THE CONTROL OF FLOAT ZONE INTERFACES BY THE USE OF SELECTED BOUNDARY CONDITIONS

By L. M. Foster and J. McIntosh
Science Applications, Inc.
2109 W. Clinton Avenue, Suite 800
Huntsville, Alabama 35805

Final Report

December 1983

Prepared for
NASA-MARSHALL SPACE FLIGHT CENTER
Marshall Space Flight Center, Alabama 35812
The main goal of the float zone crystal growth project of NASA's Materials Processing in Space Program is to thoroughly understand the molten zone/freezing crystal system and all the mechanisms that govern this system.

The purpose of this effort was to study and compute the surface boundary conditions required to give flat float zone solid-melt interfaces. The results of this study provide float zone furnace designers with better methods for controlling solid-melt interface shapes and for computing thermal profiles and gradients. Documentation and a user's guide were provided for the computer software required during this study.
One of the main goals of the Float Zone (FZ) growth project of NASA's Materials Processing in Space Program is to thoroughly understand the molten zone/freezing crystal system and all the mechanisms that govern this system. To accomplish this, the melt and interface properties, the heat and mass flows, and the dependencies of these on each other and on growth rate and g levels must be studied.

Since the float zone process involves two solid-melt interfaces, possible gas interfaces, heat and mass transfers, various driving forces and complex heating sources, an analysis of the entire process would be very complex. For an initial investigation, a more feasible approach is to examine each component of the process separately, particularly if mathematical models are to be manageable. The three principal components are: (1) the shapes of the melt and solid-melt interfaces, (2) the heat and mass transfers, and (3) the heating and cooling sources. This study combined facets of all three components.

The purpose of this 12-month effort was to study and compute the surface boundary conditions required to give flat FZ solid-melt interfaces. The successful completion of this study should provide FZ furnace designers with better methods for controlling solid-melt interface shapes and for computing thermal profiles and gradients.

This study was undertaken in two phases. The first phase was to investigate the solid zones surface boundary conditions required for flat solid-melt interfaces when given the melt zone surface boundary conditions. The second phase complemented the first and was to investigate the melt zone surface boundary conditions required for flat solid-melt interfaces if given the solid zones surface boundary conditions. Dual integral transform methods were used in both phases; in addition, the use of various numerical methods for differential equations and linear systems of equations were required.

Using NASA supplied data, the surface boundary conditions required for flat solid-melt interfaces were studied. In addition, complete documentation and a simple user's guide are provided for all the computer software required during this study.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0 INTRODUCTION</td>
<td>1-1</td>
</tr>
<tr>
<td>1.1 OVERVIEW AND STUDY DEFINITION</td>
<td>1-1</td>
</tr>
<tr>
<td>1.2 MATHEMATICAL STATEMENT OF THE CONTROL PROBLEMS</td>
<td>1-4</td>
</tr>
<tr>
<td>2.0 TWO CLASSICAL PROBLEMS</td>
<td>2-1</td>
</tr>
<tr>
<td>2.1 DESCRIPTION OF THE CLASSICAL PROBLEMS AND CHAPTER OUTLINE</td>
<td>2-1</td>
</tr>
<tr>
<td>2.2 SOLUTION OF PROBLEM P2-1</td>
<td>2-1</td>
</tr>
<tr>
<td>2.3 SOLUTION OF PROBLEM P2-2</td>
<td>2-7</td>
</tr>
<tr>
<td>2.4 AN UNSETTLING FACT WITH AN ADDED NICE SURPRISE</td>
<td>2-11</td>
</tr>
<tr>
<td>3.0 THE COOLING CONTROL FUNCTIONS</td>
<td>3-1</td>
</tr>
<tr>
<td>4.0 THE HEATING CONTROL FUNCTION</td>
<td>4-1</td>
</tr>
<tr>
<td>5.0 TEST CASES</td>
<td>5-1</td>
</tr>
<tr>
<td>5.1 SOLID REGION SURFACE CONTROL FUNCTIONS</td>
<td>5-1</td>
</tr>
<tr>
<td>5.2 MELT ZONE SURFACE CONTROL FUNCTIONS</td>
<td>5-5</td>
</tr>
<tr>
<td>6.0 FUTURE WORK AND UNRESOLVED ISSUES</td>
<td>6-1</td>
</tr>
<tr>
<td>6.1 VERIFICATION USING FREE BOUNDARY ALGORITHMS</td>
<td>6-1</td>
</tr>
<tr>
<td>6.2 MAINTAINING CURVED INTERFACES</td>
<td>6-1</td>
</tr>
<tr>
<td>6.3 NON-DIRICHLET BOUNDARY CONDITIONS</td>
<td>6-1</td>
</tr>
<tr>
<td>6.4 BASIS FUNCTIONS USED TO EXPAND THE CONTROL FUNCTIONS</td>
<td>6-2</td>
</tr>
<tr>
<td>6.5 APPLICATIONS OF LINEAR PROGRAMMING</td>
<td>6-2</td>
</tr>
<tr>
<td>6.6 MODELS AND REALITY</td>
<td>6-3</td>
</tr>
</tbody>
</table>

