alpha )</th>
<th>( g_0 )</th>
<th>( g_1 )</th>
<th>( g_2 )</th>
<th>( g_3 )</th>
<th>( g_4 )</th>
<th>((Gr_q/Re)_{max})</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>1.785E-5</td>
<td>-7.376E-4</td>
<td>5.079E-4</td>
<td>-1.122E-4</td>
<td>8.077E-6</td>
<td>1E4</td>
</tr>
<tr>
<td>0.2</td>
<td>2.888E-4</td>
<td>-1.154E-2</td>
<td>7.519E-3</td>
<td>-1.544E-3</td>
<td>1.038E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.3</td>
<td>7.247E-4</td>
<td>-3.177E-2</td>
<td>2.106E-2</td>
<td>-4.404E-3</td>
<td>3.032E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.4</td>
<td>9.130E-4</td>
<td>-4.475E-2</td>
<td>3.089E-2</td>
<td>-6.732E-3</td>
<td>4.881E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.5</td>
<td>5.880E-4</td>
<td>-4.370E-2</td>
<td>3.189E-2</td>
<td>-7.383E-3</td>
<td>5.781E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.6</td>
<td>-1.353E-4</td>
<td>-2.563E-2</td>
<td>2.235E-2</td>
<td>-6.001E-3</td>
<td>5.466E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.7</td>
<td>-1.036E-3</td>
<td>-6.055E-4</td>
<td>7.792E-3</td>
<td>-3.519E-3</td>
<td>4.413E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.8</td>
<td>-1.844E-3</td>
<td>2.768E-2</td>
<td>-9.254E-3</td>
<td>-4.421E-4</td>
<td>2.917E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.9</td>
<td>-2.555E-3</td>
<td>5.521E-2</td>
<td>-2.622E-2</td>
<td>2.726E-3</td>
<td>1.283E-4</td>
<td>1E4</td>
</tr>
</tbody>
</table>
Table 3.7

Coefficients for Eq. (3.79a) for the Equivalent Annulus for Opposing Flow

<table>
<thead>
<tr>
<th>α</th>
<th>$g_0$</th>
<th>$g_1$</th>
<th>$g_2$</th>
<th>$g_3$</th>
<th>$g_4$</th>
<th>($Gr_q/Re_{max}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>-1.840E-5</td>
<td>7.553E-4</td>
<td>-5.189E-4</td>
<td>1.143E-4</td>
<td>-8.206E-6</td>
<td>1E4</td>
</tr>
<tr>
<td>0.2</td>
<td>-3.745E-4</td>
<td>1.421E-2</td>
<td>-9.163E-3</td>
<td>1.858E-3</td>
<td>-1.228E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.3</td>
<td>-1.892E-3</td>
<td>6.834E-2</td>
<td>-4.361E-2</td>
<td>8.710E-3</td>
<td>-5.626E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.4</td>
<td>-5.867E-4</td>
<td>4.953E-2</td>
<td>-3.606E-2</td>
<td>8.214E-3</td>
<td>-6.126E-4</td>
<td>4E3</td>
</tr>
<tr>
<td>0.5</td>
<td>-2.770E-4</td>
<td>5.086E-2</td>
<td>-3.935E-2</td>
<td>9.545E-3</td>
<td>-7.652E-4</td>
<td>2E3</td>
</tr>
<tr>
<td>0.6</td>
<td>-7.547E-5</td>
<td>3.706E-2</td>
<td>-3.105E-2</td>
<td>8.169E-3</td>
<td>-7.250E-4</td>
<td>1E3</td>
</tr>
<tr>
<td>0.7</td>
<td>-1.133E-4</td>
<td>6.340E-2</td>
<td>-5.219E-2</td>
<td>1.351E-2</td>
<td>-1.172E-3</td>
<td>1E3</td>
</tr>
<tr>
<td>0.8</td>
<td>-1.224E-4</td>
<td>7.881E-2</td>
<td>-6.573E-2</td>
<td>1.724E-2</td>
<td>-1.514E-3</td>
<td>9E2</td>
</tr>
<tr>
<td>0.9</td>
<td>-7.774E-5</td>
<td>8.150E-2</td>
<td>-6.952E-2</td>
<td>1.865E-2</td>
<td>-1.684E-3</td>
<td>7E2</td>
</tr>
</tbody>
</table>
Figure 3.17 Comparison of modified and conventional friction factor ratios for the inner radius heated annulus for aiding flow.
Figure 3.18 Comparison of modified and conventional friction factor ratios for the outer radius heated annulus for aiding flow.
Figure 3.19  Comparison of modified and conventional friction factor ratios for the equivalent annulus for aiding flow.
Figure 3.20 Comparison of modified and conventional friction factor ratios for the inner radius heated annulus for opposing flow.
Figure 3.21 Comparison of modified and conventional friction factor ratios for the outer radius heated annulus for opposing flow.
Figure 3.22 Comparison of modified and conventional friction factor ratios for the equivalent annulus for opposing flow.
investigated. The result is that the modified friction factor ratio is less than the conventional one, as shown in Figure 3.19.

For opposing flow, for an inner radius heated annulus, Figure 3.20 shows that for high \( \text{Gr}_q/\text{Re} \) values, \( F \) is positive. For small values of \( \text{Gr}_q/\text{Re} \), \( F \) is negative, and thus there is a cross-over of conventional and modified friction factor ratio curves. For large values of \( \alpha \), this cross-over occurs at low values of \( \text{Gr}_q/\text{Re} \), and can not be seen in the plot. For small values of \( \alpha \), the cross-over can be clearly seen. For large values of \( \text{Gr}_q/\text{Re} \), the modified friction factor ratio is upward sloping. Figure 3.21 shows the modified friction factor ratio for the outer heated annulus in opposing flow. For this geometry, \( F \) is always positive and \((1+F)\) is always greater than \((f/f_0)\). The result is that the modified friction factor ratio curves are upward sloping. The same applies to the equivalent annulus geometry in opposing flow, as shown in Figure 3.22.

The above modified friction factor ratio results were fitted to the form

\[
\ln[(f/f_0)(1 + F)] = \sum_{l=0}^{4} q_l \lambda^l \quad (3-79b)
\]

as was done for the conventional friction factor ratio. The resulting values for \( q_l \) are listed in Tables 3.8 through 3.13.

3.4.4 Nusselt Number

The determination of the Nusselt number is trivial once the temperature field is known for the case of circumferentially uniform heat flux. The average wall temperature along the heated perimeter need not be computed using Eq. (3.63) since the temperature is constant along the wall. Instead, \( T_w \) is given by Eq. (3.53) computed at the heated wall. The Nusselt number is then given by Eq. (3.62).
Table 3.8
Coefficients for Eq. (3.79b) for the Inner Radius Heated Annulus
for Aiding Flow

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$g_0$</th>
<th>$g_1$</th>
<th>$g_2$</th>
<th>$g_3$</th>
<th>$g_4$</th>
<th>$(\text{Gr}<em>q/\text{Re})</em>{\text{max}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>1.637E-1</td>
<td>-3.319E-1</td>
<td>1.709E-1</td>
<td>-3.172E-2</td>
<td>1.948E-3</td>
<td>1E4</td>
</tr>
<tr>
<td>0.2</td>
<td>1.096E-1</td>
<td>-2.393E-1</td>
<td>1.367E-1</td>
<td>-2.822E-2</td>
<td>1.929E-3</td>
<td>1E4</td>
</tr>
<tr>
<td>0.3</td>
<td>3.065E-2</td>
<td>-9.057E-2</td>
<td>6.972E-2</td>
<td>-1.797E-2</td>
<td>1.459E-3</td>
<td>8E3</td>
</tr>
<tr>
<td>0.4</td>
<td>-3.196E-2</td>
<td>2.962E-2</td>
<td>1.372E-2</td>
<td>-8.984E-3</td>
<td>1.016E-3</td>
<td>7E3</td>
</tr>
<tr>
<td>0.5</td>
<td>-7.742E-1</td>
<td>1.183E-1</td>
<td>-2.863E-2</td>
<td>-2.006E-3</td>
<td>6.609E-4</td>
<td>6E3</td>
</tr>
<tr>
<td>0.6</td>
<td>-1.107E-1</td>
<td>1.839E-1</td>
<td>-6.049E-2</td>
<td>3.345E-3</td>
<td>3.825E-4</td>
<td>6E3</td>
</tr>
<tr>
<td>0.7</td>
<td>-1.352E-1</td>
<td>2.329E-1</td>
<td>-8.459E-2</td>
<td>7.453E-3</td>
<td>1.656E-4</td>
<td>5E3</td>
</tr>
<tr>
<td>0.8</td>
<td>-1.537E-1</td>
<td>2.698E-1</td>
<td>-1.029E-1</td>
<td>1.061E-2</td>
<td>-2.923E-6</td>
<td>5E3</td>
</tr>
<tr>
<td>0.9</td>
<td>-1.679E-1</td>
<td>2.987E-1</td>
<td>-1.176E-1</td>
<td>1.317E-2</td>
<td>-1.412E-4</td>
<td>5E3</td>
</tr>
</tbody>
</table>
### Table 3.9

Coefficients for Eq. (3.79b) for the Inner Radius Heated Annulus
for Opposing Flow

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$g_0$</th>
<th>$g_1$</th>
<th>$g_2$</th>
<th>$g_3$</th>
<th>$g_4$</th>
<th>$(\text{Gr}<em>q / \text{Re})</em>{\text{max}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>1.392E-1</td>
<td>-2.930E-1</td>
<td>1.591E-1</td>
<td>-3.114E-2</td>
<td>2.014E-3</td>
<td>5E3</td>
</tr>
<tr>
<td>0.2</td>
<td>1.501E-1</td>
<td>-3.274E-1</td>
<td>1.877E-1</td>
<td>-3.901E-2</td>
<td>2.687E-3</td>
<td>3E3</td>
</tr>
<tr>
<td>0.3</td>
<td>9.140E-2</td>
<td>-2.039E-1</td>
<td>1.218E-1</td>
<td>-2.673E-2</td>
<td>1.961E-3</td>
<td>2E3</td>
</tr>
<tr>
<td>0.4</td>
<td>1.355E-1</td>
<td>-3.025E-1</td>
<td>1.807E-1</td>
<td>-3.966E-2</td>
<td>2.909E-3</td>
<td>2E3</td>
</tr>
<tr>
<td>0.5</td>
<td>1.793E-1</td>
<td>-4.000E-1</td>
<td>2.389E-1</td>
<td>-5.239E-2</td>
<td>3.840E-3</td>
<td>2E3</td>
</tr>
<tr>
<td>0.6</td>
<td>2.175E-1</td>
<td>-4.854E-1</td>
<td>2.901E-1</td>
<td>-6.367E-2</td>
<td>4.672E-3</td>
<td>2E3</td>
</tr>
<tr>
<td>0.7</td>
<td>2.551E-1</td>
<td>-5.695E-1</td>
<td>3.404E-1</td>
<td>-7.470E-2</td>
<td>5.481E-3</td>
<td>2E3</td>
</tr>
<tr>
<td>0.8</td>
<td>2.887E-1</td>
<td>-6.445E-1</td>
<td>3.853E-1</td>
<td>-8.458E-2</td>
<td>6.208E-3</td>
<td>2E3</td>
</tr>
<tr>
<td>0.9</td>
<td>2.738E-1</td>
<td>-6.129E-1</td>
<td>3.685E-1</td>
<td>-8.154E-2</td>
<td>6.043E-3</td>
<td>1E3</td>
</tr>
</tbody>
</table>
Table 3.10

Coefficients for Eq. (3.79b) for the Outer Radius Heated Annulus for Aiding Flow

<table>
<thead>
<tr>
<th>( \alpha )</th>
<th>( g_0 )</th>
<th>( g_1 )</th>
<th>( g_2 )</th>
<th>( g_3 )</th>
<th>( g_4 )</th>
<th>( \frac{(Gr q)_{max}}{Re} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-6.770E-2</td>
<td>1.285E-1</td>
<td>-5.040E-2</td>
<td>2.845E-3</td>
<td>5.659E-4</td>
<td>2.5E3</td>
</tr>
<tr>
<td>0.1</td>
<td>-2.536E-1</td>
<td>4.816E-1</td>
<td>-2.161E-1</td>
<td>3.136E-2</td>
<td>-1.162E-3</td>
<td>3E3</td>
</tr>
<tr>
<td>0.2</td>
<td>-2.429E-1</td>
<td>4.582E-1</td>
<td>-2.032E-1</td>
<td>2.896E-2</td>
<td>-1.028E-3</td>
<td>3E3</td>
</tr>
<tr>
<td>0.3</td>
<td>-2.358E-1</td>
<td>4.423E-1</td>
<td>-1.941E-1</td>
<td>2.721E-2</td>
<td>-9.280E-4</td>
<td>3E3</td>
</tr>
<tr>
<td>0.4</td>
<td>-2.292E-1</td>
<td>4.278E-1</td>
<td>-1.860E-1</td>
<td>2.564E-2</td>
<td>-8.376E-4</td>
<td>4E3</td>
</tr>
<tr>
<td>0.5</td>
<td>-2.222E-1</td>
<td>4.122E-1</td>
<td>-1.773E-1</td>
<td>2.399E-2</td>
<td>-7.427E-4</td>
<td>4E3</td>
</tr>
<tr>
<td>0.6</td>
<td>-2.145E-1</td>
<td>3.956E-1</td>
<td>-1.682E-1</td>
<td>2.230E-2</td>
<td>-6.467E-4</td>
<td>4E3</td>
</tr>
<tr>
<td>0.7</td>
<td>-2.067E-1</td>
<td>3.790E-1</td>
<td>-1.592E-1</td>
<td>2.062E-2</td>
<td>-5.514E-4</td>
<td>4E3</td>
</tr>
<tr>
<td>0.8</td>
<td>-1.980E-1</td>
<td>3.608E-1</td>
<td>-1.496E-1</td>
<td>1.886E-2</td>
<td>-4.535E-4</td>
<td>4E3</td>
</tr>
<tr>
<td>0.9</td>
<td>-1.884E-1</td>
<td>3.409E-1</td>
<td>-1.393E-1</td>
<td>1.702E-2</td>
<td>-3.518E-4</td>
<td>4E3</td>
</tr>
</tbody>
</table>
Table 3.11

Coefficients for Eq. (3.79b) for the Outer Radius Heated Annulus for Opposing Flow

| 0   | 4.551E-1 | 0.038E00 | 6.540E-1 | -1.532E-1 | 1.266E-2 | (Grq/Re)_{max} |
| 0.1 | 1.329E-1 | -2.930E-1 | 1.591E-1 | -3.114E-1 | 2.014E-2 | 3E2            |
| 0.2 | 1.501E-1 | -3.274E-1 | 1.877E-1 | -3.901E-2 | 2.687E-3 | 4E2            |
| 0.3 | 9.140E-2 | -2.039E-1 | 1.218E-1 | -2.673E-2 | 1.961E-3 | 5E2            |
| 0.4 | 1.355E-1 | -3.025E-1 | 1.807E-1 | -3.966E-2 | 2.909E-3 | 5E2            |
| 0.5 | 1.793E-1 | -4.000E-1 | 2.389E-1 | -5.239E-2 | 3.840E-3 | 6E2            |
| 0.6 | 2.175E-1 | -4.854E-1 | 2.901E-1 | -6.367E-2 | 4.672E-3 | 7E2            |
| 0.7 | 2.551E-1 | -5.695E-1 | 3.404E-1 | -7.470E-2 | 5.481E-3 | 8E2            |
| 0.8 | 2.887E-1 | -6.445E-1 | 3.853E-1 | -8.458E-2 | 6.208E-3 | 9E2            |
| 0.9 | 2.738E-1 | -6.129E-1 | 3.685E-1 | -8.154E-2 | 6.043E-3 | 1E3            |
Table 3.12

Coefficients for Eq. (3.79b) for the Equivalent Annulus
for Aiding Flow

<table>
<thead>
<tr>
<th>α</th>
<th>$g_0$</th>
<th>$g_1$</th>
<th>$g_2$</th>
<th>$g_3$</th>
<th>$g_4$</th>
<th>$(Gr_q/Re)_{max}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>6.054E-4</td>
<td>-1.232E-3</td>
<td>6.548E-4</td>
<td>-1.304E-4</td>
<td>8.873E-6</td>
<td>1E4</td>
</tr>
<tr>
<td>0.2</td>
<td>9.180E-3</td>
<td>-1.863E-2</td>
<td>9.626E-3</td>
<td>-1.806E-3</td>
<td>1.151E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.3</td>
<td>2.475E-2</td>
<td>-5.050E-2</td>
<td>2.636E-2</td>
<td>-5.021E-3</td>
<td>3.270E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.4</td>
<td>3.602E-2</td>
<td>-7.458E-2</td>
<td>4.011E-2</td>
<td>-7.893E-3</td>
<td>5.356E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.5</td>
<td>3.837E-2</td>
<td>-8.136E-2</td>
<td>4.532E-2</td>
<td>-9.293E-3</td>
<td>6.656E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.6</td>
<td>3.226E-2</td>
<td>-7.103E-2</td>
<td>4.180E-2</td>
<td>-9.118E-3</td>
<td>7.046E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.7</td>
<td>1.847E-2</td>
<td>-4.544E-2</td>
<td>3.089E-2</td>
<td>-7.631E-3</td>
<td>6.662E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.8</td>
<td>2.888E-3</td>
<td>-1.592E-2</td>
<td>1.745E-2</td>
<td>-5.596E-3</td>
<td>5.896E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.9</td>
<td>-1.554E-2</td>
<td>1.954E-2</td>
<td>1.088E-3</td>
<td>-3.016E-3</td>
<td>4.779E-4</td>
<td>1E4</td>
</tr>
</tbody>
</table>
Table 3.13

Coefficients for Eq. (3.79b) for the Equivalent Annulus for Opposing Flow

<table>
<thead>
<tr>
<th>α</th>
<th>$g_0$</th>
<th>$g_1$</th>
<th>$g_2$</th>
<th>$g_3$</th>
<th>$g_4$</th>
<th>$(Gr/q_{Re})_{max}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>-6.206E-4</td>
<td>1.262E-3</td>
<td>-6.695E-4</td>
<td>1.330E-4</td>
<td>-9.023E-6</td>
<td>1E4</td>
</tr>
<tr>
<td>0.2</td>
<td>-1.144E-2</td>
<td>2.306E-2</td>
<td>-1.179E-2</td>
<td>2.183E-3</td>
<td>-1.367E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.3</td>
<td>-5.137E-2</td>
<td>1.026E-1</td>
<td>-5.182E-2</td>
<td>9.469E-3</td>
<td>-5.825E-4</td>
<td>1E4</td>
</tr>
<tr>
<td>0.4</td>
<td>-1.911E-1</td>
<td>3.773E-1</td>
<td>-1.871E-1</td>
<td>3.342E-2</td>
<td>-1.991E-3</td>
<td>4E3</td>
</tr>
<tr>
<td>0.5</td>
<td>-4.202E-1</td>
<td>8.353E-1</td>
<td>-4.175E-1</td>
<td>7.499E-2</td>
<td>-4.475E-3</td>
<td>2E3</td>
</tr>
<tr>
<td>0.6</td>
<td>-2.986E-1</td>
<td>6.169E-1</td>
<td>-3.267E-1</td>
<td>6.248E-2</td>
<td>-4.001E-3</td>
<td>1E3</td>
</tr>
<tr>
<td>0.7</td>
<td>-1.755E-1</td>
<td>3.783E-1</td>
<td>-2.136E-1</td>
<td>4.386E-2</td>
<td>-3.043E-3</td>
<td>1E3</td>
</tr>
<tr>
<td>0.8</td>
<td>-1.431E-1</td>
<td>3.144E-1</td>
<td>-1.830E-1</td>
<td>3.892E-2</td>
<td>-2.820E-3</td>
<td>9E2</td>
</tr>
<tr>
<td>0.9</td>
<td>-1.929E-1</td>
<td>4.233E-1</td>
<td>-2.461E-1</td>
<td>5.218E-2</td>
<td>-3.755E-3</td>
<td>7E2</td>
</tr>
</tbody>
</table>
In Figures 3.23 through 3.28, the Nusselt number is plotted against the \( \text{Gr}_q/\text{Re} \) parameter for the geometries considered. For aiding flow, a significant increase in the Nusselt number can be observed for both annular geometries when \( \text{Gr}_q/\text{Re} \) exceeds several hundred. This is due to increased convection along the heated wall. The equivalent annulus shows no significant change in Nusselt number as the \( \text{Gr}_q/\text{Re} \) parameter is increased, except for large values of \( \alpha \). For opposing flow, Nusselt numbers decrease with \( \text{Gr}_q/\text{Re} \) for the annular geometries due to decreased convection along the heated wall. As with aiding flow, significant change in the Nusselt number can be observed for the equivalent annulus except for large values of \( \alpha \). The circular tube results (Figures 3.24 and 3.27, \( \alpha = 0 \)) match that derived by Hallman [H-2]. The inner radius heated annulus results match that of Sherwin [S-2].

Lundberg, McCuen, and Reynolds [L-1] have solved for the heat transfer in annular geometries for forced convection, hydrodynamically developed, thermally developing flow. Table 3.14 compares their results for thermally developed flow with the results from the present mixed convection analysis, which were obtained by setting \( \text{Gr}_q/\text{Re} \) equal to 0.1. We see excellent agreement with the forced convection analysis for all values of \( \alpha \) reported.

3.5 Comparison of Equivalent Annulus and Infinite Rod Array

Analytical Results

Sparrow and Loeffler [S-3] solved for the forced convection friction factors for axial flow in infinite triangular and square rod array geometries. They found that as the \( (P/D) \) parameter is
Figure 3.23 Mixed convection Nusselt number for the inner radius heated annulus for aiding flow.
Figure 3.24 Mixed convection Nusselt number for the outer radius heated annulus for aiding flow.
Figure 3.25 Mixed convection Nusselt number for the equivalent annulus for aiding flow.
Figure 3.26 Mixed convection Nusselt number for the inner radius heated annulus for opposing flow.
Figure 3.27 Mixed convection Nusselt number for the outer radius heated annulus for opposing flow.
Figure 3.28 Mixed convection Nusselt number for the equivalent annulus for opposing flow.
Table 3-14
Forced Convection Nusselt Numbers for the Annulus

<table>
<thead>
<tr>
<th>α</th>
<th>Heating Location</th>
<th>Aiding</th>
<th>Opposing</th>
<th>Forced Convection Analysis [L]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.02</td>
<td>( r_1 )</td>
<td>32.704</td>
<td>32.702</td>
<td>32.70</td>
</tr>
<tr>
<td></td>
<td>( r_o )</td>
<td>4.734</td>
<td>4.734</td>
<td>4.734</td>
</tr>
<tr>
<td>0.05</td>
<td>( r_1 )</td>
<td>17.811</td>
<td>17.811</td>
<td>17.81</td>
</tr>
<tr>
<td></td>
<td>( r_o )</td>
<td>4.792</td>
<td>4.791</td>
<td>4.791</td>
</tr>
<tr>
<td>0.10</td>
<td>( r_1 )</td>
<td>11.906</td>
<td>11.906</td>
<td>11.90</td>
</tr>
<tr>
<td></td>
<td>( r_o )</td>
<td>4.834</td>
<td>4.834</td>
<td>4.834</td>
</tr>
<tr>
<td>0.25</td>
<td>( r_1 )</td>
<td>7.754</td>
<td>7.753</td>
<td>7.753</td>
</tr>
<tr>
<td></td>
<td>( r_o )</td>
<td>4.905</td>
<td>4.904</td>
<td>4.904</td>
</tr>
<tr>
<td>0.50</td>
<td>( r_1 )</td>
<td>6.184</td>
<td>6.181</td>
<td>6.181</td>
</tr>
<tr>
<td></td>
<td>( r_o )</td>
<td>5.037</td>
<td>5.036</td>
<td>5.036</td>
</tr>
</tbody>
</table>
increased, the effect of neighboring rods is reduced. This implies that when the local shear stress around the rod is normalized by the average shear stress and plotted versus $\theta$, as the (P/D) ratio increases, the angular dependence of the shear stress decreases as shown in Figure 3.29. Thus, large (P/D) ratios can be approximated by a geometry with azimuthal symmetry. This is the basis for choosing the equivalent annulus to describe rod arrays with large (P/D) ratios.

Sparrow and Loeffler [S-4] defined the equivalent annulus cell such that its flow area is equal to that of a cell of infinite rod array as shown in Figure 3.30. By equating the inner radius of the equivalent annulus with the rod radius for the interior cell, it is found that for equal flow areas,

$$\alpha = \left[ \frac{\pi}{s \tan(\pi/s)} \right]^{0.5} / (P/D)$$

(3.82)

which reduces for $s = 6$ and $s = 4$ to

$$\alpha = 0.9523/(P/D) \quad (s = 6 \text{ triangular array}) \quad (3.83a)$$

$$\alpha = 0.8862/(P/D) \quad (s = 4 \text{ square array}) \quad (3.83b)$$

Wang, et. al., [W-1] also plotted the angular dependence of shear stress along the rod surface for interior cells in a triangular array for aiding flow, as shown in Figure 3.31. He found that as the (P/D) ratio increases, the angular shear stress dependence weakens, which is consistent with the work of Sparrow and Loeffler [S-3]. Wang also found that for interior cells in aiding flow, increasing the $Gr_{q,D}/Re$ parameter resulted in a further decrease of the angular dependence of the shear stress, as shown in Figure 3.31.
Figure 3.29 Angular dependence of the normalized rod shear stress for interior cells in a (a) square and (b) triangular array for forced convection. (Taken from Ref. S-3)
Figure 3.30 Comparison of Interior Cell and Equivalent Annulus Approaches.
Figure 3.31 Angular dependence of the normalized shear stress for interior cells in a triangular array for aiding mixed convection flow. (Taken from Ref. W-1)
Since the equivalent annulus friction factor calculation is easier to perform than that for the interior cell, it is desirable to find the range of its applicability. Figures 3.32 through 3.35 directly compare the mixed convection friction factor calculated for the interior cell geometry with that for the equivalent annulus. Based on these curves, Table 3.15 was constructed, which recommends for the cases considered the minimum (P/D) ratio at which the equivalent annulus formulation accurately predicts the interior cell friction factor ratio. It can be seen that the equivalent annulus approximation can be used at a lower (P/D) ratio for triangular arrays than for square arrays. The equivalent annulus approximation also approximates aiding flow better than opposing flow. For opposing flow in a square array, it is not recommended for any (P/D) ratio.

3.6 Comparison of Analytical Results with Experimental Data

3.6.1 Friction Factor

There seems to be a scarcity of mixed convection pressure drop data in the literature. Kemeny and Somers [K-1] measured mixed convection pressure drop for aiding flow in circular tubes using water and oil as the test fluids. The friction factor ratio \( (f/f_0) \) was plotted against the \( Gr_q/Re \) parameter, and the data is grouped according to the Graetz number, defined by

\[
Gz = \frac{(RePr)}{(z/D_e)}
\]  

(3.84)

where \( z \) equals zero corresponds to the inlet of the pipe. Fully-developed flow would thus correspond to a \( Gz \) number approaching zero. Since the velocity field distortion due to buoyancy effects increases
Figure 3.32 Comparison of the mixed convection friction factor ratio for interior channels in a triangular array with the equivalent annulus prediction (aiding flow).
Figure 3.33 Comparison of the mixed convection friction factor ratio for interior cells in a square array with the equivalent annulus prediction (aiding flow).
Figure 3.34 Comparison of the mixed convection friction factor ratio for interior channels in a triangular array with the equivalent annulus prediction (opposing flow).
Figure 3.35. Comparison of the mixed convection friction factor ratio for interior channels in a square array with the equivalent annulus prediction (opposing flow).
Table 3.15

Recommended Application of Equivalent Annulus Formulation as an Approximation to Infinite Rod Bundle Geometry

<table>
<thead>
<tr>
<th>Minimum P/D Ratio</th>
<th>Aiding Flow</th>
<th>Opposing Flow</th>
</tr>
</thead>
<tbody>
<tr>
<td>Square Array</td>
<td>3.00</td>
<td>not recommended</td>
</tr>
<tr>
<td>Triangular Array</td>
<td>1.50</td>
<td>3.00</td>
</tr>
</tbody>
</table>
as the flow develops, as does the friction factor ratio \([B-1]\), one would expect the analytical solution for fully-developed flow to envelope from above the experimental data. The data is presented in Figure 3.36, along with the circular tube friction factor ratio and the modified friction factor ratio. One would expect the modified friction factor ratio to approximate the data better than the conventional friction factor ratio because Kemeny and Somers used the bulk temperature (as opposed to the spatially-averaged temperature) to calculate the gravity pressure drop in the tube. These are precisely the qualitative trends shown in the figure. The analytical results are shown only for \(Gr_q/Re\) less than 2553 because above this value, the analytical solution predicts that the flow will be reversed at the center of the tube, a situation that cannot be modeled correctly by the present analysis due to the fully-developed flow assumption. It is expected that a transition to turbulence occurs before the flow reverses at the center of the tube \([B-1]\). Kemeny and Somers, however, do not differentiate between laminar and turbulent data. In fact, they report that nearly all data presented are in the turbulent regime. The value of the analytical and experimental comparison is therefore questionable. Nevertheless, we see that the analytical prediction of modified friction factor approximates the data well for \(Gr_q/Re\) less than 1000.

Okada et al., \([O-2]\) have obtained mixed convection friction factor data in a 4x4 bare rod bundle of square array. We see in Figure 3.37 that the friction factor ratio derived here envelopes the data from above, which we explain because of developing flow effects. As with the circular tube data of Kemeny and Somers \([K-1]\), the
Figure 3.36 Experimental data compared with the analytical prediction for the mixed convection friction factor ratio for the heated circular tube for aiding flow.
(Adapted from Ref. B-1)
Figure 3.37 Experimental data compared with the analytical prediction for the mixed convection friction factor ratio for the MIT 4x4 bare rod bundle.
modified friction factor better predicts the data because the experimentalists have calculated the gravity pressure drop in the test section using the bulk temperature.

3.6.2 Nusselt Number

Hallman [H-1] obtained extensive heat transfer data for developing and fully-developed flow for a heated vertical tube with upflow (aiding flow) and downflow (opposing flow). Figures 3.38 and 3.39 are reproduced from his work, where he defined a Rayleigh number that can be related to the Grq/Re parameter by

\[ Ra_H = \frac{Grq}{Re} \]  

(3.85)

It is interesting to note that for upflow the theoretical Nusselt number predicted by Hallman's analysis very closely matches the data, even for Grq/Re values above the point the analysis predicts flow reversal at the center of tube \( (Ra_H = 2553/4 = 636) \). Since laminar flow is not expected past this point, it is not clear why the laminar flow analysis does not underpredict the heat transfer associated with the turbulent flow. For opposing flow, the analysis does underpredict the heat transfer, as shown in Figure 3.39.

Maitra and Sabbu Raju [M-3] obtained fully-developed Nusselt number data for an inner radius heated annulus in aiding flow for \( \alpha \) equal to 0.4 for aiding flow. Their analytical mixed convection laminar flow analysis results are plotted versus the Rayleigh numbers in Figure 3.40 along with the forced convection results presented by Lundberg, McCuen, and Reynolds [L-1]. Although the mixed convection analysis underpredicts the data, it accounts for the data much better than the forced convection analysis. The mixed convection analysis
Figure 3.38 Experimental data compared with the analytical prediction for the mixed convection Nusselt number for the heated circular tube for aiding flow.
(Taken from Ref. H-1)
Figure 3.39 Experimental data compared with the analytical prediction for the mixed convection Nusslet number for the heated circular tube for opposing flow. (Taken from Ref. H-1)
Figure 3.40 Experimental data compared with the analytical prediction for the mixed convection Nusselt number for the inner radius heated annulus for aiding flow. (Taken from Ref. M-3)
underprediction is believed to be caused by the presence of turbulence within the test section which would tend to increase the heat transfer. For this geometry, flow reversal was predicted here when Gr/Re is greater than 7000, and turbulence occurs after that point. Sherwin [S-2] presented experimental Nusselt number data for an annulus with $\alpha$ equal to 0.33 for opposing flow. He reported that the laminar analytical solution underpredicts the data due to turbulence.

Efthimiadis and Todreas [E-1] have obtained Nusselt number data from the interior cells of a 19-pin wire-wrapped rod bundle of triangular array. As can be seen in Figure 3.41, the data shows a stronger Gr/Re dependence than the equivalent annulus analysis. The workers report that turbulence was present for most of the data collected, as signaled by temperature fluctuation measured by thermocouples. Turbulence would increase the radial heat transport, and thus increase the Nusselt number measured. The reported data is within 20% of the analysis presented here.

3.7 Developing Flow Mixed Convection Pressure Drop in Finite Rod Bundles

3.7.1 Experimental Data

Mawatari, et al. [M-4] obtained friction factor data for mixed convection flow using water as a coolant in a 91-pin wire-wrapped bundle with P/D = 1.21 and H/D = 47.2. An increase in the $f_Re$ parameter was observed with increased rod heating. Okada, et al. [O-2] obtained mixed convection friction factor data using water as a coolant in a 19-pin wire-wrapped bundle with P/D = 1.25 and H/D = 35.
Figure 3.41 Experimental data compared with the analytical prediction for the mixed convection Nusselt number for the MIT 19-pin wire-wrapped bundle.
They also observed an increase in the fRe parameter with increased rod heating. The two bundles have similar spacings across their respective cross-sections, and thus are geometrically similar for fully-developed flow.

Figure 3.42 shows friction factor ratio data from the two bundles obtained for uniform radial (and axial) rod heating. The data is plotted versus the Grq/Re parameter. We see that the friction factor ratio increase is delayed for the 19-pin bundle. Since the two bundles are geometrically similar, it is puzzling at first glance why the Grq/Re parameter did not reduce the two sets of data. It can also be seen in Figure 3.42 that the equivalent annulus analysis model does not predict either set of data.

3.7.2 Description of Flow Redistribution Effects Within the Bundle

It was hypothesized that the buoyancy-induced flow redistribution within the bundle could be separated into two components. The first component occurs because the bundle is finite, and thus each subchannel is not identical, as it would be in an infinite array of rods. The edge cells of the bundle, because a portion of their wetted perimeter is not heated, will have a lower power-to-flow ratio than the interior cells. This causes a global redistribution of flow away from the edge cells and towards the interior of the bundle due to the increased buoyancy there. The second component of the flow redistribution occurs within a subchannel, and is that effect which is modeled by equivalent analysis. This redistribution is referred to as local redistribution of flow, and will cause the subchannel friction
Figure 3.42 ENERGY-IV prediction of the mixed convection friction factor ratio for the 19- and 91-pin wire-wrapped rod bundles.
factor to change. For an infinite array, there is no global redistribution. Since both global and local redistribution alter the velocity field within the bundle from its forced convection distribution, one expects both effects, in general, to alter the bundle friction factor. Based on the data, though, it was concluded that the local flow redistribution was not the primary contributor to the friction factor ratio increase. For the 91-pin bundle, the $Gr_q/Re$ values at which there is an increase in the bundle friction factor ratio is too low to be caused by local redistribution, which is represented by the equivalent annulus modified friction factor ratio curve. This leads to the conclusion that global redistribution is important to the prediction of the bundle friction factor ratio. Since the equivalent annulus model slightly overpredicts the 19-pin bundle data, it can be assumed that combining global and local effects would overpredict the 19-pin data by an even greater amount. Thus it was further concluded that the local flow redistribution effect was not the primary cause of the friction factor ratio increase for the 19-pin bundle. A methodology was sought to predict the global flow redistribution within the bundle and the associated bundle friction factor ratio increase.

3.7.3 ENERGY-IV Code

In order to explain the large deviations between the uniform power $f/f_0$ data of Figure 3.42 for the 19-pin and 91-pin wire-wrapped bundles, a recent version of the ENERGY-IV subchannel analysis code was applied to each case. This version was developed by Cheng [C-3] to predict inter-subchannel mixing in wire-wrapped bundles
under mixed convection conditions. The flow redistribution effects are modeled with the approximation

$$ \sum_j W_{ij} h^* = H^* \sum_j W_{ij} $$  \hspace{1cm} (3-86)

where $W_{ij}$ is the crossflow from subchannel $i$ to $j$, the $j$ subchannels surround $i$, and $h^* = h_i$ for $W_{ij} > 0$ and $h^* = h_j$ for $W_{ij} < 0$.

The enthalpy carried by the net crossflow for the subchannel, $H^*$, as proposed by Chelmer et al. [C-4] in their subchannel analysis method, is a weighted average value determined from the average enthalpies of each subchannel and its adjacent subchannels, i.e.,

$$ H = \frac{1}{2} \sum_{i=1}^I \left[ \left| X_i \right| (h+h_i) - X_i (h-h_i) \right] / \sum_{i=1}^I \left| X_i \right| $$  \hspace{1cm} (14) \hspace{1cm} (3-87)

where $X_i = \Delta m_i - \Delta m_i^*$

$$ \Delta m_i = \left. \Delta m_i \right|_{z+\Delta z} - \left. \dot{m}_i \right|_{z} $$

$\dot{m}_i = \text{subchannel mass flow rate}$

Coupling the subchannel energy equations with the subchannel axial momentum equations and one total mass balance equation, and under the assumption of constant pressure across the bundle at one axial level, the values ($\sum_j W_{ij}$) for each subchannel can be solved iteratively. The transverse momentum equation is not used because of the constant pressure assumption. This significantly reduces the computational time and storage required. The validity of this simple model for calculating flow redistribution in ENERGY-IV has been checked by comparing the axial velocity fields at different axial levels calculated by ENERGY-IV to those calculated by COBRA-WC [C-3].
A water properties package was added to ENERGY-IV to predict friction factors for the water test, and is included in Appendix C. The code was run for uniform radial power with inlet volumetric flow rates of $1.262 \times 10^{-4}$ m$^3$/s for the two bundles. The $G_{r_q}/Re$ parameter was controlled by adjusting the rod power. The ENERGY-IV predictions for the two wire-wrapped bundles are plotted in Figure 3.42. The code adequately predicts the data for $G_{r_q}/Re < 6000$ for the 19-pin bundle and follows the trend of the 91-pin data. The code was also run with the peripheral rods at lower power than the interior rods to simulate the 6% heat loss to the environment present in some of the 19-pin bundle experiments. For the range of interest, the heat loss alters the predicted $f/f_0$ ratio by less than 3%.

Forced convection friction factor correlations specific to each type subchannel were employed in all ENERGY-IV runs [C-3]. In all subchannels, the local (subchannel) $f/f_0$ ratio was taken equal to one. However, as Figure 3.42 indicates, the $f/f_0$ prediction for the whole bundle increases with increasing $G_{r_q}/Re$. This increase therefore is not due to local increase in the $f/f_0$ ratio but due to global flow redistribution among subchannels caused by thermal buoyancy effects. Consequently the differences between the 19-pin and the 91-pin data can be primarily attributed to these latter effects.
CHAPTER FOUR

TRANSIENT MIXED CONVECTION IN PARALLEL CHANNELS

4.1 Introduction

In Chapter 2, a steady-state analysis of the thermal hydraulic behavior of a parallel channel system was investigated. If a steady-state is assumed, the axial temperature profile in a channel can be analytically determined in terms of the channel flowrate. Once the temperature profile is known, the density and thus the gravity pressure drop in the channel can be calculated. This was the procedure followed to obtain the momentum-energy equation of Chapter 2.

For actual applications, it is possible that the behavior of a system of parallel channels can vary with time. For example, the inlet flowrate $m_0$ may be time dependent, as well as the individual channel heating. If the time constant associated with these boundary conditions is sufficiently small, the transient behavior can no longer be accurately modeled with quasi-steady assumptions.

The quasi-steady assumption is not valid if the time constants associated with transient behavior of the prescribed boundary conditions of the system are of the same order of magnitude as the time constants associated with the system. As an example, the transient behavior of a single heated channel is examined. Suppose the channel flowrate and heating are prescribed functions of time. These time dependent functions have associated time constants. The assumption of a steady-state axial temperature profile in the channel is only valid if the channel transit time $(L/u)$ is much smaller than the inlet flowrate and heating time constants. The momentum equation can be
thought of as a balance of inertial and resistive forces. If the inlet flowrate time constant is sufficiently large, the inertial term may be neglected, which is its quasi-steady approximation.

In this chapter, the transient forms of the mass, momentum, and energy equations will be developed for a system of parallel channels. These equations have been included in a code that models the transient thermal hydraulic behavior of a system of parallel channels connected at plena. The code is then used to predict the General Electric Shutdrown Core Coolability Model Test (SCCMT) results.

4.2 One Dimensional Conservation Equations and Numerical Methods

4.2.1 One Dimensional Continuity Equation

The transient continuity equation for axial flow through a channel (assuming no sources or sinks) is given by

\[ \frac{\partial \rho}{\partial t} + \frac{\partial}{\partial z} \rho u = 0 \]  

(4.1)

The density and velocity in Eq. (4.1) are defined at all points in the channel. For the nodding scheme considered here, the radial distribution of density and velocity within the channel need not be known so that fluid properties and velocities are only a function of the axial position \( z \) and time \( t \).

For an incompressible fluid, density variations in the continuity equation can be neglected, yielding

\[ \frac{\partial u}{\partial z} = 0 \]  

(4.2)

Equation (4.2) can then be spatially-averaged over the duct cross-section, resulting in

\[ \frac{\partial \{u\}}{\partial z} = 0 \]  

(4.3)

where the quantity in curly braces \( \{ \} \) denotes spatial averaging
over the cross section of the duct. Equations (4.2) and (4.3) say that the velocity of the fluid (and also the mass flowrate) in the channel is independent of the axial position and thus only a function of time.

4.2.2 One Dimensional Momentum Equation

4.2.2.1 Spatial Averaging

The transient momentum equation for axial flow through a channel can be written as

$$\frac{\partial}{\partial t} \rho u + \frac{\partial}{\partial z} \rho u^2 = - \frac{\partial p}{\partial z} - \rho g + \vec{\tau} \cdot \vec{\nu}$$  \hspace{1cm} (4.4)

After spatially-averaging the terms in Eq. (4.4), the result is

$$\frac{\partial}{\partial t} \{\rho u\} + \frac{\partial}{\partial z} \{\rho u^2\} = - \frac{\partial p}{\partial z} - \{\rho\}g + \{\vec{\tau} \cdot \vec{\nu}\}$$ \hspace{1cm} (4.5)

In accordance with the Boussinesq approximation, only density variations associated with the body force term of Eq. (4.5) need to be included for an incompressible fluid. The expression $\langle \rho u^2 \rangle$ can therefore be expanded to

$$\langle \rho u^2 \rangle = \rho \{u^2\}$$ \hspace{1cm} (4.6)

which is approximated by

$$\langle \rho u^2 \rangle = \rho \{u\}^2$$ \hspace{1cm} (4.7)

by assuming a uniform velocity profile across the duct. Substituting

$$\{u\} = \frac{\dot{m}}{\rho A_f}$$ \hspace{1cm} (4.8)

into Eq. (4.7) results in

$$\langle \rho u^2 \rangle = \frac{1}{\rho} \left(\frac{\dot{m}}{A_f}\right)^2$$ \hspace{1cm} (4.9)

Similarly, the expression $\langle \rho u \rangle$ can be written as

$$\langle \rho u \rangle = \frac{\dot{m}}{A_f}$$ \hspace{1cm} (4.10)
by utilizing Eq. (4.8) and neglecting density variations.

The last term of Eq. (4.5) represents the fluid-solid resistance in the channel. It is more convenient to represent this resistance in terms of a friction factor and local form loss coefficient. The result is

\[
[\ddot{\gamma}\cdot \mu \ddot{v}_u] = -\left[\frac{f}{D_e} + \frac{K}{dz}\right] \frac{1}{2\rho} \left[\frac{\dot{m}}{A_f}\right]^{\frac{1}{2}}
\]  

(4.11)

Substituting Eqs. (4.9) through (4.11) into (4.5) and multiplying by \(A_f\) yields

\[
\frac{\dot{m}}{A_f} + \frac{\partial}{\partial z} \left( \frac{\dot{m}^2}{\rho A_f} \right) = -A_f \frac{\partial}{\partial z} \left[\frac{f}{D_e} + \frac{K}{dz}\right] \frac{1}{2\rho} \frac{\dot{m}}{A_f}^{\frac{1}{2}}
\]  

(4.12)

4.2.2.2 Axial Integration (Momentum Integral)

It is possible to integrate the cross-sectionally averaged momentum equation along the channel length in order to obtain an integral momentum equation. By defining the channel pressure drop as

\[
\Delta p = p(0) - p(L)
\]  

(4.13)

and integrating Eq. (4.12) from \(z = 0\) to \(L\), one obtains

\[
\Delta p = \frac{L}{A_f} \frac{d\dot{m}}{dt} + \dot{m}^2 \cdot I_m + \frac{\dot{m}}{m} \cdot m^{1-b} \cdot I_f + \frac{\dot{m}^2}{m} \cdot I_K + I_g
\]  

(4.14)

where the friction factor is assumed to be of the form

\[
f = c Re^{-b}(f/f_o)(1 + F)
\]  

(4.15)

and \((f/f_o)(1 + F)\) is the modified friction factor ratio, which is a function of the \(Gr_q/Re\) parameter for fully-developed flow. The constants of Eq. (4.14) are of the form

\[
I_m = \frac{i}{A_f} \left[\frac{1}{\rho_L} - \frac{1}{\rho_o}\right]
\]  

(4.16)
\[
I_f = \frac{c}{2D^{1+b} e^{A_f}} \left| \int_0^L \frac{b}{\rho} (f/f_0)(1 + F) \, dz \right|
\]  (4.17)

\[
I_k = \frac{1}{2\Lambda_f^2} \left[ \Sigma(K/\rho) \right]
\]  (4.18)

and

\[
I_g = g \left| \int_0^L \rho \, dz \right|
\]  (4.19)

Typically, the density used to evaluate the integral expression in Eq. (4.19) (which is the body force term) is the bulk density as opposed to the spatially averaged density, \( \{\rho\} \). For this reason, the modified friction factor ratio was used to define the friction factor in Eq. (4.15).

The integral expressions in Eqs. (4.17) and (4.19) can be evaluated using numerical integration techniques such as the trapezoidal method, or Simpson's rule, where the integration is performed over the axial nodes for a channel.

4.2.2.3 Temporal Discretization

Now that the momentum equation has been averaged spatially and integrated along the channel length, it is desired to discretize the temporal dependence so that the transient equation can be numerically solved. This is accomplished by assuming all variables in Eqs. (4.14) and (4.19) do not vary during a time step, except for the channel flowrate, which can be approximated by

\[
\dot{m} = \frac{1}{2} (\dot{m}_j + \dot{m}_{j+1})
\]  (4.20)

where the \( j \) and \( j+1 \) subscripts refer to the present and future time levels, respectively. Defining
\[ \Delta m^* = m_{j+1}^* - m_j^* \]  

(4.21)

and substituting into Eq. (4.20) yields

\[ \dot{m} = m_j^* + \frac{1}{2} \Delta m^* \]  

(4.22)

Also, the temporal derivative is approximated by

\[ \frac{dm}{d\tau} = \frac{\Delta m}{\Delta \tau} \]  

(4.23)

The other expressions in Eq. (4.14) containing \( \dot{m} \) are expanded as

\[ m^2 = (m_j^* + \frac{1}{2} \Delta m)^2 \]  

(4.24)

\[ = m_j^2 + \Delta m \cdot m_j^* \]  

(4.25)

\[ \dot{m} \cdot \dot{m} \left|^{1-b} = \pm \left( m_j^* + \frac{1}{2} \Delta m \right)^{2-b} \right. \]  

(4.26)

\[ = \pm \left[ m_j^2 + \Delta m \cdot m_j^* \right]^{1-b} \]  

(4.27)

\[ \dot{m} \cdot \dot{m} = \pm \left( m_j^* + \frac{1}{2} \Delta m \right)^2 \]  

(4.28)

\[ = \pm \left[ m_j^2 + \Delta m \cdot m_j^* \right] \]  

(4.29)

where the (±) operator is the (+) operator if \( m_j^* \) is positive (upflow) and the (-) operator is \( m_j^* \) is negative. Substituting Eqs. (4.25), (4.28), and (4.31) into Eq. (4.14) yields

\[ \Delta p = \Delta m \cdot B + C \]  

(4.32)

where

\[ B = \frac{L}{A_f} \frac{1}{\Delta \tau} + \dot{m} \cdot I_m + \frac{2 - b}{2} \left[ m \right] \cdot \left[ 1-b \right] \cdot I_f + m \cdot I_k \]  

(4.33)

and

\[ C = m^2 \cdot I_m + m \cdot m \left[ 1-b \right] \cdot I_f + \dot{m} \cdot m \cdot I_k + I_g \]  

(4.34)

and the mass flowrates are evaluated at the present (j) time level.
4.2.2.4 **Parallel Channel Equations**

For a single channel with prescribed pressure drop \( \Delta p \), Eq. (4.32) could be solved for \( \Delta \dot{m} \), which would represent the mass flowrate increment during a time step. For a system of parallel channels, typically the pressure drop is unknown but the inlet flowrate to the system, \( \dot{m}_o \), is prescribed. Also, to account for radial pressure differences in the upper and lower plena between channels, it will be assumed that each channel has a radial pressure gradient correction term so that the channels will have the same pressure drop within an additive constant.

The continuity equation that describes the overall system mass balance is

\[
\sum_{n=1}^{N} \dot{m}_n = \dot{m}_o \tag{4.35}
\]

After differentiating this equation with respect to time, substituting Eq. (4.23), and multiplying through by \( \Delta \tau \), the continuity equation becomes

\[
\sum_{n=1}^{N} \Delta \dot{m}_n = \Delta \dot{m}_o \tag{4.36}
\]

The momentum equation becomes

\[
\Delta p + \Delta p_{rn} = \Delta \dot{m}_n \beta_n + C_n \tag{4.37}
\]

where \( \Delta p_{rn} \) is the radial pressure gradient correction term, which is prescribed. Defining \( D_n \) as below:

\[
D_n = C_n - \Delta p_{rn} \tag{4.38}
\]
Eq. (4.37) can be rewritten as

\[ \Delta p = \Delta m_n B_n + D_n \]  

(4.39)

Equations (4.36) and (4.39) comprise (N+1) equations in (N+1) unknowns (\(\Delta p\) and \(\Delta m_n\)).

### 4.2.2.5 Solution Procedure

Equation (4.39) can be written for two channels \(n\) and \(m\) and equated, yielding, after solving for \(\Delta m_m\)

\[ \Delta m_m = \frac{\Delta m_n B_n + D_n - D_m}{B_m} \]  

(4.40)

Next, Eq. (4.36) is rewritten as

\[ \Delta m_n = \Delta m_o - \sum_{m=1}^{N, m\neq n} \Delta m_m \]  

(4.41)

Substituting Eq. (4.40) into (4.41) then yields

\[ \Delta m_n = \Delta m_o - \sum_{m=1}^{N, m\neq n} \frac{\Delta m_n B_n + D_n - D_m}{B_m} \]  

(4.42)

which can be solved for \(\Delta m_m\) yielding

\[ \Delta m_n = \frac{\Delta m_o + \sum_{m=1}^{N, m\neq n} \frac{D_n - D_m}{B_m}}{1 + B_n \sum_{m=1}^{N, m\neq n} \frac{1}{B_m}} \]  

(4.43)

Equation (4.43) can be used to find \(\Delta m_n\) for all the channels.

Equation (4.39) can then be used to find \(\Delta p\). The flowrate at the new time level is subsequently given by

\[ (m_n^*)_{j+1} = (m_n^*)_j + \Delta m_n \]  

(4.44)
4.2.3 One-Dimensional Energy Equation

4.2.3.1 Spatial Averaging

The transient energy equation for axial flow through a channel can be written as

$$\frac{\partial}{\partial \tau} \rho h + \frac{\partial}{\partial z} \rho uh = -\bar{v} \cdot \bar{q''} + \frac{Dp}{D\tau} + \mu \phi^2 + q''' \quad (4.45)$$

For an incompressible fluid, density variations can be neglected for this equation. Also, the dissipation term $\mu \phi^2$ and the pressure work term $Dp/D\tau$ are negligible.

After cross-sectionally averaging the remaining terms of Eq. (4.45), the result is

$$\rho \frac{\partial}{\partial \tau} \{h\} + \frac{\partial}{\partial z} \{\rho uh\} = \frac{q'}{A_f} + \{q''''\} \quad (4.46)$$

where

$$q' = \int_{P_h} q'' dP_h \quad (4.47)$$

The RHS of Eq. (4.47) was obtained by applying the divergence theorem to the heat flux term of Eq. (4.45):

$$\int \int_{A_f} - \bar{v} \cdot \bar{q''} dA_f = \int_{P_h} q'' dP_h \quad (4.48)$$

Equation (4.46) can be multiplied by $A_f$ and rewritten as

$$\rho A_f \frac{\partial}{\partial \tau} h_b + \rho A_f \frac{\partial}{\partial \tau} (\{h\} - h_b) + m \frac{\partial}{\partial z} h_b = q' + \{q''''\} A_f \quad (4.49)$$

where the bulk enthalpy $h_b$ is defined as

$$\int \int_{A_f} \rho uh dA_f \quad (4.50)$$

$$h_b = \frac{\int_{A_f} \rho uh dA_f}{m}$$
The second term on the left hand side of Eq. (4.49) can be thought of as a correction term needed for using the bulk (as opposed to the spatially-averaged) enthalpy in the first term. For the application considered here, the detailed transient distribution of the enthalpy across the channel cross-section area is not known. This correction term will therefore be neglected. With this modification, the equation can be written as

$$\rho A_f \frac{\partial}{\partial t} h_b + \dot{m} \frac{\partial}{\partial z} h_b = q' + q''' A_f$$

(4.51)

where the volumetric heat source \(q'''\) is assumed to be spatially-averaged. The approximation of using the bulk instead of the spatially-averaged enthalpy in the transient (first) term is exact for steady-state conditions since this term will vanish.

4.2.3.2 Spatial and Temporal Discretization

In order to numerically solve the spatially-averaged energy equation, Eq. (4.51), the (axial) space and time derivatives must be replaced with finite difference expressions. The numerical method described here is the "modified Turner scheme" [T-2].

For the modified Turner scheme, the channel is assumed divided into a number of axial nodes, each node located between mesh points \(k\) and \(m\), where \(m\) can be either \(k-1\) or \(k+1\). For our notation, the axial node location will be subscripted and the time level will be superscripted. The present and future time levels are denoted \(j\) and \(j+1\), respectively.

For the modified Turner scheme, the time derivative in Eq. (4.51) is approximated by
\[ \frac{\partial}{\partial \tau} h_b = \frac{(h_b)^{j+1}_k - (h_b)^j_k}{\Delta \tau} \]  

(4.52)

The space derivative is approximated by

\[ \frac{\partial}{\partial z} h_b = \frac{(h_b)^{n+1}_k - (h_b)^{n+1}_m}{z_k - z_m} \]  

(4.53)

where \( m = k-1 \) if \( k \) increases in the direction of the flow (upflow), and \( m = k-1 \) if \( k \) decreases in the direction of the flow (downflow).

After substituting Eqs. (4.52) and (4.53) into (4.51) and solving for \((h_b)_k\), the result is

\[ (h_b)_k = \frac{\frac{q_t^i \Delta \tau}{\rho A_f} + (h_b)^j_k + \alpha (h_b)^{j+1}_m}{1 + \alpha} \]  

(4.54)

where

\[ q_t^i = q^i + q''' A_f \]

and

\[ \alpha = \frac{\frac{m}{z_k - z_m} \frac{\Delta \tau}{\rho A_f} = \frac{u}{z_k - z_m} \frac{\Delta \tau}{A_f}} \]  

(4.55)

Because the space derivative (Eq. (4.53)) is computed implicitly in time, the scheme is unconditionally stable.

4.2.4 Plena Energy Balance

Since the lower and upper plena provide the inlet temperature boundary conditions for upflow and downflow, respectively, it is necessary to know the conditions in the plena in order to model the channel behavior. If the temperature fields in the plena are not known, a first-order approach to calculate the plena conditions is to assume that the plena are perfectly mixed. An energy balance for an
adiabatic plenum with flows into and out of the liquid volume yields

\[ \rho V \frac{dh}{dt} = \sum_{\text{In}} (m\dot{h})_{\text{In}} - \sum_{\text{out}} (m\dot{h})_{\text{out}} \quad (4.56) \]

where \(V\) is the liquid volume in the plenum. The corresponding finite difference equation is

\[ \rho V \frac{h_{j+1} - h_j}{\Delta t} = \sum_{\text{In}} (m\dot{h}_{j+1})_{\text{In}} - \sum_{\text{out}} (m\dot{h}_{j+1})_{\text{out}} \quad (4.57) \]

where \(h\) and \(j+1\) refer to the present and new time step levels, respectively. Under the assumption of perfect plenum mixing, the enthalpy leaving the plenum is equal to the volume-averaged plenum enthalpy. Solving for this enthalpy at the new time step \(j+1\) yields

\[ h_{j+1} = \frac{\sum_{\text{In}} (m\dot{h}_{j+1})_{\text{In}} + \rho V \frac{h_j}{\Delta t}}{\sum_{\text{out}} (m\dot{h})_{\text{out}} + \rho V \frac{\Delta t}{\Delta t}} \quad (4.58) \]

4.2.5 Coupled Conservation Equation Solution Procedure

Now that numerical methods for the continuity, momentum, and energy equations have been presented, a methodology to solve these equations is presented. First, the mass flowrate increment for the time-step is solved for all the channels using Eq. (4.43). The mass flowrate at the new time step is then obtained by using Eq. (4.44).

The channel fluid enthalpies are updated next. If the flowrate in a channel indicates upflow (positive flowrate), the enthalpy is updated using Eq. (4.54), starting with the lowest node in the channel and marching up the channel, thereby using the lower plenum enthalpy as the boundary condition. If the flowrate indicates downflow (negative mass flowrate), then Eq. (4.54) is used again, this time starting at the highest node in the channel and marching down the
channel, thereby using the upper plenum enthalpy as the boundary condition. If this procedure is followed, all terms on the right hand side of Eq. (4.54) are explicitly known for each axial level k.

The enthalpy of the fluid in the upper and lower plenum are then updated. With channel flowrates and enthalpies now known, all variables in the right hand side of Eq. (4.58) are explicitly known, and thus it can be used to find the plena enthalpies at the new time step level.

Once the enthalpies have been updated in all the nodes, a properties package can update any other desired property in terms of the enthalpy and reference pressure.

4.3 The Code MICON

The code MICON [K-2,K-3] was written to numerically predict the transient behavior of parallel channels connected at plena. The code solves the conservation equations in the manner described in Section 4.2.2.5. In addition to the modified Turner scheme, energy equation finite difference formulations include the donor cell scheme [R-1], Turner scheme [T-3,K-4], and box method [A-1]. Also, a semi-analytical energy equation formulation based on the method of characteristics is included. Fluid property packages exist for light water, heavy water, and sodium, which the user must choose.

In addition to the conservation equations solved for the fluid regions, a transient heat conduction model for the solid structure which allows for thermal storage will be available in a future version. The model is being developed to calculate channel heat loss to the environment. It utilizes the finite difference form of the one-dimensional heat conduction equation in cylindrical co-ordinates,
and assumes negligible axial conduction and azimuthal variations. The
heat conduction model will not be described here because it is not
included in the present version of the code.

A more detailed description of MICON is presented in Appendix D,
which includes the input file data organization.

4.4 Numerical Prediction of the Shutdown Core Coolability Model Test

4.4.1 Description of Experimental Apparatus

The General Electric Shutdown Core Coolability Model Test (SCCMT)
was designed to study buoyancy-induced flow redistribution within an
LMFBR core for low flow, large power-to-flow conditions. In
particular, the objectives of the test were to determine the effect of
buoyancy-induced flow redistribution among fuel, blanket, shield, and
bypass assemblies, and to examine the thermal hydraulic core
conditions for the transition from forced convection to buoyancy
dominated flow. The SCCMT was modeled after the CDS Phase I Reactor
[S-1].

The SCCMT consists of seven parallel flow channels, connected at
the top and bottom by an inlet and outlet plenum. The dimensions of
the channels and plena are listed in Table 4.1 and a drawing of the
test section is shown in Figure 4.1.

Six of the seven flow channels are annular in geometry and
represent core assemblies. Of the six annular channels, three are
heated, representing two fuel assemblies and an internal blanket
assembly. The other three annular channels represent two radial
blanket and one shield assemblies. In addition to the annular
channels, there is a circular bypass channel. The location of these
channels as they enter the upper plenum can be seen in Figure 4.2.
Table 4.1

SCCMT Channel and Plena Dimensions

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Fuel Channels</td>
<td>2</td>
</tr>
<tr>
<td>Number of Internal Blanket Channels</td>
<td>1</td>
</tr>
<tr>
<td>Number of Radial Blanket Channels</td>
<td>2</td>
</tr>
<tr>
<td>Number of Radial Shield Channels</td>
<td>1</td>
</tr>
<tr>
<td>Number of Bypass Channels</td>
<td>1</td>
</tr>
<tr>
<td>Number of Total Flow Channels</td>
<td>7</td>
</tr>
<tr>
<td>Fuel Channel Flow Area</td>
<td>0.229 in$^2$ EA</td>
</tr>
<tr>
<td>Internal Blanket Flow Area</td>
<td>0.185 in$^2$</td>
</tr>
<tr>
<td>Radial Blanket Flow Area</td>
<td>0.185 in$^2$ EA</td>
</tr>
<tr>
<td>Radial Shield Flow Area</td>
<td>0.173 in$^2$</td>
</tr>
<tr>
<td>Bypass Flow Area</td>
<td>0.196 in$^2$</td>
</tr>
<tr>
<td>Channel Length</td>
<td>160.0 in</td>
</tr>
<tr>
<td>Heated Length</td>
<td>50.0 in</td>
</tr>
<tr>
<td>Plenum Diameter</td>
<td>25.0 in ID</td>
</tr>
</tbody>
</table>
Figure 4.1 The GE Shutdown Core Coolability Model Test (SCCMT) experiment with dimensions. (Taken from Ref. S-1)
Figure 4.2 Location of the channels for the SCCMT. (Taken from Ref. S-1)
The test loop for the experiment is shown in Figure 4.3. Demineralized water was circulated through the loop by a variable speed pump, which could be controlled to provide flowrates between 0 and 100 percent of full flow as a function of the predetermined coastdown curve. A loop heat exchanger controlled the fluid temperature at the inlet of the test section.

In addition, throttle valves were located in the seven channels. The valves were adjusted before each test to set the desired steady-state flow split among the channels. The main loop and bypass channel also had fast action isolation valves that could be operated manually during a test.

Extensive fluid temperature and flowrate measurements were made in the seven channels. Thermocouples were located at the channel inlet, outlet, and just downstream of the heated length. Bi-directional flowmeters were used to measure the low flowrates encountered. In addition, the loop included a uni-directional mainloop flowmeter, and three wattmeters to measure power inputs to the heated channels.

4.4.2 Description of Experimental Conditions

The experiment analyzed in this report corresponds to GE Shutdown Core Coolability Model Test A3, as described in [5-1]. This test was chosen to model because it included all the features of the experimental apparatus. For this test, the individual channels were throttled, the bi-directional flowmeters were operational throughout the transient, the mainloop was isolated after a flow coastdown, and the isolation valve for the bypass channel was opened during the transient. To initialize the experiment, the loop was first brought
Figure 4.3 The SCCMT loop diagram.
(Taken from Ref. S-1)
to a steady-state condition with a main loop flow of \( 7.835 \times 10^{-3} \text{ m}^3/\text{s} \) (2.07 gpm). Table 4.2 lists the associated channel heater power, steady-state volumetric flowrates, and temperatures for this test. The throttle valves in the channels were adjusted so that the ratio of the total flow through one type of assembly to the total flow through the test section matched that for the CDS Phase 1 design [S-1]. Heater powers were set at 50% of prototypical decay power levels. The isolation valve in the bypass channel was initially closed so that the initial flow in that channel was zero.

The fluid in the upper and lower plena was at 70°F (21.1°C) and 77°F (25°C), respectively. The experiments were of sufficiently short duration that these temperatures did not change appreciably during data acquisition.

After steady-state conditions were maintained for twenty seconds, the main loop pump began a coastdown, as shown in the normalized flow coastdown function of Figure 4.4. At 220 seconds into the transient, the pump had completely coasted down. The main loop isolation valve was then shut, and the isolation valve to the bypass channel manually opened. Temperature and flowrate data was recorded for an additional 380 seconds, so that the data is available for a total of 600 seconds.

4.4.3 Experimental Results

The SCCMT results showed a number of important buoyancy-induced phenomena. All channels coasted down at about the same rate during the pump coastdown, as can be seen in Figure 4.5 and 4.6. After the main loop valve was closed at 220 seconds, the channels with the largest steady-state power-to-flow ratios drew the greatest flow due to their higher temperature and thus greater fluid buoyancy. The flow in the fuel channels (F1, F2) experienced upflow while the adiabatic
Table 4.2
SCCMT Steady-State Conditions

<table>
<thead>
<tr>
<th>Channel</th>
<th>Flow (gpm)</th>
<th>Power (W)</th>
<th>Inlet Temp. (°F)</th>
<th>Outlet Temp. (°F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>0.61</td>
<td>352</td>
<td>70.4</td>
<td>73.2</td>
</tr>
<tr>
<td>F2</td>
<td>0.60</td>
<td>458</td>
<td>70.4</td>
<td>74.9</td>
</tr>
<tr>
<td>IB</td>
<td>0.50</td>
<td>202</td>
<td>70.4</td>
<td>72.6</td>
</tr>
<tr>
<td>RB1</td>
<td>0.18</td>
<td>0</td>
<td>70.4</td>
<td>72.6</td>
</tr>
<tr>
<td>RB2</td>
<td>0.15</td>
<td>0</td>
<td>69.8</td>
<td>71.0</td>
</tr>
<tr>
<td>BY</td>
<td>0</td>
<td>0</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>MAIN</td>
<td>2.07</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
Figure 4.4: Main loop coastdown curve for the SCCM1.
(Taken from Ref. 5-1)
Figure 4.5 Experimental flows in the F1, F2, IB, and Bypass channels for the SCCMT.
(Taken from Ref. S-1)
Figure 4.6 Experimental flows in the RB1, RB2, and RS channels for the SCCMT.
(Taken from Ref. S-1)
channels (RB1, RB2, and RS) experienced a slight downflow. The internal blanket channel (IB) experienced a period of flow stagnation after the coastdown, lasting until 500 seconds into the transient. A strong downflow was recorded in the bypass channel once its isolation valve was opened at 220 seconds into the transient.

Figure 4.7 shows the coolant temperatures of the heated channels at an axial location just above the heated length. The peak temperatures for the F1 and F2 channels corresponded to that time period after flow coastdown and before the natural circulation flows in the channels were established. The temperatures then stabilized at slightly lower temperatures.

For the internal blanket channel (IB), significant flow stagnation was observed. During this stagnation, the temperature fluctuated greatly. After 400 seconds, there was a rapid rise in temperature of the channel fluid. This is an interesting case because the channels with the highest steady-state temperature rise (F1 and F2) did not experience the highest peak transient temperatures.

All the heated channels show temperature fluctuations. It was reported in Ref. S-1 that injected dye at the heated channels' exit served to visualize a local flow recirculation pattern caused by the mixing of cold upper plenum fluid with hot fluid from the channel. An analysis of this pattern is presented in Chapter 6.

4.4.4 MICON Model

4.4.4.1 Noding Scheme

The MICON noding scheme used to evaluate the SCCMT results is shown in Figure 4.8. For the heated channels (F1, F2, and IB), thirteen axial nodes were used, including five for the heated length. For the unheated channels, five axial nodes were used. The upper and
Figure 4.7 Experimental peak coolant temperatures of the heated channels for the SCCMT.
(Taken from Ref. S-1)
Figure 4.8 MICON noding scheme for the SCCMT.
lower plena were each modeled as a single node, in accordance with the perfect plena mixing model incorporated into MICON.

4.4.4.2 Forced Convection Flow Resistance Modeling

The channels in the test section are all annular, with the exception of the bypass channel, which is circular. The friction factors for circular and annular channels for forced convection laminar flow are well known and can easily be derived from the Navier-Stokes equation for an incompressible fluid. Thus, it would seem that flow resistance modeling would consist of simply calculating the correct friction factor for the particular channel geometry. The forced convection friction factors used are listed in Table 4.3. Unfortunately there were other flow resistances in the channel. Throttle valves were present in the channels and were adjusted to obtain the desired flow split among the channels. Also, the flowmeters in the channels contributed to additional flow resistance.

In Ref. S-1, measurements were taken of the volumetric flow split for different values of the total pressure drop from the lower to the upper plenum. Theses curves are shown in Figure 4.9. After subtracting the gravity and friction pressure drops from these curves, a least-square power curve fit was found in Ref. S-1 for the remaining valve and flowmeter resistance. The results are listed in Table 4.4 for the F1, F2, IB, RB1, RB2, and RS channels. For the bypass channel, which was not throttled, the constants listed in the table are those reported for the bi-directional flowmeter.

In Table 4.4, the correlations for the valve and flowmeter pressure loss are of the form

\[ \Delta p_{\text{local}} = k \cdot Q^n \]  

(4.59)
Table 4.3

Forced Convection Friction Factor Correlation Coefficients Used as MICON Inputs for the SCCMT

\[ f = c \operatorname{Re}^{-b} \]

<table>
<thead>
<tr>
<th>Channel</th>
<th>Laminar</th>
<th>Turbulent</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>c1</td>
<td>c2</td>
</tr>
<tr>
<td>F1</td>
<td>95.2</td>
<td>1</td>
</tr>
<tr>
<td>F2</td>
<td>95.2</td>
<td>1</td>
</tr>
<tr>
<td>IB</td>
<td>95.6</td>
<td>1</td>
</tr>
<tr>
<td>RB1</td>
<td>95.6</td>
<td>1</td>
</tr>
<tr>
<td>RB2</td>
<td>95.6</td>
<td>1</td>
</tr>
<tr>
<td>RS</td>
<td>95.6</td>
<td>1</td>
</tr>
<tr>
<td>BY</td>
<td>64</td>
<td>1</td>
</tr>
</tbody>
</table>
Figure 4.9  Total pressure drop versus volumetric flow for the SCCMT. (Taken from Ref. S-1)
Table 4.4
Least Squares Power Fit for Channel Valve and Flowmeter Pressure Drop
for the SCCMT

\[ \Delta p = kQ^n \quad \Delta p \text{ in Pa} \]
\[ Q \text{ in } m^3/s \]

<table>
<thead>
<tr>
<th>Channel</th>
<th>( k )</th>
<th>( n )</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>2.988 \times 10^{12}</td>
<td>1.806</td>
</tr>
<tr>
<td>F2</td>
<td>2.988 \times 10^{12}</td>
<td>1.806</td>
</tr>
<tr>
<td>IB</td>
<td>9.647 \times 10^{11}</td>
<td>1.679</td>
</tr>
<tr>
<td>RB1</td>
<td>1.836 \times 10^{11}</td>
<td>1.373</td>
</tr>
<tr>
<td>RB2</td>
<td>4.196 \times 10^{11}</td>
<td>1.432</td>
</tr>
<tr>
<td>RS</td>
<td>4.714 \times 10^{11}</td>
<td>1.279</td>
</tr>
<tr>
<td>BY</td>
<td>9.573 \times 10^{12}</td>
<td>2</td>
</tr>
</tbody>
</table>
where \( Q \) is the volumetric flowrate.

MICON requires the local pressure drop to be of the form

\[
\Delta p = K \frac{C^2}{Z_p}
\]  

(4.60)

where the code was modified to allow

\[
K = C \cdot Re^{-B}
\]  

(4.61)

In addition,

\[
Q = uA_f
\]  

(4.62)

\[
G = \rho u
\]  

(4.63)

and

\[
Re = \frac{\rho u D_e}{\mu}
\]  

(4.64)

Combining Eqs. (4.59) through (4.64) and solving for \( C \) and \( B \) yields

\[
C = \frac{2kA_r^nD_e^{2-n}}{\mu^{2-n}\rho^{n-1}}
\]  

(4.65)

\[
B = 2-n
\]  

(4.66)

For water at 100°F, \( \rho = 994 \, \text{kg/m}^3 \) and \( \mu = 6.827 \times 10^{-4} \, \text{Pa-s} \). Table 4.5 lists the corresponding input parameters used by MICON, computed by substituting the fluid properties and channel geometrical parameters into Eqs. (4.65) and (4.66).

4.4.4.3 Mixed Convection Friction Factor Modeling

MICON requires as input a table of values for \( (f/f_0)(1+F) \) versus \( Gr_q/Re \) for all the channels. For the SCCMT, the F1 and F2 channels have \( \alpha = 0.6288 \) and the IB channel has \( \alpha = 0.420 \), where \( \alpha \) is the inner-to-outer radius ratio described in Chapter 3. Using the analyses of that chapter for the inner radius heat annulus, \( (f/f_0)(1+F) \) was calculated and the results were used as input to the
Table 4.5
Channel Valve and Flowmeter Pressure Drop Input Parameters for MICON
for the SCCMT

\[ K = C \cdot \text{Re} - B \]

<table>
<thead>
<tr>
<th>Channel</th>
<th>K</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>(4.466 \times 10^3)</td>
<td>0.194</td>
</tr>
<tr>
<td>F2</td>
<td>(4.466 \times 10^3)</td>
<td>0.194</td>
</tr>
<tr>
<td>IB</td>
<td>(9.241 \times 10^3)</td>
<td>0.321</td>
</tr>
<tr>
<td>RB1</td>
<td>(4.457 \times 10^5)</td>
<td>0.627</td>
</tr>
<tr>
<td>RB2</td>
<td>(3.504 \times 10^5)</td>
<td>0.568</td>
</tr>
<tr>
<td>RS</td>
<td>(5.405 \times 10^6)</td>
<td>0.721</td>
</tr>
<tr>
<td>BY</td>
<td>(3.092 \times 10^2)</td>
<td>0</td>
</tr>
</tbody>
</table>
One assumption in the derivation of the \((f/f_0)^{(1+F)}\) parameter is that of fully developed flow. We would expect that for developing flow, other important parameters would be the channel \(L/D_e\) ratio, and the fluid Prandtl number. In Ref. B-1, an attempt was made to separate the developing flow effect from the fully developed friction factor ratio for water in a heated circular tube. A friction factor ratio of the form

\[
f/f_0 = 1 + G[(f/f_0)_{FD} - 1]
\]

was presented, where \(f/f_0\) represents the average friction factor over the length, \((f/f_0)_{FD}\) represents the fully-developed friction factor ratio, and the factor \(G\), representing the geometry effect, asymptotically approaches one for large \(L/D_e\). Figure 4.10 shows the experimental data used to determine the \(L/D_e\) effect for water in a circular tube. For the SCCMT fuel channels, \(L/D_e = 516\), and for the internal blanket channel, \(L/D_e = 690\). If the circular tube geometry effect results are applicable to an inside heated annulus, Figure 4.10 shows that there should be no error when using the fully-developed analysis for these channels due to their large \(L/D_e\) ratios (\(L/D_e \gg 90\)).

4.4.5 MICON Results and Discussion

4.4.5.1 Steady-State Results

Table 4.6 compares the experimental steady-state volumetric flowrates with those calculated by MICON. There is, in general, good agreement between the two. The maximum discrepancy is the 8.33\% (9.5\times10^{-7} m^3/s) MICON underprediction for channel RB1. It must be remembered that the values for \(K\) derived in Section 4.4.4.2 were least-square fits from the experimental data over a range of pressure
Figure 4.10 Friction factor ratio data for different $L/D_e$'s.
(Taken from Ref. B-1)
Table 4.6
Steady-State Flowrates Predicted by MICON for the SCCMT

<table>
<thead>
<tr>
<th>Channel</th>
<th>Q calculated (m³/s)</th>
<th>Q experimental (m³/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>3.773E-5</td>
<td>3.785E-5</td>
</tr>
<tr>
<td>F2</td>
<td>3.779E-5</td>
<td>3.785E-5</td>
</tr>
<tr>
<td>IB</td>
<td>3.293E-5</td>
<td>3.155E-5</td>
</tr>
<tr>
<td>RB1</td>
<td>1.041E-5</td>
<td>1.136E-5</td>
</tr>
<tr>
<td>RB2</td>
<td>9.590E-6</td>
<td>9.464E-6</td>
</tr>
<tr>
<td>RS</td>
<td>2.145E-6</td>
<td>2.524E-6</td>
</tr>
<tr>
<td>BY</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>MAIN</td>
<td>1.306E-4</td>
<td>1.306E-4</td>
</tr>
</tbody>
</table>
drops. As a result, it is not reasonable to expect a perfect match of experimental and calculated flowrates for a given operating condition. We conclude that MICON adequately matches the steady-state conditions for the SCCMT.

4.4.5.2 Transient Results

Figure 4.11 compares the input and experimental coastdown of the main loop flow for the test. In MICON, the main loop flowrate fraction must be input as a function of time. The experimental and calculated flows should therefore agree exactly.

Figure 4.12 shows the MICON prediction of the volumetric flow in the F1 channel. We see excellent agreement between the calculated and experimental values. Similarly, Figure 4.13 shows the experimental and calculated flow in the F2 channel. Again, the calculated flowrates closely match those predicted by MICON.

Figure 4.14 compares the experimental and calculated flowrates for the IB channel. It can be seen that MICON overpredicts the flow in this channel after the bypass isolation valve is opened at 220 seconds. By observing the experimental data, a long period of flow stagnation can be seen. After 500 seconds, the data indicates that a small upward flow was finally established in this channel.

It must be remembered that the channel flowrate data collected is from bi-directional flowmeters located at the inlets of the channels. For one-dimensional axial flow in the channel, the flowmeter should adequately measure the flowrate. It is suspected that under low flow conditions, the hot fluid rising up the channel may interact with the colder upper plenum fluid and produce a multi-dimensional flow field.
Figure 4.11  Predicted and experimental flow in the main loop for the SCCMT.
Figure 4.12  Predicted and experimental flow in the F1 channel for the SCCMT.
Figure 4.13  Predicted and experimental flow in the F2 channel for the SCCMT.
Figure 4.14  Predicted and experimental flow in the IB channel for the SCCMT.
Under these conditions, little information can be gained from the flowmeter. This multi-dimensional flow field will be discussed in Chapter 6.

MICON predicted higher F1 and F2 channel fluid temperatures than the experimental data collected. Figures 4.15 and 4.16 show the calculated versus experimental temperatures at an axial location just above the heated length of the channels. It is not clear why this discrepancy exists in light of the good agreement of the predicted versus calculated flowrates for these channels. One explanation is that heat loss to the environment occurred in these channels, and was not modeled. Heat loss would tend to lower fluid temperatures in these channels.

There is a large discrepancy between the data and the MICON prediction for the internal blanket channel fluid temperature. Figure 4.17 shows the calculated and measured temperatures at an axial location just above the heated length. MICON predicts a large temperature excursion after the pump coastdown. The data, however, shows much lower temperatures during this period. If we limit our thinking to one-dimensional flow, this temperature difference between experimental and calculated values is impossible, since the flowmeter data shows that flow stagnation was occurring in this channel while MICON predicted weak upflow. If fluid from the upper plenum did backflow into the channel, the mixing of colder fluid from the upper plenum with the fluid in the channel would effectively act as a heat sink for the channel, thus producing lower temperatures and flowrates then predicted by the one-dimensional model. This mixing could also be the cause of the large temperature fluctuations that can be seen in
Figure 4.15 Predicted and experimental peak fluid temperatures in the F1 channel for the SCCMT.
Figure 4.16 Predicted and experimental peak fluid temperatures in the F2 channel for the SCCMT.
Figure 4.17 Predicted and experimental peak fluid temperatures in the IB channel for the SCCMT.
Figure 4.7. An analysis of the onset of this backflow is presented in Chapter 6.
CHAPTER FIVE

THE EXPERIMENTAL APPARATUS

5.1 Introduction

The purpose of constructing the test section was to observe the behavior of parallel channels connected to upper and lower plena in the mixed convection regime. The test section was constructed so that water could be used as the working fluid for design and operational simplicity. The materials of the test section were chosen so that flow visualization using dye injection or other more complicated procedures could be performed. In addition, maximum use of existing facilities in the Thermal Hydraulics Laboratory was desired.

The test section consists of three parallel channels, and is shown schematically in Figure 5.1. Two channels were designed to be heated so that power skew conditions could be experimentally simulated, and the third adiabatic. In a more specific sense, one channel could represent the high power-to-flow fuel channels and the other channel could represent the low power-to-flow blanket channels in a LMFBR core. Since the fuel and blanket channels account for about 95% of the core flow in a typical LMFBR, flow redistribution experiments were run with the adiabatic channel blocked.

5.2 Test Section Design and Construction

5.2.1 The Channels

The three channels to the test section are all of equal length. The two heated channels are annular in cross-section. Each consists of a clear, acrylic tube with a heated rod down the center. The adiabatic channel is a clear acrylic tube of circular cross-section. Clear plastic was chosen as the tube wall material because it allows flow visualization with dye injection, and is easy to machine.
Figure 5.1 Schematic drawing of the parallel channel test section.
Figure 5.2  Screw positioners for the annular channels.
The dimensions of the heated channels (CH1 and CH2) were determined from the following considerations:

(a) The channels should use the existing heated rods already available in the Thermal Hydraulics Laboratory. These rods are 1.27 cm (0.5 in) in diameter ($D_1 = 1.27$ cm).

(b) The $L/D_e$ of the channel should be large enough so that hydrodynamic developing flow effects are negligible over the length of the tube for $Re < 500$.

(c) The length of the channel should be less than 1.524 m (60 in) since this is a standard tube length.

For laminar developing flow, the entrance length $L_e$ is given by (Ref. W-3, p.338)

$$L_e/D_e \approx 0.08Re \quad (5.1)$$

For an annular duct, the equivalent diameter is given by

$$D_e = D_o - D_1 \quad \text{annular duct} \quad (5.2)$$

where $D_o$ is the inner radius of the tube wall, and $D_1$ is the rod diameter. In order that consideration (b) be satisfied, $L_e$ should be less than 20% of $L$, i.e.,

$$L > 5L_e \quad (5.3)$$

Combining Eqs. (5.1), (5.2), and (5.3) results in

$$L > 0.4 (D_o - D_1) \text{ Re} \quad (5.4)$$

At the maximum Reynolds number (from consideration (b), $Re = 500$), this becomes

$$L > 200 (D_o - D_1) \quad (5.5)$$

Equation (5.5) is satisfied for $L = 142.88$ cm (56.25 in) and $D_o = 1.91$ cm (0.75 in) utilizing $D_1 = 1.27$ cm from consideration (9).
These were the dimensions chosen for the annular channels.

In order to maintain a concentrically annular cross-section for
the heated channels, it was necessary to drill and tap holes for three
screws to position the rods in the center of the channel, as shown in
Figure 5.2. The screw positioners were located at three different
axial positions along the channel so that each channel had nine screws
positioning the rod in the tube center. The screw holes were sealed
with silicone sealant to prevent leaks.

The diameter of the adiabatic channel (CH3) was chosen so that if
the inlet flowrate to the test section were zero, and upflow existed
in the two heated channels, the downflow through the adiabatic channel
would cause essentially zero friction pressure drop for that channel.
This was accomplished by setting the channel diameter such that the
flow area was many times that for the heated channels. For a diameter
of 4.45 cm (1.75 in), the adiabatic channel flow area is 9.8 times
that of each heated channel. For these dimensions, the downflow
pressure drop would be less than 1% of the upflow pressure drop. The
channel dimensions for the test section are listed in Table 5.1.

In addition, a mechanism was provided so that the channels could
be individually blocked. For the annular channels, a tapered cork
stopper was drilled so that it could be inserted at the top of the
channels, as shown in Figure 5.3a. For the adiabatic channel, a
circular cross-section was cut from 0.64 cm (0.25 in) plastic and a
chamfer was machined into the edge so that it could accommodate an
O-ring. The O-ring was then coated with vacuum grease and inserted at
the top of the adiabatic channel in order to seal it, as shown in
Figure 5.3b.
Table 5.1

Channel Dimensions for the Parallel Channel Test Section

<table>
<thead>
<tr>
<th>Cross-sectional Shape</th>
<th>CH1 and CH2</th>
<th>CH3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rod Diameter, cm (in)</td>
<td>Annular</td>
<td>Circular</td>
</tr>
<tr>
<td>Tube Inner Wall Diameter, cm (in)</td>
<td>1.27 (0.5)</td>
<td>-</td>
</tr>
<tr>
<td>Equivalent Diameter, cm (in)</td>
<td>1.91 (0.75)</td>
<td>4.45 (1.75)</td>
</tr>
<tr>
<td>Tube Outer Wall Diameter, cm (in)</td>
<td>0.635 (0.25)</td>
<td>4.45 (1.75)</td>
</tr>
<tr>
<td>Heated Length, cm (in)</td>
<td>91.44 (36)</td>
<td>-</td>
</tr>
<tr>
<td>Total Length, cm (in)</td>
<td>142.88 (56.25)</td>
<td>142.88 (56.25)</td>
</tr>
</tbody>
</table>
Figure 5.3 Mechanism to block individual channels for the parallel channel test section.
5.2.2 The Heated Rods

A diagram of the heated rods that were available in the Thermal Hydraulics Laboratory is shown in Figure 5.4. The rods are constructed of stainless steel tube and filled with a low conductivity powder (MgO). They are rated for a maximum heating power of 2 kW each, with a uniform heat flux along the heated length. The electrical resistance is approximated 20 Ω. The material properties of the rods are not important here because only steady-state experiments were run so that the thermal inertia is neglected, and the channels are axisymmetric, so that non-radial conduction is also negligible. It was desired that the rods should not extend above the channel exit into the upper plenum so that the upper plenum flow field would not be influenced by the presence of the rods, and also so that the rods would not obstruct the flow visualization. Since the power wires for the rods are not water immersible, the rods extended into the lower plenum, and out the bottom of the lower plenum. The location at which the rod penetrated the lower plenum was sealed with a SwageLock™ compression fitting.

The lower unheated length of the rod was chosen to be the height of the lower plenum, so that no heating of the lower plenum by the rods occurred. The remaining portion of the rod consisted of 91.44 cm (36 in) of heated length, and an upper unheated length of 5.72 cm (2.25 in), for a total length of 97.16 cm (38.25 in). Since the channel length is 142.88 cm, a 45.72 cm (18 in) extender was added to the rods so that the rod extended to the top of the channel. The extender was constructed of 1.27 cm (0.5 in) O.D. stainless steel tubing. The extender and rod were joined by a stainless steel coupler, which consisted of a 1.27 cm (0.5 in) I.D., 1.59 cm
Figure 5.4 Heated rods used for the parallel channel test section.
(0.625 in) O.D. tube of 2.54 cm (1.00 in) length. This is shown in Figure 5.3. The result of attaching the extender to the rod was to produce a new rod of 91.44 cm (36 in) of heated length, and 51.44 cm (20.25 in) of unheated length.

5.2.3 The Upper and Lower Plena

The upper and lower plena were designed to be large enough so that fluid velocites were small enough so that there would be essentially no radial pressure gradients in the two plena. The material chosen was clear acrylic plastic because of its availability and transparency. A thickness of 0.64 cm (0.25 in) was adequate since the water in the test section is not pressurized. The plena are shown, with dimensions, in Figure 5.5.

In order to maintain a water level in the upper plenum, the outlet piping of the upper plenum does not travel directly downward to the drain. Instead, the path is first upwards and then downward so that the level in the upper plenum is maintained at the level of the upper bend in the pipe. This is essentially the same plumbing construction found in household sink drain "traps."

A method was devised to seal the penetration where the channels entered the plena and is shown in Figure 5.6. An annular ring was first cut from the 0.64 cm (0.25 in) plastic such that the inner diameter of the annulus equaled the outer diameter of the channel tube wall. The inner radius of the ring was then chamfered (beveled) to the shape of an O-ring. Eight holes for screws were then drilled and tapped through the ring and into the plenum wall. As these screws were tightened, the O-ring was compressed and therefore sealed the penetration.
Figure 5.5 Upper and lower plena for the parallel channel test section.
Figure 5.6 Channel penetration into the plena for the parallel channel test section.
5.3 The Flow Loop, Instrumentation, and Experimental Operation

5.3.1 Flow Loop

The purpose of the flow loop is to provide flowrate and inlet water to the test section at the desired flowrate and inlet temperature. Most of the loop was built by Wang, Rohsenow, and Todreas [W-1] for the MIT 4x4 bare rod bundle. Modifications were made by Efthimiades and Todreas [E-1] so that the loop could provide flow to the MIT 19-pin wire-wrapped rod bundle. Additional modifications were made here in order to use the loop for the parallel channel test section. Because the loop now services three different test sections, there are many components that were not used for this work. Figure 5.7 shows a simplified schematic of the loop with the components used in this work.

5.3.2 Flowmeters

Three rotameter flowmeters (FM1, FM2, and FM3) were used to measure the volumetric flowrate delivered to the test section. The flowmeters were individually calibrated in Ref. E-1 to be:

FM1: \( Q_{\text{max}} = 3.10 \times 10^{-4} \text{ m}^3/\text{s} \pm 0.75\% \)

FM2: \( Q_{\text{max}} = 1.24 \times 10^{-4} \text{ m}^3/\text{s} \pm 0.75\% \)

FM3: \( Q_{\text{max}} = 1.70 \times 10^{-5} \text{ m}^3/\text{s} \pm 0.75\% \)

where \( Q_{\text{max}} \) represents the full-scale value. FM1 and FM2 have scale divisions of 5% of full-scale, while FM3 has 1% scale divisions.

5.3.3 Thermocouples

Temperature measurements were made in the test section using copper-constantan thermocouples. Six thermocouples (TC1 through TC6) were located in the channels, at location 2.54 cm (1.00 in) from the inlet and outlet of each channel, as shown in Figure 5.8. In
Figure 5.7 Flow loop for the parallel channel test section.
Figure 5.8 Location of thermocouples for the parallel channel test section.
addition, a thermocouple (TC7) was placed in the upper plenum on the lower surface 1.27 cm (0.5 in) from where CH2 entered the plenum. Finally, a thermocouple (TC8) was used to measure the ambient air temperature.

The thermocouples were connected to a ten channel digital thermometer (Omega model 2176 A) that has a 0.1°F resolution. No ice-bath is necessary at the reference junction because the thermometer has a reference junction temperature compensation circuit that detects the temperature at the reference junction using a temperature sensitive transistor and corrects the temperature read-out. In addition, all temperatures were measured by the same digital thermometer so that bias errors caused by reference junction temperature compensation circuit errors were systematically eliminated by subtracting the readings of any two thermocouples. Since only temperature differences were used in the calculations, the accuracy of the temperature measurement approaches the resolution of the instrument.

5.3.4 Heated Rod Electrical Power

Each heated rod was heated using a Techni-power™ dual coil autotransformer variac. The two variacs used were part of a bank of variacs also used to provide electrical power for the 4×4 bare and 19-pin wire-wrapped rod bundles. The electrical power to the rods was estimated using a voltmeter across the heater cables and an ammeter placed in series with them, as shown in Figure 5.9. VM1 and VM2 measured the voltages, and AM1 and AM2 measured the currents for CH1 and CH2, respectively.

The accuracy of the voltage and current measurements was limited by the resolution of the scales of the respective voltmeter and
Figure 5.9  Power supply circuit for the parallel channel test section.
(Taken from Ref. E-1)
ammeter. The voltmeter scale has 10 V divisions, while that of the ammeter is 0.2 A. Since the electrical power to the rod can be calculated from

\[ q = E \cdot I \]  

(5.6)

the maximum error in computing the power is given by

\[ \Delta q = \Delta I \cdot E + \Delta E \cdot I \]  

(5.7)

where \( \Delta E \) and \( \Delta I \) are the errors associated with the scale resolution of the voltmeter and ammeter, respectively, and thus \( \Delta E = 10 \) V and \( \Delta I = 0.2 \) A.

In order to calculate the maximum percent error of the electrical power estimation, \( E \) and \( I \) will be eliminated from Eq. (5.7) in favor of the electrical power \( q \) and the rod electrical resistance \( R \). Ohm's law, which is

\[ E = I \cdot R \]  

(5.8)

is used in Eq. (5.6) to first eliminate \( E \), and then \( I \), producing

\[ q = E^2 / R \]  

(5.9a)

\[ q = I^2 R \]  

(5.9b)

This equation is then used to eliminate \( E \) and \( I \) from Eq. (5.7), producing

\[ \Delta q = \Delta I[q \cdot R]^{0.5} + \Delta E[\dot{q}/R]^{0.5} \]  

(5.10)

which can be rewritten as

\[ \Delta q = q^{0.5}[\Delta I \cdot R^{0.5} + \Delta E / R^{0.5}] \]  

(5.11)

Substituting \( \Delta E = 10 \) V, \( \Delta I = 0.2 \) A, and \( R = 20 \) \( \Omega \), and rearranging produces

\[ \Delta q / q = 3.130 / q^{0.5} \]  

(5.12)

The percent error in the estimation of the electrical power is thus inversely proportional to the square root of the power. If the
maximum allowable error is 5%, the heater power must be greater than 3.9 kW. Since the rods are only rated for 2kW, a large error results from using the voltmeters and ammeters for electrical power determination. More typically, the rods are heated from 100-750 W. The associated percent error range is 12.09-33.10%. For this reason, the voltmeters and ammeters were used only to estimate the power. The actual rod power used for calculations was obtained using methods that will be described in Chapter 7.

5.3.5 Heat Loss Measurement

Data was taken to insure that negligible heat loss occurred in the insulated heated channels. This was accomplished by blocking the adiabatic channel (CH3) and the annular channel not being studied so that the loop flow traveled solely up the annular channel of interest. The power to this channel was set to zero. The inlet temperature to the test section was, in general, different from that of the ambient air so that heat loss (or gain) would occur. By measuring the temperature drop and flowrate in the channel, the heat loss could be calculated.

The inlet temperature to the test section was varied so that 
\[-2.4 \leq \Delta T_{in} \leq 18.0^\circ C\], where \( \Delta T_{in} \) represents the temperature difference measured between the channel inlet and the ambient air. At a channel flowrate of \( 5.07 \times 10^{-3} \) kg/s, the maximum heat loss was measured to be approximately 2 W, which occurred for \( \Delta T_{in} = 18.0^\circ C \), for CH2. Since the heat loss measured for the two annular channels was so small (on the order of the accuracy limit imposed by the resolution digital thermometer) it can be assumed to be zero, and thus no attempt was made to correlate this heat loss.
5.3.6 Experimental Operation

In general, operation of the loop is straightforward, and consists mainly of aligning the valves so that the fluid has a direct path from the water tank to the pumps, and to the parallel channel test section. The flowrate is adjusted by opening or closing the valves just upstream of the flowmeters (V1, V2, and V3). For small flowrates (less than $2.480 \times 10^{-5}$ m$^3$/s), the pump heating of the water becomes excessive so that the bypass valve (V4) must be opened to increase the flow through the pump, thereby lowering the inlet temperature to the test section, and increasing the pump cooling by the water.

Before the water in the loop can be heated, it must be degassed. If the water is not degassed, upon heating many air bubbles will appear in the channel. This is because the air solubility decreases with increasing temperature. The degassing is therefore accomplished by increasing the water tank temperature to a higher value than the maximum temperature expected during an experiment. The water is then cooled to the ambient temperature.

The heating of the water is accomplished by injecting superheated steam into the water tank until boiling begins in the tank. Once boiling has initiated, the steam supply is terminated and the tank is allowed to cool. Water cooling typically takes about 24 hours, during which the loop is inoperable. After cooling is completed, experimental operation of the loop can be performed.
CHAPTER SIX

CHANNEL BACKFLOW FROM THE UPPER PLENUM INTO THE CHANNEL

(CHANNEL-PLENUM INTERACTION)

6.1 Introduction

Forced convection flow through a channel is predominantly axial once the flow has hydrodynamically developed, and thus one-dimensional models adequately predict this behavior. In the mixed convection regime, however, the flow pattern can be multi-dimensional, rendering one-dimensional models inadequate. The flow pattern analyzed here is a channel-plenum interaction, and can occur when a vertical duct is connected at the top of its length to a region of essentially stagnant fluid, which includes the geometry for parallel channels connected at upper and lower plena.

It was visually observed in the SCCMT experiments that for some channels, although a predominantly axial flow pattern existed in the lower part of the channel, a multi-dimensional flow pattern was observed in the upper unheated part of the channel [S-1]. Fluid from the upper plenum penetrated into the channel, even though there was net upflow in the channel. This flow pattern is referred to here as the channel backflow phenomenon. A similar flow pattern was also observed in the GE SIMONE tests, which modeled an LMFBR core under natural circulation conditions [W-5]. Unfortunately, the purpose of these tests was not to analyze the channel backflow, and as a result no measurements related to the backflow were taken.

The channel backflow phenomenon was also observed using the WANACO (Water Natural Convection) loop in Switzerland [F-1]. For this experiment, a heated rod bundle was connected in parallel with two unheated channels, as shown in Figure 6.1. Fluid from the upper plenum backflowed into the low power region of the heated bundle.
Figure 6.1  The WANACO experiment.
Large temperature fluctuations were also recorded at the outlet of the low power region of the bundle. Flowrate measurements were not made in any part of the loop.

Although the channel backflow phenomenon had been qualitatively observed previously in the GE and Swiss experiments, none of these tests were designed to collect the data necessary to quantitatively analyze the phenomena. A series of tests were therefore performed to quantitatively predict the operating conditions that cause the onset of this phenomenon by making use of the MIT parallel channel test section.

6.2 Description of MIT Onset of Channel Backflow Experiments

6.2.1 Instrumentation

In the MIT parallel channel test section, flowmeters are not present in the individual channels. One of the important parameters to measure is the flowrate entering the bottom of a channel which has upper plenum fluid penetrating at the top. This is accomplished by blocking one heated channel (CH1) and the adiabatic channel (CH3) in the test section. Having done this, the loop flowrate into the lower plenum, which can be measured using the loop flowmeters, FM2 and FM3, must equal the flow entering the bottom of the unblocked channel, and thus the channel flowrate is obtained. This reduces the parallel channel test section to one channel. The other parameters measured are the temperature of the fluid exiting the channel, and the temperature of the fluid at the bottom of the upper plenum at a location 1.27 cm (0.5 in) from the position that the channel enters the upper plenum. These temperatures are measured by thermocouples TC2 and TC7, respectively. The thermocouple in the upper plenum (TC7) was used to measure the temperature of the fluid backflowing into the channel from the upper plenum.
6.2.2 Dye Injection System

In order to detect the backflow, blue ink diluted with water was injected continuously by gravity feed into the channel by a pitot tube, as shown in Figure 6.2. The pitot tube was bent at its tip to form a right angle. The tube penetrated the upper plenum, and by virtue of its bend, injected the dye into the top of the channel. If no backflow was present, the dye would be convected out of the channel by the upflow in the channel. If backflow was occurring, flow penetrating into the channel would sweep dye with it into the channel, and the flow pattern would be illuminated by the dye.

6.2.3 Experimental Procedure and Qualitative Results

It was qualitatively observed that the channel backflow was buoyancy-induced. The buoyancy occurred because the fluid exiting the top of the channel was at a higher temperature than the fluid in the upper plenum, and thus there was a tendency for the heavier upper plenum fluid to "sink" into the channel.

Tests were run by first throttling the channel inlet flowrate (by adjusting either V2 or V3) to the desired flowrate. The rod power was then increased from zero by small increments by adjusting the variac for CH2. After this adjustment, the channel outlet temperature, upper plenum temperature, and channel flowrate were measured and recorded after a steady-state condition was achieved. The flow field was then qualitatively observed by visual inspection of the injected dye. If no backflow was observed, the channel power was increased slightly. This effectively caused higher channel outlet temperatures and thus increased the tendency for channel backflow. This process was repeated until backflow was observed. Once backflow was achieved, the
Figure 6.2 The dye injection system.
Inlet flowrate to the test section was slightly increased. This caused the backflow to cease. The process of increasing the heater power until backflow is observed was then repeated. As a result of this procedure, the conditions that caused the onset of channel backflow at different flowrates were measured. These conditions are assumed to be the arithmetic average of the data set conditions measured just before and after backflow was visually observed.

Figure 6.3 qualitatively shows the backflow into the channel. Associated with the backflow was a large fluctuation of temperature at TC3 due to the large temperature difference between the hot channel fluid and the cold plenum fluid. The dye presented evidence that the flow field was not steady. At times the colder fluid swirled around the rod, flowing down until it changed direction and was swept upwards and out of the channel by the upward flow fluid. It is hypothesized that as the plenum fluid flows downward, it is heated by the hot upflow until the temperature difference is not great enough for the plenum fluid to continue to sink, and hence the change of direction.

6.3 Onset of Channel Backflow Model Development

It is desired to obtain a first-order model to predict the onset of the channel backflow phenomenon. To do this, the backflow of fluid from the upper plenum will be idealized as a steady stream of fluid flowing axially downward, as shown in Figure 6.4.

Figure 6.5 shows the force balance for a small axial length of this stream. The stream flows downward with velocity \( u_s \), and has a cross-sectional area \( A_s \) and perimeter of \( P_s \). Its temperature is \( t_s \), which if close to the top of the channel, is the temperature of the fluid in the upper plenum. The validity of this approximation
Figure 6.3  Backflow of fluid from the upper plenum into the channel.
Figure 6.4 Idealized model for the backflow into the channel.
Figure 6.5 Force balance for the idealized model for the backflow.
decreases as the distance from the top of the channel increases. Since only the onset of the backflow is considered, and the top of the channel first experiences the backflow, the assumption is valid. The upward flow of fluid is assumed to have velocity \( u_0 \) and temperature \( t_0 \).

The force balance equation for the section of the backflow stream shown in Figure 6.4 is written as

\[
0 = \Delta p A_s + \tau \frac{P}{s} A_z - \rho_s g A_s \Delta z
\]

(6.1)

which, in the limit of \( \Delta z \) approaching zero, becomes

\[
0 = -\frac{dp}{dz} + \tau \frac{P}{s} \frac{s}{A_s} - \rho_s g
\]

(6.2)

(The sign in the first term of the RHS of Eq. (6.2) is negative since the pressure is assumed to decrease in the direction of the \( z \)-axis, which is up).

If there are no radial pressure gradients in the fluid, the axial pressure gradient in the fluid traveling up the channel must equal that of the stream of backflow. If the backflow does not greatly disturb the upflow (which it does not in the limit of the onset of backflow), this pressure gradient is given by

\[
-\frac{dp}{dz} = \rho_0 g + f \frac{\rho u_0^2}{2}
\]

(6.3)

In addition, the friction factor and Reynolds number are

\[
f = \frac{c}{Re}
\]

(6.4)

and

\[
Re = \frac{\rho_0 u_0 D_e}{\mu}
\]

(6.5)

Combining Eqs. (6.2) - (6.5) produces
\[ 0 = \rho_0 g + \frac{c\mu u_0}{2D^2_e} + \tau_s \frac{P_s}{A_s} - \rho_s g \]  

(6.6)

The fluid-to-fluid shear is denoted by \(\tau_s\). It must be proportional to the relative velocities of the upflow and backflow streams, and is therefore written as

\[ \tau_s = \frac{\mu (u_0 + u_s)}{L_s} \]  

(6.7)

where \(L_s\) is the length constant that relates the relative velocities of the fluids to the velocity gradient at the stream boundary, i.e.,

\[ L_s = \frac{u_0 + u_s}{\frac{\partial u_s}{\partial r} |_{b}} \]  

(6.8)

In addition, the stream density can be written, using the Boussinesq approximation, as

\[ \rho_s = \rho_0 (1 - \beta(t_s - t_0)) \]  

(6.9)

where \(\rho_0\) and \(t_0\) are upflow stream parameters.

For the limiting case of the onset of backflow, the backflow stream velocity \(u_s\) is negligible and thus Eq. (6.7) becomes

\[ \tau_s = \frac{\mu u_0}{L_s} \]  

(6.10)

After substituting Eqs. (6.9) and (6.10) into Eq. (6.6) and solving for the upflow velocity \(u_0\), the result is

\[ u_0 = \frac{\rho_0 g \beta(t_0 - t_s)}{c\mu u_0 + \frac{\mu P_s}{2D^2_e} + \frac{\tau_s}{L_sA_s}} \]  

(6.11)

The numerator of this expression represents the fluid buoyancy, while the first and second terms of the denominator represent the pressure gradient and shear forces, respectively.
After multiplying both sides of this equation by \( \rho_0 D_e/\mu \) and rearranging, the equation can be rewritten

\[
\frac{\rho_0 u_0 D_e}{\mu} = \frac{\rho_0^2 g \beta (t - t_s) D_e^3}{c \mu^2} + \frac{1}{2} \frac{p_s D_e^2}{c L A_s} \quad (6.12)
\]

This equation can be simplified first by recognizing that the left hand side is the Reynolds number associated with the upflow, and then by defining two new non-dimensional parameters:

\[
Gr_{\Delta t} = \frac{\rho_0^2 g \beta (t - t_s) D_e^3}{\mu^2} \quad (6.13a)
\]

\[
B = \frac{1}{2} + \frac{p_s D_e^2}{c L A_s} \quad (6.13b)
\]

Additionally, solving Eq. (6.13b) for \( L_s \) and substituting into Eq. (6.10) yields

\[
\tau_s = (B - \frac{1}{2}) \frac{u_0 \mu c A_s}{p_s D_e^2} \quad (6.14)
\]

Equation (6.13a) defines a Grashof number based on the temperature difference between the channel upflow and colder plenum fluid.

Equation (6.13b) defines the new parameter \( B \) that includes unknown parameters of the backflow stream (\( p_s, L_s, \) and \( A_s \)). \( B \) relates the fluid-to-fluid shear to the upflow velocity \( u_0 \). From Eq. (6.14), it can be seen that if \( B \) is constant, the shear is directly proportional to \( u_0 \). A value of 0.5 indicates that the fluid-to-fluid shear is negligible, and higher values indicate increasing values of the shear for a given \( u_0 \). The minimum value of \( B \) that makes physical sense is thus 0.5. Substituting Eq. (6.13) into Eq. (6.12) yields
\[ \text{Re}_b \equiv \frac{\text{Gr}_{\Delta t}}{cB} \]  

(6.15)

Equation (6.15) gives us the conditions for the onset of backflow into the channel. When the channel Reynolds number \( \text{Re} \) is less than \( \text{Re}_b \), backflow occurs.

The dimensionless parameter \( B \) is still undetermined. Since onset of backflow data has been obtained, the data will be used to fit \( B \) to \( \text{Re}_b \). Thus, Eq. (6.15) can be rewritten as

\[ \text{Gr}_{\Delta t} = c \cdot B(\text{Re}_b) \cdot \text{Re}_b \]  

(6.16)

In the most general sense, \( B \) should also be a function of the channel geometry, but since data was only collected from one geometry, it is impossible to include a geometric dependence here, if there is one.

6.4 Experimental Results and Comparison with Model

Equation (6.16) is solved for \( B \), producing

\[ B(\text{Re}_b) = \frac{\text{Gr}_{\Delta t}}{c\text{Re}_b} \]  

(6.17)

For the annular channel of the experiment, \( c=95.2 \) for laminar forced convection. The parameters \( \text{Gr}_{\Delta t} \) and \( \text{Re}_b \) were obtained from the data sets collected so that \( B \) could be calculated for each data set. \( B \) was then fitted, using the least square error procedure, to a linear function of \( \text{Re}_b \). The result is

\[ B = -0.6365 + 7.580 \times 10^{-3} \text{Re} \quad 175 < \text{Re} < 590 \]  

(6.18)

which matches the data within \( \pm 20\% \) over the range indicated.

Figure 6.6 is an onset of channel backflow map, and compares the experimental results with those predicted by the model. The agreement is within \( \pm 20\% \), which is within the error associated with the measurement. Since the onset of the backflow was determined by taking the arithmetic average of data sets collected before and after backflow
Figure 6.6  Onset of channel backflow map.
was obtained, these data sets bracket the uncertainty associated with the assumed onset of the backflow, and hence the error bars are obtained.
CHAPTER SEVEN

BUOYANCY-INDUCED FLOW REDISTRIBUTION IN TWO PARALLEL CHANNELS

7.1 Introduction

In this chapter, an experiment is described which measured the flow split between two parallel channels of different power connected only at the upper and lower plena. The transient thermal hydraulic behavior of the system of two channels was not included in the experiment, and thus CHANSOL, which includes only a steady-state formulation of the conservation equation, was used to predict the results. The experiment was performed using the MIT parallel channel experiment, with the one adiabatic channel (CH3) blocked.

7.2 Instrumentation

In order to measure the buoyancy-induced flow redistribution between the two channels in the experiment, it was necessary to determine the individual channel flowrates. Because flowmeters were not present in the individual channels, the individual channel flowrates could not be directly obtained. These channel flowrates were instead calculated from the available temperature and total flowrate data as discussed in Section 7.3.

The temperature drop for each channel could be directly measured by the thermocouples located at the inlet and outlet of the heated channels. These thermocouples are TC1 and TC4 for CH1, and TC2 and TC5 for CH2. In addition, the sum of the two channel flowrates is equal to the inlet flowrate at the lower plenum, which was measured by loop flowmeters FM2 and FM3 for the range of interest.

The electrical power to the two heated rods was monitored by VM1 and AM1 for CH1 and VM2 and AM2 for CH2. The rod heater power could then be estimated from
\[ q = E \cdot I \]  \hspace{1cm} (7.1)

where \(E\) and \(I\) are the applied voltage and current. The power calculated from Eq. (7.1) was used only to estimate the rod power, however, since the resolution of the scales of the voltmeters and ammnometers was not sufficient for accurate measurement for the range of interest, as shown in Section 5.3.4. The procedure used for obtaining the rod heater power, as well as the flow redistribution data, is explained in the following section.

7.3 Procedure to Calculate the Flow Redistribution and Channel Power

The methodology used to calculate the buoyancy-induced flow redistribution is based on the ability to predict the flow split between the two channels under forced convection conditions. Suppose the inlet flowrate is sufficiently large that buoyancy effects can be neglected. The channel energy balance can be written

\[ q_n = (\dot{m}_{nf}/\dot{m}_{of}) \dot{m}_{of} c_p \Delta t_{nf} \]  \hspace{1cm} (7.2)

where \((\dot{m}_{nf}/\dot{m}_{of})\) is the forced convection flow fraction for channel \(n\), \(\Delta t_{nf}\) is the associated temperature rise, and \(\dot{m}_{of}\) is the forced convection inlet flowrate to the test section. For two identical channels \((\dot{m}_{nf}/\dot{m}_{of})\) must equal 0.5.

Assuming now that the channel powers remain constant, the flow redistribution among the channels can be calculated for cases in which \(\dot{m}_{o}\) is reduced such that the channels are in the mixed convection. An energy balance for each channel produces

\[ q_n = \dot{m}_n c_p \Delta t_n \]  \hspace{1cm} (7.3)

Since the channel powers remained constant, Eqs. (7.2) and (7.3) can be combined, eliminating \(q_n\). After rearranging and solving for the flow fraction \((\dot{m}_n/\dot{m}_o)\), the result is
\[
\left( \frac{\dot{m}_n}{\dot{m}_0} \right) = \left( \frac{\dot{m}_{nf}}{\dot{m}_{of}} \right) \left( \frac{\dot{m}_{of}}{\dot{m}_0} \right) \left( \frac{\Delta t_{nf}}{\Delta t_n} \right)
\]

(7.4)

This analysis shows that the flow fraction for each channel can be calculated without a direct measurement of the individual channel flowrate and power.

7.4 Error Analysis

The experimental error associated with the flow fraction measurement is due to the uncertainty in each of the three factors on the RHS of Eq. (7.4). Once the uncertainty of each factor is obtained, the total uncertainty must be calculated using error propagation techniques. First, Eq. (7.4) is simplified by defining

\[
A = \left( \frac{\dot{m}_n}{\dot{m}_0} \right)
\]

(7.5)

\[
B = \left( \frac{\dot{m}_{nf}}{\dot{m}_{of}} \right)
\]

(7.6)

\[
C = \left( \frac{\dot{m}_{of}}{\dot{m}_0} \right)
\]

(7.7)

and

\[
D = \left( \frac{\Delta t_{nf}}{\Delta t_n} \right)
\]

(7.8)

The quantity A can be thought of as a function of B, C, and D. The generalized form of the standard deviation of a function \( y(x_1, x_2, x_3, \ldots, x_N) \) in terms of the standard deviation of the \( x_i \)'s is (see Ref. K-9, p.131)

\[
\sigma_y^2 = \sum_{i=1}^{N} \left( \frac{\partial y}{\partial x_i} \right)^2 \sigma_{x_i}^2
\]

(7.9)
which for our special case of

$$A = B \cdot C \cdot D$$  \hspace{1cm} (7.10)

reduces to

$$\sigma_A^2 = (CD)^2 \sigma_B^2 + (BD)^2 \sigma_C^2 + (BC)^2 \sigma_D^2$$  \hspace{1cm} (7.11)

Dividing this equation by $A^2$ produces

$$\left(\frac{\sigma_A}{A}\right)^2 = \left(\frac{\sigma_B}{B}\right)^2 + \left(\frac{\sigma_C}{C}\right)^2 + \left(\frac{\sigma_D}{D}\right)^2$$  \hspace{1cm} (7.12)

which relates the fractional errors of $A$ with those of $B$, $C$, and $D$.

The uncertainty in the value of the forced convection flow fraction is represented by $\sigma_B$ (see Eq. (7.6)). If the two channels were identical, $B$ would exactly equal 0.5 and $\sigma_B$ would equal zero. $B$ could vary from 0.5 due to either of two factors. The first is due to variations in the rod and tube diameters for the two channels, such that their flow areas and hydraulic diameters are not identical. The second factor contributing to the variation of $B$ is due to the rod being slightly eccentric in the channel (the centers of the rod and tube are not identical). As the eccentricity increases, the volumetric flow for a given pressure drop is expected to increase (see Ref. W-3, p.126). The relative eccentricity can be expressed as

$$e \equiv \frac{c}{r_o - r_1}$$  \hspace{1cm} (7.13)

where $c$ is the distance of the center of the rod from the center of the tube, as shown in Figure 7.1. The parameter $c$ can vary from zero, corresponding to the case of concentric annuli, to a maximum value of $(r_o - r_1)$ for the case of the rod touching the tube wall. The
Relative eccentricity,

\[ \alpha = \frac{c}{r_0 - r_1} \]

Figure 7.1  Effect of eccentricity on channel flowrate for annular channels. 
(Taken from Ref. W-3)
relative eccentricity therefore varies from 0 to 1. Figure 7.1 shows the effect of eccentricity on channel flowrate for a given pressure drop for different values of $\alpha = r_1/r_0$. For the heated channels in the parallel channel experiment, $\alpha = 0.667$. Assuming the relative eccentricity is under 0.20, the increase channel flow from eccentric effects is less than 1.04 times channel flow with no eccentricity. The fractional error of $B_1(q_B/B)$, is therefore assumed to be 0.04.

The uncertainty in the value of $(\dot{m}_{of}/\dot{m}_o)$ is represented by $\sigma_c$. The accuracy in both $\dot{m}_{of}$ and $\dot{m}_o$ is limited by accuracy of the flowmeter used to measure the inlet flowrate. Since only the ratio of the two quantities is necessary, errors associated with calibration of the scale will cancel. The remaining error can be from two sources. The first is due to non-linearities of the flowmeter. This effect is considered negligible since a characteristic of rotameter flowmeters is good linear behavior. The second error source is imposed by the resolution of the scale. The scales of FM2 and FM3 have full scale values of 100 and are in divisions of 5 and 1, respectively. The forced convection inlet volumetric flowrate was set to be $1.24 \times 10^{-4}$ m$^3$/s, which corresponds to a full scale reading on FM2. The fractional error on $\dot{m}_{of}$ is thus set to $5/100 = 0.05$. For mixed convection runs $\dot{m}_o$ was measured in the range of interest using FM3. At the smallest inlet flowrates used for the tests, the reading of FM3 was 35% of its full scale value. The fractional error imposed by the scale resolution at this reading is $1/35 = 0.029$. Defining
\[ E = \dot{m}_{of} \quad (7.14) \]

and

\[ F = \dot{m}_{o} \quad (7.15) \]

the associated fractional errors in \( E \) and \( F \) are therefore \( \sigma_E/E = 0.05 \) and \( \sigma_F/F = 0.029 \), respectively.

The errors associated with \( E \) and \( F \) can now be combined to give the error associated with \( C \). First, combining Eqs. (7.12), (7.14), and (7.15) produces

\[ C = \frac{E}{F} \quad (7.16) \]

Applying Eq. (7.14) to this equation and rearranging produces

\[ \frac{\sigma_C^2}{C^2} = \frac{\sigma_E^2}{E^2} + \frac{\sigma_F^2}{F^2} \quad (7.17) \]

Substituting for \( (\sigma_E/E) \) and \( (\sigma_F/F) \) produces \( (\sigma_C/C) = 0.0578 \).

The last fractional error that needs to be calculated before Eq. (7.12) can be solved for the total percent error is \( (\sigma_D/D) \). This quantity represents the errors associated with thermocouple temperature measurements. The accuracy of these measurements is limited by the accuracy of the digital thermometer and the thermocouples. The temperature error is composed of two parts. The first is bias error which is caused by errors in the temperature compensation circuit within the digital thermometer. Bias error is eliminated in the experimental procedure because only temperature differences between thermocouples are used in the calculations. The second part of the error is imposed by the resolution of the temperature measurement system, which is 0.1°F. To again simplify the nomenclature, define
\[ G = \frac{\Delta t}{n_f} \quad (7.18) \]

and

\[ H = \frac{\Delta t}{n} \quad (7.19) \]

The larger the values of \( G \) and \( H \), the smaller the fractional error.

The forced convection temperature drop is typically 5°F so that \( (\sigma_G/G) = 0.02 \). Since the temperature drop for the mixed convection runs is equal to or higher than that for forced convection, it will conservatively be assumed that \( H = G \) so that \( (\sigma_H/H) = 0.02 \), also.

The errors associated with \( G \) and \( H \) can now be combined to give the error associated with \( C \). First, combining Eqs. (7.8), (7.18), and (7.19) produces

\[ D = \frac{G}{H} \quad (7.20) \]

Applying Eq. (7.9) to this equation and rearranging produces

\[ \frac{\sigma_D^2}{D^2} = \frac{\sigma_G^2}{G^2} + \frac{\sigma_H^2}{H^2} \quad (7.21) \]

Substituting for \( (\sigma_G/G)=0.02 \) and \( (\sigma_H/H)=0.02 \) results in \( (\sigma_D/D) = 0.0283 \).

The error associated with the flow fraction measurement can now be calculated from Eq. (7.12). After substituting the values of the percent error into Eq. (7.12), one obtains \( (\sigma_A/A) = 0.0653 \). Thus, the experimental error is estimated to be \( \pm 7\% \).

7.5 Results and Discussions

Four different test runs were executed, corresponding to power skews of \( q_1/q_2 = 4.259, 2.536, 1.719, 1.186 \), with \( q_1=730 \text{W} \). The inlet volumetric flowrate was varied from a maximum value of \( 1.24\times10^{-4} \text{m}^3/\text{s} \) down to a minimum value of \( 5.96\times10^{-6} \text{m}^3/\text{s} \). The forced convection
values of the inlet flowrate ($m_{0f}$) and the channel temperature drop ($\Delta t_{nf}$) were obtained at the maximum value of the inlet flowrate. The CHANSOL simulations later executed confirmed that indeed the system was in forced convection with each channel receiving one-half the inlet flow.

In Figures 7.2 through 7.5, the flow fraction for CH1 is plotted versus the inlet flowrate. The flow fraction was calculated using Eq. (7.4) with $n = 1$. Qualitatively, it can be seen that as the inlet flow decreases, the flow fraction increases from its forced convection value of 0.5. This is due to the buoyancy-induced flow redistribution effect which redistributes flow from the cold channel (CH2) to the hot channel (CH1). For the high power skew case (Figure 7.1), $q_1/q_2 = 4.259$, the flow fraction increases to almost 0.9 at the minimum inlet flowrate.

Also shown in the figures is the CHANSOL prediction for the data. For inlet flowrates greater than 30 $q/m$, CHANSOL adequately predicts the data. For flows less than this, CHANSOL underpredicts the flow in the hot channel. The CHANSOL prediction was verified by the MICON code for $q_1/q_2 = 4.259$, as shown in Figure 7.2. MICON was only run for this case because of the CPU expense involved.

An explanation for the underprediction follows. As the inlet flowrate is decreased, the flowrate in the colder channel decreases. The outlet temperature correspondingly increases. The combination of low channel flowrate and high outlet temperature causes backflow to occur in the low power channel, as discussed in Chapter 6 and visualized by dye injection in this experiment. Once backflow starts occurring in the colder channel, the buoyancy-induced flow redistribution in the hot channel increases more than the
Flow redistribution for two parallel channels with power skew equal to 4.259.
Figure 7.3  Flow redistribution for two parallel channels with power skew equal to 2.536.
Figure 7.4  Flow redistribution for two parallel channels with power skew equal to 1.719.
Figure 7.5  Flow redistribution for two parallel channels with power skew equal to 1.186.
one-dimensional theory predicts. This is due to two factors. First, the downflow into the colder channel tends to restrict the upflow in this channel. The result is the upflow has less flow area, resulting in less flow for a given pressure drop. Secondly, the downflow in the colder channel tends to cool the fluid in this channel. This increases the gravity pressure drop in the channel, and also results in less flow for a given pressure drop.

Okada, et al., [0-2] have obtained mixed convection pressure drop data which suggests that a sharp increase in pressure drop occurs when a recirculation pattern in the upper part of a rod bundle develops. In their experiment, the cause of the recirculation pattern was a high radial power skew along the heated length. For the experiment described here, the cause of the recirculation pattern was the backflow from the upper plenum. In either case, a predominantly axial flow field was transformed into a multi-dimensional one with the advent of the recirculation pattern. Okada's pressure drop data therefore supports our hypothesis regarding increased channel pressure drop with the onset of recirculation.

The data also shows that the flow fraction of the hot channel does not increase monotonically with decreasing inlet flow. This is true for all power skews. Experimentally, backflow was observed in the hot channel in this range. It is believed that the channel backflow decreased the flow fraction in the hot channel for the same reasons outlined for the cold channel.

Figure 7.6 shows the onset of backflow map for the two channels for \( q_1/q_2 = 4.259 \). The upper half of the graph is the dimensionless
Figure 7.6 Onset of backflow map for a two-channel system of different powers.
pressure drop curve for the two channels. The curve for CH1 is below that for CH2 because it is hotter for the same Reynolds number, and thus the gravity pressure drop is lower. The lower half of the graph shows the onset of backflow curve obtained from Chapter 6 along with the operating curves for the two channels. As the channel Reynolds number decreases, the $Gr_{At}$ parameter increases since the temperature difference between the fluid exiting the channel and the fluid in the upper plenum increases. The intersection of the operating curves with the onset of backflow curve determines when backflow will begin for that channel. By extrapolating these results up to the upper part of the graph, one can determine the pressure drop for which the backflow occurs for either channel. The graph indicates that when the inlet flowrate to the system is decreased, CH2 will experience backflow first since the backflow is predicted at a higher $Ap^+$ than that for CH1. These results are consistent with those observed experimentally, as shown in Figure 7.2.

If experimentally the inlet flowrate for the buoyancy-induced flow redistribution was decreased to zero flow, the flow in one of the channels must reverse to downflow in order to satisfy the system mass balance. It is trivial to show that for identical channels with different power, the higher power channel has the strongest preference for upflow, as defined in Section 2.35. In the following chapter a visual study of the manner in which a heated channel reverses from upflow to downflow will be presented.

In summary, the deviation of the data from the one-dimensional analysis is due to the multi-dimensional backflow effect in the channel. Backflow first occurred in the low power channel and caused a corresponding decrease in the low power channel flowrate from that
predicted by the one-dimensional analysis. When backflow was established in the high power channel, a dip in the high power channel flow fraction data was observed. The decrease in channel flowrate with the onset of backflow is consistent with the increase in pressure drop observed experimentally for a rod bundle with a recirculating flow pattern at its upper length.
CHAPTER EIGHT
A VISUAL STUDY OF THE REVERSAL OF A HEATED CHANNEL
FROM UPFLOW TO DOWNFLOW

8.1 Introduction

In Chapter 6, it was reported that for low flow through a channel, it is possible that the flow will not travel axially in the channel. If the outlet temperature of the fluid in channel is sufficiently higher than that of the upper plenum, and the channel flow is sufficiently small, then a multi-dimensional flow pattern results in which cold fluid from the upper plenum penetrates into the channel. In Chapter 7, a prediction of the buoyancy-induced flow redistribution between two identical parallel channels of different power was made using the one-dimensional code CHANSOL. The code prediction did not match the experimental results when multi-dimensional flow was visually observed. This shows that one-dimensional theory can not adequately predict multi-dimensional flow.

For a heated channel with zero inlet flowrate, one-dimensional theory predicts that a steady-state is not possible, and eventually the channel will boil. With the inclusion of a multi-dimensional flow field in the channel, it is possible for counter-current flows within the channel to cool the channel sufficiently so that a steady-state may be reached prior to boiling. The upper plenum would essentially act as a heat sink for the channel. Hot rising streams of fluid would exit the channel and be replaced with cooler plenum fluid. The highest temperature that the fluid might reach would be determined by the channel heat generation rates, the channel geometry, and the temperature of the upper plenum.

Stewart, Pieczynski, and Srinivas [S-7] have presented experimental data for the velocity and temperature fields for an isolated
reactor core scale model that is heated. The scale model is based on a Westinghouse PWR. The individual assemblies in a PWR are interconnected along the length of the core (there are no "cans") so that the entire core can be thought of as an isolated channel. Figure 8.1 shows the experimental core velocities measured for an isolated reactor vessel. The cold upper plenum fluid penetrates into the low power peripheral assemblies. The heated downflow then rises up through the core center.

In this chapter, the transition from heated upflow to heated downflow will be qualitatively described. As with the experiment of Stewart, et al., steady cooling of the channel was observed for zero inlet flowrate.

8.2 Experimental Procedures

In order to observe the transition from heated upflow to downflow, the MIT parallel channel test section was used. As in the buoyancy-induced redistribution experiment described in Chapter 7, the adiabatic channel was blocked so that the test section was essentially two parallel heated channels.

The inlet volumetric flowrate to the test section was set to $2.555 \times 10^{-6} \text{ m}^3/\text{s}$ with the heater powers adjusted to $q_1 = 717.7 \text{ W}$ and $q_2 = 168.5 \text{ W}$. Dye was injected into CH2 in order to visualize the flow pattern. At this operating condition backflow was observed in CH2. The inlet flowrate to the test section was then successively decreased and the flow pattern was observed. The flow pattern was recorded by photographing the dye pattern with a Polaroid™ instant camera. The photographs were then used as an aid to draw by hand the flow patterns observed.
Figure 8.1 Core velocities measured in an isolated PWR scale model core showing recirculation from the upper plenum.
(Taken from Ref. S-7)
8.3 Results and Discussion

Figure 8.2 shows the flow patterns observed for the transition from heat upflow to downflow. In Figure 8.2a, a weak backflow is observed at the top of the channel. The dye collects at the top of the channel because the fluid velocities are relatively small compared to the dye velocity as it exits the pitot tube. As the power to the inlet flowrate to the system is further decreased, the backflow from the upper plenum penetrates deeper (lower) into the channel, as shown in Figure 8.2b and c. The backflow velocity increases and forms thin streams of fluid flowing down the channel. The stream irregularly wraps itself around the channel as shown in Figure 8.2d, producing a swirling pattern. After a distance into the channel, it reverses direction and travels up with upflow. By observing the flow pattern at the top of the channel, it was found that at a certain azimuthal position the flow was down, while elsewhere in the channel the flow was up. The azimuthal location of the downflow, however, varied with time. A similar result was observed in the WANACO experiment [F-1].

As the inlet flowrate was further decreased, the backflow penetrated deeper into the channel until injected dye at the bottom inlet of the channel showed that the net flow through the channel was essentially zero. At this point the channel was cooled totally by the backflow from the upper plenum.

As the inlet flowrate was further decreased, a net downflow was produced in the channel. Even though there was net downflow, flow did proceed up the channel in certain regions. The flow from the upper plenum was predominantly down, though, as shown in Figure 8.2e. The flow remained unsteady, and took on a serpentine character as it
Figure 8.2
Flow pattern of backflow from the upper plenum in the upper part of CH2 as it reversed from heated upflow to heated downflow.
meandered down the channel.

Figure 8.3 shows the one-dimensional dimensionless pressure drop curve for CH2 modified to account for multi-dimensional effects. As the inlet flowrate to the system of parallel channels was lowered, we proceed left on the dotted line. Point A corresponds to the onset of channel backflow, and initiates the deviation of the channel performance from one-dimensional theory. Point A can be predicted using the analysis of Chapter 6.

The dotted line, which represents the pressure drop curve when multi-dimensional effects are included, is above the one-dimensional theory curves. This is because the multi-dimensional flow tends to cool the channel, producing higher densities and thus a higher gravity pressure drop.

Point B corresponds to the minimum downflow at which one-dimensional theory can predict the channel behavior. The location of this point is not known at this time and was thus arbitrarily chosen to be the point at which one-dimensional theory predicts stable heated downflow.

Temperature measurements were made at the inlet and outlet of CH2 at the time a net downflow was recorded to see whether the first occurrence of downflow corresponded to the point where stable downflow is predicted using one-dimensional theory. The temperature measurements showed that net downflow was occurring at a much smaller downflow than allowed by the one-dimensional theory, i.e., point C. This means that there was an additional heat sink in the channel due to the backflow from the upper plenum.
Figure 8.3 Dimensionless pressure drop curve for a heated channel including multi-dimensional flow field effects.
CHAPTER NINE

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

9.1 Introduction

In this work, mixed convection effects for parallel, vertical channels have been studied. The work can be divided into the following categories:

(a) Steady, one-dimensional mixed convection flow in vertical parallel channels
(b) Steady mixed convection flow in a duct
(c) Transient, one-dimensional mixed convection flow in vertical parallel channels
(d) Multi-dimensional channel-plenum interactions
(e) Experimentally observed buoyancy-induced flow redistribution
(f) Qualitative observation of a heated channel flow reversal from upflow to downflow

The conclusions and contributions from this work are described in the next sections.

9.2 Conclusions

9.2.1 Steady, One-Dimensional Parallel Channel Flow

The coupled conservation equations that describe the mixed convection flow for parallel channels with prescribed heat flux along their length were formulated and non-dimensionalized. This formulation was then utilized to obtain expressions that predict the onset of mixed convection, flow reversal from upflow to downflow, and stability of heated and cooled flows. Accomplishments for this part of the work include
(a) The modification of the momentum-energy to account for non-uniform axial heat flux shapes by formally defining a heat flux shape factor $S$.

(b) The definition of a preference for upflow number that includes flow orificing and non-uniform axial heat flux effects.

(c) The presentation of a numerical solution procedure that solves the coupled conservation equations for an arbitrary number of channels, and the execution of the solution procedure in the CHANSOL code.

CHANSOL was then applied to the GE PRISM reactor concept in order to predict the buoyancy-induced flow redistribution during passive decay heat removal (natural circulation) of the primay loop. The results are

(d) Buoyancy-induced flow redistribution reduces the temperature rise of the hot assembly from 1.6 to 1.1 times that of the core-averaged temperature rise.

(e) The buoyancy-induced flow redistribution is relatively insensitive to the loop flowrate and core power for the conditions expected during the natural circulation.

(f) The buoyancy-induced flow redistribution is relatively insensitive to viscosity variations between channels of different axial temperature profile.

9.2.2 Mixed Convection Flow In a Duct

The steady, fully-developed conservation equations for mixed convection flow in a duct were formulated in this work. The dimensionless velocity and temperature profiles were shown to be a
function of the $\text{Gr}_q/\text{Re}$ parameter, heat flux shape, and duct geometry. The mixed convection friction factor ratio $f/f_0$ and Nusselt number $\text{Nu}$ were also dependent on these parameters. Accomplishments include

(a) The formulation of an expression for the mixed convection friction factor ratio $f/f_0$ which is based on fluid shear stress at the wall.

(b) The formulation of an expression for the dimensionless temperature using the spatially-averaged dimensional temperature as the reference temperature.

(c) The definition of a modified friction factor ratio $(f/f_0)(1+F)$ to be used in analyses where only bulk-average properties are calculated, such as lumped parameter codes.

The above formulations were then executed for geometries that exhibit azimuthal symmetry. These included the circular tube, annulus with inner and outer radii heated, and the equivalent annulus. Based on these results, the following conclusions are made:

(d) For all four geometries, velocity profiles are distorted due to buoyancy-induced flow redistribution. For aiding flow, velocities increase near the diabatic wall and decrease near the adiabatic boundary. For opposing flow, velocities decrease near the diabatic wall and increase near the adiabatic wall. The distortion increases as $\text{Gr}_q/\text{Re}$ increases.
(e) The buoyancy-induced distortion of the velocity profile causes an increase in shear at walls near which the velocity increases, and a decrease at walls near which the velocity decreases. For aiding flow, this causes \((f/f_o)\) to increase with \(Gr_q/Re\), while for opposing flow, \((f/f_o)\) decreases with \((f/f_o)\). The exception is the inner radius heated annulus with small inner/outer radius for which \((f/f_o)\) decreases with \(Gr_q/Re\) for aiding flow, and \((f/f_o)\) increases with \(Gr_q/Re\) for opposing flow. The \((f/f_o)\) results were correlated as a function of \(Gr_q/Re\) for each geometry.

(f) The modified friction factor ratio \((f/f_o)(1+F)\) results are in general different from the conventional friction factor ratio results. This is because the bulk temperature is in general different from the spatially-averaged temperature. The modified friction factor ratio may be greater or less than the conventional friction factor ratio, depending on the geometry, \(Gr_q/Re\) parameter, and orientation of the flow (aiding or opposed). The \((f/f_o)(1+F)\) results were correlated as a function of \(Gr_q/Re\) for each geometry.

(g) Nusselt numbers increase with \(Gr_q/Re\) for aiding flow, and decrease with \(Gr_q/Re\) for opposing flow for \(Gr_q/Re>100\) all geometries. The increase in Nusselt number for aiding flow is caused by the increased convection near the diabatic wall, while for opposing flow the decreased convection causes the decrease in Nusselt number.
The equivalent annulus friction factor ratio results were compared to the infinite rod array numerical results. The comparison shows that

(h) For aiding flow, the equivalent annulus matches the infinite rod array results for \( P/D > 1.50 \) for a triangular array, and \( P/D > 3.00 \) for a square array.

(i) For opposing flow, the equivalent annulus matches the infinite rod array for \( P/D > 1.50 \) for triangular arrays. At \( P/D = 3.00 \), there was significant discrepancy for square arrays, and thus it is not recommended to use the equivalent annulus model for opposing flow square arrays.

The analytical results were compared with experimental data for the friction factor ratio and Nusselt number. The results are

(j) The modified friction factor ratio predicts the circular tube data of Kemeny and Somers [K-1] better than the conventional friction factor ratio for \( Gr_d/Re < 2500 \). This is because in Ref. K-1 they used the bulk temperature to calculate the gravity pressure drop in their test section. The data lies below the analytical prediction due to buoyancy-induced turbulence, and developing flow effects.

(k) The equivalent annulus model overpredicts the friction factor ratio for the MIT 4×4 bare rod bundle [O-2], again, because of buoyancy-induced turbulence, and developing flow effects. The modified friction factor ratio predicts the data better than the conventional one because the bulk temperature was used to calculate the gravity pressure drop in the test section.
(1) Mixed convection Nusselt number data from the MIT 19-pin wire-wrapped rod bundle [E-1] was predicted by the equivalent annulus model to within ±20%. The experimental data showed a steeper increase with the $Gr_q/Re$ parameter. This discrepancy could be caused by the presence of the wire. Also, in Ref. E-1, they report turbulence for some of the tests.

Finally, an analysis was made of the mixed convection friction factor ratio data reported for finite rod bundles. Data from the MIT 19-pin bundle [O-2] and the Toshiba 91-pin bundle [M-4] were simulated using the ENERGY-IV code with a water properties package added in this work. The results show that

(m) The bundle friction factor increase with the $Gr_q/Re$ parameter for aiding flow is due to global flow redistribution effects within the bundle and is adequately predicted using the ENERGY-IV code with forced convection values for the subchannel friction factors.

(n) If the equivalent annulus model were used to estimate the subchannel friction factor, the 19-pin data would be severely overpredicted. Thus, the increase in bundle friction factor is not attributed to local (subchannel) friction factor increases.

9.2.3 Transient One-Dimensional Mixed Convection Flow

The one-dimensional conservation equations for transient mixed convection in vertical parallel channels were formulated and discretized in Refs. [K-2] and [K-3]. These equations were then incorporated into a code called MICON. Accomplishments in this work
include

(a) The formulation of the momentum-integral equation to include
the modified friction factor ratio.

MICON was then used to predict the GE Shutdown Core Coolability Model
Test (SCCMT) results. The results show that

(b) The agreement between the transient experimental data in the
MICON simulation was adequate for one-dimensional flow
through the channels. For the heated channel with weak
upflow, a large discrepancy existed between the data nd
prediction. This discrepancy was caused by multi-
dimensional flow in this channel. Colder fluid from the
upper plenum penetrated back into the channel, mixing with
the fluid in the channel and cooling it. The predicted
channel temperature was therefore much higher than
experimentally observed.

9.2.4 Multi-Dimensional Channel-Plenum Interaction

An experimental investigation was made of the channel backflow
from the upper plenum. The experiment was performed using the MIT
parallel channel test section, designed and built in this work. The
following results and conclusion are obtained:

(a) The tendency for backflow from the upper plenum into the
channel increases as the temperature difference between the
fluid in the channel and the fluid in the upper plenum
increases.

(b) The tendency for backflow to occur decreases as the channel
flowrate increases.

(c) A simple model was developed to predict the onset of the
backflow. The model assumes the forces acting on the downward stream of fluid from the upper plenum are due to fluid shear, fluid buoyancy, and the axial pressure gradient.

(d) The model predicts the onset of the channel backflow should depend on the $Gr_{At}$ and $Re$ parameters, where $Gr_{At}$ is based on the temperature difference between the fluid in the channel and the fluid in the upper plenum, and $Re$ is based on the upflow in the channel. Backflow occurs when $Re$ is less than $Re_b$, where $Re_b$ is given by

$$Re_b = \frac{Gr_{At}}{cB}$$

where $c$ is from $f=c/Re$ and $B$ is a function of $Re$.

(e) Onset of backflow data was collected using the parallel channel test section. $B$ was fitted to the data as

$$B = -0.6365 + 7.5 \times 10^{-3}Re \quad 175 \leq \Gamma \leq 590$$

An onset of channel backflow map was created based on the empirical equations.

9.2.5 Experimentally Observed Buoyancy-Induced Flow Redistribution

The MIT parallel channel test section was used to measure the buoyancy-induced flow redistribution for a two-channel system of different powers. The conclusions are:

(a) CHANSOL predicted the data when channel flows were visually observed to be one-dimensional

(b) A distinct decrease of flow occurs when backflow occurs in a channel. The flow decrease is caused by the decrease in flow area available for the upflow when backflow is
present. Also, the backflow will increase the gravity pressure drop in the channel, thereby decreasing the flow.

9.2.6 Qualitative Observation of a Heated Channel Flow Reversal From Upflow to Downflow

The MIT parallel channel test section was used to observe the way in which a heated channel reverses from upflow to downflow. The following conclusions are made:

(a) The initiation of the transition from upflow to downflow is upper plenum backflow into the channel.

(b) As the upward flow in the channel decreases, the backflow stream penetrates deeper into the channel.

(c) For a sufficiently cold upper plenum, it is possible that the heated channel will reach a steady-state even for no net flow through the channel. At this point the channel is totally cooled by backflow from the upper plenum.

(d) Stable heated downflow is possible in the region not allowed by one-dimensional theory. The reason for this is that the multi-dimensional flow field from the upper plenum acts as an additional heat sink to the channel.

9.3 Recommendations for Future Work

In this section, questions that remain unanswered are discussed. Proposals are also made to solve these questions. In addition, research questions in related fields which will help in the general understanding of the mixed convection flow regime will be presented.

- Mixed Convection Pressure Drop Measurement for Developed and Developing Flow

There is a scarcity of data in the literature for mixed
convection pressure drop in vertical duct geometries. Pressure drop data should be taken for developed and developing flow to test the validity of the fully-developed flow assumption. The annular geometry would lend itself to easy test section construction and numerical modeling.

- Mixed Convection Pressure Drop Numerical Prediction for Developing Flow

The pressure drop data collected should be compared with numerical predictions for developing flow using a finite difference scheme. A two-dimensional, azimuthally symmetric analysis would be sufficient to predict the developing pressure drop for the azimuthally symmetric geometries studied in Chapter 3.

- Buoyancy-Induced Turbulence Studies

Significant amounts of experimental work need to be done to more fully understand the buoyancy-induced turbulence phenomenon. This work is necessary because it will determine the range of validity of the laminar analyses, and will provide insight on a type of turbulence which is of a significantly different character than that for forced convection. An experiment should be built that maximizes the measurability of the fluid thermal hydraulic parameters, such as the temperature and velocity fields, so that gross and local variations can be measured and visualized. A start was made in Ref. R-3 to measure the buoyancy-induced turbulence for a rod array geometry. We propose here an even simpler geometry, such as parallel plates, to characterize the basic nature of the turbulence, which is still not understood.
- Multi-Dimensional Flow Through a Channel

For low Reynolds number flow through a geometry (Re<500), there is the possibility of multi-dimensional flows developing due to local heat sources and sinks. A start was made in this work to characterize the flow pattern that develops when the heat sink is cold fluid in the upper plenum, a situation which led to the channel backflow from the upper plenum. In Ref. T-4, a recirculation pattern is researched for the MIT 4×4 bare rod bundle. The heat sink in this case is a cold duct wall. What needs to be shown is that these recirculation patterns add, and not subtract, from the coolatability of the channel. Since during a transient the net flow in a channel may be essentially zero, an experimental study of the thermal hydraulic behavior of a heated rod bundle for zero inlet flow is of interest. Boundary conditions for the bundle would include heat loss along the duct wall (to model inter-assembly heat transfer) and the upper plenum fluid temperature. The experimental analysis could then be complemented by a numerical simulation by a code such as TEMPEST [T-6]. A pressure drop versus flowrate curve should also be constructed that includes the effects of recirculating flows.

- Numerical Study of Energy Equation Formulations for Transient Flow in a Diabatic Channel

Included in the MICON code are several formulations for the discretized energy equation. Among the formulations is one that is based on the method-of-characteristics. A comparison should be made of the different methods for selected transients for a single diabatic channel. Performance of the different formulations should be judged based on numerical stability, numerical diffusiveness, and computational time.
REFERENCES


M-1 P.M. Magee, Personal Communication to V. Iannello, Fall 1985.


Appendix A

CHANSOL Input Description and Listing

CHANSOL is a small code designed to solve for the steady-state flow distribution among parallel channels connected at upper and lower plena. Since it is so small, it was implemented in an IBM Personal Computer. CHIN.DAT is designated as the input file and CHOUT.DAT as the output file. The input description and program listing follow.
Input Description

The input file may be in free-format. Parameters in square brackets are the Chapter 2 equivalents of the CHANSOL variables.

Card 1:  n, chlen, grav
          n = number of channel types [N]
          chlen = channel length (m) [L]
          grav = g*cosθ (m/s²) [g]

Card 2:  t₀, rho₀, cp, beta, rmu
          t₀ = reference temperature (°C) [t₀]
          rho₀ = reference density (kg/m³) [ρ₀]
          beta = thermal expansivity (°C⁻¹) [β]
          rmu = reference viscosity (Pa-s) [μ₀]
          cp = specific heat (J/kg/°C) [cₚ]

Cards 3.1:  num(i), diameq(i), aflow(i), s(i), i=1,n
          num(i) = number of channels of type i [yₙ]
          diameq(i) = equivalent diameter of channels of type i [Dₚ]
          aflow(i) = flow area of channels of type i [Aₚ]
          s(i) = heat flux shape for channels of type i [Sₚ]

Cards 4.1:  cfric(i), rndfric(i), rklass(i), tlow(i), tup(i), phi(i), i=1,n
          cfric(i) = c from f=c/Re b for channels of type i [cₙ]
          rndfric(i) = b from f=c/Re b for channels of type i [bₙ]
          rklass(i) = form losses for channels of type i [∑Kₖₙ]
          tlow(i) = temperature of fluid from lower plenum entering channels of type i (°C) [t₁ₙ]
          tup(i) = temperature of fluid from upper plenum entering channels of type i (°C) [tₚ]
          phi(i) = φ defined from uₜ/ρ = uₜ/ρ₀(1-ϕ(t-t₀)) for channels of type i [ϕ]

Card 5:  control, itlim
          control = Reynolds number error allowed for convergence
          itlim = maximum number of iterations allowed

Card 6:  w₀
          w₀ = 100% inlet flowrate (kg/s)

Card 7:  q₀(i), i=1,n
          q₀(i) = 100% power for channels of type i (W)

Card 8:  wfrac, pfrac
          wfrac = defined by wflow0 = wfrac*w₀
          pfrac = defined by qheat(i) = pfrac*q₀(i)
          wflow0 = Inlet flow rate [mₙ₀]
          qheat(i) = channel power for channels of type i [qₙ]

Card 8 is repeated with different values of wfrac and pfrac for as many cases as the user wishes to run. The program will stop on an end-of-file indication.
program chansol
  dimension rey(20), theta(20), delta(20), gamma(20), geom(20),
    & diameq(20), aflow(20), qheat(20), rkloss(20),
    & wflow(20), s(20), fun(23), xl(20), xu(20), cfric(20), rnfric(20),
    & tup(20), tlow(20), phi(20), xi(20), wvec(20), num(20), q0(20)
  common/nondiml/rey, theta, delta, gamma, geom, xl, xu,
    & dpstar, rey0, xi, num
  common/dim/diameq, aflow, qheat, beta, rmu, cp, rkloss, cfric,
    & tup, tlow, t0, rho0, wflow, wflow0, de0, aflow0, chlen, deltap,
    & grav, phi
  common/convor/contol, iter, itlim
  common/press/fun

  c
  c*** This program finds the flow split for n channels connected
  c*** in parallel to common plena. The program makes the equal
  c*** pressure drop assumption. In addition, channels that draw
  c*** from the same plenum can have different input temperature
  c*** boundary conditions. No matrix algebra is involved in any
  c*** part of the program. Victor Iannello, March 1984.
  c
  c*** chin.dat is the input file (channel 7) and chout.dat is the
  c*** output file (channel 8). In addition, output is displayed
  c*** at the terminal.
  c***
  open(7, file='chin.dat')
  open(8, file='chout.dat', status='new')
  c
  c*** Subroutine READIT reads in the data and non-dimensionalizes
  c*** it.
  c
  idat=0
  100  continue
    call readit (n, rnfric, s, ncase, idat, q0, w0)
    if(idat.eq.2) stop
  c
  c*** Subroutine SOLVER solves the n+1 coupled equations.
  c
  call solver (n, rnfric, s)
  c
  c*** Subroutine results formats and prints the output.
  c
  call results(n, rnfric, s)
  goto 100
  200  continue
  1000 format(/,80(''),/80(''))
  end
subroutine readit(n,rnfric,s,ncase,idat,q0,w0)
  dimension rey(20),theta(20),delta(20),gamma(20),geom(20),
  & diameq(20),aflow(20),qheat(20),rkloss(20),
  & wflow(20),s(20),xl(20),xu(20),cfrc(i),rnfrc(i),
  & tup(20),tlow(20),phi(20),xi(20),num(20),q0(20)
  common/nondiml/rey,theta,delta,gamma,geom,xl,xu,
  & dpstar,rey0,xi,num
  common/dim/diameq,aflow,qheat,beta,rmu,cp,rkloss,cfrc,
  & tup,tlow,t0,rho0,wflow,wflow0,de0,aflow0,chlendeltap,
  & grav,phi
  common/conver/contol,iter,ilim

  c
  c*** This subroutine reads and non-dimensionalizes the input.
  c

  if(idat.eq.1) goto 500
  idat=1

  100 continue
  read (7,*) n,chlen,grav
  write(8,1105) n,chlen,grav
  write(*,1105) n,chlen,grav

  110 continue
  read (7,*) t0,rho0,cp,beta,rmu
  write(8,1115) t0,rho0,beta,rmu,cp
  write(*,1115) t0,rho0,beta,rmu,cp

  200 continue
  write(8,1200)
  write(*,1200)
  read (7,*) (num(i),diameq(i),aflow(i),s(i),i=1,n)
  write(8,1210)
  write(*,1210)

  230 continue
  do 245 i=1,n
  write(8,1220) i,num(i),diameq(i),aflow(i),s(i)
  write(*,1220) i,num(i),diameq(i),aflow(i),s(i)

  245 continue
  read (7,*) (cfrc(i),rnfrc(i),rkloss(i),tlow(i),tup(i),
    & phi(i),i=1,n)
  write(8,1230)
  write(*,1230)
  write(8,1240)
  write(*,1240)

  246 continue
  do 250 i=1,n
  write(8,1250) i,cfrc(i),rnfrc(i),rkloss(i),tlow(i),tup(i),
    & phi(i)
  write(*,1250) i,cfrc(i),rnfrc(i),rkloss(i),tlow(i),tup(i),
    & phi(i)

  250 continue
  read (7,*) contol,ilim
  write(8,1305) contol,ilim
  write(*,1305) contol,ilim
Calculate the equivalent diameter for the entire system.

wetpsum=0.
aflowsm=0.
do 330 i=1,n
   wetpsum=wetpsum+4*aflow(i)/diameq(i)*num(i)
aflowsm=aflowsm+aflow(i)*num(i)
330 continue
   aflow0=aflowsm
de0=4*aflowsm/wetpsum
write(8,1355) aflow0,de0
write(*,1355) aflow0,de0

Non-dimensionalize the channel parameters.

do 350 i=1,n
   rey(i)=100.0
   delta(i)=cfric(i)*rmu**2/2/diameq(i)**3/rho0**2/grav
   gamma(i)=rkloss(i)*rmu**2/2/diameq(i)**2
   & /chlen/rho0**2/grav
   geom(i)=aflow(i)/aflow0*de0/diameq(i)
xu(i)=beta*(tup(i)-t0)
xl(i)=beta*(tlow(i)-t0)
x(i)=phi(i)/beta
350 continue

dpstar=0.

Read in the operating conditions cases by first reading the full flowrate (w0) and channel powers q0(i).

read (7,*) w0
read (7,*) (q0(i),i=1,n)
500 continue

For each case, read in the flow fraction (wfrac) and power fraction (pfrac), and then calculate the absolute values of qheat(i), theta(i), wflow0, and rey0.

read (7,*,end=550) wfrac,pfrac
do 510 i=1,n
   qheat(i)=q0(i)*pfrac
   theta(i)=beta*qheat(i)*diameq(i)/aflow(i)/cp/rmu
510 continue
   wflow0=w0*wfrac
   rey0=wflow0/aflow0*de0/rmu
return
550 continue
idat=2
return
c
 c*** Format statements.
c
1100 format(i5,2e10.3)
1105 format(//' SYSTEM GEOMETRY',/,
 & ' number of channel types = ',i10,/, 
 & ' channel length (m) = ',1pe10.3,/, 
 & ' acc due to gravity (m/s2)= ',1pe10.3)
1110 format(5e10.3)
1115 format(//' FLUID PROPERTIES',/,
 & ' ref temp (C) = ',f10.3,2x,' ref dens (kg/m3)= ',f10.3,/, 
 & ' beta (/C) = ',1pe10.3,2x,' ref visc (Pa-s) = ',e10.3,/, 
 & ' spec heat (J/kg/C)= ',e10.3)
1200 format(//' CHANNEL GEOMETRY')
1210 format(2x,'Chan Type',7x,'Num',3x,'Eq D(m)',2x,'Area(m2)',3x,
 & 'Flx Shp')
1220 format(1x,i5,5x,i10,3(1pe10.3))
1230 format(//' CHANNEL FLOW RES & TEMP PARAMETERS (f=c/Re^n)')
1240 format(2x,'Chan Type',9x,'c',9x,'n',4x,'K-loss',3x,'Tlow(C)',
 & 4x,'Tup(C)',3x,'Phi(/C)')
1250 format(1x,i5,5x,6(1pe10.3))
1255 format(://' ini flow= ',1pe10.3,2x,'qheat = ',0pf10.3,/, 
 & 'tlow = 'f10.3,2x,'tup = ',f10.3)
1300 format(e10.3,15)
1305 format(//' ITERATION INFO',/,' convergence tolerance = ', 
 & 1pe10.3,2x,'iteration limit = ',13)
1310 format(3e10.3)
1355 format(//' MISC SYSTEM INFO',/,
 & ' total flow area (m2) = ',1pe10.3,/, 
 & ' hydraulic diam (m) = ',1pe10.3)
end

subroutine solver (n,rnfri,s)
dimension rey(20),theta(20),delta(20),gamma(20),geom(20),
& fun(23),dfdre(23),s(20),x1(20),xu(20),rnfri(20),xi(20),num(20)
common/nondiml/rey,theta,delta,gamma,geom,x1,xu,
& dpstar,rey0,x1,num
common/conver/contol,iter,ltlim
c
 c*** This subroutine solves the n+1 simultaneous equations to
c*** calculate the flow split and temperature rise for each
c*** channel.
c
n1=n+1
iterout=0
iterout is the counter for the outer (pressure) iteration
loop. iterin is the counter for the inner (rey(i)) loop.
The solution procedure is to choose dpstar and then calculate
the individual rey(i)'s based on this estimate. The mass
balance equation is then checked. If not satisfied within
the allowed tolerance, a new dpstar is chosen and the pro-
cedure continues.

continue
Start the outer (pressure) loop.
ierror=0
do 1000 i=1,n

Do the inner iteration loop n times (once per channel)
in order to calculate the individual channel flows.
iterin=0
if(rey(i).lt.0.) goto 500

Upflow case.
continue
rey(i)=abs(rey(i))
su=s(i)

Newton's method is used to solve for rey(i) for channels.
fun(i) is the momentum-energy equation and dfdre(i)
is its derivative with respect to rey(i). A positive
rey(i) is upflow.

vfac=1-xl(i)*x1(i)-theta(i)*su/rey(i)*x1(i)
fterm=delta(i)*rey(i)**(2-rnfric(i))
fun(i)=(-xl(i)-theta(i)/rey(i)*su) +
& fterm*vfac +
& (gamma(i)*rey(i)*rey(i)) - dpstar

dfdre(i)=(theta(i)*su/rey(i)/rey(i)) +
& ((2-rnfric(i))*delta(i)*rey(i)**(1-rnfric(i)))*vfac +
& (theta(i)*su/rey(i)/rey(i))*fterm +
& (2*gamma(i)*rey(i))

if(iterin.ge.itlim) then
  ierror=1
  goto 1000
endif
continue

 Obtain the new estimate of rey(i).
renew=rey(1) - fun(1)/dfdre(1) * 0.60  
error=abs(renew - rey(1))  
if(error.lt. contol .and. renew .ge. 0.0) then  
   rey(1)=renew  
   goto 1000  
endif  
rey(i)=renew  
iterin=iterin + 1  
if(rey(i).lt.0.0 .and. iterin.ge.(itlim/2)) goto 500  
goto 200  

c

*** Downflow case.  
c

500  
continue  
c

*** Newton's method is used to solve for rey(i) for channels.  
*** fun(i) is the momentum-energy equation and dfdre(i)  
*** is the derivative with respect to rey(i). A positive  
*** rey(i) is downflow in this section.  
c

sd=1.0-s(1)  
continue  
rey(i)=abs(rey(i))  
fterm=delta(i)*rey(i)**(2-rnfri(i))  
vfac=1-xu(i)*xi(i)-theta(i)*sd/rey(i)*xi(i)  
run(i)=(-xu(i)-theta(i)/rey(i)*sd) -  
& fterm*vfac  
& (gamma(i)*rey(i)*rey(i)) - dpstar  
c

dfdre(i)=(theta(i)*sd/rey(i)/rey(i)) -  
& ((2-rnfri(i))*delta(i)*rey(i)**(1-rnfri(i)))*vfac -  
& (theta(i)*sd/rey(i)/rey(i)*xi(i))*fterm -  
& (2*gamma(i)*rey(i))  
c

if(iterin.ge.itlim) then  
   ierror=1  
   rey(i)=rey(i)  
   goto 1000  
endif  
continue  
c

*** Obtain the new estimate of rey(i).  
c

renew=rey(i) - fun(1)/dfdre(1) * 0.60  
error=abs(renew - rey(i))  
if(error.lt. contol .and. renew .ge. 0.0) then  
   rey(i)=renew  
   goto 1000  
endif  
rey(i)=renew  
iterin=iterin + 1  
if(rey(i).lt.0.0 .and. iterin.gt.(itlim/2)) goto 200  
goto 600
Now that the channel flow rates are known for this dpstar, we check to see if continuity is satisfied. If not, we obtain a new estimate of dpstar using Newton's method on the continuity equation.

```c
fsum=0.0
dfsum=0.0
do 1500 i=1,n
   if (rey(i).lt.0.) goto 1200
   upflow
      fsum=fsum+rey(i)*geom(i)*num(i)
      dfsum=dfsum+geom(i)/dfdre(i)*num(i)
go to 1500
1200 continue
   downflow
      fsum=fsum+rey(i)*geom(i)*num(i)
      dfsum=dfsum-geom(i)/dfdre(i)*num(i)
1500 continue
```

We check for convergence. If not achieved, obtain new estimate of dpstar.

```c
fun(nl)=fsum-rey0
dfdre(nl)=dfsum
error=abs(fun(nl))
if(error.lt.contol .and. ierror.eq.0) goto 2000
dpstar=dpstar-fun(nl)/dfdre(nl)*0.90
iterout=iterout+1
if(iterout.lt.itlim) goto 50
```

If the continuity equation is not satisfied after itlim outer loop iterations, print error message and call results.

```c
write(*,3000) rey0
write(8,3000) rey0
2000 continue
return
```

Format statements follow.

```c
format(/'/5x,'***convergence not achieved for this','
&' 'channel ordering***',/8x,'rey0= ',f10.3)
end
```
subroutine results(n,rnfrc,s)
dimension rey(20),theta(20),delta(20),gamma(20),
& geom(20),rmat(23,23),diameq(20),aflow(20),qheat(20),
& rkloss(20),wflow(20),s(20),xl(20),xu(20),cfric(20),rnfrc(20),
& tup(20),tlow(20),phi(20),xi(20),num(20)
dimension tempout(20),tempin(20),fun(23)
common/nondim1/rey,theta,delta,gamma,geom,xl,xu,
& dpstar,rey0,xi,num
common/dim/diameq,aflow,qheat,beta,rmu,cp,rkloss,cfric,
& tup,tlow,t0,rho0,wflow,wflow0,de0,aflow0,chlen,deltap,
& grav,phi
common/conver/contol,iter,itolim
common/press/fun

c
*** This subroutine outputs the input data, non-dimensionalized
*** input data, and calculation results.
c
do 100 i=1,n
wflow(i)=rey(i)*aflow(i)*rmu/diameq(i)
if(rey(i).ge.0.) then
  tempout(i)=tlow(i)+theta(i)/rey(i)/beta
  tempin(i)=tlow(i)
else
  tempout(i)=tup(i)-theta(i)/rey(i)/beta
  tempin(i)=tup(i)
endif
100 continue
deltap=(dpstar+1.)*rho0*grav*chlen

c
*** Print out the results.
c
550 continue
write(*,1900)
write(8,1900)
write(*,2000) wflow0,rey0,deltap,dpstar
write(8,2000) wflow0,rey0,deltap,dpstar
write(*,2100)
write(8,2100)
c
do 600 i=1,n
write(*,2200) i,qheat(i),wflow(i),tempin(i),tempout(i)
write(8,2200) i,qheat(i),wflow(i),tempin(i),tempout(i)
600 continue
write(*,2300)
write(8,2300)
c
do 700 i=1,n
dpl=dpstar+fun(i)
write(*,2400)i,rey(i),theta(i),delta(i),gamma(i),xl(i),xu(i),xi(i)
write(8,2400)i,rey(i),theta(i),delta(i),gamma(i),xl(i),xu(i),xi(i)
700 continue
return
c
*** Format statements follow.
c
1900 format(/30x,'FLUID CONDITIONS',/30x,16('-'))
2000 format(13x,'Inlet Flow (kg/s) =',1pe10.3,5x,
& 'Inlet Rey=',0pf10.3,/,
& 13x,'Delta-P (Pa) =',1pe10.3,5x,
& 'Dpstar =',1pe10.3)
2100 format(/,1x,'Chan Type',2x,'Power(W)',2x,'Fl(kg/s)',4x,'Tin(C)',
& 3x,'Tout(C)')
2200 format(i5,5x,4(1pe10.3))
2300 format(/,' Chan Type',6x,'Rey',5x,'Theta',5x,'Delta',5x,'Gamma',
& 8x,'X1',8x,'Xu',8x,'Xi')]}
2400 format(i5,4x,7(1pe10.3))
end
Appendix B

ANNO and ANNA Codes

ANNO and ANNA are small codes implemented on the MIT Multics computer system designed to solve the fully-developed, mixed convection conservation equations for geometries with azimuthal symmetry. ANNA solves for the aiding flow case and ANNO for the opposing flow case. Both codes use an input file of the same format so that identical cases can be run for aiding and opposing by running the appropriate codes. The format of the input file is free format. The input file is assumed to be file09 and the output file10.

For Bessel function calculations, the appropriate NAG or IMSL subroutine is called. The subroutines called for the specific Bessel functions are

ANNA: MMKEL0: ber₀, bei₀, ker₀, kei₀
       MMKELD: ber₀', bei₀', ker₀', kei₀'

ANNO: S17aef:J₀
       S17aff:J₁
       S17acf:Y₀
       S17adf:Y₁
       S18aef:I₀
       S18aff:I₁
       S18acf:K₀
       S18adf:K₁

In addition, IMSL routine leqt2f solves the four simultaneous boundary condition equations for both ANNA and ANNO.
Input Description

Card1: ngroqre, ibc, igrav, astar
ngroqre = number of $Gr_q/Re$ cases considered
  1 inner radius heated annulus
ibc =  2 outer radius heated annulus
       3 circular tube
       4 equivalent annulus
igrav = 0 no printing at terminal
       1 with printing at terminal
astar = inner/outer radius ratio

Card2: grovev(1), i=1,ngroqre
grovev(1) = $Gr_q/Re$ value corresponding to case 1
program anna
  double precision x,xl,xo,ber,bei,rker,kerl,berp.
  & beil,kerlp,kerlp,xad,xh
  dimension a(4,4),b(4),wkarea(28),gorev(50),tb(101)
  common/block/ igrav
  c
  c***This program calculates the mixed convection velocity profiles
  c***for flow in annular and circular channels. The possible
  c***combinations are:
  c*** 1bc=1 Annular channel with heated inside perimeter
  c*** 1bc=2 Annular channel with heated outside perimeter
  c*** 1bc=3 Circular channel (with heated outside perimeter)
  c*** 1bc=4 Annular channel with a no shear outer boundary
  c***Gore is the Gr/Re parameter, and aster is the inside to
  c***outside radius ratio for the annulus (=0 for circular
  c***channel). Azimuthal symmetry is assumed. Also, igrav=1
  c***for aiding flow, and igrav=-1 for opposing flow.
  c
  rewind (9)
  pi=3.1415927
  read (9,*) ngore,1bc,igrav,aster
  read (9,*) (gorev(i),i=1,ngore)
  c
  do 9999 istar=1,9
  c
  aster=istar/10.
  c
  c***phopw is the heated to wetted perimeter ratio
  c
  if(1bc.eq.3) aster=0.
  if(1bc.eq.4) then
    ph=2*0*pl*aster
  else
    ph=2*0*pl*(aster+1.0)
  endif
  if(1bc.eq.1 .or. 1bc.eq.4) then
    ph=2*0*pl*aster
  else
    ph=2*0*pl
  endif
  phomw=ph/pw
  af=pl*(1.0-aster*aster)
  de=4.0*af/pw
  c
  do 700 ntrial=1,ngore
  gore=gorev(ntrial)
  if(igrav.eq.1) write(6,1105)
  write(10,1105) 1105 format(55('**'))
  if(igrav.eq.1) write(6,1100) gore,aster,phopw,1bc
  write(10,1100) gore,aster,phopw,1bc
  1100 format(5X,'gr/rev'=',f10.4,5X,'aster=',f10.4,
    & /5X,'ph/w'=',f10.4,5X,'1bc=',110,110,
    & /5X,'grav = aiding')
  c
  c***The correlating parameter is eta
  c
  eta4=gore*4.*phopw
  eta=eta4**0.25
  c
  c***Rl and ro are the inside and outside radii, respectively.
c*** divided by the hydraulic radius (=Do-Di).
c
d=2.*(1.-astar)
if(ibc.eq.4) d=2.*(1.-astar*astar)/astar
ri=astar/de
ro=1./de
xi=ri*eta
xo=ro*eta
xad=xo
if(ibc.eq.2 .or. ibc.eq.3) xad=x1
xh=x1
if(xad.eq.x1) xh=xo
if(ibc.eq.3) goto 200

c*** Set up the matrix to calculate the constants based on
the four boundary conditions. They are:
ibc=1: u(r1)=u(ro)=q(ro)=t(ro)=0.
ibc=2: u(r1)=u(ro)=q(r1)=t(r1)=0.
ibc=3: u'(r1)=u(ro)=q(r1)=t(r1)=0.
where u' denotes the gradient of u.

c 1er=0
rnrm2=j0(x1)/y0(x1)
rnrm3=j0(x1)/i0(x1)
rnrm4=j0(x1)/k0(x1)

c a(1,1)= j0(x1)
a(1,2)= y0(x1)*rnrm2
a(1,3)= i0(x1)*rnrm3
a(1,4)= k0(x1)*rnrm4

if(ibc.ne.4) goto 125

c a(2,1)= -j1(xo)
a(2,2)= y1(xo)*rnrm2
a(2,3)= i1(xo)*rnrm3
a(2,4)= -k1(xo)*rnrm4
goto 126

125 continue
a(2,1)= j0(xo)
a(2,2)= y0(xo)*rnrm2
a(2,3)= i0(xo)*rnrm3
a(2,4)= k0(xo)*rnrm4

126 continue
a(3,1)= j1(xad)
a(3,2)= y1(xad)*rnrm2
a(3,3)= i1(xad)*rnrm3
a(3,4)= -k1(xad)*rnrm4

c a(4,1)= j1(xh)
a(4,2)= y1(xh)*rnrm2
a(4,3)= i1(xh)*rnrm3
a(4,4)= -k1(xh)*rnrm4

118 continue
write(*,120) ((a(i,j),j=1,4),i=1,4)
120 format(4(ipe10,3))
b(1)=0.
b(2)=0.
b(3)=0.
b(4)=grore/eta/eta/eta/eta
if(lbc.eq.2) b(4)=b(4)

*** Subroutine leqt2f is an IMSL routine that solves linear
*** simultaneous equations. The coefficients are left
*** in vector b.

m=1
n=4
la=4
ldgt=0
ier=0
call leqt2f(a,m,n,la,b,ldgt,wkarea,ier)

b(2)*b(2)*rnorm2
b(3)*b(3)*rnorm3
b(4)*b(4)*rnorm4
goto 250

continue
b(1)=-10(xh)*grore/eta/eta/eta
   b(1)=b(1)/((j0(xh)*i1(xh)-10(xh)*j1(xh))
b(2)=0.0
b(2)*j0(xh)*grore/eta/eta/eta
b(3)=b(3)/((j0(xh)*i1(xh)-10(xh)*j1(xh))
b(4)=0.0

continue

*** The friction factor based on the fluid shears (gradients) at the
*** channel walls is calculated. Notice that it is two times greater
*** than the pressure gradient factor. This is because our choice of
*** the zero temperature is the area-averaged one. The friction factor
*** is assumed to be of the form f=c/Re for mixed convection, and
*** for purely forced (isothermal) convection.

twi=0.
if(lbc.eq.3) goto 310
rj1=j1(x1)
ry1=y1(x1)
il1=i1(x1)
rl1=k1(x1)
twi=-b(1)*rj1+b(2)*ry1+b(3)*ril1+b(4)*rl1
continue

rj1=j1(x0)
ry1=y1(x0)
il1=i1(x0)
rl1=k1(x0)
two=-b(1)*rj1+b(2)*ry1+b(3)*ril1+b(4)*rl1
if(lbc.eq.4) goto 335

c=eta*8.+(twi*astar/(1.+astar)-two/(1.+astar))
goto 340

continue

335 c=eta*8.*twi
340 continue
r1=c/2.0

*** Output the results for the whole channel.
if(grav.eq.1) write( 6,1500)
write(10,1500)
1500 format(10x,"rstar",5x,"r/de",6x,"ustar",5x,"tstar",5x,"qstar",
& /10x,45(*-*)

do 400 i=0,10
   rstar=1/10.
   r=r*i*rstar*(ro-ri)
   x=r*eta
   ier=0
   if(ibc.eq.3) goto 370
   rj0= j0(x)
   rj1= j1(x)
   ry0= y0(x)
   ry1= y1(x)
   ri0= 10(x)
   ri1= 11(x)
   rk0= k0(x)
   rk1= k1(x)
   goto 380
370 continue
   rj0=j0(x)
   rj1=j1(x)
   ry0=0d0
   ry1=0d0
   ri0=10(x)
   ri1=11(x)
   rk0=0d0
   rk1=0d0
380 continue
   u= b(1)*rj0+b(2)*ry0+b(3)*ri0+b(4)*rk0
   q= b(1)*rj1+b(2)*ry1+b(3)*ri1+b(4)*rk1
   q=q*eta*eta*eta/grove
   if(ibc.eq.2 .or. ibc.eq.3) q=-q
   t=b(1)*rj0-b(2)*ry0+b(3)*ri0+b(4)*rk0
   t=(-r1-t*eta*eta)/grove
   if(dabs(x-xh)).le.1d-6) twall=t
   if(grav.eq.1) write( 6,2000) rstar,r,u,t,q
write(10,2000) rstar,r,u,t,q
2000 format(5x,5f10.3)
400 continue
   if(grav.eq.1) write( 6,2200) c,r1
write(10,2200) c,r1
2200 format(5x,"f*re="",f10.3,5x,"press grad fact=",f8.3)
c***The forced convection friction factor for the circular tube (ibc=3)
c***is fo=64./Re, while the value for the annulus (ibc=1 or ibc=2) is
c
ciso=64.
if(ibc.eq.3) goto 450
if(ibc.eq.4) goto 445
bracket=1.*astar**4*(1.*astar**2.)*2.)/alog(1.*astar)
ciso=(ciso*(1.*astar**2.)*(1.*astar)**2.)/bracket
goto 450
445 continue
bracket=0.75-astar**4/4.+astar**4+1./astar
 cis0=16.*(1.-astar**astar)**3/astar**2/bracket
450 continue
f0fiso=c/ciso

c
if(grav.eq.1) write(6,2300) cis0,f0fiso
write(10,2300) cis0,f0fiso
2300 format(5x,"f0f=",f10.3,5x,"f/f0=",f10.3)
c

***Calculate the Nusselt number rnu by first finding the bulk
***temperature by volumetric flow weighting the integral and
***then computing rnu from the bulk and wall temperatures.

***
do 600 i=1,101
r=r+((1-1)*(ro-ri))/100.0
x=r*eta
ier=0

c
if(lbc.eq.3) goto 570
rj0= j0(v)
rj1= j1(x)
ry0= y0(s)
ry1= y1(x)
r10= 10(x)
ri= i1(x)
rk0= k0(x)
 rk1= k1(x)
goto 580
570 continue
rj0= j0(x)
rj1= j1(x)
ry0= 0d0
ry1= 0d0
r10= 10(x)
ri= i1(x)
 rk0= 0d0
 rk1= 0d0

c
580 continue
u= b(1)+rj0+b(2)+ry0+b(3)+r10+b(4)+rk0
t=-b(1)+rj0+b(2)+ry0+b(3)+r10+b(4)+rk0
t=-t+eta*eta/grore
 tb(1)=u+t+r
600 continue
dr=(ro-ri)/100.0
 tbulk=trapint(tb,dr,101)*2.0*pi*des/af
 rnu=1.0/(twall-tbulk)
 if(grav.eq.1) write(6,3000) tbulk,rnu
 write(10,3000) tbulk,rnu
3000 format(5x,"tbulk=",fpe10.3,5x,"Nu = ",0pf10.3)
 bigf=-2.0*grore*tbulk/c
 frm=+f0fiso+(1.0+bigf)
 if(grav.eq.1) write(6,3100) bigf,frm
 write(10,3100) bigf,frm
3100 format(5x,"f* ",fpe10.3,5x,*(f/fo)(1+f)*",0pf10.3)
700 continue

c
continue
stop
end
real function j0(x)
double precision x
j0=s17ae(x,ier)
return
end
real function j1(x)
double precision x
j1=s17aff(x,ier)
return
end
function y0(x)
double precision x
y0=s17acf(x,ier)
return
end
function y1(x)
double precision x
y1=s17adf(x,ier)
return
end
real function l0(x)
double precision x
l0=s18ae(x,ier)
return
end
real function l1(x)
double precision x
l1=s18aff(x,ier)
return
end
real function k0(x)
double precision x
k0=s18acf(x,ier)
return
end
real function k1(x)
double precision x
k1=s18adf(x,ier)
return
end
function trapint(y,dx,npoints)
dimension y(101)
sum=0.0
npoint1=npoints-1
do 100 i=2,npoint1
   sum=sum+2*y(i)
   100 continue
sum=sum+y(1)+y(npoints)
trapint=sum*dx/2.0
return
end
program anno
  double precision x, xi, xo, rj0, rj1, r10, r11, rk0, rk1, ry0, ry1,
    & xed, xh, order, yn(2)
  real *4 j0, j1, k0, k1, 10, 11, y0, y1
  dimension a(4,4), b(4), wkarea(28), gровер(50), tb(101)
  c
  c***This program calculates the mixed convection velocity profiles
  c***for flow in annular and circular channels. The possible
  c***combinations are:
  c*** ibc=1 Annular channel with heated inside perimeter
  c*** ibc=2 Annular channel with heated outside perimeter
  c*** ibc=3 Circular channel (with heated outside perimeter)
  c*** ibc=4 Annular channel with a no shear outer boundary
  c*** Grore is the Gro/Re parameter, and aster is the inside to
  c***outside radius ratio for the annulus (=0 for circular
  c***channel). Azimuthal symmetry is assumed. Also, igrav=1
  c***for aiding flow, and igrav=-1 for opposing flow.
  c
  rewind (9)
  pi=3.1415927
  read (9,*) ngore, ibc, igrav, aster
  read (9,*) (gровер(l), l=1,ngore)
  c
  do 9999 1istar=1,9
  c
  c*** phopw is the heated to wetted perimeter ratio
  c
  if(ibc.eq.3) aster=0.
  if(ibc.eq.4) then
    pw=2.0*pi*aster
  else
    pw=2.0*pi*(1.0+aster)
  endif
  if(ibc.eq.1 .or. ibc.eq.4) then
    ph=2.0*pi*aster
  else
    ph=2.0*pi
  endif
  af=pi*(1.0-aster*aster)
  de=4.0*af/pw
  phopw=ph/pw
  c
  do 700 ntrial=1,ngore
  ngore=gровер(ntrial)
  if(igrav.eq.1) write( 6,1105)
  write(10,1105)
1105 format(55('**'))
  if(igrav.eq.1) write( 6,1100) grore, aster, phopw, ibc
  write(10,1100) grore, aster, phopw, ibc
1100 format( 5x,'gr/re*  ',f10.4,5x,'aster* ',f10.4, &
      /5x,'ph/pw*  ',f10.4,5x,'ibc = ',i10, &
      /5x,'grav = opposing')
  c
  c*** The correlating parameter is eta
  c
  eta4=grupo+phopw
  eta=eta4**0.25
  c
  c*** R1 and ro are the inside and outside radii, respectively,
c***divided by the hydraulic radius (=Do/Di).
c    ri=astar/de 
    ro=1./de 
    xi=ri*eta 
    xo=ro*eta 
    xad=xo 
    if(IBC.eq.2 .or. IBC.eq.3) xad=x1 
    xh=x1 
    if(xad.eq.x1) xh=xo 
    if(IBC.eq.3) goto 200 
c
    c***Set up the matrix to calculate the constants based on 
c***the four boundary conditions. They are: 
c***IBC=1: u(r1)=u(ro)=q(ro)=t(ro)=0. 
c***IBC=2: u(r1)=u(ro)=q(r1)=t(r1)=0. 
c***IBC=3: u'(r1)=u(ro)=q(r1)=t(r1)=0. 
c***where u' denotes the gradient of u. 
c    IER=0 
    " 
c    CALL MMKEL0(X1,BER,BEL,RKER,RKEL,IER) 
    RNAME=ABS(BER/RKER) 
    A(1,1)=BER 
    A(1,2)=BEL 
    A(1,3)=RKER*RNAME 
    A(1,4)=RKEL*RNAME 
    " 
c    IF(IBC.NE.4) GOTO 125 
    CALL MMKELD(XO,BER,BEL,RKER,RKEL,IER) 
    GOTO 126 
    125 CALL MMKEL0(XO,BER,BEL,RKER,RKEL,IER) 
    126 CONTINUE 
    A(2,1)=BER 
    A(2,2)=BEL 
    A(2,3)=RKER*RNAME 
    A(2,4)=RKEL*RNAME 
    " 
    CALL MMKELD(XAD,BERP,BELP=RKER,RKELP,IER) 
    A(3,1)=BELP 
    A(3,2)=BERP 
    A(3,3)=RKELP*RNAME 
    A(3,4)=RKER*RNAME 
    " 
    CALL MMKELD(XH,BERP,BELP=RKER,RKELP,IER) 
    A(4,1)=BELP 
    A(4,2)=BERP 
    A(4,3)=RKELP*RNAME 
    A(4,4)=RKER*RNAME 
    " 
    118 CONTINUE 
    " 
    B(1)=0. 
    B(2)=0. 
    B(3)=0. 
    B(4)=-GROE/ETA/ETA/ETA 
    IF(IBC.EQ.1 .OR. IBC.EQ.4) B(4)=-B(4) 
    " 
    c***Subroutine leqt2f is an IMSL routine that solves linear 
c***simultaneous equations. The coefficients are left
c***in vector b.
c
m=1
n=4
la=4
ldgt=0
ier=0
call leqt2f(a,m,n,la,b,ldgt,wkarea,ier)
b(3)=b(3)*rnorm
b(4)=b(4)*rnorm
if(ier.ne.3) goto 250
c
200 continue
call mmkel0(xh,ber,be1,rrkr,rrke1,ier)
call mmkeld(xh,berp,be1p,rkerp,rrke1p,ier)
b(1)=be1*grore/eta/eta/eta
b(1)=be1*(berp+be1+be1p)
b(2)=ber*grore/eta/eta/eta
b(2)=berp+be1+be1p
b(3)=0d0
b(4)=0d0
250 continue
C
C***The friction factor based on the fluid shears (gradients) at the
C***channel walls is calculated. Notice that it is two times greater
C***than the pressure gradient factor. This is because our choice of
C***the zero temperature is the area-averaged one. The friction factor
C***is assumed to be of the form f=c/Re for mixed convection, and
C***for ciso/Re for purely forced (isothermal) convection.
C
twi=0.
if(ier.eq.3) goto 310
call mmkeld(xi,berp,be1p,rkerp,rrke1p,ier)
twi=b(1)+berp+b(2)+be1p+b(3)*rkerp+b(4)*rrke1p
310 continue
C
call mmkeld(xo,berp,be1p,rkerp,rrke1p,ier)
two=b(1)+berp+b(2)+be1p+b(3)*rkerp+b(4)*rrke1p
if(ier.eq.4) goto 335
c=eta*8.*(twi+aestar/(1+aestar)-two/(1+aestar))
goto 340
335 c=eta*8.*twi
340 continue
r1=c/2.0
C
C***Output the results for the whole channel.
c
if(igrav.eq.1) write(6,1500)
write(10,1500)
1500 format(10x,"rstar",5x,"r/de",6x,"ustar",5x,"tstar",5x,"qstar",&
10x,45("-"))
do 400 i=0,10
rstar=i/10.
r=r1+rstar*(ro-r1)
x=x*eta
ier=0
c
call mmkel0(x,ber,be1,rrkr,rrke1,ier)
call mmkeld(x,berp,be1p,rkerp,rrke1p,ier)
c
u = b(1)*ber + b(2)*bei + b(3)*rker + b(4)*rkei
q = -b(1)*bei + b(2)*ber - b(3)*rkei + b(4)*rker
q = q*eta*eta/eta/grore
if(ibc.eq.2 .or. ibc.eq.3) q = -q
t = -b(1)*bei + b(2)*ber - b(3)*rkei + b(4)*rker
t = (-r1-t*eta*eta)/grore
if((dabs(x-xh)).lt.1e-6) twall = t
if(igrav.eq.1) write(6,2000) rstar,r,u,t,q
write(10,2000) rstar,r,u,t,q
2000 format(5x,5f10.3)
400 continue

if(igrav.eq.1) write(6,2000) c,r1
write(10,2200) c,r1
2200 format(5x,"f*re = ",f10.3) x," press grad fact = ",f8.3)
c
***The forced convection friction factor for the circular tube (ibc=3)
c***is fso=64./Re, while the value for the annulus (ibc=1 or ibc=2) is
c
ciso=64.
if(ibc.eq.3) goto 450
if(ibc.eq.4) goto 445
bracket=1. -astar**4. -((1. -astar+2.)**2.)/alog(1./astar)
ciso=ciso*(1. -astar+2.)**2./bracket
goto 450
445 continue
bracket=0.75-astar**4/4.+astar*alog(1./astar)
ciso=16.*((1.-astar)*astar)**3/astar**2/bracket
450 continue
fofiso=c/ciso

if(igrav.eq.1) write(6,2300) ciso,fofiso
write(10,2300) ciso,fofiso
2300 format(5x,"f*fo= ",f10.3) x,"f/fo= ",f10.3)
c***Compute the Nusselt number rnu by first calculating the bulk
c***temperature by volumetric flow weighting the integral and then
c***calculating rnu from tbulk and twall.
c***
do 600 i=1,101
r=r1+(i-1)/100.0*(ro-r1)
x=r*eta
ier=0
c
call mmkelo(x,ber,bei,rker,rkei,ier)
call mmkeo(x,ber,bei,rker,rkei,ier)
c
c
u = b(1)*ber + b(2)*bei + b(3)*rker + b(4)*rkei
t = -b(1)*bei + b(2)*ber - b(3)*rkei + b(4)*rker
t = (-r1-t*eta*eta)/grore
tb(1) = u*t+r
tb(1) = u*t+r
600 continue
dr=(ro-r1)/100.0
rbulk=trapez(tb,dr,101)*de=de/af*2.0*pi
rnu=1./(twall-tbulk)
if(igrav.eq.1) write(6,3000) rbulk,rnu
write(10,3000) rbulk,rnu
3000 format(5x,"rbulk= ",p10.3,5x,"nu = ",pf10.3)
bigf=2.0*rbulk*grore/c
frmod=(1.0+bigf)*fofiso
if(igrav.eq.1) write(6,3100) bigf,frmod
write(10,3100) bigf,frmod
3100 format(5x,",F=".ipe10.3,5x,"(f/fo)(1+F)=".opf10.3)
C 700 continue
C
9999 continue
stop
end
function trapint(y,dx,npoints)
dimension y(101)
npoint1=npoints-1
sum=0.0
do 100 i=2,npoint1
    sum=sum+2.0*y(i)
100 continue
trapint=(sum+y(1)+y(npoints))/2.0*dx
return
end
Appendix C

FIT Code

FIT is a small code that was used to fit the friction factor ratio to the form

$$\frac{4}{n=1} \sum q_i \lambda_i^1$$

(C.1)

where

$$\lambda = \ln(1 + Gr_q/Re)$$

(C.2)

and \( r \) is either \((f/f_0)\) or \((f/f_0)(1+F)\). The program minimizes the error function

$$E = \sum_{n=1}^N \left( \sum_{i=0}^4 q_i \lambda_i^1 - r_n \right)^2$$

(C.3)

where \( N \) data pairs \((\lambda_n, r_n)\) are input. The function \( E \) is minimized with respect to \( q_0, q_1, q_2, q_3, \) and \( q_4 \) and thus there will be five equations to be solved simultaneously. Minimizing \( E \) with respect to \( q_k \) produces

$$\frac{\partial E}{\partial q_k} = 2 \sum_{n=1}^N \left[ \sum_{i=0}^4 q_i \lambda_i^1 - r_n \right] \lambda_k^1 \lambda_n^1 = 0 \quad 0 \leq k \leq 4$$

(C.4)

After rearranging, this equation becomes

$$\sum_{n=1}^N \left( \sum_{i=0}^4 q_i \lambda_i^1 \right) \lambda_k = \sum_{n=1}^N \lambda_k \lambda_n \quad 0 \leq k \leq 4$$

(C.5)

Written in matrix form, Eq. (C.5) is

$$A \times q = C$$

(C.6)

where
\[ A = \begin{bmatrix} N & \Sigma \lambda_n & \Sigma \lambda_n^2 & \Sigma \lambda_n^3 & \Sigma \lambda_n^4 \\ \Sigma \lambda_n & \Sigma \lambda_n^2 & \Sigma \lambda_n^3 & \Sigma \lambda_n^4 & \Sigma \lambda_n^5 \\ \Sigma \lambda_n^2 & \Sigma \lambda_n^3 & \Sigma \lambda_n^4 & \Sigma \lambda_n^5 & \Sigma \lambda_n^6 \\ \Sigma \lambda_n^3 & \Sigma \lambda_n^4 & \Sigma \lambda_n^5 & \Sigma \lambda_n^6 & \Sigma \lambda_n^7 \\ \Sigma \lambda_n^4 & \Sigma \lambda_n^5 & \Sigma \lambda_n^6 & \Sigma \lambda_n^7 & \Sigma \lambda_n^8 \end{bmatrix} \]  

\[ a_1 \] 

\[ a_2 \] 

\[ a_3 \] 

\[ a_4 \]  

and 

\[ C = \begin{bmatrix} \Sigma r_n \\ \Sigma r_n \lambda_n \\ \Sigma r_n \lambda_n^2 \\ \Sigma r_n \lambda_n^3 \\ \Sigma r_n \lambda_n^4 \end{bmatrix} \]  

(C.7a) 

(C.7b) 

(C.7c) 

All summations in Eq. (C.7) run from \( n=1 \) to \( N \). 

Equation (C.6) is solved in FIT by the IMSL library leq2f, which finds the solution of simultaneous linear equations.
Input Description

The input format is free-format. FIT assumes file35 is the input file, and outputs results to file36. Since free formatting is used, do not use a card identifier in any column or it will be read.

Card1: title
title = label of less than 80 characters for data set

Card2: qrqre(i), fratio(i)
qrqre(i) = Grq/Re for data set
fratio(i) = friction factor ratio for data set

Card 2 is repeated for each data pair. The end of input data is signaled by a data pair of qrqre(i) = fratio(i) = -1. A maximum of 100 data pairs is allowed.

The user may fit additional data sets in the same computer run by sequentially supplying Cards 1 and 2 for each data set desired. The program will automatically stop when an end-of-file is encountered when attempting to read Card 1 (the title) for a data set.
program fit
    implicit real*4(a-h,o-z)
    character*80 title
    dimension a(5,5),c(5),warea(150),grqre(100),fratio(100)

    rewind(35)
    read(35,45,end=5000) title
    write(36,50) title

    format(a80)
    format(/a80)
    i=1
    100 continue

    read (35,*) grqre(i),fratio(i)
    if(grqre(i).lt.0) goto 200
    if(grqre(i).lt.0.) grqre(i)=id-3
    write(36,110) grqre(i),fratio(i)
    110 format(5x,Grq/Re=",1pd10.3, f`fratio=",1pd10.3)
    i=i+1
    goto 100

    npoints=1-1

    continue

    c
    200 continue
    npoints=1-1

    c
    perform summations for matrices.

    300 continue
    grqre1=0d0
    grqre2=0d0
    grqre3=0d0
    grqre4=0d0
    grqre5=0d0
    grqre6=0d0
    grqre7=0d0
    grqre8=0d0

    c
    fsum1=0d0
    fsum2=0d0
    fsum3=0d0
    fsum4=0d0
    fsum5=0d0

    c
    do 400 i=1,npoints
        gr=alog(grqre(i)+1)
        grqre1=grqre1+gr
        grqre2=grqre2+gr+gr
        grqre3=grqre3+gr+gr+gr
        grqre4=grqre4+gr+gr+gr+gr
        grqre5=grqre5+gr+gr+gr+gr+gr
        grqre6=grqre6+gr+gr+gr+gr+gr+gr
        grqre7=grqre7+gr+gr+gr+gr+gr+gr
        grqre8=grqre8+gr+gr+gr+gr+gr+gr+gr

    c
    fr=alog(fratio(i))
fsun1 = fsun1 + fr
fsun2 = fsun2 + fr * gr
fsun3 = fsun3 + fr * gr * gr
fsun4 = fsun4 + fr * gr * gr * gr
fsun5 = fsun5 + fr * gr * gr * gr

400 continue

c*** Set up matrices.

c***

a(1,1) = npoints
a(1,2) = grqre1
a(1,3) = grqre2
a(1,4) = grqre3
a(1,5) = grqre4

c

a(2,1) = grqre1
a(2,2) = grqre2
a(2,3) = grqre3
a(2,4) = grqre4
a(2,5) = grqre5

c

a(3,1) = grqre2
a(3,2) = grqre3
a(3,3) = grqre4
a(3,4) = grqre5
a(3,5) = grqre6

c

a(4,1) = grqre3
a(4,2) = grqre4
a(4,3) = grqre5
a(4,4) = grqre6
a(4,5) = grqre7

c

a(5,1) = grqre4
a(5,2) = grqre5
a(5,3) = grqre6
a(5,4) = grqre7
a(5,5) = grqre8

c

c(1) = fsun1

b(2) = fsun2

b(3) = fsun3

b(4) = fsun4

b(5) = fsun5

c*** Solve the system using the IMSL routine leqt2f

c***

m = 1
n = 5
ia = 5
ldgt = 0
ier = 0
call leqt2f(a, m, n, ia, c, ldgt, wkrea, ier)
if (ier .ne. 0) write(36, 500) ier

500 format(5x, '***ERROR CONDITION--IER= ', i3, '***')

c***

c*** Print out the results

c***

write(36, 1000) c(1), c(2), c(3), c(4), c(5)

1000 format(1x, 'a0=' ', lpe10.3, 2x, 'a1=' ', lpe10.3, 2x, 'a2=' ', lpe10.3
\begin{verbatim}
\& 2x,'a3=','fpe10.3,2x,'a4=','fpe10.3)
goto 10
5000 continue
stop
end
\end{verbatim}
D.1 Introduction

In this Appendix, the numerical methods incorporated into the MICON code are described. MICON was written by Dr. Sorel Kaizerman while he was associated with the MIT Nuclear Engineering Department in 1983-4. The code was debugged and implemented on the MIT Multics computer system as part of this work. Much of the content of this appendix was directly taken from Refs. K-2 and K-3, or slightly modified to incorporate changes made as part of this work. The nomenclature used in this appendix is consistent with that used in programming the MICON code, and may or may not correspond with the nomenclature used with the earlier chapters of this report. In this way, the appendix stands alone as a reference for the MICON code.
D.2 Mathematical Model for the Fluid

D.2.1 Fluid Momentum-Integral Equation

D.2.1.1 Formulation

By averaging the momentum transport equation over a cross-sectional area, $A$, perpendicular to the vertical $z$-axis, the following one-dimensional momentum is obtained:

$$\frac{\partial W}{\partial t} + \frac{\partial}{\partial z} \left( W^2 \rho A \right) + A \frac{\partial p}{\partial z} + \frac{1}{A} \frac{F \cdot W}{2 \rho De} + A \rho q = 0 \quad (D.1)$$

In Eq. (D.1), all the quantities are averaged values over the cross sectional area, $A$. In this equation, $W$ is the mass flow rate, i.e., $W = A \rho v_z = AG$.

In this equation, the friction term appears in a usual form. Here, the non-dimensional coefficient, $F$, includes both the friction coefficient, $f$, and the local pressure loss coefficient. In Eq. (D.1), $De$ is the hydraulic diameter and $q$ is the acceleration due to gravity.

Additionally, the fluid is assumed incompressible and therefore the continuity equation is written as

$$W(z,t) = W(t) \quad (D.2)$$

The friction coefficient, $f$, is assumed to be given by:

$$f = C_1 Re^{-C_2} (f/f_0)(1+F) = C_1 \left( \frac{WDe}{\mu A} \right)^{-C_2} (f/f_0)(1+F) \quad (D.3)$$

The equations are now written for a parallel channel system consisting of $I$ channels plus one bypass channel. The bypass channel can be identical to the other channels, but is labeled as such in the formulation here.

The area averaged momentum equation, Eq. (D.1) can then be integrated over the channel length and the integral momentum equation is obtained, for a certain channel, $i$, as follows:
\[(P_o - P_L)_I = \frac{L_1}{A_1} \frac{dW_1}{dt} + W_1^2 I_m,1 + \]
\[w_1 \left| W_1 \right|^{1-C_2} I_{f,1} + W_1 \left| W_1 \right| I_{k,1} + I_{q,1} \]  

(D.4)

where \(o\) and \(L\) refer to the bottom and top of the channel, respectively,

\[I_{m,1} \equiv \frac{1}{A_1^2} \left| \frac{3}{\rho} \left( \frac{1}{\rho} \right) \frac{dz}{dz} \right|_1 = \frac{1}{A_1^2} \left[ \frac{1}{\rho_L} - \frac{1}{\rho_o} \right] \]  

(D.5)

\[I_{f,1} \equiv \frac{C_1}{2} \left[ \frac{1}{D_{1+2+C_2+2-C_2}} \left| \left( \frac{v}{\rho} \right) (f/f_o)(1+F)dz \right|_1 \right] \]  

(D.6)

\[I_{k,1} \equiv \frac{1}{2A_1^2} \left[ \frac{\sum(k/\rho)}{\rho} \right]_1 = \frac{1}{2A_1^2} \left[ (k/\rho)_{\text{contraction}} + (k/\rho)_{\text{expansion}} + (k/\rho)_{\text{local}} \right]_1 \]  

(D.7)

\[I_{q,1} \equiv q \left| \frac{dz}{dz} \right|_1 \]  

(D.8)

The temporal derivative in Eq. (D.4) is approximated by:

\[\frac{dW_1}{dt} = \frac{W_1(t+\Delta t) - W_1(t)}{\Delta t} = \frac{\Delta W_1}{\Delta t} \]  

(D.9)

All other terms in Eq. (D.4) are assumed to be averaged values over the time interval \((t, t+\Delta t)\). The terms depending on the flow rate, averaged over the time interval, are approximated by:

\[\bar{F}(W) = \frac{1}{2} [F(W+\Delta W) + F(W)] = \frac{1}{2} \left[ F(W) + \Delta W \frac{dF(W)}{dW} + F(W) \right] = \]

\[= F(W) + \frac{1}{2} \Delta W \frac{dF(W)}{dW} \]  

(D.10)

where \(W\) and \(F(W)\) are values at the old time step. Therefore:
\[
\begin{align*}
\overline{W_1^2} &= W_1\,|W_1| + |W_1|\Delta W \\
\overline{W_1^{1-C_2}} &= W_1|W_1|^{1-C_2} + \frac{2-C_2}{2} |W_1|^{1-C_2}\Delta W \\
\overline{W_1^2} &= W_1^2 + W_1\Delta W
\end{align*}
\]

Using Eq. (D.11) in Eq. (D.4) produces:

\[
(P_o - P_L) = \Delta W_1 B_1 + C_1
\]

where:

\[
B_1 = B_1(t) = \frac{L_1}{A_1} \frac{1}{\Delta t} + W_1^*I_{m,1} + \frac{2-C_2}{2} |W_1|^{1-C_2}I_{f,1} + |W_1|I_{k,1}
\]

\[
C_1 = C_1(t) = W_1^2 \cdot I_{m,1} + W_1^*|W_1|^{1-C_2}I_{f,1} + W_1^*|W_1|I_{k,1} + I_{q,1}
\]

\[
W_1 = W_1(t)
\]

We assume now that in either the upper or the lower plenum a certain radial pressure gradient \(\Delta P_1\) can exist and the difference between the pressure in the lower plenum \(P_L\) and the pressure in the upper plenum \(P_U\) for an arbitrary channel \(i\) is given by:

\[
(P_D - P_U)_i = (P_D - P_U)_b + \Delta P_1
\]

where \(\Delta z_{U}\) is the vertical distance between the hot leg center line and the core outlet module, while \(\Delta z_D\) is the vertical distance between the core inlet module and the cold leg centerline.

For the bypass channel:

\[
(P_D - P_U)_b = (P_o - P_L)_b + q(\rho_u \cdot \Delta z_U + \rho_D \cdot \Delta z_D + \rho_u \cdot \Delta z_F)
\]
where $\Delta z_F$ is the height of water in the upper plenum over the hot
leq centerline.

From Eqs. (D.12), (D.14), (D.15) and (D.16) we obtain:

$$
\Delta W_1 B_1 + C_1 - \Delta P_1 = \Delta W_2 B_2 + C_2 - \Delta P_2 = \ldots
$$

$$
= \Delta W_1 B_1 + C_1 - \Delta P_1 = \ldots = \Delta W_I B_I + C_I - \Delta P_I
$$

$$
= \Delta W_b B_b + C_b + q \rho U \Delta z_F \quad (I \text{ equations}) \quad (D.17)
$$

where $I$ is the total number of channels in the core.

The overall mass balance equation is:

$$
\sum_{i=1}^{I} W_i + W_b + W_r = 0 \quad (D.18)
$$

where $W_r$ is the inlet flowrate to the system.

**Observation:** For our sign convention, the mass flowrate is
positive for upflow and negative for downflow.

Subtracting Eq. (D.18) written at two successive times $t$ and $t+\Delta t$
yields

$$
\sum_{i=1}^{I} \Delta W_i + \Delta W_b + \Delta W_r = 0 \quad (D.19)
$$

If the inlet mass flowrate is known, we have $I+1$ equations (Eqs.
(D.17) and (D.19)) with $I+1$ unknowns ($\Delta W_{i=1,2,\ldots,I}$ and $\Delta W_b$). The
equations can be solved as follows:

$$
\Delta W_m = \Delta W_1 \frac{B_1}{B_m} + \frac{(C_i - \Delta P_i) - (C_m - \Delta P_m)}{B_m} \\
(m = 1, 2, \ldots, I, \ m \neq i) \quad (D.20a)
$$

$$
\Delta W_b = \Delta W_1 \frac{B_1}{B_b} + \frac{(C_i - \Delta P_i) - (C_h + q \rho U \Delta z_F)}{B_h} \quad (D.20b)
$$

Introducing Eqs. (D.20) and (D.18) we finally obtain:
\[
\Delta W_{i} = \left[ \sum_{m=1}^{L} \frac{1}{B_{m}} \Delta I_{m} (C_{i} - \Delta P_{i}) (C_{m} - \Delta P_{m}) + \frac{1}{B_{b}} (C_{i} - \Delta P_{i}) (C_{b} + g_{0} U \Delta z_{f}) \right] \left( 1 + B_{i} \right) \frac{1}{B_{m}} \frac{1}{B_{b}} 
\]

\( i = 1, 2, ..., I \)  

\[
\Delta W_{b} = [\Delta W_{r} + \sum_{i=1}^{I} \Delta W_{i}] 
\]

(D.22)

Then

\[
W_{i}(t+\Delta t) = W_{i}(t) + \Delta W_{i}, \quad i=1, 2, ..., I 
\]

(D.23)

\[
W_{b}(t+\Delta t) = W_{b}(t) + \Delta W_{b} 
\]

(D.24)

D.2.1.2 Integration Methods

The terms \( I_{f, i} \) and \( I_{q, i} \) of the integral momentum equation (Eqs. (D.6 and D.8)) contain integrals of the form: \( \int_{a}^{b} f(x) \, dx \). These integrals are solved numerically for each equidistant mesh point axial region along each channel using either:

--- Simpson's rule (for even number of intervals in a region):

\[
\int_{a}^{b} f(x) \, dx = \frac{1}{3} \sum_{i=1}^{I} \Delta z_{i} \left[ f(x_{1}) + 4 \, S_{e} + 2 \, S_{o} + f(x_{N}) \right] 
\]

(D.25)

where

\( I = \) total number of (equidistant mesh points) regions.

\( \Delta z_{i} = \) mesh interval in a certain region \( i \).

\( S_{e} = \sum_{n=2}^{N-1} f(x_{n}), \quad n = \text{even} \)
\[ S_0 = \sum_{n=3}^{N-2} f(x_n) \quad , \quad n = \text{odd} \]

\( N \) = total number of mesh points in a region (odd number).

--- or the trapezoidal rule:

\[ \int_{a}^{b} f(x) \, dx = \sum_{i=1}^{N-1} \frac{\Delta z_1}{2} \left[ f(x_1) + 2S + f(x_N) \right] \quad (D.26) \]

where

\[ S = \sum_{n=2}^{N-1} f(x_n) \]

D.2.2 Fluid Energy Equation Formulations

A number of numerical methods for the solution of energy equation have been introduced in MICON. In this section they are only presented for application to the decoupled solution of the energy equation.

The energy transport equation when averaged over a cross-sectional area \( A \) yields for one-dimensional flow:

\[ \frac{\partial F}{\partial t} + v \frac{\partial F}{\partial z} = Q \quad (D.27) \]

where \( F \) can be either the liquid temperature, \( T \), or enthalpy, \( H \), and \( Q \) has the appropriate dimensions.

If an implicit coupling at the coolant - wall tube boundary is used, the characteristic velocity, \( v \), is given by:

\[ v = E_1 \cdot \frac{W}{\rho A} \quad (D.28) \]

and the net source term, \( Q \), is given by:

\[ Q = E_2 \cdot q_h^i - E_3 \quad (D.29) \]
where

\[ E_1 \equiv \left[ 1 + \frac{2\pi r}{A} \cdot \Delta t \cdot \frac{1}{\rho c_p} \frac{dq''}{dt} \right]^{-1} \]  \hspace{1cm} (D.30)

\[ E_2 \equiv \begin{cases} \frac{E_1}{(A \cdot \rho \cdot c_p)} & \text{, if } F \equiv T \\ \frac{E_1}{(A \cdot \rho)} & \text{, if } F \equiv H \end{cases} \]  \hspace{1cm} (D.31)

\[ E_3 \equiv \begin{cases} 2\pi r \cdot q'' \cdot \frac{E_1}{(A \cdot \rho \cdot c_p)} & \text{, if } F \equiv T \\ 2\pi r \cdot q'' \cdot \frac{E_1}{(A \cdot \rho)} & \text{, if } F \equiv H \end{cases} \]  \hspace{1cm} (D.32)

and \( r \) is the rod diameter.

If the indices \( i, k, \) and \( n \) denote the channel number, the axial level and the (old) time level, respectively, then:

\[ v = v^n_{i,k} \text{ (or } v^{n+1}_{i,k} \text{, or } v^{n+1/2}_{i,k} \text{) = characteristic velocity.} \]

\[ W = W^n_i \text{ (or } W^{n+1}_i \text{, or } W^{n+1/2}_i \text{) = mass flow rate in channel } i. \]

\[ A = A_i = \text{flow area of channel } i \text{ (constant along the channel).} \]

\[ \rho = \rho^n_{i,k} = \text{density.} \]

\[ q'_{h,i,k} = q^{n}_{h,i,k} \text{ (or } q^{n+1}_{h,i,k} \text{ or } q^{n+1/2}_{h,i,k} \text{) = heat rate per unit length from the heated element to the coolant.} \]

The numerical schemes used in MICON to solve Eq. (D.27) in each channel \( i \) are:

- donor-cell (upwind differencing),
- Turner,
- modified Turner (implicit donor cell),
- box.

**Donor Cell Scheme [R-1]**

In this scheme Eq. (D.27) is assumed to be written for an arbitrary control volume, centered at the axial mesh point \( k \) and be
delimited by \( k-1/2 \) and \( k + 1/2 \).

The time derivative is approximated by:

\[
\frac{\partial F}{\partial t} \approx \frac{F_{k}^{n+1} - F_{k}^{n}}{\Delta t}
\]  

(D.33)

The space derivative is approximated by:

\[
\frac{\partial F}{\partial z} = \frac{F_{k}^{n} - F_{m}^{n}}{z_{k} - z_{m}}
\]  

(D.34)

where \( z_{k-1} < z_{k} < z_{k+1} \), and

\[
k - 1, \text{ if } v > 0
\]

\[
m = k + 1, \text{ if } v < 0
\]

The velocity \( v \) and the heat source term can be evaluated at mesh point \( k \) or averaged over the segment \((z_{k}, z_{m})\), at the time level \( n \), \( n+1/2 \), or \( n+1 \) (in the decoupled solution of energy equation, the mass flow rate has still been computed from the momentum equation and the liquid density can be assumed constant in the energy equation).

This is a first order accuracy scheme, i.e. the truncation error is of order 0 (\( \Delta t, \Delta z \)). The scheme is conditionally stable, the stability condition being:

\[
\alpha \equiv \frac{v}{z_{k} - z_{m}} \frac{\Delta t}{\Delta z} \leq 1
\]  

(D.35)

If the velocity \( v \) and the mesh step size are spatially constant, the time step can be imposed such that \( \alpha = 1 \). In this case the scheme is exact, but unfortunately, for a multi-channel system with a different liquid velocity in each channel, this approach can not be applied. At the other limit (\( \alpha \to 0 \)) the scheme is also exact, but in the domain \( 0 < \alpha < 1 \) this scheme introduces numerical diffusion, the effect of
which must be verified.

Substituting Eqs. (D.33 - D.35) into Eq. (D.33) results in:

\[ F_{k}^{n+1} = (1 - \alpha) F_{k}^{n} + \alpha F_{m}^{n} + \Delta t \cdot Q \quad \text{(D.36)} \]

**Turner Scheme [T-3, K-4]**

In this scheme Eq. (D.27) is assumed to be written for an arbitrary control volume between two consecutive mesh points, \( k \) and \( m \) where \( m \) can be either \( k - 1 \) or \( k + 1 \).

The time derivative is approximated by:

\[ \frac{\partial F}{\partial t} = \frac{1}{2} \left[ \left( \frac{\partial F}{\partial t} \right)_{k} + \left( \frac{\partial F}{\partial t} \right)_{m} \right] = \frac{F_{k}^{n+1} - F_{k}^{n} + F_{m}^{n+1} - F_{m}^{n}}{2\Delta t} \quad \text{(D.37)} \]

The space derivative is approximated by:

\[ \frac{\partial F}{\partial z} = \frac{F_{k}^{n+1} - F_{m}^{n+1}}{z_{k} - z_{m}} \quad \text{(D.38)} \]

where again:

\[ k - 1 \quad \text{, if } v > 0 \]
\[ m = k + 1 \quad \text{, if } v < 0 \]

The velocity \( v \) and the heat source term \( Q \) are evaluated at the new time level, \( n + 1 \), and are axially averaged over the segment \((z_{k}, z_{m})\).

The truncation error for this scheme is of order \( O(\Delta t, \Delta z^{2}) \) and it is an unconditionally stable scheme.

Substituting Eqs. (D.37) and (D.38) into (D.27), using the definition of \( \alpha \) from Eq. (D.35) yields:

\[ F_{k}^{n+1} = \frac{1}{1 + 2\alpha} \left[ F_{k}^{n} + F_{m}^{n} - (1 - 2\alpha) F_{m}^{n+1} \right] + \frac{2 \cdot \Delta t \cdot Q}{1 + 2\alpha} \quad \text{(D.39)} \]
This scheme is also diffusive. It is interesting to note that for $\alpha = 0.5$ the "donor cell" scheme and the "Turner" scheme are identical.

**Modified Turner Scheme** [T-2, K-4]

This is an implicit donor cell scheme. The time derivative is given by Eq. (D.33). The space derivative is approximated by Eq. (D.34), but with $F_k$ and $F_m$ evaluated now at the time level $n+1$, instead of $n$.

The truncation error is of order $O(\Delta t, \Delta z)$ and the scheme is unconditionally stable.

Substituting the finite difference expressions into Eq. (D.27) results in:

$$F_k^{n+1} = \frac{1}{1+\alpha} \left[ F_k^n + \alpha F_m^{n+1} \right] + \frac{\Delta t \cdot Q}{1+\alpha}$$  \hspace{1cm} (D.40)

This scheme is also diffusive.

**Box Scheme** [A-1]

This is a center-time, center-space scheme in the $t-z$ plane. The time derivative is given by Eq. (D.37), while the space derivative is approximated by:

$$\frac{\partial F}{\partial z} = \frac{1}{2} \left( \frac{F_k^{n+1} + F_k^n}{z_k - z_m} - \frac{F_m^{n+1} + F_m^n}{z_k - z_m} \right)$$  \hspace{1cm} (D.41)

The truncation error is of order $O(\Delta t^2, \Delta z^2)$, the scheme being unconditionally stable.

Substituting the finite difference expressions in Eq. (D.27) yields
\[ F_k^{n+1} = \frac{1}{1+\alpha} \left[ (1-\alpha)(F_k^n - F_m^{n+1}) + (1+\alpha) F_m^n \right] + \frac{2 \cdot \Delta t \cdot Q}{1+\alpha} \quad (D.42) \]

If \( \alpha = 1 \), the "box" scheme is identical with the "donor cell" scheme, both of them being exact. If \( \alpha \neq 1 \), the numerical diffusion is generally not negligible, and unrealistic results may be obtained.

D.2.3 Ambient (Environmental) Temperature Calculation

The temperature of the gas in the environmental atmosphere associated with each channel and each axial level can be updated using a simple non-convective energy equation:

\[ \frac{dT}{dt} = \frac{q_{\text{right}} \cdot Pe \cdot L}{V_e \cdot \rho \cdot c} \quad (D.43) \]

where

\[ \frac{dT}{dt} = \frac{T_k^{n+1} - T_k^n}{\Delta t} \]

\( q_{\text{right}} \) = heat flux at the outer wall (right hand side) of the solid structure

\( Pe \) = outer wall perimeter.

\( L \) = channel's length.

\( V_e \) = volume of air in the environmental atmosphere, attributed to one channel.

\( T, \rho, c \) = air temperature, density, and specific heat, respectively.

D.2.4 Plena Enthalpy Calculation

For estimating the coolant enthalpies in the plena, perfect mixing is assumed. From an overall energy balance for each (adiabatic) plenum,

\[ \rho V \frac{dH}{dt} = \sum_{\text{in}} (\text{WH}) - \sum_{\text{out}} (\text{WH}) \quad (D.44) \]

where \( V \) is the volume of liquid in the plenum, which is discretized to
\[
\rho V \frac{H_{n+1} - H_n}{\Delta t} = \sum_{\text{in}} (WH)^{n+1} - \sum_{\text{out}} (WH)^{n+1}
\]  \hspace{1cm} (D.45)

D.3 Convective Heat Transfer Constitutive Equations

The convective heat transfer coefficient at the inner wall (left hand side) of the solid structure is calculated using a correlation of the general form:

\[
\text{Nu} = a_1 \text{Re}^{a_2} \text{Gr}^{a_3} \text{Pr}^{a_4}
\]  \hspace{1cm} (D.46)

where

\[ a_1, a_2, a_3, \text{ and } a_4 \text{ are input constants} \]

\[ \text{Nu} \equiv \frac{hD_e}{k} \quad \text{(Nusselt number)} \]

\[ \text{Re} \equiv \frac{|W| \cdot D_e}{A u} \quad \text{(Reynolds number)} \]

\[ \text{Gr} \equiv g D e^3 \frac{\rho^2}{\mu^2} \beta \Delta T_w \quad \text{(Grashof number)} \]

\[ \text{Pr} \equiv \frac{c_p \mu}{k} \quad \text{(Prandtl number)} \]

\[ \beta = -\frac{1}{\rho} \frac{\partial \rho}{\partial T} \bigg|_p = -\frac{c_p}{\rho} \frac{\partial H}{\partial T} \bigg|_p \]

\[ \Delta T_w = |T_w - T| \]

\( h \) = convective heat transfer coefficient

\( De \) = hydraulic diameter

\( A \) = flow area

\( W \) = flow rate
\( g = \text{acceleration due to gravity} \)

\( T_w = \text{wall temperature} \)

\( T, H, P, \rho, c_p, \beta, k, \mu = \text{fluid temperature, specific enthalpy, pressure, density, specific heat, coefficient of thermal expansion, thermal conductivity and dynamic viscosity, respectively.} \)

Equation (D.46) is applied for two sub-cases:

- If \( a_2 \neq 0 \), the heat transfer coefficient, \( h \), is calculated from: \( \text{Nu} = a_1 \text{Re}^{a_2} \text{Pr}^{a_4} \). The input coefficients \( a_1, a_2, \) and \( a_4 \) may be different in the laminar flow regime (\( \text{Re} \leq \text{Re}_1 \)) and in the turbulent flow regime (\( \text{Re} \geq \text{Re}_2 \)), where \( \text{Re}_1 \) and \( \text{Re}_2 \) are also input constants. For the transition regime (\( \text{Re}_1 < \text{Re} < \text{Re}_2 \)), the Nusselt number is assumed to be linearly dependent on the Reynolds number connecting the Nu values at \( \text{Re}_1 \) and \( \text{Re}_2 \).

- If \( a_2 = 0 \), the heat transfer coefficient, \( h \), is calculated from: \( \text{Nu} = a_1 \text{Gr}^{a_3} \text{Pr}^{a_4} \). The input coefficients \( a_1, a_3, \) and \( a_4 \) may differ if \( \text{GrPr} \leq (\text{GrPr})_1 \), \( (\text{GrPr})_1 < \text{GrPr} < (\text{GrPr})_2 \), or \( \text{GrPr} \geq (\text{GrPr})_2 \), where \( (\text{GrPr})_1 \) and \( (\text{GrPr})_2 \) are also input values.

D.4 The Equation of State and Liquid Thermophysical Properties

D.4.1 Introduction

In the MICON code, the energy equation can be solved for the enthalpy \( (H) \), or temperature \( (T) \) as the primary dependent variable, and thus, the caloric equation of state must be in the form of \( T = T(H,P) \), or \( H = H(T,P) \) if the energy equation is solved for the enthalpy, or temperature, respectively. The code can be used to analyze the behavior of systems with one of the following liquids: Sodium, \( H_2O \), or \( D_2O \). For certain liquids (\( H_2O \) and \( D_2O \)), the polynomial fits found in the literature are used in the form \( T = T(H,P) \), whereas others (sodium) are
of the form \( H = H(T, P) \). If the polynomial fit for a certain liquid property is not explicitly expressed, the Newton-Raphson iteration method is applied. For example, in a water system, if the energy equation has been solved for \( T \) as a primary dependent variable, the caloric equation of state originally written as:

\[
T = \sum_{i=0}^{I} q_i(P) \cdot H^i
\]  

(D.48)

is recast as:

\[
f(H) \equiv \sum_{i=0}^{I} q_i(P) \cdot H^i - T
\]  

(D.49)

Using the Newton-Raphson method:

\[
H_m = H_{m-1} - f(H_{m-1}) \cdot \left\{ \frac{\partial f}{\partial H} \right\}_T^{-1}_{H_{m-1}} =
\]

\[
= H_{m-1} - \sum_{i=0}^{I} q_i(P) \cdot H_{m-1}^i - T \cdot \left\{ \sum_{i=0}^{I} i \cdot q_i(P) \cdot H_{m-1}^{i-1} \right\}^{-1}
\]  

(D.50)

where "m" is the iteration count.

The convergence occurs if:

\[
\left| f(H_m) \right| \leq \varepsilon_T,
\]  

(D.51)

where "\( \varepsilon \)" is the accepted relative error of convergence. The relatively mild nonlinearity of the direct fit results in only 2 or 3 iterations being needed to attain a more than adequate accuracy (\( \varepsilon = 10^{-4} \)).
After the temperature (or enthalpy) has been computed from the caloric equation of state all other thermo-physical properties are obtained from their corresponding fits.

The next section lists the liquid thermodynamics and transport properties and the associated polynomial fits. All properties are in SI units.

D.4.2 Polynomial Fits

**Light Water Properties [K-4, G-1, A-1]**

\[
T = \sum_{i=0}^{3} (a_i \cdot P + b_i) \cdot H^i
\]  
(D.52)

where

\[
\begin{align*}
a_0 &= -1.59542 \cdot 10^{-7} & b_0 &= +2.72911 \cdot 10^{+2} \\
a_1 &= -5.19625 \cdot 10^{-13} & b_1 &= +2.39491 \cdot 10^{-4} \\
a_2 &= +1.20635 \cdot 10^{-18} & b_2 &= +5.96601 \cdot 10^{-12} \\
a_3 &= -5.60257 \cdot 10^{-25} & b_3 &= -1.31469 \cdot 10^{-17}
\end{align*}
\]

\[
\rho = \begin{cases} 
    f_1 + f_2 \cdot H^2 + f_3 H^4, & \text{if } H \leq 6.513 \cdot 10^5 \\
    f_4 + f_5 (H-f_6)^{-1}, & \text{if } H > 6.513 \cdot 10^5
\end{cases}
\]  
(D.53)

where
\[ f_1 = +9.99650 \times 10^{+2} \quad +4.97370 \times 10^{-7} \cdot p \]
\[ f_2 = -2.58470 \times 10^{-10} \quad +6.17670 \times 9^{-19} \cdot p \]
\[ f_3 = +1.26960 \times 10^{-22} \quad -4.92230 \times 10^{-31} \cdot p \]
\[ f_4 = +1.48864 \times 10^{+3} \quad +1.33890 \times 10^{-6} \cdot p \]
\[ f_5 = +1.46950 \times 10^{+9} \quad +8.85736 \times 10^{0} \cdot p \]
\[ f_6 = +3.20372 \times 10^{+6} \quad +1.20483 \times 10^{-2} \cdot p \]

**Specific heat at constant pressure \((J/kg \cdot K)\)**

\[ c_p \equiv \left( \frac{\partial H}{\partial T} \right)_p = \left( \frac{\partial T}{\partial H} \right)_p = \left[ \sum_{i=1}^{3} (a_i \cdot p + b_i) \cdot H^{i-1} \right]^{-1} \quad (D.54) \]

The specific heat at constant pressure has been directly derived from Eq. (D.52).

**Thermal expansion coefficient \((^\circ K^{-1})\)**

\[ \beta = -c_p \cdot p^{-1} \cdot R_H \quad (D.55) \]

where

\[ R_H \equiv \left( \frac{\partial \rho}{\partial H} \right)_p = \begin{cases} 2f_2 \cdot H + 4f_3 \cdot H^3, & \text{if } H \leq 6.513 \times 10^5 \\ -f_5 \cdot (H-f_6)^{-2}, & \text{if } H > 6.513 \times 10^5 \end{cases} \quad (D.56) \]

\( R_H \) has been directly derived from Eq. (D.53).

**Thermal conductivity \((W/m \cdot {^\circ K})\)**

\[ k = \sum_{i=0}^{3} a_i \cdot X_i \quad (D.57) \]

where

\[ X = H/5.815 \times 10^{+5} \]
and

\[ a_0 = +5.73740 \times 10^{-1} \quad a_2 = -1.45470 \times 10^{-1} \]

\[ a_1 = +2.53610 \times 10^{-1} \quad a_3 = +1.38750 \times 10^{-2} \]

**Dynamic viscosity (Pa-s)**

\[
\mu = \sum_{i=0}^{4} \left[ a_i \cdot X^i - b_i \cdot Y^i \cdot (P-P_1) \right] , \quad \text{if} \quad H < H_1
\]

\[
\mu = \sum_{i=0}^{3} \left[ c_i \cdot H^i + d_i \cdot H^i \cdot (P-P_1) \right] , \quad \text{if} \quad H_1 < H < H_2 \quad \text{(D.58)}
\]

\[
\mu = \sum_{i=0}^{4} \left[ f_i \cdot Z^i \right] , \quad \text{if} \quad H > H_2
\]

where

\[ X = q_0 \cdot (H-q_1) \]

\[ Y = q_2 \cdot (H-q_3) \]

\[ Z = q_4 \cdot (H-q_5) \]

and

\[ a_0 = +1.29950 \times 10^{-3} \quad b_0 = -6.59590 \times 10^{-12} \]

\[ a_1 = -9.26400 \times 10^{-4} \quad b_1 = +6.76300 \times 10^{-12} \]

\[ a_2 = +3.81050 \times 10^{-4} \quad b_2 = -2.88830 \times 10^{-12} \]

\[ a_3 = -8.21940 \times 10^{-5} \quad b_3 = +4.45250 \times 10^{-13} \]

\[ a_4 = +7.02240 \times 10^{-6} \]

\[ c_0 = +1.45260 \times 10^{-3} \quad d_0 = -3.80640 \times 10^{-11} \]

\[ c_1 = -6.98810 \times 10^{-9} \quad d_1 = +3.92850 \times 10^{-16} \]

\[ c_2 = +1.52100 \times 10^{-14} \quad d_2 = -1.25860 \times 10^{-21} \]
\[ c_3 = -1.23030 \times 10^{-20} \quad d_3 = +1.28600 \times 10^{-27} \]

\[ f_0 = +3.02600 \times 10^{-4} \quad g_0 = +8.58130 \times 10^{-6} \]

\[ f_1 = -1.83660 \times 10^{-4} \quad g_1 = +4.26590 \times 10^{-4} \]

\[ f_2 = +7.56710 \times 10^{-5} \quad g_2 = +6.48450 \times 10^{-6} \]

\[ f_3 = -1.64790 \times 10^{-5} \quad g_3 = +5.53590 \times 10^{-4} \]

\[ f_4 = +1.41650 \times 10^{-6} \quad g_4 = +3.89210 \times 10^{-6} \]

\[ g_5 = +4.01470 \times 10^{-5} \]

\[ P_1 = +6.89460 \times 10^{-5} \]

\[ H_1 = +2.76000 \times 10^{-5} \]

\[ H_2 = +3.94000 \times 10^{-5} \]

**Heavy water properties [K-5]**

The relationships of \( \text{D}_2\text{O} \) properties are derived based on the corresponding \( \text{H}_2\text{O} \) properties multiplied by adequate coefficients or functions.

Let us define:

\[ (F)_R \equiv (F)_{\text{D}_2\text{O}} / (F)_{\text{H}_2\text{O}} \quad \text{(D.59)} \]

where \( F \) is any property.

In the following, the coefficients or the functions that must multiply the \( \text{H}_2\text{O} \) properties to obtain the corresponding \( \text{D}_2\text{O} \) properties are presented.

**Temperature**

\[ (T)_R = 1.0115 \quad \text{(D.60)} \]
**Density**

<table>
<thead>
<tr>
<th>H[MJ/kg]</th>
<th>0</th>
<th>0.419</th>
<th>0.628</th>
<th>0.837</th>
<th>1.047</th>
<th>1.256</th>
<th>1.360</th>
<th>1.465</th>
</tr>
</thead>
<tbody>
<tr>
<td>(ρ)_R [-]</td>
<td>1.1070</td>
<td>1.1065</td>
<td>1.1035</td>
<td>1.1000</td>
<td>1.0950</td>
<td>1.0865</td>
<td>1.0790</td>
<td>1.0650</td>
</tr>
</tbody>
</table>

(D.61)

**Specific heat at constant pressure**

(c_p)_R = 1.0  

(D.62)

**Thermal expansion coefficient**

The thermal expansion coefficient of D_2O is obtained using Eq. (D.55). The values of (c_p)_D_2O and (ρ)_D_2O are obtained by from Eqs. (D.54), (D.62) and (D.53), (D.61), respectively. The value of (R_H)_D_2O is obtained by using Eq. (D.56) and the following correction function:

<table>
<thead>
<tr>
<th>H[MJ/kg]</th>
<th>&lt;0.42</th>
<th>0.54</th>
<th>&gt;1.50</th>
</tr>
</thead>
<tbody>
<tr>
<td>(R_H)_R [-]</td>
<td>1.12</td>
<td>1.18</td>
<td>1.20</td>
</tr>
</tbody>
</table>

(D.63)

**Thermal conductivity**

<table>
<thead>
<tr>
<th>T[°K]</th>
<th>303</th>
<th>333</th>
<th>373</th>
<th>473</th>
<th>553</th>
<th>623</th>
</tr>
</thead>
<tbody>
<tr>
<td>(k)_R [-]</td>
<td>0.969</td>
<td>0.952</td>
<td>0.933</td>
<td>0.905</td>
<td>0.865</td>
<td>0.853</td>
</tr>
</tbody>
</table>

(D.64)

**Dynamic viscosity**

<table>
<thead>
<tr>
<th>T[°K]</th>
<th>303</th>
<th>333</th>
<th>373</th>
<th>473</th>
<th>553</th>
<th>623</th>
</tr>
</thead>
<tbody>
<tr>
<td>(μ)_R [-]</td>
<td>1.29</td>
<td>1.18</td>
<td>1.16</td>
<td>1.12</td>
<td>1.11</td>
<td>1.10</td>
</tr>
</tbody>
</table>

(D.65)

**Sodium properties [S-5]**
Enthalpy (J/kg)

\[ H = \sum_{i=0}^{3} a_i \cdot T^i \]  \hspace{1cm} (D.66)

where

\[ a_0 = -6.75080 \cdot 10^{-4} \]
\[ a_2 = -4.16720 \cdot 10^{-1} \]
\[ a_1 = +1.63010 \cdot 10^{+3} \]
\[ a_3 = +1.54270 \cdot 10^{-4} \]

Density (kg/m³)

\[ \rho = \sum_{i=0}^{2} f_i \cdot T^i + c^{-2} \cdot (P - P_1) \]  \hspace{1cm} (D.67)

where

\[ f_0 = +1.00420 \cdot 10^{+3} \]
\[ c^{-2} = +2.0 \cdot 10^{-7} \]
\[ f_1 = -2.13900 \cdot 10^{-1} \]
\[ p_1 = +1.5 \cdot 10^{+5} \]
\[ f_2 = -1.10460 \cdot 10^{-5} \]

Specific heat at constant pressure (J/kg °K)

\[ c_p \equiv \left( \frac{\partial H}{\partial T} \right)_p = \sum_{i=1}^{3} i \cdot a_i \cdot T^{i-1} \]  \hspace{1cm} (D.68)

The specific heat at constant pressure has been directly derived from Eq. (D.66).

Thermal expansion coefficient (°K⁻¹)

\[ \beta = -p^{-1} \cdot R_T \]  \hspace{1cm} (D.69)

where

\[ R_T \equiv \left( \frac{\partial p}{\partial T} \right)_p = \sum_{i=1}^{2} i \cdot f_i \cdot T^{i-1} \]  \hspace{1cm} (D.70)

\[ R_T \] has been directly derived from Eq. (D.67).
Thermal conductivity (W/m·°K)
\[ k = \sum_{i=0}^{3} a_i \cdot T^i \]  \hspace{1cm} (D.71)

where
\[ a_0 = +1.10450 \times 10^{+2} \quad a_2 = +1.54300 \times 10^{-5} \]
\[ a_1 = -6.51120 \times 10^{-2} \quad a_3 = -2.46170 \times 10^{-9} \]

Dynamic viscosity (Pa·s)
\[ \mu = \sum_{i=1}^{3} a_i / i^i \]  \hspace{1cm} (D.72)

where
\[ a_0 = +3.65220 \times 10^{-5} \quad a_2 = -4.56880 \times 10^{+1} \]
\[ a_1 = +1.66260 \times 10^{-1} \quad a_3 = +2.87330 \times 10^{+4} \]

The thermo-physical properties of air have been introduced in MICON as simple constants. They will be modeled in a future version.

The properties of solid materials (\( \rho, c, k \)) have also been supplied for UO\(_2\) and Zr only. The properties of other materials must be introduced by the user.

D.5 MICON - Code Description

The MICON-1 (Mixed Convection Channels - version 1) code was developed for the analysis of the Shutdown Core Coolability Model Test results. The code, which has about 4000 statements, contains a MAIN module and 30 other subroutines or functions. It has been written in FORTRAN-IV language, being compiled on the MIT (Multics) Honeywell machine. The block diagram of MICON-1 code is shown in Fig. D.1 and the subroutine and function modules follow.
Figure D.1 Block diagram of MICON.
This is the most complete text of the thesis available. The following page(s) were not included in the copy of the thesis deposited in the Institute Archives by the author:

346
21. TRAPEZ : Solves numerically integral \([F(x)\cdot dx]\) between integration limits \(a\) and \(b\) using the trapezoidal rule.

22. ENTEL : Calculates the enthalpy of fluid as a function of pressure and temperature.

23. TEMPER : Calculates the temperature of fluid as function of enthalpy.

24. DENS : Calculates the density of fluid as function of pressure and enthalpy (or temperature).

25. SPHEAT : Calculates the specific heat of fluid as function of pressure and enthalpy (or temperature).

26. VISC : Calculates the dynamic viscosity of fluid as function of pressure and enthalpy (or temperature).

27. THCON : Calculates the thermal conductivity of fluid as function of enthalpy (or temperature).

28. THEXP : Calculates the thermal expansion coefficient of fluid as function of pressure, enthalpy (or temperature), density and specific heat.

29. ROC : Calculates the volumetric heat capacity of solid materials as function of temperature.

30. RK : Calculates the thermal conductivity of solid materials as function of temperature.
D.6 Input Description

This section outlines the input deck for the MICON code. Included are the format statements and input lists associated with all of the "read" statements contained in the code. The code uses formatted input for columns 1-72, and therefore card numbers (or any other kind of identifiers) can be located in columns 73-80. A suggested card numbering scheme is also included.

The heat conduction model referred to in the input description for the code is for future versions. The user should therefore set lcondu<0 and lhtr<0.

This version of MICON has been implemented on the Multics computer system at M.I.T. The input file is assumed to be file01 and the output file is file02. These can be altered to other filenames by adding the appropriate "open" statements in the source code.
Card 0:  format (7i10)  ichan, jcanmx, kaxmax, nregmx, ntimeh  
ichan = number of channels  
jcanmx = maximum number of heat conduction nodes  
kaxmax = maximum number of axial mesh points  
ntimeh = maximum number of pairs of points of the form (value, time) in the method of characteristics  
formulation of the energy equation  

Cards 1.1, 1.2, 1.3, 1.4:  format (14i5)  lstedy, lplenu, plend, lfluid,  
lmomen, lenerg, lcondu, lcond, lconr, lhtr, lhtr1, lhtrr, liliq,  
lmomfl, lform, letopf, litolen, leamb, ldtecon, lrest, lunits.  
lpas, kpas, jpas, itermx, itptr1, itnr2  

-1  only transient calculation  
  lstedy =  0  only steady calculation  
     1  steady and transient calculation  
     >0  calculate energy equation for upper plenum  
   lplenu =  <0  use input data for properties of upper plenum fluid  
     <1  no fluid calculation  
      1  with fluid calculation  
     >1  dummy  
   lmomen =  dummy  
     <1  no fluid energy equation calculation  
        1  box scheme  
        2  Turner scheme  
        3  Modified Turner scheme  
        4  Donor cell with W=W^n  
   lenerg =  5  Donor cell with W=(1/2)(W^{n+1}+W^n)  
        6  Donor cell with W=W^{n+1}  
        7  M.O.C., trapezoidal integration, with  
            non-linear effects  
        8  M.O.C., Simpson's rule integration, with  
            non-linear effects  
        9  M.O.C., trapezoidal integration, no non-linear  
            effects  
       10  M.O.C., Simpson's rule integration, no non-  
           linear effects  
   lcondu =  <0  no conduction calculation  
                   >0  with conduction calculation  
     <0  T^{n+1}_w, T^n_{fluid}  at left hand side of structure  
     lcond =  0  T^n_w, T^n_{fluid}  at left hand side of structure  
           >0  T^{n+1}_w, T^{n+1}_{fluid}  at left hand side of structure
<0  \( T_{n+1}^w, T_n^w \) fluid at right hand side of structure

\[ lconr = 0 \]
\( T_n^w, T_n^w \) fluid at right hand side of structure

>0  \( T_{n+1}^w, T_n^w \) fluid at right hand side of structure

if \( lconl > 0 \), then \( lconr < 0 \)

\[ lhtr = \begin{cases} 
\leq 0 & q''_{left} \text{ input, } q''_{right} \text{ input} \\
>0 & q''_{left} \text{ calculated by } q''_{left} = h_{left} (T_{fl} - T_{w, in}) \\
& q''_{right} \text{ calculated by } q''_{right} = h_{right} (T_{w, ext} - T_{amb}) 
\end{cases} \]

\[ lhtrl = \begin{cases} 
\leq 0 & h_{left} \text{ input} \\
>0 & h_{left} \text{ calculated} 
\end{cases} \]

\[ lhtrr = \begin{cases} 
\leq 0 & h_{right} \text{ input} \\
>0 & h_{right} \text{ calculated} 
\end{cases} \]

1  \( \text{H}_2\text{O} \)

2  \( \text{Na} \)

3  \( \text{Air} \)

4  \( \text{D}_2\text{O} \)

\[ l\text{mom} l = \begin{cases} 
\leq 0 & \text{no momentum flux term in momentum equation} \\
>0 & \text{with momentum flux term in momentum equation} 
\end{cases} \]

\[ l\text{form} = \begin{cases} 
\leq 0 & \text{form losses neglected in momentum equation} \\
>0 & \text{form losses included in momentum equation} 
\end{cases} \]

\[ \leq -2 \] solve energy equation for \( H; (H_{i, k})_{\text{initial input}} \)

\[ letopt = \begin{cases} 
-1 & \text{solve energy equation for } T; (T_{i, k})_{\text{initial input}} \text{ for } 0 \text{ or } 1 \\
0 & \text{solve energy equation for } T; (T_{i, k})_{\text{initial input}} = T_{\text{init}} \\
\geq 2 & \text{solve energy equation for } H; (H_{i, k})_{\text{initial input}} = H_{\text{init}} 
\end{cases} \]

\[ l\text{tolen} = \begin{cases} 
\leq 0 & \text{no limitation on variation of fluid properties per time step} \\
>0 & \text{with limitation on variation of fluid properties per time step} 
\end{cases} \]

\[ l\text{eamb} = \begin{cases} 
\leq 0 & \text{no updating of ambient temperature} \\
>0 & \text{with updating of ambient temperature} 
\end{cases} \]

\[ l\text{dtcon} = \begin{cases} 
\leq 0 & \text{no limitation of variation of fluid properties per steady-state iteration} \\
>0 & \text{with limitation on variation of fluid properties per steady-state iteration} 
\end{cases} \]
lrest = dummy
lunits = dummy
ipas = print data for channels i=1,lchan,ipas
kpas = print data for axial levels k=1,kk(i),kpas
jpas = print data for radial nodes j=1,jmax(1,k),jpas
itermx = maximum number of steady-state iterations
itpr1 = print steady-state iterations between iter=1,itermx
itpr2 = print steady-state iterations between
iter=itpr2,itermx

Cards 2.1.1: format (7d10.3) c1lam(i), c2lam(i), c1tur(i), c2tur(i),
reli1

f0 = c1/Re^2
c1lam(i)= c1 for laminar regime
c2lam(i)= c2 for laminar regime
c1tur(i)= c1 for turbulent regime
c2tur(i)= c2 for turbulent regime
reli1= laminar regime for Re<relim;turbulent regime for
Re>>relim

Card 2.2.1: format (7d10.3) a11, a21, a31, a41, reli1, grpr1

Nu = a1 Re^a2 Gr^a3 Pr^a4 for Re>>0; Nu=a1 Gr^a3 Pr^a4 for Re=0
a11 = a1 for laminar regime
a21 = a2 for laminar regime
a31 = a3 for laminar regime
a41 = a4 for laminar regime
reli1 = laminar regime for Re<relim1 and Re>>0
grpr1 = laminar regime for Gr•Pr<grpr1 and Re=0

Card 2.2.2: format (7d10.3) a12, a22, a32, a42, reli2, grpr2

Nu = a1 Re^a2 Gr^a3 Pr^a4 for Re>>0; Nu=a1 Gr^a3 Pr^a4 for Re=0
a12 = a1 for turbulent regime
a22 = a2 for turbulent regime
a32 = a3 for turbulent regime
a42 = a4 for turbulent regime
reli2 = turbulent regime for Re<relim2 and Re>>0
grpr2 = turbulent regime for Gr•Pr<grpr2 and Re=0

natural convection regime for Gr•Pr>grpr2 and Re=0
Card 2.2.3: format (7d10.3) a13, a23, a33, a43, rnurig

\[ \text{Nu} = a_1 \text{Re}^{a_2} \text{Gr}^{a_3} \text{Pr}^{a_4} \text{ for } |\text{Re}| > 0; \text{Nu} = a_1 \text{Gr}^{a_3} \text{Pr}^{a_4} \text{ for } \text{Re} = 0 \]

a13 = a1 for natural convection regime
a23 = a2 for natural convection regime
a33 = a3 for natural convection regime
a43 = a4 for natural convection regime
rnurig = Nu for outer wall/ambient boundary

Cards 3.1.1: format (7d10.3) kcon(i), kexp(i), kloc(i)

\[ \text{k_{con}(i)} = \text{contraction pressure loss factor for channel } i \]
\[ \text{k_{exp}(i)} = \text{expansion pressure loss factor for channel } i \]
\[ \text{k_{loc}(i)} = \text{local pressure loss factor for channel } i \]

Card 3.2: format (1415) nfric(i)

nfric(i) = number of \((\text{Gr}_q/\text{Re}, (f/f_0)(1+F))\) pairs for channel i

Cards 3.3.1: format (6d10.3) ftab(i,k), k=1, 2*nfnc(i)

\[ \text{ftab}(i,k) = \frac{\text{Gr}_q}{\text{Re}} \text{ for } (k/2+1/2)\text{th pair for channel } i \text{ for odd } k \]
\[ = (f/f_0)(1+F) \text{ for } (k/2)\text{th pair for channel } i \text{ for even } k \]

Card 4: format (7d10.3) tolw, tolh, tolh, tolc, tolp, tolx

tolw = fluidrate relative change tolerance per time step
tohl = fluid enthalphy relative change tolerance per time step
tolt = fluid temperature relative change tolerance per time step
tolc = structure relative change tolerance per time step
tolp = plena temperature relative change tolerance per time step
tolx = property relative change tolerance per steady-state iteration

Cards 5.1: format (6d10.3,110) l(1), rl(1), re(1), rheat(1), a(1), de(1), nreg(1)

format (d10.3,6i10) (lreg(1,n),kax(1,n),jmax(1,n), mater(1,n)), n=1,nreg(1)
l(i) = length of channel i (m)
ri(i) = inner radius of structure wall for channel i (m)
re(i) = outer radius of structure wall for channel i (m)
rheat(i) = heater (rod) radius for channel i (m)
a(i) = flow area for channel i (m²)
def(i) = hydraulic diameter for channel i (m)
nreg(i) = number of fluid axial regions for channel i
lreg(i,n) = length of region n for channel i (m)
kax(i,n) = number of axial mesh points for region n for channel i
jmax(i,n) = number of radial mesh points for region n for channel i
mater(i,n) = structure material for region n for channel i

Card 6: format (7d10.3) volu, vold, dzu, tamb, cpext, volamb, tsolid

volu = upper plenum volume (m³)
vold = lower plenum volume (m³)
dzu = height of upper plenum outlet above core channel outlet (m)
tamb = ambient temperature (K)
cpext = ambient fluid specific heat (J/kg/K)
volamb = ambient fluid volume per channel (m³)
tsolid = structure temperature (K)

Card 7: format (2d10.3, 5i/o) wretin, tvalvr, lvalvr

wretin = inlet flowrate (kg/s)
tvalvr = time that valve for bypass channel (i=nchan) changes position
lvalvr = 0 initially bypass valve closed
1 initially bypass valve opened

Card 8: format (7d10.3) pinit, [tinit if |letopt|=1, einit if |letopt|=2], cpinit, tref, eref, hconvl, hconvr

pinit = initial pressure (Pa)
tinit = initial fluid temperature (K)
einit = initial fluid enthalpy (J/kg)
cpinit = initial fluid specific heat (J/kg/K)
tref = reference temperature (K)
eref = reference enthalpy (J/kg)
hconvl = initial left hand side heat transfer coefficient (W/m²/K)
hconvr = initial right hand side heat transfer coefficient (W/m²/K)
Cards 9.1: format (7d10.3) \{tem(i,k) if letopt=-1, ent(i,k) if letopt=-2\}, n=1, klast(i)
(omit if letopt>0)

\( \text{tem(i,k)} \) = fluid temperature at axial position \( k \) for channel \( i \) (K)

\( \text{ent(i,k)} \) = fluid enthalpy at axial position \( k \) for channel \( i \) (J/kg)

\( \text{klast(i)} \) = number axial mesh points for channel \( i \)

Cards 10.1: format (6d10.3, l10) W(i), delp(i), deltu(i), deltd(i),
gcost(i), timev(i), lvalv(i)
\text{format (d10.3,6110) q1heat(i,n), linteg(i,n), n=1,nreg(i)}
\( w(i) \) = flowrate for channel \( i \) (kg/s)
\( delp(i) \) = radial pressure gradient in plena for channel \( i \) (Pa)
\( deltu(i) \) = radial temperature gradient in upper plenum for channel \( i \) (K)
\( deltd(i) \) = radial temperature gradient in lower plenum for channel \( i \) (K)
\( gcost(i) \) = gravity*\( \cos \theta \) for channel \( i \) (m/s\(^2\))
\( timev(i) \) = time valve for channel \( i \) changes position (s)
\( lvalv(i) = 0 \) initially valve for channel \( i \) closed
\text{= 1 initially valve for channel \( i \) open}
\( q1heat(i,n) \) = linear heat rate for region \( n \) for channel \( i \) (W/m)

1 trapezoidal rule Integration method for
2 Simpson rule region \( n \) in channel \( i \)
3 dummy

Card 11: format (14i5) npairf, npairp

\( \text{npairf} \) = number of (flow fraction, time) pairs
\( \text{npairp} \) = number of (power fraction, time) pairs
flow fraction = inlet flow fraction
power fraction = heater power fraction (assumed equal for all channels)

Card 11.1: format (6d10.3) flowfh(i), i=1,2*npairf

\( \text{flowfh(i)} \) = flow fraction for \((i/2+1/2)\)th pair for odd \( i \)
time for \((i/2)\)th pair for even \( i \)
Card 11.2: format (6d10.3) powfrh(i), i=1,2*npairh

\[ \text{powfrh}(i) = \text{power fraction for (i/2+1/2)th pair for odd } i \]
\[ = \text{time for (i/2)th pair for even } i \]

Card 12: format (6d10.3) {tupper,tdown if |letopt|=1; hupper,hdown if |letopt|=2}

\[ \text{tupper} = \text{upper plenum temperature (K)} \]
\[ \text{tdown} = \text{lower plenum temperature (K)} \]
\[ \text{hupper} = \text{upper plenum enthalpy (J/kg)} \]
\[ \text{hdown} = \text{lower plenum enthalpy (J/kg)} \]

Card 13: format (14i5) npairku, npaird

\[ \text{npairu} = \text{number of (upper plenum property fraction, time) pairs} \]
\[ \text{npaird} = \text{number of (lower plenum property fraction, time) pairs} \]
\[ \text{plenum property} = \text{temperature if } |\text{letopt}|=1 \]
\[ = \text{enthalpy if } |\text{letopt}|=2 \]

Card 13.1: format (6d10.3) etupfh(i), i=1,2*npairu

\[ \text{etupfh}(i) = \text{upper plenum property fraction for (i/2+1/2)th pair for odd } i \]
\[ = \text{time for (i/2)th pair for even } i \]

Card 13.2: format (6d10.3) etdpfh(i), i=1,2*npaird

\[ \text{etdpfh}(i) = \text{lower plenum property fraction for (i/2+1/2)th pair for odd } i \]
\[ = \text{time for (i/2)th pair for even } i \]

Card 14: format (7d10.3) time0, timef, dtsted, dttran

\[ \text{time0} = \text{initial time for transient calculation (s)} \]
\[ \text{timef} = \text{final time for transient calculation (s)} \]
\[ \text{dtsted} = \text{steady state iteration time step (s)} \]
\[ \text{dttran} = \text{transient time step (s)} \]

Card 15: format (14i5) ntexec, ntprin, ntplot, ntdump

\[ \text{ntexec} = \text{number of execution time intervals} \]
\[ \text{ntprin} = \text{number of printing intervals} \]
\[ \text{ntplot} = \text{number of plotting intervals} \]
\[ \text{ntdump} = \text{number of dumping intervals} \]
Card 15.1.1: format(7d10.3) texec(n), n=1,ntexec+1
   texec(n) = time at the start of execution time interval n
   texec(ntexec+1) = time at the end of the last time interval

Card 15.1.2: format(7d10.3) dtexec(n), n=1,ntexec
   dtexec(n) = transient time step for execution time interval n

Card 15.2.1: format(7d10.3) tprin(n), n=1,ntprin+1
   tprin(n) = time at the start of printing interval n
   tprin(ntprin+1) = time at the end of the last print interval

Card 15.2.2: format(7d10.3) dtprin(n), n=1,ntprin
   dtprin(n) = printing time step for printing interval n

Card 15.3.1: format(7d10.3) tplot(n), n=1,ntplot
   tplot(n) = time at the start of plotting interval n
   tplot(ntplot+1) = time at the end of the last plotting interval

Card 15.3.2: format(7d10.3) dtplot(n), n=1,ntplot
   dtplot(n) = plotting time step for plotting interval n

Card 15.4.1: format(7d10.3) tdump(n), n=1,ntdump
   tdump(n) = time at the start of dumping interval n
   tdump(ntdump+1) = time at the end of the last dumping interval

Card 15.4.2: format(7d10.3) dtdump(n)
   dtdump(n) = dumping time step for dumping interval n
program micon
implicit real*8(a-h,o-z)
parameter (ichand=7,kaxmaxd=13,jconmx=5,nrregmx=3,nthmehd=20,
& nthmhd2=40)

dimension t(ichand,kaxmaxd,jconmx),told(ichand,kaxmaxd,jconmx),
& rtube(ichand,nrregmx,jconmx)

dimension z(ichand,kaxmaxd),tem(ichand,kaxmaxd),ent(ichand,kaxmaxd),
& p(ichand,kaxmaxd),ro(ichand,kaxmaxd),cp(ichand,kaxmaxd),
& temold(ichand,kaxmaxd),entold(ichand,kaxmaxd),roold(ichand,kaxmaxd),
& cpold(ichand,kaxmaxd),q2left(ichand,kaxmaxd),q2right(ichand,kaxmaxd),
& htt(ichand,kaxmaxd),htar(ichand,kaxmaxd),text(ichand,kaxmaxd),
& roext(ichand,kaxmaxd),z0(ichand,kaxmaxd),t0(ichand,kaxmaxd),
& ettm0(ichand,kaxmaxd),tregim(ichand,kaxmaxd),dz(ichand,nrregmx),
& drtube(ichand,nrregmx),qvol(ichand,nrregmx),qtheat(ichand,nrregmx),
& linteg(ichand,nrregmx),kmn(ichand,nrregmx),kmx(ichand,nrregmx),
& jmax(ichand,nrregmx),mater(ichand,nrregmx),avect(kaxmaxd,jconmx),
& fveck(kaxmaxd,jconmx)

dimension argf(kaxmaxd),argg(kaxmaxd),dtwtfl(kaxmaxd),
& dtwtfr(kaxmaxd),dq2tfl(kaxmaxd),dq2tfr(kaxmaxd),
& avect(jconmx),bvect(jconmx),cvect(jconmx),dvect(jconmx)

dimension vtime(ichand,kaxmaxd,nthmehd),qtime(ichand,kaxmaxd,nthmehd),
& qbtme(ichand,kaxmaxd,nthmehd),gtime(ichand,nthmhd2),
& etttime(ichand,nthmehd),etttime(ichand,nthmehd),
& nptime(ichand)

common/aaa/chlieng(7),rtube(7),rtubes(7),rnest(7),aflow(7),
& del(7),nrreg(7),klast(7),w(7),wold(7),delp(7),deltu(7),
& deltd(7),delhu(7),delhd(7),gcost(7),time(7),tvoid(7),
& rintm(7),rintf(7),rintk(7),rintt(7),rkcon(7),rkexp(7),
& rklc(7),bmm(7),cmom(7),lvalv(7),lvoid(7),tttime(7),
& ttttime(7),lregim(7)

common/logkey,istedy,lplenu,lplend,ilfluid,lmomen,lenerg,icondu,
& icon1,iconn,ihtr,ihtr1,ihtr2,lliq,lmomfl,lform,leptf,letabs,
& lamb,ldtcon,lrest,lunit1,ipas,kpas,ipas,iterm,ltpr1,ltpr2,
& lstep

common/times/ttime0,tme,tme0,dtstde,dttran,time,dt

common/fricco/c1am(7),c2lam(7),citur(7),c2tur(7),relim(7)

common/geomin/volu,vold,dzu,tupper,hdowl,hdow,ruper,
& lamb,cpext,volamb,tsolid,retin,浰ilit,einit,cpinit,
& tref,erf,hconv1,hconvr,roinit,wret,wreol,tvalv,tvold,
& lvalv,lvold,tupold,tupold,tdo1,hdold,rdow

common/toler/tolw,tolh,tolt,tolc,tolp,txl,tolx,oluten,itolw,
& itolw,itolt,itolc,itolp,itolx

common/timeeh/texch(10),dtexch(9),tprin(10),dtprin(9),tplot(10),
& dtpit(10),tdump(10),dtdump(9),tprint(200),tplot(200),tdump(200),
& nxex,aex,ntprin,ntplot,ntdump,nex2,nexcl,nprin2,nplot2,
& nplot1,ndump2,ndump1

common/idim/ichan,jconmx,kaxmax,nrregmx,nthmehd,nthmhd2
common/hystet/etdphf(200), etdpfh(200), turef, huref, & tdref, hdref, npairr, npaird

common/hystor/flowfh(200), powfrh(200), npairf, npairp

common/htcoef/a11, a21, a31, a41, a12, a22, a32, a42, & a13, a23, a33, a43, relim1, relim2, grpr1, grpr2, & rnuvrig

common/fricr/ftab(7, 50), nfric(7)

rewind(1)
call idmen

call inpinit(told, rtube, z, tem, ent, p, ro, cp, temold, entold, & text, roext, htl, htr, q2left, q2righ, dz, drtube, qvol, qheat, kmin, & kmax, linteg, matr, jmax, z0, to, roold, cpold, lregim, chleng, & rtube, rheat, aflow, de, nreg, ttime0, ttime0, ttime, lregim, klast, & rkcon, rkep, rloc, w, wold, delp, delu, deltd, delhu, delhd, gcost, & tmev, tvoid, lvalv, lvoid, dtwtf, dtwtr, dq2tf, dq2tr)

call timer1

call steady(told, rtube, z, tem, ent, p, ro, cp, & temold, entold, roold, cpold, q2left, q2righ, htl, htr, text, & roext, z0, to, ttime0, lregim, dz, drtube, qvol, qheat, & linteg, kmin, kmax, jmax, matr, evect, fvect, argf, argg, & dtwtf, dtwtr, dq2tf, dq2tr, bveck, cveck, dvect, & ttime, qatime, qbtine, gtime, etutim, ettim, nptime, & isted, ichan, jconmx, kmaxmax, nregmx, ntimem, ntimh2)

call trans(told, rtube, z, tem, ent, p, ro, cp, & temold, entold, roold, cpold, q2left, q2righ, htl, htr, text, & roext, z0, to, ttime0, lregim, dz, drtube, qvol, qheat, & linteg, kmin, kmax, jmax, matr, evect, fvect, argf, argg, & dtwtf, dtwtr, dq2tf, dq2tr, bveck, cveck, dvect, & ttime, qatime, qbtine, gtime, etutim, ettim, nptime, & isted, ichan, jconmx, kmaxmax, nregmx, ntimem, ntimh2)

stop
end

subroutine momint(z, p, tem, ent, ro, linteg, kmin, kmax, & chleng, aflow, de, nreg, klast, rkcon, rkep, rloc, w, & delp, gcost, tmev, lvalv, rintm, rintf, rintg, rinti, lregim, & tvoid, lvoid, wold, b, c, argf, argg, ichan, kaxmax, & nregmx, qheat, rtube, rheat)

implicit real * 8(a-h, o-z)

*** this routine solves the momentum integral equation for
*** each channel (i=1, ichan) and returns the flow rates w(1).

*** sk, jan84.

dimension z(ichan, kaxmax), p(ichan, kaxmax), tem(ichan, kaxmax),
& ent(ichan, kaxmax), ro(ichan, kaxmax), linteg(ichan, nregmx),
& kmin(ichan, nregmx), kmax(ichan, nregmx)

dimension chleng(ichan), aflow(ichan), de(ichan), nreg(ichan),
& klast(ichan), rkcon(ichan), rkep(ichan), rloc(ichan), w(ichan),
& delp(ichan), gcost(ichan), tmev(ichan), lvalv(ichan), rintm(ichan),
&
& rintf(ichan), rintk(ichan), rintg(ichan), tvold(ichan), lvoid(ichan),
  wold(ichan), b(ichan), c(ichan), argf(kaxmax), argg(kaxmax),
  & tregl(ichan), q(ichan), rheat(ichan), nregm(ichan), rtube1(ichan), rheat(ichan)

common/logkey/1stedy, lplenu, lplend, lfluid, lmoment, lenerg, lcondv, lcondr, lcondl, lhrtr, lhrtr1, lhrtr2, lhrtr3, liiq, lmonfl, lform, letopt, letabs,
  liamb, lidtrc, lrest, lunits, lipas, kpas, jpas, itermx, itpr1, itpr2, lstep

common/timeO, timef, dtsted, dttran, time, dt

common/fricco/c1lam(7), c2lam(7), c1tur(7), c2tur(7), relim(7)

common/geomnu/volu, vold, dzum, tupper, hupper, tdown, hdown, rupper,
  & tamb, cpx, vold, tsoild, wretin, pinit, tinit, einit, cpx, pinit,
  & tref, eref, hconvl, hconvr, roinit, wret, wred, tvalvr, tvalvd,
  vlvorl, lvoid, tupold, hupold, tdold, hddold, rodown

common/hyst/or/flowfl(200), powfrh(200), npairf, npairf

common/toler/tolw, toilh, tolw, tolw, tolx, toix, ltolen,
  & ltolw, ltolh, ltolw, ltolc, ltolp, ltolx

data zero, one, two, thr, four, eps/0.d0, 1.d0, 2.d0, 3.d0, 4.d0, 5.d0, 6.d0,
  ltolw=0

do 500 i=1, ichan
  kk=klast(i)
  rmax=rregm(i)
  rintm(i)*zero
  rintf(i)*zero
  rintk(i)*zero
  rintg(i)*zero
  c3=rkconf1
  c4=rkexp1
  sumf=zero
  sumg=zero

***step 1. computation of integral terms: rint-m, f, k, g(i)

***if(lmonfl.le.0) momentum flux term is neglected
***if(lform.le.0) form loss term is neglected

if(lmonfl) 130, 130, 120
  continue
  rintm(i) = (one/ro(1,kk)-one/ro(1,1))/aflow(i)/aflow(i)
130  continue
  if(lform) 170, 170, 140
140  continue
  roavg=(ro(1,1)+ro(1,kk))/two
  if(w(i)) 150, 170, 150
150  continue
  rintk(i) = rkloc1(i)/roavg/two/aflow(i)/aflow(i)
170  continue
  pav = (p(1,1)+p(1,kk))/two
  hav = (ent(1,1)+ent(1,kk))/two
  tav = (tem(1,1)+tem(1,kk))/two
  roaq = (ro(1,1)+ro(1,kk))/two
  rmuav = visc(pav, tav, hav, roaq, lliq)
  reyav = dabs(w(i))**2/flow(i)/aflow(i)/rmuav
  rintn = c*rmuav*c4/two/aflow(i)**(two-c4)
rintn=rintn/de(1)**c4/roav
if(reav-relim(1)) 200,200,210
200 continue
c1=c1lam(i)
c2=c2lam(i)
ifregim(1)=1
goto 220
210 continue
c1=c1tur(i)
c2=c2tur(i)
ifregim(1)=2
220 continue
exmpf=c1/two/de(1)**(one+c2)/aflow(1)**(two-c2)
do 350 n=1,nmax
   k1=kmin(1,n)
k2=kmax(1,n)
kax=k2-k1+1
zdow=z(1,k1)
ztop=z(1,k2)
lint=1+leng(1,n)
do 240 k=k1,k2
   rmiu=visc(p(1,k),tem(1,k),ent(1,k),ro(1,k),liq)
   cond=thcon(p(1,k),tem(1,k),ent(1,k),ro(1,k),liq)
   beta=thexp(p(1,k),tem(1,k),ent(1,k),liq)
   fac=frac0(g(1),q,h,heal(1,n),de(1),rheat(1))
   &rtube(1,aflow(1),gcost(1),ro(1,k),beta,cond,rmiu,ichan,i)
   argf(k-k1+1)=rmiu**c2/ro(1,k)*fac
   argg(k-k1+1)=ro(1,k)
240 continue
   goto (260,270,280), lint
260 continue
rintf=trapez(zdow,ztop,argf,kax,lint(1,n))
rintgn=trapez(zdow,ztop,argf,kax,lint(1,n))
goto 300
270 continue
rintf=imps(zdow,ztop,argf,kax,lint(1,n))
rintgn=imps(zdow,ztop,argf,kax,lint(1,n))
goto 300
280 continue
c***dummy now***
   stop
300 continue
sumf=sumf+rintf
sumg=sumg+rintgn
350 continue
rintf(i)=exmpf*sumf
rintg(i)=gcost(i)*sumg
   c
   c***step2. evaluation of te expression of b(i) and c(i) in the eqn
   c***po(i)-p(i)=dw(i)+b(i)+c(i)
dabswl=dabs(w(t))
   if(dabswl.lt.eps) dabswl=eps
term1=chleg(1)/aflow(1)/dt
term2=w(t)+rintm(1)
term3=dabsw1***(one-c2)*rintf(i)
term4=dabsw1*rintk(1)
term5=dabswl***(one-c4)*rintn
b(i)=term1+term2+(two-c2)/two*term3+term4+(two-c4)/two*term5
c(i)=w(t)+(term2+term3+term4+term5)*rintg(1)
   c
c***step 3. find if channel i is open or closed at actual time
    if(time-timev(i)) 400,370,370
 370    continue
    timev(i)=timef+eps
    if(lvalv(i)) 380,380,390
 380    continue
    lvalv(i)=1
    goto 400
 390    continue
    lvalv(i)=0
 400    continue

c***finished with steps 1-3 for channel i. continue with next
    c***channel.
 500    continue

c***repeat step 3 for return line
    if(time-tvalvr) 550,510,510
 510    continue
    tvalvr=timef+eps
    if(lvalvr) 520,520,530
 520    continue
    lvalvr=1
    goto 550
 530    continue
    lvalvr=0
 550    continue

c***step 4. computation of incremental flow rate during
    c***actual time step interval and finish with the computation
    c***of flow rate in each channel i at the end of the time step.
    wrnew=wrnew polymalflow, time, npair, memorf)*dble(lvalvr)
    dwret=wrnew-wret
    ichan1=ichan-1
    deltpb=c(ichan)
    sumdwli=zero
    do 800 r=1,ichan1
    if(lvalv(i)) 600,600,610
 600    continue
    sumdwli=sumdwli-w(i)
    w(i)=zero
    goto 800
 610    continue
    sumnom=zero
    sumden=zero
    cimdpl=c(i)-delp(i)*deltpb
    do 650 m=1,ichan1
    if(m-i) 620,650,620
 620    continue
    expr=dble(lvalv(m))/b(m)
    cmmdpm=c(m)*delp(m)*deltpb
    sumnom=sumnom+(cimdpl*cmmdpm)*expr
    sumden=sumden+expr
 650    continue
    expr=dble(lvalv(ichan))/b(ichan)
    cbpdpb=c(ichan)*gcost(ichan)*rouper*du
    sumnom=sumnom+(cimdpl*cbpdpb)*expr
sumden = sumden + expr
\(dw_i = -(d\text{wret} + \text{sumn0m})/(\text{one} + b(i) \times \text{sumden})\)
sumdwi = sumdwi + dwi
if (itolen. le. 0) goto 700
if (dabs(w(i)). lt. eps) goto 700
if (dabs(dwi/w(i)). gt. tolw) ltolw = 1

### Compute the flow rates in the core channels (i = 1, ichan)

### At the new time level

700 continue
w(i) = w(i) + dwi
800 continue

### Compute the flow rates in bypass channel (i = ichan) and in
### the return line at the new time

820 continue
w(ichan) = zero
goto 880
840 continue
i = ichan
dwi = -(d\text{wret} + \text{sumdwi})
if (itolen. le. 0) goto 860
if (dabs(w(i)). lt. eps) goto 860
if (dabs(dwi/w(i)). gt. tolw) ltolw = 1
860 continue
w(i) = w(i) + dwi
880 wret = wnew
return
end

function fratio(w, qiheat, de, rheat, rtube1, aflow, grav, & ro, beta, cond, rmiu, ichan, i)

### The function fratio finds the mixed convection to forced convection
### friction factor ratio as a function of the Grf/Re parameter. The
### value of Grf/Re is first calculated based on the external heat
### source, channel flowrate, and local properties. The value for
### fratio is then obtained by interpolating or extrapolating from
### the table ftab. For an annulus, the heated perimeter is assumed to
### be based on the heater (rod) radius. If the heater radius equals
### zero, the tube inner radius is used. VI May 1985.

implicit real*8(a-h.o-z)
dimension table(50)
common/fricr/ftab(7,50), nfric(7)
data pi, eps, zero, one, two/3.14159d0, 1.d-6, 0.d0, 1.d0, 2.d0/
fratio = one
reyn = w*de/aflow/rmiu
if (dabs(qiheat). lt. eps) return
if (rheat. le. eps) rheat = rtube1
q2heat = qiheat/two/pi/rheat
gro = ro*ro*grav*beta*de**4*q2heat/cond/rmiu/rmiu
gro = grq/reyn
if (dabs(gro). lt. eps) return
if(gore.lt.zero) goto 500
klast=nfric(1)
klast2=klast+2
klast1=klast2-1
if(gore.le.ftab(1,klast)) goto 50
fratio=ftab(1,klast2)
return

50 continue
if(gore.ge.ftab(1,1)) goto 60
fratio=ftab(1,2)
return

60 continue

C***Function polate requires that the table be in the form y1,x1,y2,x2,...
C***The xy pairs in ftab must therefore be inverted.
C
do 100 k=1,klast
    table(2*k-1)=ftab(1,2*k)
    table(2*k)=ftab(1,2*k-1)
100 continue
fratio=polate(table,gore,klast,memorp)
return

500 continue
C***dummy for now
return
end

function simps(a,b,f,n,lerr)
C
C***the function simps uses simpon's rule to numerically
C***calculate the integral of f(x) between integration limits
C***a and b. sumevn is the sum of all f(x(j)) for even
C***j (2,le.j,le.(n-1)), while sumodd is the sum of all
C***f(x(j)) for odd j (3,le.j,le.(n-2)). h is the step size
C***between adjacent x(j). h=(b-a)/(n-1). n (n.ge.3) is
C***the total number of mesh points in the interval a,b
C***(odd) while n-1 is the total number of mesh intervals
C***(even). if n=3, sumodd=zero.
C***sk,dec83.
C
implicit real*8(a-h,o-z)

dimension f(n)
data zero,one,two,three,four/0.0,1.0,2.0,3.0,4.0,5.0/

C***Initialization
C
nn=(n-1)/2
nn=2*nn+1
if(nn.ne.nn) goto 60
if(n-3) 50,10,10
10 continue
nml=n-1
nn2=n-2
h=(b-a)/dbl(nn1)
sumevn=zero
sumodd=zero

C
C***evaluate sumevn and sumodd
C
do 20 j=2,nml,2
sumevn=sumevn+f(j)
if(n.1e.3) goto 40
do 30 j=3,nm2,2
30 sumodd=sumodd+f(j)
c
***return estimated value of the integral
40 continue
  simps=h/thre*(f(1)+four*sumevn+two*sumodd+f(n))
return
50 continue
  ierr=1
  write(2,80) n,a,b
  format(//10x,"****error in simps. number of mesh points less & than",\ 
  13,"a","a",1pd12.5,"b",1pd12.5)
return
60 continue
  ierr=1
  write(2,90) n,a,b
  format(//10x,"****error in simps. total number of mesh points is & even",\ 
  13.,"a",1pd12.5,5x,"b",1pd12.5)
return
end

function trapez(a,b,f,n,ierr)
c
***the function trapez uses the trapezoidal rule to numerically
***calculate the integral of f(x) between the integration limits
***a and b. sum is the sum of all f(x(j)) for 2.1e.j.1e.(n-1).
***h is the step size between adjacent x(j). h=(b-a)/(n-1).
***n-1 is the total number of mesh intervals. if n=2, sum=zero.
implicit real*8(a-h,o-z)
dimension f(n)
data zero,one,two,thre,four/0.d0,1.d0,2.d0,3.d0,4.d0/
c
***initialization
10 continue
  nm1=n-1
  h=(b-a)/dble(nm1)
  sum=zero
  if(n.2) 50,40,20
c
***evaluate sum
20 continue
  do 30 j=2,nm1
30 sum=sum+f(j)
c
***return estimated value of the integral
40 continue
  trapez=h/two*(f(1)+two*sum+f(n))
return
50 continue
  write(2,80) n,a,b
  format(//10x,"****error in trapez. number of mesh points less than"
& 2*****.,//10x,"n"==13,"a"==1,pf(12.5,5x,"b"==1,pf(12.5))
stop
end

function polate(xy,xx,nn,kk)
implicit real*8(a-h,o-z)

c***linear interpolation and extrapolation routine
c
dimension xy(200)
c
c***xy is a table of y(1),x(1),y(2),x(2),....y(nn),x(nn)
c***xx is the given value of x
c***nn is the number of entries in xy
c***kk is both the position guess and the final value
c
data zero/0.d0/
c
x=xx
n=nn
m=ials(n)
k=kk
if(k.lt.1) k=1
if(k.ge.m) k=m-1

c***is constant wanted
c
if(m-1) 20,30,50
20 continue
polate=zero
return
30 continue
polate=xy(1)
return.
c
c***loop to decrease index
c
c
50 continue
if(xy(2+k)-x) 70,70,60
60 continue
k=k-1
if(k) 90,90,50

c
c***loop to increase index
c
c
70 continue
if(x-xy(2+k+2)) 120,120,80
80 continue
k=k+1
if(k-m) 70,100,100

c
c***setup extrapolation index
c
c
90 continue
k=1
goto 120
100 continue
k=m-1
c
c***get answer
c
120 continue
   kk=k
   pulate=xy(2*k-1)+(x-xy(2*k))*((xy(2*k+1)-xy(2*k-1))
   & /((xy(2*k+2)-xy(2*k))
   return
end

subroutine timer1

***this routine computes from time intervals of printing
***of printing (tprint), and from delta t print (dtpin1) the exact times
***sk jan84.

implicit real*8(a-h,o-z)

c data eps /1.d-5/
nprint2=1
tprint(n)=tprint(1)
do 500 i=1,200
   do 300 n=1,nprint
      if((tprint(i-1).gt.(tprint(n)-eps)).and.
         & ((tprint(i-1).lt.(tprint(n+1)-eps))) goto 100
   goto 300
100 continue
   tprint(i)=tprint(i-1)+dtpin(n)
goto 400
300 continue
400 nprint=nprint+1
   if((tprint(i).gt.(tprint(nprint+1)-dtpin(nprint)+eps))
      & return
500 continue
   return
end

subroutine idimen

***this routine defines the overall dimension of the problem.
***sk jan84.

c ichan=number of channels (including bypass channel)
c jconnx=number of radial mesh points in solid structures
ckaxmax=max number of axial mesh points
cknregm=max number of axial regions

c common/idim/ichan,jconnx,kaxmax,nregmx,ntimeh,ntimh2

c write(2,150)
read (1,100) ichan,jconnx,kaxmax,nregmx,ntimeh
ntimh2=ntimeh+2
write(2,200) ichan,jconnx,kaxmax,nregmx,ntimeh,ntimh2
return
100 format(7110)
150 format(/5x,"input return",/5x,12("--"),/5x,12("--"),/5x,"problem"
& dimension data",/5x,22("--"),/)
200 format(5x,"number of channels (including bypass) is ichan",
& 12,/5x,"maximum number of radial mesh points in solid"
& structure is jconnx",12,/5x,"maximum number of axial
& mesh points is kaxmax = 12, /5x, *maximum number of axial
& region is nregmx = 12, /5x, *ntimh = 13, /5x, *ntimh2 = 13
end
subroutine condid(t, told, rtube, drtube, qvol, he, hf, te, tf, q2left,
& q2righ, e, f, kmin, kmax, jmax, mater, nreg, dtwf, dtwtfr, dq2tf1,
& dq2tf2, a, b, c, d, 1con, lconr, 1, lstep, 1chan, jcomx, kaxmax, nregmx)

C   ***this routine solves the one-dimensional heat conduction
C   ***equation in cylindrical coordinates (assuming azimuthal
C   ***symmetry and no axial conduction).  sk jan84.
C implicit real *8(a-h, o-z)
C  dimension told(1chan, kaxmax, jcomx)
C  dimension t(1chan, kaxmax, jcomx), rtube(1chan, nregmx, jcomx),
& drtube(1chan, nregmx), qvol(1chan, nregmx), he(1chan, kaxmax),
& hf(1chan, kaxmax), te(1chan, kaxmax), tf(1chan, kaxmax), q2left
& (1chan, kaxmax), q2righ(1chan, kaxmax), e(1chan, kaxmax, jcomx),
& f(1chan, jcomx), kmin(1chan, nregmx), kmax(1chan, nregmx),
& jmax(1chan, nregmx), mater(1chan, nregmx), nreg(1chan),
& dtwf(1chan, kaxmax), dtwtfr(1chan, kaxmax), dq2tf1(1chan, kaxmax),
& dq2tf2(1chan, kaxmax), a(jcomx, b(jcomx), c(jcomx), d(jcomx))
C  common/time, time0, timef, dt, dtstep, dttran, time, dt
C  common/toler/tolw, tolh, tolt, tolx, tolp, tolx, ltolw,
& ltohl, ltoht, ltoxt, ltopx, ltolx
C data zero, one, two, thrw, four, half/0.d0, 1.d0, 2.d0, 3.d0, 4.d0, 5.d0/
& nmax=nreg(1)
C  goto (50, 650), 1step
50  continue  do 500 n=1, nmax
  if(n-1) 60,60,70
60  continue  k1=kmin(1,n)
  goto 80
70  continue  k1=kmin(1,n)+1
  continue
80  continue  k2=kmax(1,n)
  jmaxr=jmax(1,n)
  matermater(1,n)
  drsqo4=drtube(1,n)+drtube(1,n)/four
  do 500 k=k1, k2
  C  C
  C   ***set up coeff a, b, c, d(j) in finite difference heat cond eqn:
  C   ***a(j)*t(j+1)+b(j)*t(j)+c(j)*t(j-1)=d(j)
  C  it therefore performs the elimination and implicit fluid-
  C   ***structure coupling (if desired) steps of the heat cond eqn
  C   ***solution.
  C
   do 300 j=1, jmaxr
   C
   C   ***set up coefficients
   C
   rjdr=rtube(1,n,j)*drtube(1,n)
   rjodr=rtube(1,n,j)/drtube(1,n)
if(j-1) 120,120,160
continue
if(1con1) 130,140,130
continue
bleft=two*rtube(i,n,j)*dt*he(i,k)
dtwf(r(k)=bleft
dleft=zero
if(1con1,lt.0) dieleft=bleft*te(i,k)
goto 150
continue
bleft=zero
dleft=two*rtube(i,n,j)*dt*q2left(i,k)
continue
c(j)=zero
goto 180
continue
r2left=rjttdr-drsq4
rodr=e=jojdr-half
taveg=(t(i,k,j-1)+t(i,k,j))/two
rocleft=roc(taveg,mat)
rlleft=rlk(taveg,mat)
fluxtr=two+rodr=rlleft*dt
bleft=rocleft+r2left+fluxtr
dleft+r2left+(rocleft*t(i,k,j)+qvol(1,n)*dt)
c(j)=fluxtr
continue
if(j-jmaxr) 190,210,210
continue
r2right=rjttdr-drsq4
rodr=ojdr-half
taveg=(t(i,k,j)+t(i,k,j+1))/two
rocright=roc(taveg,mat)
rlright=rlk(taveg,mat)
fluxtr=two+rodr+rlright*dt
a(j)=fluxtr
bright=rocright+r2right+fluxtr
dright=r2right+(rocright*t(i,k,j)+qvol(1,n)*dt)
goto 250
continue
a(j)=zero
if(1conr) 220,230,220
continue
bright=two*rtube(i,n,j)*dt*hf(1,k)
dtwfr(r(k)=bright
dright=zero
if(1conr,lt.0) dright=bright+tf(1,k)
goto 250
continue
bright=zero
dright=two*rtube(i,n,j)*dt*q2right(i,k)
continue
b(j)=bleft+bright
dright=dleft+dright
continue
c
****elimination and fluid-structure coupling steps

****at a certain axial level k, we must compute the vectors
****e(j,k) and f(j,k) that satisfy the recursion relations
****t(k,1)*e(k,j)+t(k,j-1)*f(k,j) for backward elimination
and forward substitution, or \( t(k,j) = e(k,j) \cdot t(k,j+1) + f(k,j) \)

for forward elimination and backward substitution. This step is finished with the computation of derivative \( d\alpha_{psic2}/dt\text{fluid} \) that is needed in the solution of the fluid energy equation if an implicit fluid-solid structure at either left or right hand side is used.

\[
j_{max} = j_{maxr} + 1
\]

if (iconr) 320, 320, 400

if (iconr. eq. 0) an explicit fluid-structure at the right and side is used (i.e. \( t\text{fluid} \) at this side appears at the old time step).

320 continue

dtw\text{fr}(k) = zero
dq2\text{fr}(k) = zero

backward elimination

\[
e(k, j_{max}) = c(j_{max}) / b(j_{max})
f(k, j_{max}) = d(j_{max}) / b(j_{max})
do 340 jj = 2, j_{max} - 1
\]

\[
den = b(jj) \cdot a(jj) + e(k, j+1)
e(k, j) = c(jj) / den
f(k, j) = (d(jj) \cdot a(jj) + f(k, j+1)) / den
\]

continue

\[
den = b(1) \cdot a(1) + e(k, 2)
t(k, 1) = (d(1) \cdot a(1) + f(k, 2)) / den
\]

if (iconl) 360, 360, 370

if (iconl. eq. 0) an explicit fluid-solid structure at the left hand side is also used. Therefore \( d(twall) / d(t\text{fluid}) \)

\[d(\alpha_{psic2}) / d(t\text{fluid}) = 0.\] if (iconl. gt. 0), \( d(twall) / d(t\text{fluid}) = 2 \cdot r1 \cdot dt \cdot he(1) \cdot a(2)\) and \( d(\alpha_{psic2}) / d(t\text{fluid}) = he(1 \cdot d(twall) / d(t\text{fluid}))\).

360 continue

dtw\text{fl}(k) = zero
dq2\text{fl}(k) = zero
goto 450

370 continue

dtw\text{fl}(k) = dtw\text{fl}(k) / den
dq2\text{fl}(k) = he(1, k) \cdot (one - dtw\text{fl}(k))
goto 450

400 continue

if (iconr. gt. 0) an implicit fluid-solid structure at right hand side is used (i.e. \( t\text{fluid} \) at this side appears at the new time step).

dtw\text{fl}(k) = zero
dq2\text{fl}(k) = zero

forward elimination

\[
e(k, 1) = a(1) / b(1)
f(k, 1) = d(1) / b(1)
do 420 jj = 2, j_{max} - 1
\]

\[
den = b(jj) \cdot c(jj) + a(k, j-1)
\]
e(k, j) = a(j)/den
f(k, j) = (d(j) + c(j) * f(k, j-1))/den
continue
j = jmax - 1

den = b(j) - c(j) * e(k, j-1)
t(1, k) = (d(j) + c(j) * f(k, j-1))/den
dtwfcc(k) = dtwtfrc(k)/den

d2tfrc(k) = -hf(k, k) * (one - dtwtfrc(k))

450 continue

C*** the steps of elimination and fluid structure coupling,
C*** at a certain axial level k, are finished. It continues
C*** now at the next axial level.

C
500 continue
if (icon1) 550, 550, 600
550 continue
if (iconr) 650, 650, 600
600 continue
lstep = 1
C
C*** if at either left or right hand side, an implicit fluid-solid
C*** structure is used, go to solve the energy equation, and then come
C*** back for final substitution.
C
return
650 continue
C
C*** the last step -- substitution
C
do 900 n = 1, nmax
if (n-1) 670, 670, 680
670 continue
k1 = kmin(1, n)
900 continue
k2 = kmax(1, n)
jmax = jmax(1, n)
do 900 k = k1, k2
if (iconr) 750, 750, 800
750 continue
C
C*** if (iconr.le.0), i.e., if an explicit fluid-solid structure
C*** coupling at the right hand side is used, for both subcases
C*** explicit or implicit coupling at the left hand side, a
C*** backward elimination has been used; therefore, a forward
C*** substitution will be used, too.
C
770 continue
go to 850
800 continue
C
C*** if (iconr.gt.0), i.e., implicit fluid-solid structure at right
C*** hand side is used, a forward elimination has been used;
C*** therefore a backwards elimination will be used, too.
c t(i,k,jmaxr) = t(i,k,jmaxr) + dtwtr(k) * tf(i,k)
do 820 jj = 2, jmaxr
   j = jmaxr - jj + 1
t(i,k,j) = e(k,j) * t(i,k,j+1) + f(k,j)
820 continue
850 continue

c c*** the substitution step at a certain axial level k is finished.
c it now continues at the next axial level.
c
900 continue
   1stel = 1
   ltolc = 0
   if (1tolc .le. 0) goto 990
do 950 n = 1, nmax
   if (n .eq. 1) 920, 920, 930
920 continue
   k1 = kmin(i, n)
goto 940
930 continue
   k1 = kmin(i, n) + 1
940 continue
   k2 = kmax(i, n)
   jmaxr = jmax(1, n)
do 950 k = k1, k2
   do 950 j = 1, jmaxr
tdifer = t(i, k, j) - tdol(1, k, j)
   if (dabs(tdifer) .gt. tolc) ltolc = 1
   if (ltolc .eq. 1) goto 990
950 continue
990 continue
return
end

subroutine inplin(t, tol, rtube, z, tem, ent, p, ro, cp, temid, entid, &
text, roext, htl, htr, q2left, q2right, dz, dtrube, qvol, q1heat, kmin, &
kmax, lnteg, matr, jmax, zo, t0, roold, cpold, lregin, chlen,
&
rtube, rtube, rheat, aflow, de, nreg, ttime0, ttiold, lregin,
&
klast, rkcon, rkeps, rkloc, w, wold, delp, delu, delt, del, deiu, delh,
&
gcost, timev, tvold, lvold, dtwtr, dq2tf1, dtwtrf, dq2tfr)

implicit real*8(a-h, o-z)

*** this is the input initialization routine, sk jan84.

dimension t(ichan, kaxmax, jconmx), told(ichan, kaxmax, jconmx), &
rtube(ichan, nregmx, jconmx)
dimension z(ichan, kaxmax), tem(ichan, kaxmax), ent(ichan, kaxmax), &
p(ichan, kaxmax), ro(ichan, kaxmax), cp(ichan, kaxmax), temid(ichan, &
kaxmax), entid(ichan, kaxmax), text(ichan, kaxmax), roext(ichan, &
kaxmax), htl(ichan, kaxmax), htr(ichan, kaxmax), q2left(ichan, kaxmax), &
q2right(ichan, kaxmax), dz(ichan, nregmx), dtrube(ichan, nregmx), &
qvol(ichan, nregmx), q1heat(ichan, nregmx), kmin(ichan, nregmx), &
kmax(ichan, nregmx), lnteg(ichan, nregmx), mater(ichan, nregmx), &
jmax(ichan, nregmx), roold(ichan, kaxmax), cpold(ichan, kaxmax), &
zo(ichan, kaxmax), t0(ichan, kaxmax), lregin(ichan, kaxmax)
dimension chlen(ichan), rtube(ichan), rtrube(ichan), rheat(ichan), &
aflow(ichan), delchan, nreg(ichan), klast(ichan), rkcon(ichan),
**General Model Data**

```
read (1, 1000) istedy, lplenu, lplend, lfluid, lmoment, lenerg, liconn, liconr, lhttr, lhttr, lliq, lmoment1
& lunit, lpas, kpas, jpas, itermx, lptr1, lptr2
if(lenerg.ge.7) litol=0
lconr=asin0(0, lconr)
if(lfluid.gt.1) lfluid=1
if(lunit.gt.0) lunit=0
if(lhttr.lt.-2) lhttr=-2
if(lhttr.eq.0) lhttr=1
if(lhttr.gt.2) lhttr=2
if(liconr.gt.0 .and. lconr.gt.0) lconr=-lconr
```
& rhheat(i), aflow(i), de(i), nreg(i)
rmx*nreg(i)
kmin(1,1)=1
z(1,1)=zero
do 300 n=1,nmax
read (1,1600) reglen,kaxreg,jmax(1,n),mater(1,n)
if(kaxreg.lt.2) kaxreg=2
if(jmax(1,n).lt.2) jmax(1,n)=2
if(nmax.eq.1) reglen=chieng(1)
dz(1,)=reglen/1.dble(kaxreg-1)
klast(1)=kmin(1,n)+kaxreg-1
if(klast(1).gt.kaxmax) goto 920
kmax(1,n)=klast(1)
if(n.ne.nmax) kmin(1,n+1)=kmax(1,n)
k1=kmin(1,n)+1
k2=kmax(1,n)
do 200 k=k1,k2
200 z(1,k)=z(1,k-1)+dz(1,n)
  jmaxr=jmax(1,n)
  if(jmaxr.gt.jconmx) goto 930
  drtube(1,n)=(rtube(1,1)-rtube(1,1))/dble(jmaxr-1)
do 250 j=1,jmaxr
250 rtube(1,n,j)=rtube(1,1)+rtube(1,n)*dble(j-1)
300 continue
kk=klast(1)
  if(dabs(z(1,kk)-chieng(1)).gt.(tol*chieng(1))) goto 940
write(2,1650)
  if(lunits.eq.0) write(2,1700)
  if(lunits.gt.0) write(2,1730)
do 400 n=1,nmax
  if(k1=kmin(1,n)
  k2=kmax(1,n)
  jmaxr=jmax(1,n)
  write(2,1750) n, dz(1,n), k1, k2, z(1,k1), z(1,k2), drtube(1,n),
  & jmaxr, rtube(1,n,1), rtube(1,n, jmaxr), mater(1,n)
400 continue

c***geometrical definition of plena and initial data for ambient atm,
c***solid structure, and return line
read (1,1150) volu, void, dzu, tamb, cpext, volamb, tsolid
read (1,11620) wretin, tvalvr, lvvalvr
write (2,1800) volu, void, dzu, tamb, cpext, volamb, tsolid, wretin,
& tvalvr, lvvalvr
wret=wretin
dwret=zero
if(tvalvr.gt.1) lvvalvr=1
if(tvalvr.lt.0) lvvalvr=0
wreold=wretin
tvold=tvalvr
lvold=lvvalvr


c c***initial properties and operational conditions


c goto (500,550), etcabs
continue
read (1,1150) pinit, tinit, cplinit, tref, href, hconvl, hconvr
& elit=entali(pinit, tinit, li, liq)
&href =entali(pinit, tref, li, liq)
goto 600
if (lenegrit > 1) lenegrit = 1
letabs = labs(letopt)
1step = 1
write(2, 1050)
write(2, 1100) lstedy, lplenu, lplend, lifluid, lmomen, lenegrit, lcondu,
& lcrnt, lconr, lhtr, lhtr1, lhtr2, linq, lmom1, liform, letopt, itolen,
& leamb, ltdcon, lrest, lunits, lps, kpas, kps, jpas, ltermx, ltpfr, ltpfr2,
& lstep, letabs

c
### general constant data

do 50 i = 2, 1, ichan
read (1, 1150) c1lam(i), c2lam(i), c1tur(i), c2tur(i), relim(i)
if (dabs(c2lam(i) - c2tur(i)) > eps .and. c1lam(i) / c1tur(i) > 0)
& (c2tur(i) - c2lam(i) > 0))
50 continue
read (1, 1150) a11, a21, a31, a41, relim1, grpr1
read (1, 1150) a12, a22, a32, a42, relim2, grpr2
read (1, 1150) a13, a23, a33, a43, rrunig
write(2, 1210) a11, a21, a31, a41, a12, a22, a32, a42, a13,
& a23, a33, a43, relim1, relim2, grpr1, grpr2, rrunig
write(2, 1250) 1, rrunig, rrunig, c1lam(i), c2lam(i),
& c1tur(i), c2tur(i), relim(i)
do 100 i = 1, ichan
read (1, 1150) nkfric(i), ftab(k), k = 1, kfric(i)
write(2, 1255) 1
write(2, 1256) ftab(k), k = 1, kfric(i)
do 110 i = 1, ichan
kfric(i) = ftab(k) - kfric(i)
write(2, 1256) ftab(k) - kfric(i)
do 110 i = 1, ichan
read (1, 1150) tolw, tolh, tolkt, tolct, tolup, tolx
write(2, 1280) tolw, tolh, tolkt, tolct, tolup, tolx

c
tolw = 0
tolh = 0
tolkt = 0
tolct = 0
tolup = 0
tolx = 0

c
### geometrical data

do 400 i = 1, ichan
write(2, 1350) 1
read (1, 1400) chleng(i), rtube1(i), rtube2(i), rheat(i), afrow(i),
& de(i), nreg(i)
if (de(i) > 1) nreg(i) = 1
if (de(i) > 1) nregmx() goto 910
if (afrow(i) > 1) afrow(i) = 1
if (de(i) > 1) de(i) = 2
if (nreg(i) > 1) nreg(i) = 1
if (de(i) > 1) de(i) = 2
write(2, 1500) chleng(i), rtube1(i), rtube2(i), rheat(i),
& de(i), afrow(i), nreg(i)
write(2, 1550) chleng(i), rtube1(i), rtube2(i), rheat(i),
& de(i), afrow(i), nreg(i)
550 continue
read (1,1150) pinit,einit,cpinit,tref,eref,hconvr,hconvl,tinit*temper(pinit,einit,111q)
tref =temper(pinit,eoref,111q)
600 continue
roinit=dens(pinit,tinit,einit,111q)
roupper=roinit
tupper=tinit
hupper=einit
tdown=tinit
hdown=einit
write(2,1850) pinit,tinit,einit,cpinit,tref,eref,hconvr,hconvl,
& hconvr,roinit
sumwi=zero
do 717 i=1,ichan
kk=klast(i)
nmax=nreg(i)
610 continue
if(lsetopt) 610,660,660
620 continue
goto (620,640), letabs
630 continue
read (1,1150) (tem(i,k),k=1,kk)
do 630 k=1,kk
ent(i,k)=ental(pinit,tem(i,k),111q)
640 continue
goto 705
650 continue
read (1,1150) (ent(i,k),k=1,kk)
do 650 k=1,kk
tem(i,k)=temper(pinit,ent(i,k),111q)
660 continue
goto 705
670 continue
goto (670,690), letabs
680 continue
do 680 k=1,kk
ent(i,k)=tinit
tem(i,k)=ental(pinit,tem(i,k),111q)
690 continue
goto 705
700 continue
do 700 k=1,kk
ent(i,k)=einit
tem(i,k)=temper(pinit,ent(i,k),111q)
705 continue
do 710 k=1,kk
temold(i,k)=tem(i,k)
tenold(i,k)=ent(i,k)
p(i,k)=pinit
ro(i,k)=dens(pinit,tem(i,k),ent(i,k),111q)
cp(i,k)=spheat(pinit,tem(i,k),ent(i,k),111q)
z0(i,k)=zero
t0(i,k)=zero
roold(i,k)=ro(i,k)
cpold(i,k)=cp(i,k)
1regim(i,k)=0
text(i,k)=tamb
roext(i,k)=dens(pinit,tamb,hdummy,3)
htl(i,k)=hconvl
htr(i,k)=hconvr
q2left(i,k)=htr(i,k)*(tem(i,k)-tsolid)
q2righ(i,k)=htr(i,k)*(tsolid-text(i,k))
if(lhs.eq.0) q2left(i,k)=zero
if(lhs.eq.0) q2righ(i,k)=zero
if(lhs.lt.1) q2left(i,k)=zero
if(lhs.lt.1) q2righ(i,k)=zero
710 continue
  do 715 k=1,kaxmax
  dtwtf1(k)=zero
dtwthr(k)=zero
dq2tf1(k)=zero
dq2tfhr(k)=zero
715 continue
717 continue
  do 800 l=1,lchan
  nmax=nreg(l)
goto (720,730), letabs
720 continue
  read (1,1400) w(l),delpl(l),deltu(l),deltd(l),gcost(l),
  & timev(l),lvalv(l)
delhu(l)+deltu(l)+cpinit
delhd(l)+deltd(l)+cpinit
goto 740
730 continue
  read (1,1400) w(l),delpl(l),delhu(l),delhd(l),gcost(l),
  & timev(l),lvalv(l)
deltu(l)+delhu(l)+cpinit
deltfd(l)+delhd(l)+cpinit
740 continue
  if(lhs.ne.lchan) goto 750
  w(l)=sumwl-wretin
750 continue
  sumwl=sumwl+w(l)
  if(lvalv(l).lt.0) lvalv(l)=0
  if(lvalv(l).gt.1) lvalv(l)=1
  wold(l)=w(l)
  tvald(l)=timev(l)
  lvald(l)=lvalv(l)
  iregim(l)=0
  do 790 n=1,nmax
  jmax=jmax(1,n)
  read (1,1600) q1heat(l,n),lnteg(l,n)
  qvol(l,n)=zero
  if(lnteg(l,n).gt.2) lnteg(l,n)=2
  if(n-l) 760,770,770
760 continue
  k1=kmin(l,n)
goto 780
770 continue
  k1=kmin(l,n)+1
780 continue
  k2=kmax(l,n)
c c==check if the number of axial levels in each axial zone
c==is even or odd. If it is even, simpson's rule can not
c==be used for the numerical integration
c kax=kmax(l,n)-kmin(l,n)+1
kk=(kax-1)/2
kk=2*kk+1
if(kax.ne.kk) lnteg(i,n)=1
do 790 k=k1,k2
    do 790 j=1,jmaxx
        t(1,k,j)=tsolid
    enddo
    told(i,k,j)=t(1,k,j)
continue
790 continue

*** the initial state of channel i was defined. go to the next channel.

continue
800 continue
    do 830 i=1,ichan
        kk=klast(i)
        if(lunits.ie.0) write(2,1870) i
        if(lunits.gt.0) write(2,1880) i
        do 830 k=1,kk
            write(2,1900) k,p(i,k),tem(i,k),ent(i,k),ro(i,k),cp(i,k),
                & text(i,k),roext(i,k),hti(i,k),htr(i,k),q2left(i,k),q2righ(i,k)
        enddo
    enddo
830 continue
    if(lunits.ie.0) write(2,1920)
    if(lunits.gt.0) write(2,1930)
    do 850 i=1,ichan
        write(2,1940) i,w(i),delta(i),deltau(i),delta(i),delhu(i),delh(i),
            & gcost(i),timev(i),lvalv(i)
    enddo
850 continue
    do 870 i=1,ichan
        if(lunits.ie.0) write(2,1960) i
        if(lunits.gt.0) write(2,1970) i
        nmax=nreg(i)
        do 870 n=1,nmax
            write(2,1980) n,q1heat(i,n),qvol(i,n),lnteg(i,n)
        enddo
870 continue
    write(2,1990) tsolid

*** fractional flow rate in return line and fractional power histories

read (1,1000) npairf, npairp
npairf=max0(npairf,2)
npairp=max0(npairp,2)
npairf=min0(npairf,100)
npairp=min0(npairp,100)
nflow=2*npairf
npw2=2*npairp
read (1,2000) (flowfh(i), i=1,nflow2)
read (1,2000) (powfrh(i), i=1,npw2)
flowfh(1)=one
flowfh(2)=zero
powfrh(1)=one
powfrh(2)=zero
write(2,2010) npairf
write(2,2020) (flowfh(i), i=1,nflow2)
write(2,2030) npairp
write(2,2020) (powfrh(i), i=1,npw2)

*** initial thermodynamic state of liquid in plena
*** and fractional temperature (or enthalpy) histories
*** in upper and lower plena

goto (880,890) iltabs
880 continue
read (1,2000) tupper,tdown
hupper=ental(pinit,tupper,111q)
hdown=ental(pinit,tdown,111q)
goto 900
890 continue
read (1,2000) hupper,hdowm
upper=temper(pinit,hupper,111q)
tdown=temper(pinit,hdowm,111q)
900 continue
rouper+dens(pinit,tupper,hupper,111q)
rodown+dens(pinit,tdown,hdowm,111q)
cupper+sheat(pinit,tupper,hupper,111q)
cpdowm+sheat(pinit,tdown,hdowm,111q)
read (1,1000) npairu,npaird
npairu=max0(npairu,2)
npaird=max0(npaird,2)
npairu=min0(npairu,100)
npaird=min0(npaird,100)
netu2=2*npairu
netd2=2*npaird
read (1,2000) (etuph(i),i=1,netu2)
read (1,2000) (etdpfh(i),i=1,netd2)
etuph(i)*one
etuph(2)*zero
etuph(i)*one
etdpfh(2)*zero
turef=tupper
huref=hupper
tdref=tdown
tudref=tupper
hupold=hupper
tdold=tdown
hdold=hdowm
write(2,2035) pinit,tupper,tdown,hupper,hdowm,
\ rouper,rodown,cupper,cpdowm
write(2,2040) npairu
write(2,2020) (etuph(i),i=1,netu2)
write(2,2045) npaird
write(2,2020) (etdpfh(i),i=1,netd2)
c
***time and time steps data

read (1,1150) time,tmef,dtsted,dttran
time=time0
dt=dtsted
if(lstedy.lt.0) dt=dttran
write(2,2050) time0,tmef,dtsted,dttran,time,dt
do 905 i=1,ichan
time0(i)=time0
ttmold(i)=time0
905 continue
read (1,1000) ntevec,ntrin,nplot,ntdump
ntevec=min0(ntevec,9)
ntrin=min0(ntrin,9)
nplot=min0(nplot,9)
ntdump=min0(ntdump,9)
ntevec=max0(ntevec,1)
ntrin=max0(ntrin,1)
nplot=max0(nplot,1)
ntdump=max0(ntdump,1)
ntex1=ntexec+1
ntpr1=ntprin+1
ntpl1=ntplot+1
ntdu1=ntdump+1
read(1,1150) (texec(n),n=1,ntex1)
read(1,1150) (dtexec(n),n=1,ntexec)
read(1,1150) (tprin(n),n=1,ntpr1)
read(1,1150) (dtprin(n),n=1,ntprin)
read(1,1150) (tplot(n),n=1,ntpl1)
read(1,1150) (dtplot(n),n=1,ntplot)
read(1,1150) (tdump(n),n=1,ntdu1)
read(1,1150) (dtdump(n),n=1,ntdump)
texec1=time0
tprin1=time0
tplot1=time0
tdump1=time0
texec(ntex1)=time0
tprin(ntex1)=time0
tplot(ntpl1)=time0
tdump(ntdu1)=time0
write(2,2100) (texec(n),n=1,ntex1)
write(2,2100) (dtexec(n),n=1,ntexec)
write(2,2100) (tprin(n),n=1,ntpr1)
write(2,2200) (dtprin(n),n=1,ntprin)
write(2,2100) (tplot(n),n=1,ntpl1)
write(2,2250) (dtplot(n),n=1,ntplot)
write(2,2100) (tdump(n),n=1,ntdu1)
write(2,2300) (dtdump(n),n=1,ntdump)
return
c****print errors
c910 continue
write(2,9100) 1,nreg(i),nregmx
stop
920 continue
write(2,9200) 1,klast(i),kaxmax
stop
930 continue
write(2,9300) 1,n,jmax(1,n),jconnx
stop
940 continue
write(2,9400) 1,z(i,kk),chieng(i)
stop
c1000 format(4(3f18.14,1x),/)
1050 format(/5x,"general model data. /5x,18("--",/)
1100 format(1x,1f10.0,3x,1f10.0,1x,"plenu=".1f10.0,1x,"plend=".1f10.0,1x,
& 1x,"iplenu=".1f10.0,1x,"iplend=".1f10.0,1x,
& 1x,"i1 fluid=".1f10.0,1x,"i1 momenr=".1f10.0,1x,"i1 energ=".1f10.0,1x,
& /10x,"i1 condu=".1f10.0,1x,"i1 conl=".1f10.0,1x,"i1 conr=".1f10.0,1x,
& 1x,"i1 htrr=".1f10.0,1x,"i1 htrf=".1f10.0,1x,"i1 form=".1f10.0,1x,
& 1x,"i1 etopt=".1f10.0,1x,"i1 tolenc=".1f10.0,1x,"i1 leamn=".1f10.0,1x,
& /10x,"i2 condw=".1f10.0,1x,"i2 rest=".1f10.0,1x,"i2 units=".1f10.0,1x,
& 1x,"i2 pfas=".1f10.0,1x,"i2 kpas=".1f10.0,1x,"i2 jpas=".1f10.0,1x,
& /10x,"i2 term1=".1f10.0,1x,"i2 term2=".1f10.0,1x,"i2 trpr1=".1f10.0,1x,
& /10x,"i2 trpr2=".1f10.0,1x,"i2 trtr=".1f10.0,1x,
& 1x,"i2 step=".1f10.0,1x,"i2 letabs=".1f10.0,1x,
1150 format(7d10.3)
1200 format(/5x,"general constants. /5x,17("--",/),
This is the most complete text of the thesis available. The following page(s) were not included in the copy of the thesis deposited in the Institute Archives by the author:
& /5x,"initial temperature",
& /5x,"initial enthalpy",
& /5x,"initial specific heat",
& /5x,"reference temperature",
& /5x,"reference enthalpy",
& /5x,"initial htc at left hand side",
& /5x,"initial htc at right hand side",
& /5x,"initial density",

1870 format(/5x,"initial state of the fluid in channel i = ",
& i3," at each axial level k",
& /5x,66("-"),/2x,"axial",4x,"pressure",2x,
& "temperat",2x,"enthalpy",3x,"density",2x,
& "sp. heat",2x,"ext. temp.",1x,"ext. dens.",1x,"h.t.c. lef",1x,
& "h.t.c. rig",1x,"h. flux 1",1x,"h. flux r",
& /2x,"level",5x,*(n/m2),4x,
& "(deg k)",3x,*(j/kg),4x,"(kg/m3)",2x,
& "(j/kg/k)",3x,"(deg k)",3x,"(kg/m3)",2x,"(w/m2/k)",2x,
& "(w/m2/k)",3x,"(w/m2)",4x,"(w/m2)/
"

1880 format(/5x,"initial state of the fluid in channel i = ",
& i3," at each axial level k",
& /5x,66("-"),/2x,"axial",4x,"pressure",2x,
& "temperat",2x,"enthalpy",3x,"density",2x,
& "sp. heat",2x,"ext. temp.",1x,"ext. dens.",1x,"h.t.c. lef",1x,
& "h.t.c. rig",1x,"h. flux 1",1x,"h. flux r",
& /2x,"level",5x,*(psi),5x,
& "(deg f)",2x,"(btu/lbm)",1x,"(1bm/ft3)",1x,
& "btu/lbm/f",2x,"(deg f)",2x,"(1bm/ft3)",1x,"btu/f2/f/h",1x,
& "btu/f2/f/h",1x,"btu/f2/h",1x,"btu/f2/h/"
"

1900 format(15,5x,11(ipd10.3))
1920 format(/5x,"other initial state and operational data for channel"
& 1",/5x,54("-"),
& /2x,"channel",2x,"flow rate",1x,"dp upper",
& 2x,"dt upper",2x,"dt lower",2x,"dh upper",
& 2x,"dh lower",2x,"grav=cost",1x,"time valv",1x,"valve st",
& /2x,"number",4x,"(kg/s)",6x,"(-)",4x,"(deg k)",3x,
& "(deg k)",4x,"(j/kg)",4x,"(j/kg)",4x,"(m/s2)",
& 5x,"(s)",7x,"(-)"

1930 format(/5x,"other initial state and operational data for channel"
& 1",/5x,54("-"),
& /2x,"channel",2x,"flow rate",1x,"dp upper",
& 2x,"dt upper",2x,"dt lower",2x,"dh upper",
& 2x,"dh lower",2x,"grav=cost",1x,"time valv",1x,"valve st",
& /2x,"number",4x,"(1bm/s)",5x,"(-)",4x,"(deg f)",3x,
& "(deg f)",3x,"(btu/lbm)",1x,"(btu/lbm)",2x,"(ft/s2)",
& 4x,"(s)",7x,"(-)"

1940 format(15,5x,8(ipd10.3),15)
1960 format(/5x,"power distribution and integration methods for channel"
& 1",/5x,54("-"),
& /2x,"reg",3x,"ext. heat int. heat integr",
& /2x,"no.",4x,"(w/m)",5x,"(w/m3)",5x,met,
"

1970 format(5x,"power distribution and integration methods for channel"
& 1",/5x,54("-"),
& /1x,"reg",2x,"ext. heat int. heat integr",
& /1x,"no.",1x,"(btu/ft3/h)",1x,"btu/ft3/h",3x,met,
"

1980 format(15,5x,2(ipd10.3),15)
1990 format(/5x,"the initial temperature in all solid-structures is"
& taken to be t(i,k,j)="ipd10.3)
2000 format(8d10.3)
2010 format(/5x,"fractional flow rate in return line versus"
& t*e.,/5x,46("-"),/5x,"(number of pair of points in the table"
& is npairf="*,13,*",//5(1x,"flow fr.*".4x,"time*.3x),/)
2020 format(10(lpd10.3))
2030 format(//5x,"fractional power in heated rods versus time",
& /5x,44("".),/5x,"(number of pair of points in the table is",
& npairf="*,13,*",//5(1x,"power fr.*".3x,"time*.3x),/)
2035 format(//5x,"initial properties of liquid in plena",
& /5x,37("".),/)
2040 format(//5x,"fractional temperature (or enthalpy)",
& " in upper plenum versus time",
& /5x,64("".),/5x,"(number of pairs of points in the",
& " table is npairu="*,13,*")",
& //5(lx,"etup fr.*".4x,"time*.3x),/)
2045 format(//5x,"fractional temperature (or enthalpy)",
& " in lower plenum versus time",
& /5x,64("".),/5x,"(number of pair of points in the",
& " table is npaird="*,13,*")",
& //5(lx,"etow fr.*".3x,"time*.3x),/)
2050 format(//5x,"time and time step data",/5x,23("".),
& /5x,"initial time 
& /5x,"final time 
& /5x,"delta t steady 
& /5x,"delta t transient (initial)",
& /5x,"actual time 
& /5x,"actual delta t",
& /5x,10(lpd10.3))
2100 format(//5x,"time interval",/6x,10(lpd10.3))
2150 format(5x,"delta t execution",/7x,9(lpd10.3))
2200 format(5x,"delta t print 
& /7x,9(lpd10.3))
2250 format(5x,"delta t plot 
& /7x,9(lpd10.3))
2300 format(5x,"delta t dump 
& /7x,9(lpd10.3))
9100 format(//5x,"*****ERROR IN INPUT DATA*****",
& in channel 1="*,12,*", number pf axial
& regions input is nreg(1)="*,12,*", when maximum
& number of regions accepted is only nregmx="*,12,*")
9200 format(//5x,"*****ERROR IN INPUT DATA*****",
& in channel 1="*,12,*", number of axial mesh points
& is kmax(l)="*,12,*", wwhen maximum number of axial mesh
& points accepted is only kaxmax="*,12,*")
9300 format(//5x,"*****ERROR IN INPUT DATA*****",
& in channel 1="*,12,*", and region n="*,12,*", number
& of radial mesh points is jmax(1,n)="*,12,*", when maximum
& number of radial mesh points accepted is only jconmx="*,
& 12,*")
9400 format(//5x,"*****ERROR IN INPUT DATA*****",
& in channel 1="*,12,*", the sum of region heights z(l,kk)="*,
& d10.3,"is not equal with channel length chleng(1)="*,
& d10.3,"/)
c end

subroutine energiz, tem, ent, ro, cp, q2left, q1heat, kmin, kmax,
& rtube, afld, nreg, klast, w, wold, deltu, deltd, delhu, delhd,
& r=2lfl., i, ican, kmax, nregmx)
implicit real*8(a-h,o-z)

****solution of energy equation for channel l, using different
****eulerian numerical methods.  sk, feb84.

dimension z(ichan,kaxmax),tem(ichan,kaxmax), ent(ichan,kaxmax),
 & ro(ichan,kaxmax), cp(ichan,kaxmax), q2left(ichan,kaxmax),
 & q1heat(ichan,nregmx), km(ichan,nregmx), kmax(ichan,nregmx)

dimension rtube(ichan), aflow(ichan), nreg(ichan), klast(ichan),
 & w(ichan), wold(ichan), deltu(ichan), deltd(ichan),
 & delhu(ichan), delhd(ichan), dq2tfl(kaxmax)

common/logkey/istedy, lplenu, lplend, lfluid, lmoment, lenerg, lcondu,
 & lcon1, lconr, lhtr,r, lhtr_l, llir, llmq, lform, letopt, letabs,
 & lmeb, ldtcon, lrest, lunits, lpas, kpas, jpas, ltermx, ltpfr1, ltpfr2,
 & lstep

c
common/times/time0, timef, dttstd, dtttran, time, dt

c
common/geomin/volu, void, duv, tupper, hupper, tdow, hdow, rouper,
 & tamb, cpxt, volamb, tsolid, wretin, pinit, tininit, eininit, cplinit,
 & tref, reref, hcov1, hcovr, roinit, wret, wrold, tvalvr, tvrdo,
 & lvalvr, lvrd, tupold, hupold, tdold, hdold, rodow

c
common/hystor/flowrh(200), powfrh(200), npairf, npairrp

c
common/toler/ tolw, tolh, tolx, tolto, tolpc, tolpl,itol, itolen,
 & ltolw, ltolh, ltolx, ltolpl, ltolpc, ltolpl

c
data zero, one, two, thre, four, twopi/0.0d0, 1.0d0, 2.0d0, 3.0d0, 4.0d0,
 & 6.2831853907d0/

c
if(lenerg.lt.1) return

c
ltol=0
ltolh=0
kk=klast(1)
nmax=nreg(1)

d therm=twopi*rtube(i)/aflow(1)
timold=time-dt

if(timold.lt.zero) timold=zero

powfr=polate(powfrh, timold, npairf, memorp)

powfrh=polate(powfrh, time, npairrp, memorp)

goto(30, 30, 30, 30, 40, 40, 50, 40, 50, 50, 50, 50, 30, 30, 30, 30, 30, 30)

30 flowr=w(i)
goto 60

40 flowr=wold(1)
goto 60

50 flowr=(w(i)+wold(1))/two

60 continue

****if the flow in channel l is upward oriented, the energy equation
****is integrated from k*1 to k*kk, using lower plenum b.c.'s and
****vice versa

do 500 nn=1,nmax

efflow) 70,80,80

continue
n=nmax-nn+1

70 goto 90
80  continue
n=nn
90  continue
  k1=kmin(1,n)+1
  k2=kmax(1,n)
  source=q1heat(1,n)*powfro
  sourcn=q1heat(1,n)*powfrn
do 450  ki=k1,k2
    if(flowr) 100,110,110
100  continue
  k=k1+k2-k1-1
  m=k+1
  goto 120
110  continue
  k=k1
  m=k-1
120  continue

**preparation of the coefficients in energy equation and
 setup boundary conditions**

```c
expr1=geomf*dt*(dq2tf1(k)/rho(1,k)+dq2tf1(m)/rho(1,m)
  & /cp(1,m))/two
expr2=two/sflow(1)/(rho(1,k)+rho(1,m))
if (letabs=1) 130,130.250
```

**set up b.c. and solve the energy equation using the temp or
 enthalpy as primary dependent variable if letabs=1, or letabs=2.**

```c
130  continue
  expr3=expr2*two/(cp(1,k)+cp(1,m))
  expr4=geomf*(q2left(1,k)/rho(1,k)+q2left(1,m)/rho(1,m)
  & /cp(1,m))/two
    if(flowr) 180,200,200
180  continue
    if(k-kk+1) 220,190,190
190  continue
    etim=tem(1,kk)
    tem(1,kk)=tupper+deltu(1)
    goto 220
200  continue
    if(k-2) 210,210,220
210  continue
    etim=tem(1,1)
    tem(1,1)=tdown+deltd(1)
220  continue
    etik=tem(1,k)
    etimn=tem(1,m)
    goto 320
```

```c
250  continue
  expr3=expr2
  expr4=geomf*(q2left(1,k)/rho(1,k)+q2left(1,m)/rho(1,m))/two
    if(flowr) 270,290,290
270  continue
    if(k-kk+1) 310,280,280
280  continue
    etim=ent(1,kk)
```
ent(1,kk)=hupper+delhu(1)
goto 310
290 continue
if(k-2) 300,300,310
300 continue
s.etim=ent(i,i,1)
ent(i,i)=hdown+delhd(i)
310 continue
etik=ent(i,k)
etimn=ent(i,m)
c
c
320 continue
setold=etik+etim
detold=etik+etim
expr2=expr2/(one+expr1)
expr3=expr3/(one+expr1)
expr4=expr4/(one+expr1)
c
c
***solution of energy equation using various numerical schemes

c
goto(330,340,350,360,370,380,360,370,380,380,380), lenerg
330 continue
c
c
***box scheme
c
source=(sourco+sourcn)/two
expr1=expr2/(z(1,k)-z(1,m))*dt
expr2=expr1*w(1)
expr1=expr1+weul(i)*detold
expr3=(expr3+source-expr4)*dt*two
etikn=(setold+expr3+etimn*(expr2-one)-expr1)/(one+expr2)
goto 400
340 continue
c
c
***turner scheme
c
source=sourcn
expr1=expr2/(z(1,k)-z(1,m))*dt
expr2=expr1*w(1)*two
expr3=(expr3+source-expr4)*dt*two
etikn=(setold+expr3+etimn*(expr2-one))/(one+expr2)
goto 400
350 continue
c
c
***modified turner (full implicit donor cell) scheme
c
source=sourcn
expr1=expr2/(z(1,k)-z(1,m))*dt
expr2=expr1*w(1)
expr3=(expr3+source-expr4)*dt
etikn=(etik+expr3+etimn*expr2)/(one+expr2)
goto 400
360 continue
c
c
***full explicit donor cell
c
source=(sourco+sourcn)/two
expr1=expr2/(z(1,k)-z(1,m))*dt
expr1=expr1*wold(i)*detold
expr3=(expr3*source-expr4)*dt
etkn=etkn+expr3-expr1
goto 400
370 continue

c***donor cell scheme with mass flow rate at n+1/2
	source=(source+source)/two
expr1=expr2/(z(1,k)-z(1,m))*dt
expr1=expr1*wl(i)*detold
expr3=(expr3*source-expr4)*dt
etkn=etkn+expr3-expr1
goto 400
380 continue

c***semi-implicit donor cell scheme (mass flow rate in the convection
c***term at new time step)
	source=(source+source)/two
expr1=expr2/(z(1,k)-z(1,m))*dt
expr1=expr1*wl(i)*detold
expr3=(expr3*source-expr4)*dt
etkn=etkn+expr3-expr1

c
400 continue

c

C***set up tem(1,k)=etkn, or ent(1,k)=etkn', if the energy eqn has
C***been solved for temperature or enthalpy, respectively.

if (etabs-1) 410, 410, 430

410 continue

if (itolen, le 0) goto 420
if (dabs(etkn-entkn), gt tolt*etkn) litol=1

420 continue
etlm=tem(1,k)
entlm=entlm(1,k)
etkn=etkn
430 continue

if (itolen, le 0) goto 440
if (dabs(etkn-entkn), gt tolh*etkn) litolh=1

440 continue
etlm=ent(1,k)
entlm=ent(1,k)
etkn=etkn

C***continue with the next lower (if w(1).lt.0), or higher
C***if w(1).ge .0) axial level k of the axial region n.

C
450 continue

c
C***continue with the next lower (if w(1).lt.0), or higher
C***if w(1).ge .0) axial region n.

500 continue

return
end

subroutine enerlt(z, tem, ent, ro, cp, q2left, q1heat, kmin, & kmax, rube, afw, nreg, klkast, w, wold, z0, t0, etlm0, ttime, & delu, delu, delh, delh, dql2fl, vtime, qtime, qtime, & entlm, etlm, gtime, nptime, l, ichan, kaxmax, nregmx, & ntimeh, ntimh2)
implicit real*8(a-h,o-z)

**solution of transient energy equation using the method of characteristics. sk, as, march 84.**

dimension z(iChan,kaxmax), tem(iChan,kaxmax), ent(iChan, & kaxmax), ro(iChan,kaxmax), cp(iChan,kaxmax), q21left(iChan, & kaxmax), z0(iChan,kaxmax), t0(iChan,kaxmax), ettim0(iChan, & kaxmax), q1heat(iChan,nregmx,kmin(iChan,nregmx),kmax & (iChan,nregmx)
dimension rtube(iChan), aflow(iChan), nreg(iChan), ttime0(iChan), & klast(iChan), w(iChan), wold(iChan), deltu(iChan), & delt(iChan), deldh(iChan), dq2tf1(kaxmax)

dimension vtime(iChan,kaxmax,ntimeh), qatime(iChan, & kaxmax,ntimeh), qbtimel(iChan,kaxmax,ntimeh), gtime & (iChan,ntimh2), etutim(iChan,ntimeh), ettim(iChan, & ntimeh), nptime(iChan)

dimension dummyv(4)

common/logkey/lstedy,lplenu,lplend,lf, li, lmlen,
& laneg, lcond, lcon, lntr, lntr1, lntr2, lllq,
& lmonfl, lform, liopt, letabs, lambd, ldtcon, lrest,
& lunits, lphas, lphas, ltermx, ltrpl, ltrpl2, lstep

common/times/time0, timef, dtsted, dttran, time, dt

common/geoin/volu, void, azu, topp, hupper, tdow, hdown, rouper,
& tmb, cpex, ola, tsoill, wretin, pinit, tinit, einit, cplnit,
& tref, eref, hconv, hconvr, roinit, wret, wreol, tvalr, tvr,
& tvld, tvrd, tupdold, hupdold, tdold, hdold, rodown

common/hystor/flowfrh(200), powfrh(200), npairf, npairp

common/timelh/texec(10), dtrexec(9), tprin(10), tplot(10),
& dplot(10), tdump(10), tdump(9), tprin(200), tplot(200), tdump(200),
& ntrexec, nprin, nplot, ntoexec, nexec2, nexec1, nprin, nplot, nplotf,
& nplot, ndump2, ndump1

data zero, one, two, three, four, twopl, eps
& /0.0, 1.0, 2.0, 3.0, 4.0, 5.0, 6.283185307d0, 1.d-6/
l00h=0
l01t=0

***check for stagnation or flow reversal

time0=ttimel0(i)
if(abs(w(i))<eps) 120, 140, 140
continue

time0=time

**time0(i)=time0

***stagnation conditions in channel i.

call enerls(z, tem, ent, ro, cp, q21left, q1heat, kmin, kmax,
& rtube, aflow, nreg, klast, w, z0, t0, ettim0, deltu, delt, & delh, deldh, dq2tf1, vtime, qatime, qbtimel, etutim, ettim,
& gtime, nptime, l, iChan, kaxmax, nregmx, ntimeh, notimh)
return

continue
dtn=dt
if(w(i)*wold(i)) 150, 170, 170
continue
c

*** stagnation conditions in channel i have occurred in the
*** interval (time-dt,time). the time of stagnation is obtained
*** by a linear interpolation in time of the old and new values
*** of flow rates. this time is redefined now as time0 and
*** and wold(i)=w(i,time0)*0.
c

timold=time-dt
dw=w(i)-wold(i)
time0=(w(i)+timold-wold(i)*time)/dw
ttime0(i)=time0
wold(i)=zero
dt=time0-timold
call enerls(z,tem,ent,ro,cp,q2left,qtheat,kmin,kmax,
& rtube1,aflow,nreg,klast,wold,z0,t0,etstim,deltu,deltd,
& delhu,delhd,q2tfl,vtime,qatime,qotime,etutim,etltime,
& gtime,ntime,i,ichan,kaxmax,nregmx,ntimeh,ntimh2)
dt=time-time0
continue

kk=klast(i)
rmx=nreg(i)
if(ntime(i).lt.ntimeh) goto 200
continue

*** if(nptime.eq.ntimeh) move the last nptime-1 arrays down
*** by one unit, keep the value of arrays at nptime=1 (time=
*** time0) and fill the last cells by the values of arrays
*** at the actual time (ntime=ntimeh)
c

nptf=ntimeh-1
do 190 npt=2,nptf
ntime2=2+npt
ntime=ntime2-1
gtime(i,ntime1)=gtime(i,ntime1+2)
gtime(i,ntime2)=gtime(i,ntime2+2)
etutim(i,npt)=etutim(i,npt+1)
etltime(i,npt)=etltime(i,npt+1)
do 190 k=1,kk
vtime(i,k,npt)=vtime(i,k,npt+1)
qatime(i,k,npt)=qatime(i,k,npt+1)
qotime(i,k,npt)=qotime(i,k,npt+1)
continue
nptime(i)=nptf
continue
doomf=twopi*rtube1(i)/aflow(i)
nptime(i)=nptime(i)+1
ntime2=2+nptime(i)
ntime=ntime2-1
gtime(i,ntime1)=w(i)/aflow(i)
gtime(i,ntime2)=time
npt=nptime(i)
gtime=gtime(i,ntime1)
pftime=poline(powfrn.time,npairp,memorp)
memop=memop
continue
c**prepare boundary conditions for next time steps
   if (w(i)) 220,120,240
220 continue
   c***down flow
   mm=1
   ma=0
   mb=1
   ka=kk
   goto 260
240 continue
   c***up flow
   mm=-1
   ma=-1
   mb=0
   ka=1
260 continue
   t0(i,ka)=time
   expr1=one/ro(i,ka)/(one+geom*dt+dq2tfl(ka)/
      & ro(i,ka)/cp(1,ka))
   expr3=expr1
   expr4=expr1*q2left(1,ka)
   if (etabs(i)) 270,270,280
270 continue
   expr3=expr3/cp(1,ka)
   expr4=expr4/cp(1,ka)
   etltim(1,npt)=tupper+delu(i)
   etltim(1,npt)=tdown+deltd(i)
   tem(1,ka)=etltim(1,npt)
   if (mm.eq.1) tem(1,ka)=etltim(1,npt)
   goto 290
280 continue
   etltim(1,npt)=hupper+delhu(i)
   etltim(1,npt)=hdown+delhd(i)
   ent(1,ka)=etltim(1,npt)
   if (mm.eq.1) ent(1,ka)=etltim(1,npt)
290 continue
   vtime(i,ka,npt)=expr1
   qtime(i,ka,npt)=expr3
   qbtime(i,ka,npt)=expr4
   ztim=z(i,ka)

C c***at the current time level, for each axial level k=2,3,...
c***,kk-1,kk, if the flow is up, or k=kk-1,kk-2,...,3,2,1 if
c***the flow is down, first find the characteristic trajectory
c***that begins at the point k at the current time and then
c***integrate the energy eqn along this characteristic.
C
c c0 2000 nn=1,nmax
   if (w(i)) 310,120,220
310 continue
   n=nmax-nn+1
   goto 330
320 continue
   n=nn
330 continue
   k1=kmin(1,n)+1
   k2=kmax(1,n)
   do 1900 k1=k1,k2
      rint2=zero
rint2o=zero
tup=time
delt=dt
tlow=tup-delt
nlim=n
gtup=gttime
pfrup=pftime
memorp=memorp
npto=npt
ksum=1
if(w(i)) 350,120,360
350 continue
k=k+k2-ki-f
goto 370
360 continue
k=ki
370 continue
klim=k
expr1=one/ro(i,k)/(one+geom*dt*dq2tf1(k)/ro(i,k)/cp(i,k))
expr3=expr1
expr4=expr1*q2left(i,k)
if(isabs-1) 380,380,390
380 continue
expr3=expr3/cp(i,k)
expr4=expr4/cp(i,k)
390 continue
vtim(i,k,npt)=expr1
qatime(i,k,npt)=expr3
qbtime(i,k,npt)=expr4
zup(z(i,k))
z0(i,k)=zup
t(i,k)=time0
imat=1
420 continue
c
***for each axial level go back in time on the trajectory
***until it intersects either the t or z axis

imat=imat
zup=zup
npto=npto-1
q1z=zero
tlow=dsmax1(tlow,time0)
delt=tup-tlow
pfrlow=polate(powfrh,tlow,npairp,memorp)
if(tlow-gttime(1,4)) 440,460,460
440 continue
c
***if time0<gttime(1,2).le.tlow.it.gttime(1,4) use a linear interpolation
***polation in time to evaluate gtlow
c
do 450 ln=1,4
dummyy(ln)+gttime(1,ln)
gtlow=polate(dummyy,tlow,2,memtim)
goto 480
460 continue
ntime=2*npto-1
gtlow=gttime(1,ntime)
c
480 continue
expr10=expr1
expr30=expr3
expr40=expr4
exprlt=(gtup+gtlow)*delt/two
if((lenerg.eq.8 .or. lenerg.eq.10).and.(lmet.eq.1))
& exprs=(gtup0+four*gtup+gtlow)*delt/thre
if(w(i)) 490, 120, 500
490 continue

C***down flow. when go back in time decrease k.
C
nlow=nlim
nup=nmax
goto 510
500 continue
C
C***up flow. when go back in time, decrease k.
C
nlow=1
nup=nlim
510 continue
do 800 nnt=nlow,nup
if(w(i)) 520, 120, 530
520 continue
nt=nnt
klow=kmin(1,nt)+1
if(nt.eq.nlim) klow=klim+1
kup=kmax(1,nt)
kb=kup
530 continue
nt=nlow+nup-nt
klow=kmin(1,nt)+1
kup=kmax(1,nt)
if(nt.eq.nlim) kup=klim
kb=klow-1
540 continue
nlim=nt
do 750 kkt=klow,kup
if(w(i)) 550, 120, 560
550 continue
kt=kkt
goto 570
560 continue
kt=klow+kup-kkt-1
570 continue
klim=kt-mm

C****four options are used in the solution of the energy
C****equation by met. of char.:
C****lenerg =7--use trapezoidal rule for integration in time
C**** and take into account the non-linear effects
C**** (i.e., the change in space and time of ro,cp,q2,
C**** etc.)
C****lenerg =8--use simpson's rule for integration in time and
C**** take into account the non-linear effects
C****lenerg =9--use trapezoidal rule and neglect nonlinearities
C**** (ro,cp,q2, etc., are the values at point k, time t)
C****lenerg =10--use simpson's rule and neglect non-linearities
C
if(energ-8) 580, 580, 620
continue
ksum=ksum+1
if(tup-gtime(1,4)) 590, 600, 600
continue
dummyv(1)=vtime(1,kt,1)
dummyv(3)=vtime(1,kt,2)
dummyv(2)=gtime(1,2)
dummyv(4)=gtime(1,4)
x expr10=polate(dummyv, tup, 2, memtim)
dummyv(1)=qtime(1,kt,1)
dummyv(3)=qtime(1,kt,2)
ex pr30=polate(dummyv, tup, 2, memtim)
dummyv(1)=qbtim e(1,kt,1)
dummyv(3)=qbtim e(1,kt,2)
ex pr40=polate(dummyv, tup, 2, memtim)
goto 610
600 continue
expr10=vtime (1,kt,npt0+1)
ex pr30=qtime(1,kt,npt0+1)
ex pr40=qbtim e(1,kt,npt0+1)
610 continue
expr1=expr1+expr10
expr2=expr2+expr30
expr4=expr4+expr40
c
dfik=one/dble(ksum)
expr10=expr1*dfik
expr30=expr3*dfik
expr40=expr4*dfik
620 continue
c
***find the trajectory in the time interval (time, time-delt). 
c****rint1 defines integral (val(t)), from t*time+delt to t* 
c****time.
if(energ.eq.8 .or. energ.eq.10) goto 630
c
*** if energ=7, or energ=9, use trapezoidal rule for integration 
c**** in time.
rint1=expr1+expr10
zin=zup-rint1
goto 680
630 continue
c
*** if energ=8, or energ=10, use simpson’s rule for integration 
c**** in time as follows: for the first time interval use the trap-
c****pezoidal rule (rint1=(v(t)+v(t-delt))*delt/2). If this traj 
c**** did not intersect either the t or z axis, and if delt does not 
c**** change in the next time interval use simpson’s rule-- 
c****(rint1=(v(t)+4*v(t-delt)+v(t-3*delt))*delt/3). next time 
c****interval use the trap rule again-- 
c****(rint1=rint1+(v(t+2*delt)+v(t-3*delt))*delt/2).
c****next time interval, if it is possible, use simpson’s rule 
c****(rint1=rint1+(v(t+2*delt)+4*(t+3*delt)+v(t+4*delt))*delt/3), etc

c
if(time) 640, 640, 650
640 continue
zupper=zup
delt0=delt
expr1a=expr10
gtup0=gtup
rint1=expr1t*expr10
zin=zup-rint1
goto 680
continue
if(dabs(delt-delt0).ie.eq.) goto 660
lmet=1.0
lmet=1.0
goto 640
continue
expr1a=(expr10+expr1a)/two
rint1=expr1s*expr1a
zin=zupper-rint1
continue

C***check if the trajectory intersected the time line, time-delt
C***in the spatial domain at kt. If not, decrease kt for up flow,
C***or increase kt for down flow.
C
if(w(i)) 700,120,710
continue
if(z(i,kt)-zin) 750,900,900
continue
if(z(i,kt)-zin) 900,900,750
C
750 continue

C***the trajectory did not intersect the line time-delt in
C***the axial region nt. Before decreasing the axial region,
C***compute the heat rate added to the fluid in the axial region
C***nt (*dprime*dz) and then continue with the next nt.
C
delz=zup1-z(i,kb)
zup1=z(i,kb)
qiz=qiz+q1heat(i,nt)*delz
C
800 continue

C***the trajectory intersects the line (z-zlim) axis.
C***in this case f(t,z)=f(t0(t,z),zlim)+int(q)*dt, where
C***t0 is the time when the characteristic intersects the
C***time axis. The integral int(q) is carried out between
C***the limits t0 and t.
C
delz=zup-zlim
qiz=qiz/delz
C
C***compute the time when the characteristic intersects the
C***t axis using a linear interpolation in the t-z plane.
C
rint1=zup-zlim
t0(i,k)=tlow*delt*(one-delz/rint1)
delt=tup-t0(i,k)
z0(i,k)=zlim
C
C***compute the integral (q)=integral(q1*expr3/aflow-expr4*
C***geomf) over the actual time interval and then add this
C***integral to its value of previous time intervals.
C
C***if the trajectory intersects the z-zlim axis, the trap
c****rule must be used to evaluate the integral (q) in the
  c****last interval.
  c
  expr3=q1z*expr30/aflow(1)
  expr4=expr40*geomf
  pfrlow=pole(t,powfrh,t0(i,k),npairp,memorp)
  rint2=((pfrup+pfrlow)*expr3/two*expr4)*delt+rint2
  c
  c****set-up boundary conditions, ie, f(t0,zlim)
  c
  if ((tup-eps).gt.gtime(1,4)) goto 820
  c
  c****gtime(1,2).lt.tup.le.gtime(1,4)
  c****gtime(1,2).le.tlow.lt.gtime(1,4)
  c
  dummyv(2)=gtime(1,2)
  dummyv(4)=gtime(1,4)
  goto 830
  c
  820 continue
  c
  c****tup.gt.gtime(1,4)
  c****tlow.ge.gtime(1,4)
  dummyv(2)=tlow
  dummyv(4)=tup
  830 continue
  c
  if (w(1)) 840,120,850
  c
  840 continue
  dummyv(1)=etutim(1,npto)
  dummyv(3)=etutim(1,npto+1)
  goto 860
  c
  850 continue
  dummyv(1)=etutim(1,npto)
  dummyv(3)=etutim(1,npto+1)
  etbc=pole(dummyv,t0(i,k),2,memtim)
  c
  c****compute f(t,z) and then repeat the computations for the
  c****next k
  c
  etnew=etbc+rint2
  c
  870 continue
  tem(1,k)=etnew
  goto 1900
  c
  880 continue
  ent(1,k)=etnew
  goto 1900
  c
  900 continue
  c
  c****the trajectory intersects the line time-delt into the
  c****axial region nt, between axial levels kt and kt-mm
  c****(mm=1, if w.lt.0, and mm=-1, if w.gt.0)
  c
  delz=zup1-zin
  q1z=q1z+q1hheat(i,nlim)*delz
  delz=zup-zin
  q1z=q1z/delz
  exprj1=q1z*expr30/aflow(1)
expr4=expr4+geonf
if(lenenr.eq.8 .or. lenenr.eq.10) goto 930
C
C***if lenenr=7, or lenenr=9, use the trap rule to compute
C***integral (q) in the time interval (t-delt,t)
C***
C
rint2=((pfrup+pfrlow)*expr3/two-expr4)*delt+rint2
goto 980
930 continue
C
C***if lenenr=8, or lenenr=10, use simpson's rule to
C***compute integral (q) in the time interval (t-2*delt, t).
C**** the rule is used similarly as for computation
C***of trajectory (see explanations after 620 continue)
C
if(imet) 940,940,950
940 continue
delt0=dlr
expr3a=expr3
expr4a=expr4
pfrup=pfrup
rint2=((pfrup+pfrlow)*expr3/two-expr4)*delt+rint2
goto 980
950 continue
if(dabs(delt-delt0).le.eps) goto 960
imet=imet
rint2o=rint2
goto 940
960 continue
expr3a=(expr3+expr3a)/two
expr4a=(expr4+expr4a)/two
rint2=((pfrup+four+pfrup+pfrlow)*expr3a/thre-two
& *expr4)*delt+rint2o
rint2o=rint2
980 continue
C
C***check if the characteristic intersected the z (time=time0)
C***axis
C
if(tlow-(time0+eps)) 1100,1100,1000
1000 continue
C
C***the characteristic did not yet intersect either t(z=zl)zlim)
C***or z(time=time0) axis. therefore reduce the time by delt
C***and repeat the calculation
C
ksum=1
if(tlow-gtime(1,4)) 1020,1030,1030
1020 continue
dummyv(1)=ttime(1,klim,1)
dummyv(2)=gtime(1,2)
dummyv(3)=ttime(1,klim,2)
dummyv(4)=gtime(1,4)
expr1=polate(dummyv,tlow,2,memtim)
dummyv(1)=qtime(1,klim,1)
dummyv(3)=qtime(1,klim,2)
expr3=polate(dummyv,tlow,2,memtim)
dummyv(1)=qbtme(1,klim,1)
dummyv(3)=qbtme(1,klim,2)
expr1=polate(dummyv, tlow, 2, memtim)
goto 1050
1030 continue
expr1=vtime(1, klim, npto)
expr3=qatime(1, klim, npto)
expr4=qbtimex(1, klim, npto)
1050 continue
tup=tlow
tfordt=tup-eps
do 1070 nxy=1, ntexec
delt=ntexec(nxy)
if(tfordt.lt.texec(nxy+1)) goto 1080
1070 continue
tlow=tup-delt
zup=z1n
gtup=gtlow
pfup=pf1ow
if(npto.lt.2) npto=2
goto 420
1100 continue
c
*** the characteristic intersects the z(t* time) axis.
*** at the point z0(t, z)=z1n. in this case f(t, z)=f(time0,
***+z0)+int(q), where the integral is represented by int2.
c
t0(1, k)=time0
z0(1, k)=z1n
if(dabs(gtlow)-eps) 1120, 1130, 1130
1120 continue
c
*** stagnant conditions in channel, at time=time0. in this
*** case, compute the initial condition for energy equation
*** (atic) using a linear interpolation between the values of
*** enthalpy (or temperature) in the adjacent neighbor mesh
c*** points to the point z1n=z0(1, k)
c
dummyv(1)=ettim0(1, klim+ma)
dummyv(2)*z(1, klim+ma)
dummyv(3)=ettim0(1, klim+mb)
dummyv(4)=z(1, klim+mb)
dummyv(4)=z(1, klim+mb)
etic=polate(dummyv, z1n, 2, memtim)
goto 1300
1130 continue
c
*** solve the steady state energy equation from the channel
*** entrance to the point where the trajectory intersects
*** the z(t* time) axis. the solution will be used as
*** the initial condition for the transient energy equation.
c
ksum=1
qiz=zero
if(gtlow) 1150, 1120, 1160
1150 continue
etb=etutim(1, 1)
expr1=vtime(1, kk, 1)
expr3=qatime(1, kk, 1)
expr4=qbtimex(1, kk, 1)
ntlow=nlime
num=rmax
\begin{verbatim}
   delzt=\textstyle z(1,kk)-z_n
   \textbf{\textbf{goto}} 1170
1160 \textbf{\textbf{continue}}
   \textbf{\textbf{etbc=etbc+tim(1,1)}}
   expr1=vtim(1,1,1)
   expr3=qatim(1,1,1)
   expr4=qbtim(1,1,1)
   nlow=1
   nup=nlim
   delzt=z_n-z(1,1)
1170 \textbf{\textbf{continue}}
do 1250 nnt=nlow,nup
   \textbf{\textbf{if}}(gltlow) 1180,1120,1190
1180 \textbf{\textbf{continue}}
   ntt=nlow+nup-nnt
   klow=kmin(1,ntt)+1
   if(ntt.eq.nlim) klow=klim+1
   kup=kmax(1,ntt)
   delzt=z(1,kup)-z(1,klow-1)
   if(ntt.eq.nlim) delzt=z(1,kup)-z(1,klow-1)
   q1z=qi+z+q\textbf{\textbf{heat}}(1,ntt)*delzt
   \textbf{\textbf{goto}} 1200
1190 \textbf{\textbf{continue}}
   ntt=nnt
   klow=kmin(1,ntt)+1
   kup=kmax(1,ntt)
   if(ntt.eq.nlim) kup=klim
   delzt=z(1,kup)-z(1,klow-1)
   if(ntt.eq.nlim) delzt=z(1,kup)-z(1,klow-1)
   q1z=qi+z+q\textbf{\textbf{heat}}(1,ntt)*delzt
1200 \textbf{\textbf{continue}}
do 1250 kkt=klow,kup
   \textbf{\textbf{if}}(gltlow) 1210,1120,1220
1210 \textbf{\textbf{continue}}
   kt=klow+kup-kkt-1
   \textbf{\textbf{goto}} 1230
1220 \textbf{\textbf{continue}}
   kt=kkt
1230 \textbf{\textbf{continue}}
   ksum=ksum+1
   expr1=expr1+vtim(1,kt,1)
   expr3=expr3+qatim(1,kt,1)
   expr4=expr4+qbtim(1,kt,1)
1250 \textbf{\textbf{continue}}
   q1z=qi+z+delzt
   df1k*one/\textbf{\textbf{dble}}(ksum)
   expr1=expr1*df1k
   expr3=expr3*df1k/aflow(1)
   expr4=expr4*df1k/geom
   vaveg=expr1*gltlow
   qaveg=q1z+pflow*expr3*expr4
   etic=qaveg/dabs(vaveg)*delzt+etbc
1300 \textbf{\textbf{continue}}
   etnew=etic+rnt2
   \textbf{\textbf{if}}(letabs-t) 1320,1320,1330
1320 \textbf{\textbf{continue}}
   t\textbf{\textbf{em}}(1,kk)=etnew
   \textbf{\textbf{goto}} 1900
1330 \textbf{\textbf{continue}}
   ent(1,kk)=etnew
\end{verbatim}
c
***continue with the next k in the region n

1900 continue

2000 continue

dt=dt0
return
end

subroutine enerls(z,tem,ent,ro,cp,q2left,qiheat,kmin,
& kmax,rtube1,aflow,nreq,klast,w,z0,t0,etttm0,deltu,
& delt,deltu,deltl,dq2tf1,vtime,qtime,quntime,etutim,
& etttim,gtim,e,fntime,ichan,kaxmax,nregmx,ntimeh,
& ntimeh2)

implicit real*8(a-h,o-z)

***solution of steady state or stagnant conditions energy

***equation. sk,march84.

dimension z0(ichan,kaxmax),to(ichan,kaxmax),etttim0
& (ichan,kaxmax),z(ichan,kaxmax),
& tem(ichan,kaxmax),ent(ichan,kaxmax),ro(ichan,kaxmax),
& cp(ichan,kaxmax),q2left(ichan,kaxmax),qiheat(ichan,
& nregmx),kmin(ichan,nregmx),kmax(ichan,nregmx)

dimension rtube1(ichan),aflow(ichan),nreq(ichan),
& klast(ichan),w(ichan),deltu(ichan),deltl(ichan),
& dq2tf1(kaxmax),deltl(dichan),deltl(dichan)

dimension vtime(ichan,kaxmax,ntimeh),qtime(ichan,
& kaxmax,ntimeh),quntime(ichan,kaxmax,ntimeh),gtim
& (ichan,ntimeh2),etttim(ichan,ntimeh),atim(ichan,
& ntimeh),e,fntime(ichan)

c
common/logkey/lstedy,iplenu,iplenu,iplenu,iplenu,iplenu,iplenu,iplenu,iplenu,iplenu,
& lconu,lconr,lconl,ltim,ltim,ltim,ltim,ltim,ltim,ltime,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
& ltime,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,ltim,
nmax=nreg(1)
kk=klast(1)
powfr=polate(powfrh, time0, npairp, memorp)
to(i,1)=time0
to(i,kk)=time0
if(dabs(w(1))-eps) 700, 50, 50
  continue
if(1etabs-1) 60, 60, 70
  continue
etutim(i,1)=tupper+deltu(i)
etutim(i,1)=tdown+deldt(i)
goto 80
  continue
etutim(i,1)=hupper+delu(i)
etutim(i,1)=hdown+deldh(i)
  continue
  c
  c***set up boundary conditions and begin the summation
  c***in v average and q average expressions
  c
  if(w(1)) 120, 700, 150
  c***down flow
  120 continue
  klim=kk
  klim=kk+1
  sum1=one/ro(i,kk)/(one+geomf*dt*dq2tfl(kk)/ro(i,kk)
&/cp(1,kk))
  sum2=sum1
  sum3=q2left(1,kk)*sum1
  sum4=zero
  if(1etabs-1) 130, 130, 140
  continue
  etbc=tupper+deltu(1)
  tem(i,kk)=etbc
  ettim0(i,kk)=etbc
  sum2=sum2/cp(1,kk)
  sum3=sum3/cp(1,kk)
  etutim(i,1)=etbc
  ettim(i,1)=tdown+deldt(1)
goto 200
  continue
  etbc=hupper+delu(1)
  ent(i,kk)=etbc
  ettim0(i,kk)=etbc
  etutim(i,1)=etbc
  ettim(i,1)=hdown+deldh(1)
goto 200
  c***up flow
  150 continue
  klim=1
  klim=0
  sum1=one/ro(i,1)/(one+geomf*dt*dq2tfl(1)/ro(i,1)/
& cp(i,1))
  sum2=sum1
  sum3=q2left(1,1)*sum1
  sum4=zero
  if(1etabs-1) 160, 160, 170
  continue
  etbc=tdown+deldt(1)
  tem(i,1)=etbc
ettim0(1,1)=etbc
sum2=sum3/cp(1,1)
sum3=sum3/cp(1,1)
ettim1(1,1)=tupper+deltu(i)
ettim1(1,1)=etbc
goto 200
170 continue
etbc=hdwn+deldh(i)
et(1,1)=etbc
ettim0(1,1)=etbc
ettim1(1,1)=hupper+deltu(1)
ettim1(1,1)=etbc
c
***compute f(t<0,z=0)=qaveg/vaveg*z0+f(0)
200 continue
vtimex1,klim,1)=sum1
qtime1,klim,1)=sum2
qbtimex1,klim,1)=sum3
do 500 nx=1,nnmax
if(w()) 220,700,230
220 continue
nx=nnmax-nx+1
fot 240
230 continue
nx=nn
240 continue
k1=kmin(1,n)+1
k2=kmax(1,n)
sources=qheatmix(1,n)*powfr
do 500 k1=k1,k2
if(w()) 300,700,310
300 continue
k=k1+k2-k1+1
m=k+1
goto 320
310 continue
k=k1
m=k-1
320 continue
tot(1,k)=time0
z0(1,k)=z(1,k)
expr=one/ro1(1,k)/(one+geom*dt+d2tf1(k)/ro1(2,k)/cp(1,k))
kdif=k-kmin
rdif=abs(kdif)
rkdif=abs(rk0)
sum1=sum1+expr
vtimex1,k,1)=expr
if(lt tabs.1e-1) expr=expr/cp(1,k)
qtime1(k,1)=expr
sum2=sum2+expr
expr=expr*q2left1(1,k)
qbtimex1,k,1)=expr
sum3=sum3+expr
delzkm=z(1,k)-z(1,m)
sum4=sum4+source*delzkm
vaveg=sum4*gradtime1(1,1)/rk0
qaveg=sum4/(z1,k)-z(1,klim))
expr3a=sum2/rk0/aflow(i)
expr4a=sum3/rk0/geom
a=vaveg*qaveg*expr3a-expr4a
etik = qaveg/vaveg*(z(1,k)-z(1,k1m))+etbc
if(letabs-1) 350,350,360
350 continue
  tem(1,k)=etik
goto 370
360 continue
  ent(1,k)=etik
370 continue
  ett1m0(1,k)=etik

*** continue with next axial level

500 continue
  return

*** solution of energy equation for stagnation conditions

700 continue

760 continue
  k1 = kmin(1,n)
goto 780
770 continue
  k1 = kmin(1,n)*1
780 continue
  k2 = kmax(1,n)
  source = q1heat(1,n)*powfr
do 850 k = k1,k2
  to(1,k) = time0
  z0(1,k) = z(1,k)
  expr1 = one+geomf*dt*dq2tfl(k)/ro(1,k)/cp(1,k)
  expr1 = one/ro(1,k)/expr1
  expr3 = expr1
  expr4 = expr1*q2left(1,k)
if(letabs-1) 810,810,820
810 continue
  expr3 = expr3/cp(1,k)
  expr4 = expr4/cp(1,k)
  tem(1,k) = tem(1,k)+dt*(source*expr3/aflow(1)-
  & expr4*geomf)
  ett1m0(1,k) = tem(1,k)
goto 830
820 continue
  ent(1,k) = ent(1,k)+dt*(source*expr3/aflow(1)-
  & expr4*geomf)
  ett1m0(1,k) = ent(1,k)
830 continue
  vtime(1,k,1) = expr1
  qatime(1,k,1) = expr3
  qbtme(1,k,1) = expr4
850 continue

*** continue with next axial region

900 continue
if(letabs-1) 920,920,930
920 continue
etutim(i,1)=tupper+deltu(i)
etutim(i,1)=tdown+deltd(i)
goto 940
930 continue
etutim(i,1)=hupper+deldh(i)
etutim(i,1)=hdown+deldh(i)
940 continue
return
end

subroutine eplena(tem,ent,cp,klast,w,ichan,kaxmax)
c implicit real=8(a-h,o-z)
c c***this routine solves the thermal energy equation for the
c***upper and lower plena. sk,mar84.
c dimension tem(ichan,kaxmax),ent(ichan,kaxmax),cp(ichan,
& kaxmax),klast(ichan),w(ichan)
c common/logkey/lstedy,iplenu,iplend,ifluid,lmomen,lenerg,
& lcondu,lconl,lconr,lhtr,lhtrl,lhtrr,l11q,lmomfl,lform,
& latopt,letabs,leamb,ltdcon,lrest,lunits,lpas,lpas,lpas,
& itermx,lpri,lpri2,istep
c common/timef/time0,timdf,dtsed,dtttran,time,dt
c common/geomin/volu,vold,dzu,tupper,hupper,tdown,hdown,rouper,
& tend,cpext,volamb,tsolid,wretin,pinit,tinit,einit,cpinit,
& tref,eref,hconvl,hconvr,roinit,wret,wreol,tvalvr,tvoldr,
& lvalvr,lvoldr,tupold,hupold,told,hold,rodold

c common/hystet/etupfh(200),etdpfh(200),turef,tereфр,tdref,hdef,
& npairu,npaird
c common/toler/tolw,tolh,tolt,tolc,tlol,tolx,ltolen,ltolw,
& ltolh,ltolt,ltolc,ltolp,ltolx
c data zero,one,two,three,four,0.0d0,1.0d0,2.0d0,3.0d0,4.0d0/
c c***initialization
c if(1.etabs-1) 110,110,130
110 continue
if(iplenu.lt.1) tupper=turef*polate(etupfh,time,npairu,memetu)
hupper=entl(pinit,tupper,l11q)
if(iplend.lt.1) tdown=tdref*polate(etdpfh,time,npaird,memetd)
hdown=entl(pinit,tdown,l11q)
goto 150
130 continue
if(iplenu.lt.1) hupper=turef*polate(etupfh,time,npairu,memetu)
tupper=temper(pinit,hupper,l11q)
if(iplend.lt.1) hdown=tdref*polate(etdpfh,time,npaird,memetd)
tdown=temper(pinit,hdown,l11q)
c
150 continue
rouper=dens(pinit,tupper,hupper,l11q)
rodown=dens(pinit,tdown,hdown,l11q)
cpupper=sheat(pinit,tupper,hupper,l11q)
cpdown=sheat(pinit,tdown,hdown,l11q)
if(lplen1.lt.1 .and. lplenu.lt.1) return

sumup=zero
sumun=zero
sumdp=zero
sumdn=zero
do 600 i=1,lchan

kk=klast(i)
if(w(i)) 200,600,400
200 continue

***down flow in channel i. compute sumun and sumdn.

if(lplenu.lt.1) goto 250
sumun=sumun+w(i)
250 continue
if(lplen1.lt.1) goto 600
if(letabs-1) 270,270,280
270 continue
sumdn=sumdn+w(i)*cp(i,1)*tem(i,1)
goto 600

280 continue
sumdn=sumdn+w(i)*ent(i,1)
goto 600

400 continue

***up flow in channel i. compute sumup and sumdp

if(lplenu.lt.1) goto 450
if(letabs-1) 420,420,430
420 continue
sumup=sumup+w(i)*cp(i,kk)*tem(i,kk)
goto 450

430 continue
sumup=sumup+w(i)*ent(i,kk)
450 continue
if(lplen1.lt.1) goto 600
sumdp=sumdp+w(1)

600 continue

if(wret) 620,800,700
620 continue

***down flow in return line. compute sumun and sumdn

if(lplenu.lt.1) goto 650
sumun=sumun+wret
650 continue
if(lplen1.lt.1) goto 800
if(letabs-1) 670,670,680
670 continue
sumdn=sumdn+wret*cpinlt*tref
goto 800

680 continue
sumdn=sumdn+wret*eref
goto 800

700 continue


c***up flow in return line. compute sumup and sumdp.
c
if(iplenu.lt.1) goto 750
if(ietabs-1) 720,720,730
720 continue
sumup=sumup+wret*cpinit*treff
goto 750
730 continue
sumup=sumup+wret*eref
750 continue
if(iplend.lt.1) goto 800
sumdp=sumdp+wret
800 continue
itolp=0
c
c***computes the values of tupper or hupper and tdown, or
c***hdown solving the energy balance equation: ro*vol*
c**(dh/dtime)=sumin(w*h)-sumout(w*h). the values of
c***the fluid enthalpy in plena are treated implicitly

(ie,are taken at the new time level in the flux terms)
c
c***upper plenum

if(iplenu.lt.1) goto 850
denom=rouper*volu
if(ietabs-1) 810,810,820
810 continue
tupper=tupper+dt*(sumup+sumun*cpupper+tupper)/denom/cpupper
if(itolen.le.0) goto 850
if(dabs(tupper-tupold)/tupold.gt.tolp) itolp=1
goto 850
820 continue
hupper=hupper+dt*(sumup+sumun*hupper)/denom
if(itolen.le.0) goto 850
if(dabs(hupper-hupold)/hupold.gt.tolp) itolp=1
850 continue


c***lower plenum

if(iplend.lt.1) goto 900
denom=rodown*volid
if(ietabs-1) 860,860,870
860 continue
tdown=tdown+dt*(sumdn+sumdp*cpdown+tdown)/denom/cpdown
if(itolen.le.0) goto 900
if(dabs(tdown-tdold)/tdold.gt.tolp) itolp=1
goto 900
870 continue
hdown=hdown+dt*(sumdn+sumdp*hdown)/denom
if(itolen.le.0) goto 900
if(dabs(hdown-hdold)/hdold.gt.tolp) itolp=1
900 continue
return
end

subroutine state(temp,ent,p,ro,cp,klst,l,ichan,kaxmax)
c
implicit real*8(a-h,o-z)
c
c***this routine computes the fluid thermodynamic properties,
c***ie, h=hi(p,t), or t=t(p,h),ro=ro(p,t), or ro=ro(p,h),
c*** cp = cp(p, t), or cp = cp(p, h), for each axial level k=1, kk,
c*** and in each channel i=1, ichan. sk, mar84.
c
dimension tem(ichan, kaxmax), ent(ichan, kaxmax), p(ichan, kaxmax).
& ro(ichan, kaxmax), cp(ichan, kaxmax), klast(ichan)

c
common/logkey/1stedy, iplenu, lplenl, lfluid, lmoment, lenereg,
& lcond, lcondl, lconr, lhtr, lhtri, lhtrr, lliq, lmonfl, lform,
& ltemp, letabs, leamb, ldcon, lrest, lunits, lpsa, kpas, jpas,
& ltermx, ltrpr1, ltrpr2, lstep

c
kk = klast(i)
do 500 k = f, kk
if (letabs - 1) 100, 100, 200
100 continue
ent(1, k) = ental(p(1, k), tem(1, k), lliq)
goto 300
200 continue
tem(1, k) = temper(p(1, k), ent(1, k), lliq)
goto 300
300 continue
ro(1, k) = dens(p(1, k), tem(1, k), ent(1, k), lliq)
cp(1, k) = sphet(p(1, k), tem(1, k), ent(1, k), lliq)
500 continue
return
end

function dens(p, t, h, i)
implicit real*(8(a-h, o-z))
c
*** this function computes the density as a function of pressure and
*** enthalpy (or temperature) for different fluids (si units):
c*** if i = 1, h20 liquid
** if i = 2, liuid metal
*** 3, air
*** 4, d20 (corrections h20-d20)
c
*** refs: guppy et al, bnl-nureg-51650, sk et al, tned-r/616, a schor,
c*** scd thesis
*** sk, apr84.
c
dimension a(6), b(6), c(3), rolr(16)
c
data init /0/
c
if (init) 100, 100, 200
100 continue
c
init = 1

goto (120, 140, 160, 120), 1
120 continue

c
hlim = 6.51300d05

c
a(1) = 9.99650d+02
a(2) = 2.58470d-10
a(3) = 1.26960d-22
a(4) = 1.48864d+03
a(5) = 1.46950d+09
a(6)=+3.20372d+06
b(1)=+4.97370d-07
b(2)=+6.17670d-19
b(3)=+4.92230d-31
b(4)=+1.33890d-06
b(5)=+8.85736d 00
b(6)=+1.20483d-02

if(1.eq.1) goto 200
c***d20 correction table: roir(1)=rol,r(1),roir(2)=h1(1),
c***roir(3)=rol,r(2),roir(4)=h1(2),etc, where rol,r=rol(d20)/
c***rol(h20)

c
roir( 1)=+1.1070d 00
roir( 2)=+0.0000d 00
roir( 3)=+1.1065d 00
roir( 4)=+0.4190d+06
roir( 5)=+1.1035d 00
roir( 6)=+0.6280d+06
roir( 7)=+1.1000d 00
roir( 8)=+0.8370d+06
roir( 9)=+1.0950d 00
roir(10)=+1.0470d+06
roir(11)=+1.0865d 00
roir(12)=+1.2560d+06
roir(13)=+1.0790d 00
roir(14)=+1.3600d+06
roir(15)=+1.0650d 00
roir(16)=+1.4650d+06
c
go to 200
c140 continue
c
def=1.5d+05
c1m2=2.0d-07
c
a(1)=+1.00420d+03
a(2)=-2.13900d-01
a(3)=+1.01460d-05
c
go to 200
c160 continue
ro=1.0d0
c
200 continue
go to (300,400,500,300),1
300 continue
if(h-h1m) 320,320,350
320 continue
do 330 i=1,3
330 c(i)=a(i)+b(i)*p
ro=c(i)+h*h*(c(2)+h*h*c(3))
if(1-1) 600,600,380
c
350 continue
do 360 i=4,6
360 c(i-3)=a(i)+b(i)*p
ro = c(1) + c(2)/(h - c(3))
if(1 - 1) 600, 600, 380
380 continue
rod2h2 = polate(ro1r, h, 8, memro)
ro = ro + rod2h2
goto 600
400 continue
ro = a(1) + t*(a(2) + t*a(3)) + clm2*(p-pref)
goto 600
500 continue
c
*** dummy here. ro = 1
ro = 1.0d0
600 continue
dens = ro
return
end
function temper(p, h, l)
implicit real *8(a-h-o-z)

*** this function computes the fluid temperature as a function
*** of pressure and enthalpy. It is similar to the function
*** ental, therefore use explanation of "ental"; however,
*** for liquid metal, the polynomial fits are in the form
*** h = h(p, t), and iterative method is used now for this liquid
*** to evaluate t. sk, apr84.
c
dimension a(4), b(4), c(4)
c
data zero, one, two, thre, four, init/0.0d0, 1.0d0, 2.0d0, 3.0d0, 4.0d0, 0/
c
if(init) 100, 100, 200
100 continue
c
init = 1
c
goto (120, 140, 160, 120), 1
c
120 continue
c
td2h2 = 1.0115d0
c
a(1)*=-1.59542d-07
a(2)*=-5.19625d-13
a(3)*=1.20535d-18
a(4)*=-5.60257d-25
c
b(1)*=2.72911d+02
b(2)*=2.39491d-04
b(3)*=5.96601d-12
b(4)*=-1.31469d-17
c
goto 200
c
140 continue
c
t = 7.0d+02
eps=1.0d-06

a(1)=-6.75080d+04
a(2)=+1.63010d+03
a(3)=-4.16720d-01
a(4)=+1.54270d-04

goto 200

continue

c***air-dummy here

t=400.d0

goto 200

continue

goto (300.400,500.300).1

do 320 i=1,4
    c(i)=a(i)+p+b(i)
    t=c(1)+h*(c(2)+h*(c(3)+h*c(4)))
    if(1.eq.4) t=t+td2h2
    goto 600

400 continue

conv=eps+h

continue

ft=a(1)+t*(a(2)+t*(a(3)+t*a(4)))
ft=ft/h

if(dabs(ft).lt.conv) goto 600
df/dt=a(2)+t*(two*a(3)+t*three*a(4))

500 continue

c***dummy here

t=400.0d0

600 continue

temper=t

end

function ental(p,t,l)

implicit real*8(a-h,o-z)

c***this function computes the enthalpy as a function of
pressure and temperature, for different fluids:
c***if l=1, h2o liquid

c***if l=2, liquid metal

c***if l=3, air (dummy here)
c***if l=4, d2o (corrections h2o-d20)
c***since for water the polynomial fits are given in the

c***form t=t(p,h), a newton-raphson iterative method is

c***used to evaluate h=h(p,t). units:p(pa),t(deg k),h(j,kg).
c***sk,apr84.

dimension a(4),b(4),c(4)
data zero, one, two, three, four, init/0.d0, 1.d0, 2.d0, 3.d0, 4.d0, 0/

if(init) 100, 100, 200
100 continue

init=1

goto (120, 140, 160, 120), 1

120 continue

h=1.0d+05
eps=1.0d-06
td2h2=1.0115d0

a(1)=-1.59542d-07
a(2)=-5.19625d-13
a(3)=1.20635d-18
a(4)=-5.60257d-25

b(1)=2.72911d+02
b(2)=2.39491d-04
b(3)=5.96601d-12
b(4)=-1.31469d-17

***for h20: t=sumi(c(i)*h**(i-1)), where
***c(i)=a(i)*p+b(i), and i=1,2,3,4
***for d20: t=(h20)*td2h2*t(h20)
***refs: thor code manual, sk, scd, sk et al., tned-r/616.

the iterations are carried out as follows:
***define f(h)=sumi(c(i)*h**(i-1))-t and
***dfdh(h)=sumi((i-1)*c(i)*h**(i-2)), then
***convergence criteria: abs(f(h(i)))<t.eeps*t

goto 200

140 continue

a(1)=-6.75080d+04
a(2)=1.63010d+03
a(3)=-4.16720d-01
a(4)=1.54270d-04

***for liquid metals: h=sumi(a(i)*t**(i-1)), where
***i=1,2,3,4. ref: a. schor, mit scd thesis.

goto 200

160 continue

***air- dummy here

h=1.0d+07
200 continue

goto (300, 400, 500, 300), 1

300 continue
conv=eps*t
   do 320 i=1,4
320   c(i)=a(i)+p+b(i)
c
350 continue
   fh=c(1)+h*(c(2)+h*(c(3)+h*c(4)))
   if(1.eq.4) fh=fh+td2h2
   fh=fh-t
   if(dama(fh).lt.conv) goto 600
   dfdh=c(2)+h*(two*c(3)+thre*c(4))
   if(1.eq.4) dfdh=dfdh+td2h2
   h=h-fh/dfdh
   goto 350
c
400 continue
   h=a(1)+t*(a(2)+t*(a(3)+t*a(4)))
   goto 600
500 continue

c***dummy here
c
600 continue
   ental=h
   return
end

function visc(p,t,n,ro,1)
implicit real*8(a-h,o-z)
c
***this function computes the dynamic viscosity as a
***function of pressure and enthalpy (or temperature)
c***for different fluids:
c***if i=1, h2o liquid
***if i=2, liquid metal
***if i=3, air (dummy here)
c***if i=4, d2o (corrections h2o-d2o)
c***units: p(pa), t(deg k), h(j/kg), ro(kg/m3), miu(n*s/m2).
c***refs: asc code manual; sk, et al, tmad-r/616; a.schor,
c***sce thesis.
c***sk,apr84.
c
dimension rmuur(12)
c
data one,init/1.00,0/
c
if (init) 100,100,200
c
100 continue
c
init=1
c
120 continue
c
p1=6.8946d05
h1=2.7600d05
h2=3.9400d05
a0=+1.2995d-03
da1=-9.2640d-04
da2=+3.8105d-04
da3=-8.2194d-05
da4=+7.0224d-06

c
b0=-6.5959d-12
b1=+6.7630d-12
b2=+2.8883d-12
b3=+4.4525d-13

c
c0=+1.4526d-03
c1=-6.9881d-09
c2=+1.5210d-14
c3=-1.2303d-20

c
d0=-3.8064d-11
d1=+2.9285d-16
d2=-1.2586d-21
d3=+1.2860d-27

c
f0=+3.0260d-04
f1=-1.8366d-04
f2=+7.5671d-05
f3=-1.6479d-05
f4=+1.4165d-06

c
g0=+8.5813d-06
g1=+4.2659d+04
g2=+6.4845d-06
g3=+5.5359d+04
g4=+3.8921d-06
g5=+4.0147d+03

c
if(1.eq.1) goto 200

c***d20 correction table: rmiur(1)=miu,r(1)
c***rmiur(2)=t(1),rmiur(3)=miu,r(2), rmiur(4)=
c***t(2), etc., where miu,r=miu(d20)/miu(h20)

c
rmiur( 1)=+1.2900d 00
rmiur( 2)=+3.0300d+02
rmiur( 3)=+1.1800d 00
rmiur( 4)=+3.3300d+02
rmiur( 5)=+1.1600d 00
rmiur( 6)=+3.7300d+02
rmiur( 7)=+1.1200d 00
rmiur( 8)=+4.7300d+02
rmiur( 9)=+1.1100d 00
rmiur(10)=+5.5300d+02
rmiur(11)=+1.1000d 00
rmiur(12)=+6.2300d+02

c
goto 200

c
140 continue

c
b0=+3.6522d-05
b1=+1.6626d-01
b2=-4.5688d+01
b3=+2.8733d+04

goto 200

160  continue
rmiu=1.0d-07

200  continue

goto (300,400,500,300),1

300  continue

if(h.gt.h1) goto 320

x=g0*(h-g1)
y=g2*(h-g3)
rmiu=a0+x*(a1+x*(a2+x*(a3+x*a4)))-
   & (b0+y*(b1+y*(b2+y*b3)))*(p-p1)

if(1-1) 600,600,380

320  continue

if(h.gt.h2) goto 350

rmiu=c0*h*(c1*h*(c2*h*c3))+
   & (d0+h*(d1+h*(d2*h*d3)))*(p-p1)

if(1-1) 600,600,380

350  continue

z=g4*(h-g5)
rmiu=f0+z*(f1+z*(f2+z*(f3+z*f4)))+
   & (f6+h*(f7+h*(f8+h*f9)))*(p-p1)

if(1-1) 600,600,380

380  continue
rmd2h2=polate(rmiur,t,6,memmiu)
rmiu=rmiu+rmd2h2
goto 600

400  continue

x=one/t

rmiu=b0+x*(b1+x*(b2+x*b3))

goto 600

500  continue

c***dummy here

c
rmiu=1.0d07

600  continue

visc=rmiu
return
end

function thcon(p,t,h,ro,1)
c implicit real*8(a-h,o-z)
c
***this function computes the thermal conductivity as a
***function of enthalpy (or temperature) for different fluids:
c***if l=1, h20 liquid
***if l=2, liquid metal
***if l=3, air (dummy here)
c***if l=4, d20 (corrections h20-d20)
c***units:p(ka), t(deg k), h(j/kg), ro(kg/m3), k(w/m-deg k)
c***refs: ssc code manual, sk et. al., nted-r/616, a. schor ssc
c***thesis.
c***sk apr84.
c
dimension rkr(12)

data init/0/

c if(init) 100,100,200

c 100 continue

c init=1

goto (120,140,160,120),1

120 continue

c hl=+5.850d05

c a0=+5.7374d-01
a1=+2.5361d-01
a2=+1.4547d-01
a3=+1.3875d-02

c if(1.eq.1) goto 200

c***d20 correction table: rkr(1)=k,r(1), rkr(2)=t(1),
c***rkr(3)=k,r(2), rkr(4)=t(2), etc., where k,r=k(d20)/
c***k(h20).
c
rkr( 1)=+0.9690d 00
rkr( 2)=+3.0300d+02
rkr( 3)=+0.8650d 00
rkr( 4)=+3.3300d+02
rkr( 5)=+0.9330d 00
rkr( 6)=+3.7300d+02
rkr( 7)=+0.8050d 00
rkr( 8)=+4.7300d+02
rkr( 9)=+0.8630d 00
rkr(10)=+5.5300d+02
rkr(11)=+0.8530d 00
rkr(12)=+6.2300d+02

c goto 200

140 continue

c a0=+1.1042d+02
a1=+6.5112d-02
a2=+1.5430d+05
a3=-2.4617d-09

go to 200

c 160 continue
   rk=4.0d-02

c 200 continue

goto(300,400,500,300),1

c 300 continue

c  x=h/hf
   rk=a0+x*(a1*x*(a2+x*a3))
   iff(1-1) 600,600,350

c 350 continue

c  rkd2h2=polate(rkr,t,6,memk)
   rk=rk*rkd2h2

c  goto 600

c  400 continue

c  rk=a1+t*(a1+t*(a2+t*a3))

c  goto 600

c  500 continue

c  c***dummy here

c   rk=4.0d-02

c  600 continue
   thcon=rk
   return
   end

function spheat(p,t,h,1)
   implicit real*8(a-h.o-z)
   c***this function computes the specific heat as a function
   c***of pressure and enthalpy (or temperature). for different
   c***fluids
   c***if l=1, h2o liquid
   c***if l=2, liquid metal
   c***if l=3, air (dummy here)
   c***if l=4, d20 (corrections h2o-d20)
   c***units: p(pa). t(deg k). h(j/kg). cp(j/kg/deg k)
   c***cp=(dh/dt) at constant pressure is directly derived from
   c***the polynomial fits given in functions "ental" or "temper."
   c***sk apr84.
   c
dimension a(3),b(3),c(3)

data one.init/1.00,0.0/

c   if(init) 100,100,200
100    continue
     c
     init=1
     c
     goto (120, 140, 160, 120), 1
     c
120    continue
     c
     cpd2h2=1.0d0
     c
     a(1)=-5.19625d-13
     a(2)=+2.41270d-18
     a(3)=-1.68077d-24
     c
     b(1)=+2.39491d-04
     b(2)=+1.19320d-11
     b(3)=-3.94407d-17
     c
     goto 200
     c
140    continue
     a(1)=+1.63010d+03
     a(2)=-8.33440d-01
     a(3)=+4.62810d-04
     c
     goto 200
     c
160    continue
     c***air-dummy here
     c
     cp=5.0d+03
     c
200    continue
     c
     goto (300, 400, 500, 300), 1
     c
300    continue
     c
     do 320 i=1,3
320    c(1)=a(i)+p+b(i)
     c
     cp=omac/(c(1)+h*(c(2)+h*c(3)))
     if(1.eq.4) cp=cp*cpd2h2
     goto 600
     c
400    continue
     cp=a(i)+t*(a(2)+t*a(3))
     goto 600
     c
500    continue
     c***dummy here. cp=5000.
     c
     cp=5.0d+03
     c
600    continue
     sphere=cp
     return
end
subroutine fluidi(z, tem, ent, p, ro, cp, q2left, z0, t0, ettim0,
& qIheat, lInteg, kmir, kmax, chleng, rtubei, aflow, de, nreg, klast,  
& w, wold, delp, delt, deltd, delhu, delhd, gcost, ttime0, lregim,      
& tsvl, tvoid, rintv, rintf, rintk, rintg, rkcon, rkexp, rkloc,  
& bmom, cmom, lval, lvold, argf, argg, dq2tlf,  
& vtime, qtime, qbtme, gtime, etumin, etutim, rtime, ntime,   
& lenerg, leqfl, lated, lichan, kaxmax, nregax, ntimeh, 
& ntim2)  
implicit real*8(a-h,o-z)  
c  
c***this routine solves the fluid balance equations in their simplest  
c***form, i.e., assuming incompressible fluid, solves first the momentum  
c***integral equation and later on a decoupled energy equation for each  
c***channel. In the last step, the thermodynamic properties are updated  
c***using a state equation. sk, spr84.  
c  
dimension zl(ichan,kaxmax), teml(ichan,kaxmax), entl(ichan,kaxmax),  
& pl(ichan,kaxmax), rol(ichan,kaxmax), cp(ichan,kaxmax),      
& q2leftl(ichan,kaxmax), zol(ichan,kaxmax), t0l(ichan,kaxmax),  
& ettim0l(ichan,kaxmax), qIheat(ichan,nregax), lInteg(ichan,nregax),      
& kminl(ichan,nregax), kmaxl(ichan,nregax)  
c  
dimension chlengl(ichan), rtubei(ichan), aflowl(ichan), de(ichan),  
& nregl(ichan), klast(ichan), wt(ichan), wold(ichan), delpl(ichan),  
& delt(ichan), deltd(ichan), delhu(ichan), delhd(ichan), gcostl(ichan),  
& tsvl(ichan), tvoidl(ichan), rintvl(ichan), rintf(ichan), rintk(ichan),  
& rintgl(ichan), rkconl(ichan), rkexp(ichan), rkloc(ichan), bmoml(ichan),  
& cmomm(ichan), lvall(ichan), lvoldl(ichan),  
& argfl(kaxmax), arggl(kaxmax), dq2tfl(kaxmax), ttime0l(ichan),  
& lregiml(ichan)  
c  
dimension vtime(ichan,kaxmax,ntimeh), qatime(ichan,kaxmax,ntimeh),  
& qbtme(ichan,kaxmax,ntimeh), gtime(ichan,ntim2),  
& etutiml(ichan,ntimeh), ettim(ichan,ntimeh), ntime(ichan)  
c  
if(leqfl-1) 100,100,200  
100 continue  
c  
c***when leqfl=1, call momentum integral routine, returning the  
c***flow rates in each channel.  
call momint(z,p,tem,ent,ro,lInteg,kmin,kmax,  
& chleng, aflow, de, nreg, klast, rkcon, rkexp, rkloc, w,  
& delp, gcost, ttime, lvold, rintm, rintf, rintk, rintg, lregim,  
& tvoid, lvold, wold, bmom, cmom, argf, argg, lchan,  
& kmax, nregax, qIheat, rtubei, rheat  
return  
200 continue  
c  
c***when leqfl=2, for each channel i, call one of the energy  
c***equation routines, i.e., energ, if 1.0 . energ . 1.0, enerls,  
c***or enerlt, for steady or transient conditions, respectively,  
c***if 7.0 . energ . 10.0, then it computes all other thermo-  
c***dynamic properties using the state routine.  
c  
if(lenerg-6) 300,300,400  
300 continue  
c  
call energ(e,tem,ent,ro,cp,q2leftl,gIheat,  
& kminl,kmaxl,rtubeil,aflowl,nregl,klast,lwold,  
& delt, deltd, delhu, delhd, dq2tfl,l, lichan, kaxmax,  
& ntiml, ntimeh), ettiml, etumin, ttime0l, lichan, kaxmax,  
& tsvl, tvoidl, rintm, rintf, rintk, rintg, lregim)
& nregmx)
goto 600
C
400 continue
C
if (listed .ne. 0) goto 500
call enters (z, tem, ent, ro, cp, q2left, q1heat,
& kmin, kmax, rtube1, aflow, nreg, klast, w, z0, to, ettim0,
& deltu, deltld, delhu, delhd, dq2tf1, vtime, qatime, qotime,
& etutim, etlim, gtime, nptime, i, ichan, kaxmax, nregmx,
& ntimeh, ntimh2)
goto 600
C
500 continue
C
call enterlt (z, tem, ent, ro, cp, q2left, q1heat,
& kmin, kmax, rtube1, aflow, nreg, klast, w, wold, z0, to,
& ettim0, ttime0, deltu, deltld, delhu, delhd, dq2tf1, vtime,
& qatime, qotime, etutim, etlim, gtime, nptime, i, ichan,
& kaxmax, nregmx, ntimeh, ntimh2)
C
600 continue
C
call state (tem, ent, p, ro, cp, klast, l, ichan, kaxmax)
return
end
subroutine stabill (ro, dz, kmin, kmax, aflow, w, wold, nreg,
& dt, lennerg, ldt, ichan, kaxmax, nregmx)
C
implicit real *8 (a-n, o-z)
C
*** this routine checks if the courant stability condition
C
*** (alpha = dt/dz .le. 1) is violated, when an explicit
C
*** treatment of the energy flux term is used (lennerg=4,5,6).
C
*** if yes, then reduces dt such that alpha .le. 1, for each
C
*** axial region and each channel.  sk, apr84.
C
dimension ro (ichan, kaxmax), dz (ichan, nregmx),
& kmin (ichan, nregmx), kmax (ichan, nregmx), aflow (ichan),
& w (ichan), wold (ichan), nreg (ichan)
C
data one, two/0.95d0, 2.0d0 /
C
ldt=1
C
do 200 i=1, ichan
nmax=nreg (i)
if (lennerg .eq. 4) g=wold (i)/aflow (i)
if (lennerg .eq. 5) g=(wold (i)+w (i))/two/aflow (i)
if (lennerg .eq. 6) g=w (i)/aflow (i)
C
do 100 n=1, nmax
k1=kmin (i, n)
k2=kmax (i, n)
roavg=(ro (i, k1)+ro (i, k2))/two
v=g/roavg
v=dsbs (v)
alp ha=v*dt/dz (i, n)
if (alpha .lt. one) goto 100
dt=one*dz (i, n)/v
.
ldt=2
100 continue

***continue with next channel

200 continue
return
end

function rk(t,m)

implicit real*8(a-h,o-z)

***this function computes the thermal conductivity rk as a function of temperature (t)
***for different solid materials (m).
***units: t(deg k),rk(w/m-k). sk apr84

data init/0/
if(init) 100,100,300
100 continue

init=1

***rkbrsi=rk(w/m-k)/rk(btu/ft-hr-f)
***t(f)=tbs11+t(k)-tbs12
***t(c)=t(k)-tck

rkbrsi= 6.23067d+03
tbs11= 1.80000d 00
tbs12= 4.59670d+02
tck= 2.73150d+02

m

***m=1, uo2, ref: nureg/cr-0665,la-7777-ms
***m=2, zr, ref: nureg/cr-0665,la-7777-ms
***m=3, dummy now

120 continue

    c1=+4.040d+01
c2=+4.640d+02
c3=+1.216d-04
c4=+1.867d-03
c5=+1.911d-02
c6=+2.580d 00
c7=+5.800d-04
c8=+5.000d-02
c9=+1.000d+02
t1=+1.650d+03	one=1.0d0
one=+5.0d-02

goto 300
140 continue

b1=+7.510d 00
b2=+2.090e-02
b3=−1.450e+05
b4=+7.870e-09

c  goto 300
c
160  continue
c****dummy
c
300  continue
c
   goto (350,400,450,450,450),m

c 350  continue
tc=t-tck
c0=c6+c7*tc
c=c9*(one-c0*oneftd)/(one-c0*c8)
ca=c4*tc
c=exp(ca)
   if(tc>t1) 360,360,380
360  continue
rk=c*(c1/(c2+tc)+c3*ca)
return
380  continue
rk=c*(c5+c3*ca)
return

c 400  continue
c
   rk=b1+tt*(b2+t*(b3+t*b4))
return
450  continue
rk=one
c
return
end
function roc(t,m)
c
implicit real*8(a-h,o-z)
c
****this function computes the volumetric heat capacity
c****(roc=roc*cp) as a function of temperature (t) for different
c****solid materials (m)
c****units: t(deg k), roc(j/m3-k), sk apr84.
c
dimension cpa(13),cpb(28)
data init/0/
c
   if(init) 100,100,300

c 100  continue
init=1
c
c****rocbsl=roc(j/m3-k)/roc(btu/f3-f)
c****t(f)=tbsi1+t(k)-tbsi2
c
   rocbsl=+6.7066e+04
tbsi1=+1.8000e+00
tbsi2=+4.5967e+02

c
goto (120, 140, 160, 160, 160), m

c c***m=1, uo2, ref: nureg/cr-0665, la-7777-ms

c c***m=2, zr, ref: nureg/cr-0665, la-7777-ms

c c***m=3-5, dummy now

c
120 continue

c cpa( 1)++1.54960d+01
 cpa( 2)++1.91450d+01
 cpa( 3)++7.84730d-04
 cpa( 4)++5.64370d+06
 cpa( 5)++5.35285d+02
 cpa( 6)++3.76946d+04
 cpa( 7)++1.98700d+00
 cpa( 8)++1.00000d+00
 cpa( 9)++2.00000d+00

c rouo2 ++1.09700d+04

c
go 300

c
140 continue

c cpb( 1)++2.81000d+02
 cpb( 2)++3.00000d+02
 cpb( 3)++3.02000d+02
 cpb( 4)++4.00000d+02
 cpb( 5)++3.81000d+02
 cpb( 6)++6.40000d+02
 cpb( 7)++3.75000d+02
 cpb( 8)++1.09000d+03
 cpb( 9)++5.02000d+02
 cpb(10)++1.09300d+03
 cpb(11)++5.90000d+02
 cpb(12)++1.13000d+03
 cpb(13)++6.15000d+02
 cpb(14)++1.13300d+03
 cpb(15)++7.19000d+02
 cpb(16)++1.15300d+03
 cpb(17)++8.16000d+02
 cpb(18)++1.17300d+03
 cpb(19)++7.70000d+02
 cpb(20)++1.19300d+03
 cpb(21)++6.19000d+02
 cpb(22)++1.21300d+03
 cpb(23)++4.69000d+02
 cpb(24)++1.23300d+03
 cpb(25)++3.56000d+02
 cpb(26)++1.24800d+03
 cpb(27)++3.56000d+02
 cpb(28)++1.30000d+03

c rorh ++6.50000d+03

go 300

c
160 continue

c c***dummy

c
300 continue
c
goto (350, 400, 450, 450, 450), m
350   continue
   c
   cpa(10)=cpa(5)/t
   cpa(11)=dexp(-cpa(10))
   cpa(12)=cpa(8)/cpa(7)/t
   cpa(13)=dexp(-cpa(12))
   c
   cpuo2=cpa(1)*(cpa(2)*cpa(10)+cpa(10)+cpa(11))
   &   /(cpa(11)-cpa(8))/(cpa(11)-cpa(8))
   &   +cpa(9)*cpa(3)*t+cpa(4)*cpa(12)/t*cpa(13))
   c
   roc=rouo2*cpuo2
   return
400   continue
   cprz=polate(cpb, t, 14, memcpb)
   c
   roc=rozr*cprz
   return
450   continue
   roc=1.0d+06
   return
end
function thexp(p, t, h, l)
   implicit real*8(a-h, o-z)
   c
   c*** this function computes the thermal expansion coef. (beta)
   c*** as a function of pressure and enthalpy (or temperature)
   c*** for different fluids (all properties are si units):
   c*** if l=1, h2o liquid
   c*** if l=2, liquid metal
   c*** if l=3, air
   c*** if l=4, d2o (corrections h2o-d2o)
   c*** next relationships are used:
   c*** beta=one/ro*(dro/dt) at constant p for liquid metal
   c*** beta=cp/ro*(dro/dh) at constant p for h2o
   c*** (dro/dt) or (dro/dh) are derived directly from relations
   c*** given in function dens, when the corrections h2o-d2o
   c*** for (dro/dh) are taken from s.k. et. a1., tned-r/616.
   c*** sk, apr 84.
   c
   dimension a(4), b(4), c(2), rh1r(10)
   c
   data init/0/
   c
   if(init) 100, 100, 200
100   continue
   c
   init=1
   c
   goto (120, 140, 160, 120), 1
120   continue
h1m=+6.51300d+05

a(1)=+5.16940d-10
a(2)=+5.07840d-22
a(3)=+1.46950d+09
a(4)=+3.20372d+06

b(1)=+1.23934d-18
b(2)=+1.98692d-30
b(3)=+8.85736d 00
b(4)=+1.20483d-02

if(1.eq.1) goto 200

c***d20 correction table: rhrl(1)=rhl,r(1),rhrl(2)=h2(1),
c***etc., where rhl,r=rhl(d20)/rhl(h20).

  rhrl( 1)=1.12d 00
  rhrl( 2)=0.00d 00
  rhrl( 3)=1.12d 00
  rhrl( 4)=0.42d+06
  rhrl( 5)=1.18d 00
  rhrl( 6)=0.54d+06
  rhrl( 7)=1.20d 00
  rhrl( 8)=1.50d+06
  rhrl( 9)=1.20d 00
  rhrl(10)=2.00d+06

goto 200

goto 200

c continue

c a(1)=+2.13900d-01
a(2)=+2.20920d-05

goto 200

c continue
beta=+1.0d-03

c continue

goto (300,450,500,300),1

continue
if(h-h1m) 320,320,350

continue
do 330 i=1,2

  c(1)=a(1)+b(1)*p
  rh=h+(c(1)+h+h*c(2))
  if(1-1) 400,400,380

continue
do 330 i=3,4

  c(2)=a(1)+b(1)*p
  rh=c(1)/(h-c(2))/(h-c(2))
  if(1-1) 400,400,380

continue

rh2h2=pole(rhrl,n.5,mamrhl)
rh=rhd2h2

400 continue

C***beta=-cp/ro*(dro/dn)
C
C cp=sphat(p,t,h,1)
ro=dens(p,t,h,1)
beta=-rh*cp/ro
C
goto 600

450 continue
rt=a(1)+tt*a(2)
ro=dens(p,t,h,1)
C
C***beta=-1/ro*(drodt)
C
beta=-rt/ro
C
goto 600

500 continue
C
C***dummy here. beta=1.d-03
C
beta=1.d-03
C
600 continue
thexp=beta
return
C
C subroutine steady(t,told,rtube,z,tem,ent,p,ro,cp,
& temold,entold,roold,cpold,q2left,q2righ,ht1,ht2,text,
& roext,z0,t0,etim0,lregim,dz,dtube,qvol,q1heat,
& linterg,kmin,kmax,jmax,materv.evect.fvect,
& argf,argv,dtwtr,dqtfl,dq2tf1,dq2tf2,
& avect,bvect,cvect,dvect,
& vtime,qatime,qbtime,gttime,etutim,etitim,ntime,
& lated,ichan,jconmx,kaxmax,nregmx,ntimeh,ntimh2)
C
C implicit real*8(a-h,o-z)
C
C***steady state computation routine. sk may84.
C
dimension t(ichan,kaxmax,jconmx),told(ichan,kaxmax,jconmx),
& rtube(ichan,nregmx,jconmx)
C
dimension z(ichan,kaxmax),tem(ichan,kaxmax),ent(ichan,kaxmax),
& p(ichan,kaxmax),ro(ichan,kaxmax),cp(ichan,kaxmax),
& temold(ichan,kaxmax),entold(ichan,kaxmax),roold(ichan,kaxmax),
& cpold(ichan,kaxmax),q2left(ichan,kaxmax),q2righ(ichan,kaxmax),
& ht1(ichan,kaxmax),ht2(ichan,kaxmax),text(ichan,kaxmax),
& roextl(ichan,kaxmax),zo0(ichan,kaxmax),to0(ichan,kaxmax),
& etim0(ichan,kaxmax),lregim(ichan,kaxmax),dz(ichan,nregmx),
& dtube(ichan,nregmx),qvol(ichan,nregmx),q1heat(ichan,nregmx),
& linterg(ichan,nregmx),kmin(ichan,nregmx),kmax(ichan,nregmx),
& jmax(ichan,nregmx),materv(ichan,nregmx),evect(kaxmax,jconmx),
& fvect(kaxmax,jconmx)
C
dimension argf(kaxmax),argv(kaxmax),dtwtr(kaxmax),
& dtwtr(kaxmax),dq2tf1(kaxmax),dq2tf2(kaxmax),
& avect(jconmx),bvec(jconmx),cvect(jconmx),dvect(jconmx)
c

dimension vtime(ichan, kaxmax, ntimeh), qatime(ichan, kaxmax, ntimeh),
& qbttime(ichan, kaxmax, ntimeh), gttime(ichan, ntimeh2),
& etutim(ichan, ntimeh), etltime(ichan, ntimeh),
& nptime(ichan)

common/aan/chlen(7), rtube(7), rtubes(7), rheat(7),
& aflow(7), de(7), nreg(7), klast(7),
& w(7), wold(7), delp(7), delu(7),
& delt(7), delhu(7), delhd(7), gcost(7),
& timev(7), tvoid(7), rintm(7), rintf(7),
& rink(7), rintg(7), rkcon(7), rkexp(7),
& rklc(7), bmm(7), cmom(7), lvalv(7),
& lvoid(7), ttimeo(7), ttiold(7), tregim(7)

common/logkey/lstedy, ilplen, ilplend, ilfluid, ilmomem, ilenerg,
& ilcondu, ilconl, ilconr, lhtr, lhtrl, lhtrp, likiq, lilmomfl, ilform,
& ileopt, ilatbs, ileamb, ildtcon, ilrest, ilunits, ilpas, ilpas,
& iltermx, ilptr1, ilptr2, ilstep

common/time/s/time0, timef, dtstated, dtttran, time, dt

common/geomin/vol, vold, dzu, tupper, hupper, tdown, hdown, rrouper,
& lamb, cpext, volam, tsolid, wretin, pinit, tinmt, ainmt, cpinit,
& tref, ref, hconv1, hconvr, rconvit, wret, wraold, tvvalv, tvvold,
& lvalv, tvvold, tupold, hupold, tdold, hdold, rodown

common/toler/to/70, tolh, tolto, tolc, tolpl, tolx, iltolen, iltolw, iltolh,
& iltol, iltolc, iltolp, iltolx

common/time/texec(10), dexec(9), tprin(10), dtprin(9), tplot(10),
& dtpolt(10), tdump(10), dtump(9), tprint(200), tplot(200), tdump(200),
& nexec, nprin, nplot, ndump, nexec2, nexeccl, nprin2, nprin1, npplot2,
& npplot1, ndump2, ndump1

data zero, two, eps/0.0d0,2.0d0,1.0d-05/

if(lstedy.lt.0) return

lsted=0
ilotr=1tolen
iltolen=1ldcon
do 500 iter=1, 1termx
iltol=0
dt=dtstated
continue

call core(t, told, rtube, z, tem, ent, p, ro, cp, q2left, q2righ,
& h1, htr, text, roext, z0, to, ettm0, dz, drtube, qvol, qheath,
& linteg, kmn, kmax, jmax, mater, evect, fvect,
& argf, argg, dtwtf1, dtwtf1r,
& dq2dif, dq2fr, evct, bvecl, cvect, dvect, dregim,
& vtime, qatime, qbttime, gttime, etutim, etltime, nptime,
& listed, ichan, jconxx, kaxmax, nregmx, ntimeh, ntimeh2)

call eplea(tem, ent, cp, klast, w, ichen, kaxmax)

call enviro(t, text, roext, htr, q2righ, kmin, kmax,
& jmax, chlen, rtube, nreg, ichen, jconxx, kaxmax, nregmx)

if(iltolen.lt.1) goto 350
if((iltolw.eq.0) and (iltolen.eq.0) and (iltolp.eq.0) and
& (1tolc.eq.0).and.(1tolp.eq.0)) goto 350

C
C***if during an iteration step the change in any parameter is greater
C***than its prescribed tolerance, all variables are assigned their
C***old value and are recomputed with dt=dt/2.
C
C  tupper=tupold
  hupper=hpold
  tdown=tdold
  hdown=hdold
  lvalvr=lvold
  tvalvr=tvold
  wret=wreold
  dt=dt/two
  do 300 i=1,ichan
    lvalv(i)=lvold(i)
    tmev(i)=tvold(i)
    w(i)=wold(i)
    ttmev(i)=ttold(i)
    c
    nmex=nreg(i)
    c
  do 300 n=1,nmax
    if(n-1) 110,110,120
   110 continue
     k1=kmin(1,n)
     goto 130
   120 continue
     k1=kmin(1,n)+1
   130 continue
     k2=kmax(1,n)
     jmax=jmax(1,n)
     do 300 k=1,k2
       temv(1,k)=temold(1,k)
       entv(1,k)=entold(1,k)
       rov(1,k)=roold(1,k)
       cpv(1,k)=cpold(1,k)
     do 300 j=1,jmax
       t(1,k,j)=told(1,k,j)
     300 continue
     goto 100
C
C***keep in memory the values of the main variables at the end
C***of an iteration step
C
C  continue
  if(dabs(tupold-tupper).gt.(tolx*tupper)) itolx=1
  if(dabs(hpold-hupper).gt.(tolx*hupper)) itolx=1
  if(dabs(tdold-tdown).gt.(tolx*tdown)) itolx=1
  if(dabs(hdold-hdown).gt.(tolx*hdown)) itolx=1
  if(dabs(lvold-lvalvr).gt.dabs(tolx*wret)) itolx=1
  tupold=tupper
  hpold=hupper
  tdold=tdown
  hdold=hdown
  lvold=lvalvr
  tvold=tvalvr
  wreold=wret
C
  do 400 1=1,ichan

C
void(i)=ivalv(i)
tvoid(i)=timev(i)
if(dabs(wold(i)-w(i)).gt.dabs(tolx+w(i))) tolx=1
wold(i)=w(i)
ttold(i)=ttimeto(i)
nmax=nreg(i)
do 400 n=1,nmax
if(n-i) 360,360,370
360 continue
k1=kmin(1,n)
goto 380
370 continue
k1=kmin(1,n)+1
380 continue
k2=kmax(1,n)
jmaxx=jmax(1,n)
do 400 k=k1,k2
if(dabs(temold(1,k)-tem(1,k)).gt.(tolx*tem(1,k))) tolx=1
if(dabs(entold(1,k)-ent(1,k)).gt.(tolx*ent(1,k))) tolx=1
temold(1,k)=tem(1,k)
entold(1,k)=ent(1,k)
roold(1,k)=ro(1,k)
ccold(1,k)=cc(1,k)
do 400 j=1,jmaxx
if(dabs(told(1,k,j)-t(1,k,j)).gt.(tolx*t(1,k,j))) tolx=1
told(1,k,j)=t(1,k,j)
400 continue
if(tolx.eq.0) goto 450
if(iter.gt.itpr1.and.iter.lt.itpr2) goto 500
450 continue

***print the main variables

call writea(t,z,tem,ent,p,ro,q2left,q2right,ht1,htr,
& text,z0,t0,1regim,kmin,kmax,jmaxx,wx,deltu,deltl,delu,
& delhd,nreg,klast,ivalv,lregim,ltime,dt,tupper,hupper,
& tdown,hdqw,wr,ifluid,icondu,iter,ipas,kpas,aps,
& i-chan,jconmx,kasmax,nregmx)
if(tolx.eq.0) goto 600
500 continue

***the convergence of all main parameters has occurred
***in the limit of tolerance (tolx), or the minimum number
***of iterations permitted (itermx) has been exceeded.

600 continue

***if lstedy=0 (only steady state computation) the execution
***is finished

if(lstedy.eq.0) stop
itolen=itolfr
return
end
subroutine trans(t,told,rtube,z,tem,ent,p,ro,cp,
& temold,entold,roold,ccold,q2left,q2right,ht1,htr,text,
& roext,z0,t0,etlim0,1regim,dz,drtube,qvol,qheat,
& linteger,kmin,kmax,jmaxx,mater,evect,fvect,
& argf,argg,dtwfl,dtwfr,dq2tf1,dq2tf1,dq2tf1,
& avect,bvect,cvect,dvect,
& vtime, qtime, qtime, gtime, etutim, etim, nptime,
& listed, ichen, jconmx, kaxmax, nregmx, ntimeh, ntimeh2)

implicit real*8(a-h, o-z)

****transient computation routine. sk may84.

dimension t(ichen, kaxmax, jconmx), told(ichen, kaxmax, jconmx),
& rtube(ichen, nregmx, jconmx)

dimension z(ichen, kaxmax), tem(ischen, kaxmax), ent(ischen, kaxmax),
& p(ischen, kaxmax), ro(ischen, kaxmax), cp(ischen, kaxmax),
& temold(ischen, kaxmax), antold(ischen, kaxmax), roold(ischen, kaxmax),
& cpold(ischen, kaxmax), q2left(ischen, kaxmax), q2right(ischen, kaxmax),
& htl(ischen, kaxmax), htr(ischen, kaxmax), text(ischen, kaxmax),
& roext(ischen, kaxmax), zo(ischen, kaxmax), tol(ischen, kaxmax),
& etimold(ischen, kaxmax), lregim(ischen, kaxmax), dz(ischen, nregmx),
& drtube(ischen, nregmx), qvol(ischen, nregmx), qtheat(ischen, nregmx),
& lintag(ischen, nregmx), kmin(ischen, nregmx), kmax(ischen, nregmx),
& jmax(ischen, nregmx), mater(ischen, nregmx), evec(ischen, kaxmax, jconmx),
& fvec(kaxmax, jconmx)

dimension argf(kaxmax), argg(kaxmax), dtwtf1(kaxmax),
& dtwtf1(kaxmax), dq2tfr(kaxmax), dq2tfr(kaxmax),
& avect(jconmx), bvect(jconmx), cvect(jconmx), dvec(jconmx)

dimension vtime(ischen, kaxmax, ntimeh), qatime(ischen, kaxmax, ntimeh),
& gtime(ischen, kaxmax, ntimeh), gttime(ischen, ntimeh2),
& etutim(ischen, ntimeh), etutim(ischen, ntimeh),
& nptime(ischen)

common/asx/chieng(7), rtube(7), rtube(7), rheat(7),
& aflow(7), del(7), nreg(7), klant(7),
& w(7), vola(7), dela(7), del(7),
& delt(7), delhu(7), delhd(7), gcst(7),
& timev(7), tvoid(7), rintm(7), rintf(7),
& rintk(7), rintg(7), rkon(7), rkexp(7),
& rkloc(7), bnmom(7), cnom(7), lvalv(7),
& lvoid(7), ttime(7), ttold(7), tregim(7),
common/logkey/lasted, plenu, plend, plful, plmom, plener,
& lcond, lcon, lcon, lht, lhtlin, lhtlin, ltiq, ltmom, lform,
& letopt, letabs, leamb, ltdcon, lrest, lunits, lps, kps, jps,
& itermx, itpr, itpr, itstep

common/time0, timef, dtst, dttran, time, dt

common/geomin/volu, vold, dzu, tupper, tupper, tdown, hdown, roupper,
& tmb, cmext, voldamb, tsolid, wretin, ninit, tinit, einit, cpinit,
& tref, eref, hconvl, hconvr, roinit, wret, wredold, tvalvr, tvoid,
& lvalvr, lvold, tupold, hupold, tdold, hdold, rodow

common/toler/tolw, tolh, tol, tolw, tolh, ltol, ltol, ltol, ltol,
& ltol, ltoic, ltoic, ltoic

common/timem/texexc(10), dtexexc(9), tprin(10), dtprin(9), tplot(10),
& dtplot(10), tdump(10), dtdump(9), tprin(200), tplot(200), tdump(200),
& nteexc, ntprin, ntplot, ndump, naxec2, naxec1, nprin2, nprint1, nplot2,
& nplot1, ndump2, ndump1

data zero, two, eps/0.0, 2.000, 1.00-05/
if(listed.1t.0 .and. lenrg.gt.6)
& call enersls(z,tem,ent,ro,cp,q2left,q1heat,
& kmin,kmax,rtube,aflow,nreg,klast,w,z0,t0,ettim0,
& deltu,deltd,delhu,delhd,dq2tf1,vtime,qatim,qttime,
& etutim,etitim,gttime,ntime,1,ichan,kaxmax,nregmx,ntimeh,
& ntimh2)

call writea(t,z,tem,ent,p,ro,q2left,q2righ,ht1,htr,
& text,z0,t0,lregim,kmin,kmax,jmax,w,deutu,deltd,delhu,
& delhd,nreg,klast,lvalv,lregim,time,dt,tupper,hupper,
& tdown,hdown,wret,lfuid,icondu,iter,ipas,kpas,jpas,
& ichan,jconmx,kaxmax,nregmx)
1sted=1
timold=time
nexec=1
100 continue

ccc****find the execution time step (dt).

do 150 n=nexec,1,nexec
nexec=n
dt=dtexec(n)
if((time+eps).lt.texec(n+1)) goto 200
150 continue
200 continue

if((time-timeold).ge.eps) return

call core(t,told,rtube,z,tem,ent,p,ro,cp,q2left,q2righ,
& htr,x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x14,x15,x16,
& x17,x18,x19,x20,x21,x22,x23,x24,x25,x26,x27,x28,
& x29,x30,x31,x32,x33,x34,x35,x36,x37,x38,x39,x40)
& if(linteg,kmin,kmax,jmax,mater,evect,fvect
& p argf,argg,dtwtf1,dtwtr,
& dq2tf1,dq2trf,avect,b vect,cvect,veci,regim,
& vtime,qatime,qt time,gtt ime,etutim, etitim, ntime,
& 1sted,ichan,jconmx,kaxmax,nregmx,ntimeh,ntimh2)
call eplena(tem,ent,cp,klast,w,ichan,kaxmax)
call enviro(t,text,roext,htr,q2righ,kmin,kmax,
& jmax,chlen,rtube,nreg,ichan,jconmx,kaxmax,nregmx)

if(1tolen.1t.1) goto 350
if((1tolv.eq.0).and.(1tolh.eq.0).and.(1tol1.eq.0).and.
& (1tolc.eq.0).and.(1tolp.eq.0)) goto 350

ccc****if during an iteration step the change in any parameter is greater
ccc****than its prescribed tolerance, all variables are assigned their
ccc****old value and are recomputed with dt=dt/2.

dt=dt/two
 tupper=tupold
 hupper=hu pold
 tdown=told
 hdown=hdold
 lvalv=lvold
 tvalv=tvold
 wret=wreold
do 300 i=1,ichan
lvalv(i)=lvoid(1)
timev(i)=tvold(1)
w(i)=wold(1)
tttime0(1)=ttold(1)
c
nmax=nreg(1)
c
do 300 n=1,nmax
if(n-1) 250,250,260
250 continue
k1=kmin(1,n)
goto 270
260 continue
k1=kmin(1,n)+1
270 continue
k2=kmax(1,n)
jmaxr=jmax(1,n)
do 300 k=k1,k2
tam(1,k)=temold(i,k)
etn(1,k)=entold(i,k)
ro(1,k)=roold(i,k)
cp(1,k)=cpold(i,k)
do 300 j=1,jmaxr
(t(1,k,j)=told(i,k,j))
300 continue
c
goto 200
c
***keep in the memory the values of the main variables
***at the end of a time step.
c
350 continue
tupold=tupper
hupold=hupper
tdold=tdown
hdold=hdown
lvoid=lvalvr
tvoid=tvold
wreold=wret
ttimold=time
do 400 i=1,ichan
lvoid(i)=lvalv(i)
tvoid(i)=timev(i)
wold(i)=w(i)
ttold(i)=tttime0(1)
c
nmax=nreg(1)
c
do 400 n=1,nmax
if(n-1) 360,360,370
360 continue
k1=kmin(1,n)
goto 380
370 continue
k1=kmin(1,n)+1
380 continue
k2=kmax(1,n)
jmaxr=jmax(1,n)
do 400 k=k1,k2
temoid(1,k)=tem(1,k)
entoid(1,k)=ent(1,k)
rooid(1,k)=ro(1,k)
cpoid(1,k)=cp(1,k)
do 400 j=1,jmax
	old(k,j)=t(1,k,j)
400 continue

if time=tprint(n), where tprint() is a time of
c***print-out data, call write

n=nprint2

c***Dump channel temperature and flowrate information to output
c***files.

icpi=ichan+1
write(20,1000) time,(w(i),i=1,ichan),wret
do 425 l=1,ichan
ifile=l+20
kk=klast(l)
write(ifile,1000) time,(tem(l,k),k=1,kk)
425 continue

if(n,gt,nprint) stop
if((time-tprint(n)+eps).lt.zero) goto 100
nprint2=nprint+1

call write(t,tz,tem,ent,ro,q2left,q2right,hti,htl,
& t,text,z0,ro,qleft,w,qmax,kmin,kmax,jmax,w,delwu,delwd,delhu,
& delhd,nreg,klast,ivalv,iregim,itime,dt,tupper,htpper,
& tdown,hdown,wret,ifluid,lcondu,iter,ipas,kpas,jpas,
& ichan,jconmx,kaxmax,nregmx)
goto 100
500 continue

continue

return

format(20(1p10.3))
end

subroutine enviro(t,text,roext,hti,q2right,kmin,kmax,
& jmax,chieng,rtubee,nreg,ichan,jconmx,kaxmax,nregmx)

implicit real*8(a-h-o-z)

this routine computes the environmental (air) temperature
from a simplified energy equation (without convection
term), the density of air (vs. temp), and then the convective
heat transfer coefficient and the heat flux at the structure-
environment boundary (right-hand side) sk may84.

dimension t(ichan,kaxmax,jconmx)

dimension text(ichan,kaxmax),roext(ichan,kaxmax),hti(ichan,kaxmax),
& q2right(ichan,kaxmax),kmin(ichan,nregmx),kmax(ichan,nregmx),
& jmax(ichan,nregmx)

dimension chieng(ichan),rtubee(ichan),nreg(ichan)

common/logkey/istedy,iplenu,iplend,ifluid,imomen,ilenerr,lcondu,
& iconi,iconr,ihtr,ihtr1,ihtr2,il11q,ilmomf1,ilform,leoopt,leotab,leamb,ldtcon,lrest,lunits,ipas,kpas,jpas,itermx,iltpr1,iltpr2,
& listep

common/time0,timef,dtsed,dtran,time,dt

common/geomin/volu,vold,dsu,tupper,hupper,tdown,hdow,rouper,
& tmax,cepex,volamb,tshow,tsolid,wretin,pinit,tinit,einit,cpinit,
& tref,aref,hconvl,hconvr,roinit,wret,wreol,tdvalv,tdvold,tdvold,
& tvlvr,lvold,tupold,hupold,tdold,hdold,rdold

common/htcoef/a11,a21,a31,a41,a12,a22,a32,a42,a13,a23,a33,a43,
& relini,relim2,grpr1,grpr2,runrig

data four,twopl/4.0d0,6.28d0/

if(ileamb.lt.1 .and. lhtr.lt.1) return

d0 500 i=1,lchan
geomf=volamb/twopl/rtube(1)/ch1eng(1)
nmax=nreg(1)
d0 500 n=1,nmax
if(n.ne.1) goto 110,110,120

continue
k1=kmin(1,n)
goto 130

continue
k1=kmin(1,n)+1

continue
k2=kmax(1,n)

jmax=jmax(1,n)
do 400 k=k1,k2
if(ileamb.lt.1) goto 200

text(1,k)*text(1,k)+dt*q2righ(1,k)/geomf/roext(1,k)
& /cepex
roext(1,k)=dens(pdummy,text(1,k),hdummy,3)

continue
if(lhtr.lt.1) goto 400
if(lhtr.gt.1) goto 300
defext=four-geomf
rk=thcon(pdummy,text(1,k),hdummy,rdum,3)
htr(1,k)=runrig*rk/defext

continue
q2righ(1,k)*htr(1,k)+t(t1,k,jmaxr)-text(1,k)

400 continue

c***continue with next region and next channel

c
500 continue
return
end

subroutine htlef(t,tem,ent,p,ro,cp,q2left,htl,1regim,kmin,kmax,w,ch1eng,aflow,de,gcost,nreg,
& lhtr,lhtr1,il11q,1,ichan,jconmx,kaxmax,nregmx)
c implicit real*8(a-h,o-z)

c***this routine computes the convective heat transfer coef.
c***and the heat flux at the liquid-solid interface (left side)
c***sk, may84.
dimension t(ichan,kaxmax,jconmx)
c
dimension tem(ichan,kaxmax),ent(ichan,kaxmax),p(ichan,kaxmax),
& ro(ichan,kaxmax),cp(ichan,kaxmax),htl(ichan,kaxmax),
& q2left(ichan,kaxmax),lreglm(ichan,kaxmax),kmin(ichan,nregmax),
& kmax(ichan,nregmax)
c
dimension w(ichan),chlen(g(ichan),aflow(ichan),de(ichan),
& gcost(ichan),nreg(ichan)
c
common/htco6f/a11,a21,a31,a41,a12,a22,a32,a42,a13,a23,
& a33,a43,relm1,relm2,grpr1,grpr2,nnurig
c
data eps/1,d-06/
if (lt/hr.lt.1) return

c***computation of the heat transfer coeff. at left hand side from:
c***nu=a1*(re=a2)*(gr=a3)*(pr=a4), where nu=htl*de/k,
c***re=abs(w)*de/(aflow*miu); gr=grav*(1+3)*(ro*2)/(miu*2)*
c***beta=abs(tw-tfluid); pr=cp*miu/k.
c
gde=aflow(1)*dabs(w(1))
mmn*nreg(1)
g13=gcost(1)+chlen(1)+nreg(1)+chlen(1)+chlen(1)
do 500 n=1,mmn
if(n-1) 110,110,120
110 continue
k1=kmin(1,n)
goto 130
120 continue
k1=kmin(1,n)+1
130 continue
k2=kmax(1,n)
do 500 k=k1,k2
if(lhr1.lt.1) goto 480
rmiu=visc(p(1,k),tem(1,k),ent(1,k),ro(1,k),liiq)
rk=thcon(p(1,k),tem(1,k),ent(1,k),ro(1,k),liiq)
beta=thexpfp(1,k),tem(1,k),ent(1,k),ro(1,k),liiq)
r=gcde/rmiu
pr=cp(1,k)*rmiu/rk
dtwal=tem(1,k)–t(1,k,1)
dtwal=dabs(dtwal)
gr=13*ro(1,k)*ro(1,k)/rmiu*rmiu*beta*dtwal
c
if(re–relm2) 200,250,250
200 continue

c***laminar, or transition laminar-turbulent flow regimes

c
lreglm(1,k)=1
a2=a21
if(re.lt.eps.or.dabs(a2).lt.eps) goto 300
rec=dmini(re,relm1)
rea2=rea2
a1=a11
a3=a31
a4=a41
gra3=gra3
pra4=pra4
rwi=a1+rea2*gra3+pra4
rnul=rnu

if(re-relim1) 450,450,250
continue

***turbulent, or transition laminar-turbulent flow regimes.

lregim(1,k)=3
a2=a22
if(re.lt.eps .or. debs(a2).lt.eps) goto 300
rec=dmax1(re, relim2)
rea2=rec*a2
a1=a12
a3=a32
a4=a42
gra3=gra*a3
pra4=pra*a4
rnu=a1*rea2*gra3*pra4
rnur=rnu

if(re-relim2) 270,450,450
continue

***transition laminar-turbulent flow regime. calculate nu
***number using a linear interpolation as function of re
***number between the values of nu(relim1) and nu(relim2).

lregim(1,k)=2
delnu=rnur-rnul
daire=relim2-relim1
rnu=rnul+(re-relim1)*delnu/daire
goto 450

continue

c***nu=a1*(gra*a3)*(pra*a4)

gpr=gpr
if(gpr-gpr2) 330,390,390
continue
if(gpr-gpr1) 350,350,370
continue

lregim(1,k)=4
a1=a11
a3=a31
a4=a41
goto 420

continue

lregim(1,k)=5
a1=a12
a3=a32
a4=a42
goto 420

continue

lregim(1,k)=6
a1=a13
a3=a33
a4=a43
c
420 continue
rnu=nu**(gr**a3)*(pr**a4)
450 continue
c**compute the convective heat transfer coef: h=nu*k/d
c**and the heat flux: q=h*(tfluid-twall)
c
480 continue
q2left(1,k)=ht1(1,k)*((temp(1,k)-t(1,k,1))
500 continue
return
end
subroutine core(t,told,rtube,z,tem,ent,p,ro,cp,q2left,q2right,
& h1,h2,tex,roext,z0,t0,etim0,dz,drtube,qvol,qiheat,
& lntec,kmin,kmax,jmax,mater,evect,fvect,
& argf,argc,dwtfl1,dwtfr,
& dq2tl1,dq2tr,avect,bvect,cvect,dvect,iregim,
& vtime,qttim,qttime,gttime,etim1,etitim,ntime,
& lated,ichan,jconnx,kaxmax,nregmx,ntimeh,ntimh2)

c implicit real*8(a-h,o-z)
c**this is the core module’s main routine, sk apr84.
c
dimension t(ichan,kaxmax,jconnx),told(ichan,kaxmax,jconnx),
& rtube(ichan,nregmx,jconnx)
c
dimension z(ichan,kaxmax),temp(ichan,kaxmax),ent(ichan,kaxmax),
& pl(ichan,kaxmax),ro(ichan,kaxmax),cp(ichan,kaxmax),
& q2left(ichan,kaxmax),q2right(ichan,kaxmax),ht1(ichan,kaxmax),
& h1(hchan,kaxmax),tex(ichan,kaxmax),roext(ichan,kaxmax),
& z0(ichan,kaxmax),t0(ichan,kaxmax),etim0(ichan,kaxmax),
& dz(ichan,nregmx),drtube(ichan,nregmx),qvol(ichan,nregmx),
& qiheat(ichan,nregmx),lntag(ichan,nregmx),kmin(ichan,kaxmax),
& kmax(ichan,nregmx),jmax(ichan,nregmx),mater(ichan,nregmx),
& evect(kaxmax,jconnx),fvect(kaxmax,jconnx),iregim(ichan,kaxmax)
c
dimension argf(kaxmax),argc(kaxmax),dwtfl1(kaxmax),dwtfr(kaxmax),
& dq2tl1(kaxmax),dq2tr(kaxmax),avect(jconnx),bvect(jconnx),
& cvect(jconnx),dvecj(jconnx)
c
dimension vtime(ichan,kaxmax,ntimeh),qttim(ichan,kaxmax,ntimeh),
& qttime(ichan,kaxmax,ntimeh),gttime(ichan,ntimh2),etitim1,
& t(ichan,ntimeh),etim1(ichan,ntimeh),ntimh2(ichan)
c
common/sea/chieng(7),rtube(7),rtubes(7),rheat(7),
& aflow(7),del(7),nreg(7),klast(7),w(7),void(7),delp(7),
& delu7(7),deltd(7),delu7(7),delu7(7),gcost(7),
& timev(7),tvoid(7),rntim(7),rntime(7),
& rintk(7),rintg(7),rkcon(7),rkepm(7),rkexp(7),rkloc(7),
& bmon(7),cmom(7),lvalv(7),tvoid(7),ttime0(7),ttime1(7),
& iregim(7)
c
common/logkey/istedy,iplenu,iplend,ifluid,limomen,lemerg,
& icoud,iconl,iconn,ihtr,ihtr1,ihtr2,ilq,ilmonfl,ilform,
& letopt,letab,leamb,ldtcon,ldrest,luqts,lpas,kpas,jpas,
& iterx,itrpr1,itrpr2,istep
common/time0, timef, dttimed, dttrant, time, dt

if(ifluid.eq.1) 200, 100, 1000

if(fluid.eq.1) no fluid computation
if(fluid.eq.1) solves first the mom int equation
for all core and then a decoupled energy equation
for each channel "i" in the core.
if(fluid.gt.1) are options to be used in the future for
solving the fluid balance equations. Now is dummy.

100 continue

if(lenerg.eq.4) the courant stability condition is not
violated. if yes, reduce dt for further computations.

if(lenerg.eq.4) 150, 120, 150
120 continue

call stab11(ro, dz, kmin, kmax, aflow, w, wold, nreg,
& dt, lenerg, ldt, lchan, kaxmax, nregmx)

150 continue

solve the mommentum integral equation

lcqfl=1

if(fluid(1, int, p, ro, cp, q2left, z0, t0, ettim0,
& q1heat, int, m, kmax, ch1eng, rtube1, aflow, b, wold, nreg, klast,
& w, wold, delp, delt, deld, delhu, delhe, gcost, ttim0, lregim,
& ttime, tvoid, rintf, rintk, rintg, rkcon, rkexp, rkloc,
& bmom, cmom, lvalv, lvoid, argf, argg, dq2tf1,
& vtime, qatime, qbtim, qatim, ettim, ettim, nptim,
& lenerg, lcqfl, lsted, ldummy, lchan, kaxmax, nregmx, ntimeh,
& ntimh2)

if(lenerg.ne.5 .and. lenerg.ne.6) goto 200

check whether for semi-implicit eqn solns (lenerg=5 or 6),
the courant stability condition is not violated. If
yes (ldt.ne.1), reduce dt and solve again the mom int
eqn.

call stab11(ro, dz, kmin, kmax, aflow, w, wold, nreg,
& dt, lenerg, ldt, lchan, kaxmax, nregmx)

if(ldt.ne.1) goto 150

200 continue

for each channel "i", solves simultaneously the
heat conduction equation in the wall (if,icondu,ne,0)
and energy and state equations in the fluid (if,ifluid,
ne,0).

do 500 i=1,lchan

if(icondu) 250, 250, 220

if
```c
220 continue
250 continue
270 continue
300 continue
350 continue
370 continue
400 continue
430 continue
450 continue
500 continue
1000 continue
```
c
*** dummy

return
end

subroutine write (t, z, tem, ent, p, ro, q2left, q2right,
& h1, h2, text, z0, t0, lregim, kmin, kmax, jmax,
& w, delt, delt, delh, dreg, klast, livalv, lregim,
& time, dt, tupper, tdown, hdwn, wret,
& ffluid, lcondu, iter, ipas, kpas, jpas,
& ichan, jconnx, kaxmax, nregmx)

   implicit real*8 (a-h, o-z)

   c
   *** this is the code output print-out routine.
   c
   sk, may84.
   c
   dimension t(ichan, kaxmax, jconnx)
   c
   dimension z(ichan, kaxmax), tem(ichan, kaxmax), ent(ichan, kaxmax),
   & p(ichan, kaxmax), ro(ichan, kaxmax), q2left(ichan, kaxmax),
   & q2right(ichan, kaxmax), h1(ichan, kaxmax), h2(ichan, kaxmax),
   & text(ichan, kaxmax), z0(ichan, kaxmax), t0(ichan, kaxmax),
   & lregim(ichan, kaxmax), kmin(ichan, nregmx), kmax(ichan, nregmx),
   & jmax(ichan, nregmx)
   c
   dimension w(ichan), delt(ichan), delt(ichan), delh(ichan),
   & delh(ichan), nreg(ichan), klast(ichan), livalv(ichan),
   & lregim(ichan)
   c
   ipl+ichan+1
   write(2,1000) iter, time, dt
   write(2,1100)
   do 200 i=1, ichan
   tu+tupper+deltu(i)
   hu+hupper+deltu(i)
   td+tdown+delt(1)
   hd=hdwn+delt(1)
   write(2,1200) i, w(i), tu, td, hu, hd, livalv(i), lregim(i)
   200 continue
   write(2,1200) ipl, wret
   do 500 i=1, ichan, ipas
   write(2,1300) i
   if(ifluid, it, i) goto 350
   write(2,1400)
   kk=klast(i)
   do 300 k=1, kk, kpas
   write(2,1500) k, z(1,k), p(1,k), tem(1,k), ent(1,k),
   & ro(1,k), text(1,k), h1(1,k), h2(1,k),
   & q2left(1,k), q2right(1,k), z0(1,k), t0(1,k), lregim(1,k)
   300 continue
   c
   350 continue
   if(lcondu, it, i) goto 500
   write(2,1600) i, (j, j=1, jconnx, jpas)
   nmax=nreg(i)
   do 450 n=1, nmax
   jmaxmax(jmax(i, n))
   kj=kmax(i, n)
   if(n-1) 370, 370, 380
370 continue
   k1=kmin(1,n)
goto 400
380 continue
   k1=kmin(1,n)+1
400 continue
do 450 k=k1,k2,kpas
   write(2,1700) k,(t(i,k,j),j=1,jmaxr,jpas)
450 continue

C***continue with next channel 1
C
500 continue
return
990 format(10,1pd10.3))
1000 format(///5x,46("-"),/5x,"iter="13,5x,
& "time="1pd10.3,5x,"dt="1pd10.3,
& /5x,46("-"),/)
1100 format(///5x,"flow rate in channels and plena's
& temperatures and enthalpies"/5x,61("-"),/2x,
& "channel flow rate t up plen t low plen",2x,
& "h up plen h low plen channel flow",4x,
& "no. (kg/s) (k) (k)*,8x,
& "(j/kg) (j/kg) state regim"/)
1200 format(///6.4x,1pd10.3,4(1x,1pd11.4),16,17)
1300 format(///5x,"fluid data for channel 1="13,
& /5x,28("-"),/)
1400 continue(///2x,"k","4x,"high","4x,"pressure","4x,
& "temper.","3x,"enthalpy","4x,"density","4x,"tem air",
& 3x,"htc left","2x,"htc right","2x,"q2 left","3x,
& "q2 right","5x","z0","8x","t0","4x,"ht",
& /8x,"(m)","5x","(n/m2)","7x","(k)","6x","(j/kg)","5x,
& ") (kg/m3)","6x","(k)","5x","(w/m2-k)","2x","(w/m2-k)","3x,
& ") (w/m2)","4x","(w/m2)","5x","(m)","7x","(s)","4x","rg"/)
1500 format(///5x,"temperature profile in the solid structure
& of channel 1="13,2x,"(deg k)",/5x,68("-"),
& //2x,"k",11(2x,"t(1,k")",12,"=")
1700 format(13,11(1x,1pd10.3))
end