**APPENDIX**

<table>
<thead>
<tr>
<th>APPENDIX</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>A THE OWNERS MANUAL</td>
<td>A-1</td>
</tr>
<tr>
<td>B CONVERGENCE OF EQUATION (2.3.12)</td>
<td>B-1</td>
</tr>
<tr>
<td>C COMPUTER CODE LISTS</td>
<td>C-1</td>
</tr>
</tbody>
</table>
LIST OF ILLUSTRATIONS

FIGURE PAGE
1-1 Illustration of the Float Zone Process. ............... 1-3
1-2 FZ Model. ........................................ 1-5
2-1 Generalized Melt Zone ................................ 2-2
2-2 Generalized Lower Solid Region. ..................... 2-7
2-3 Nominal and Perturbed Surface Temperature ........... 2-12  
2-4 Influence of Perturbations of the Surface Temperature on the Thermal Gradient. ......................... 2-12
2-5 Material and Perturbed Surface Temperatures ........... 2-14
2-6 Influence of Perturbations of the Surface Temperature on the Thermal Gradients ......................... 2-14
3-1 FZ Lower Solid Region Problem ........................ 3-1
4-1 FZ Melt Zone Problem .................................. 4-1
5-1 Melt Zone Surface Temperature Distribution. .......... 5-2
5-2 Solid Regions' Surface Control Functions ............... 5-2
5-3 Modification of Lower Solid Region's Surface Control Functions .............................................. 5-4
5-4 Solid Region's Surface Temperature ..................... 5-6
5-5 Melt Zone Surface Control Function ..................... 5-7

TABLE PAGE
5-1 MATERIAL AND SYSTEM PARAMETERS ...................... 5-1
5-2 RELATIVE DIFFERENCES BETWEEN THE REQUIRED INTERFACE GRADIENTS AND THOSE RESULTING FROM THE USE OF THE SOLID REGIONS' SURFACE CONTROL FUNCTIONS .................. 5-3
5-3 MATERIAL AND SYSTEM PARAMETERS ...................... 5-5
5-4 RELATIVE DIFFERENCES BETWEEN THE REQUIRED INTERFACE GRADIENTS AND THOSE RESULTING FROM THE USE OF THE MELT ZONE SURFACE CONTROL FUNCTION .................. 5-8
1.0 INTRODUCTION

1.1 OVERVIEW AND STUDY DEFINITION

Silicon (Si) is used in a wide variety of electronic devices including high power rectifiers, space solar cells, infrared detector arrays and high density integrated circuits. The three principal industrial methods for growing silicon crystal ingots or boules are the float zone (FZ), Czochralski (Cz), and cold crucible methods. Because molten silicon acts as a universal solvent, Cz grown Si is plagued with crucible contamination which is intolerable for high performance optical and infrared devices. However, because the FZ process is containerless, crucible contaminants are avoided. Other advantages of FZ growth include uniformity of axial resistivity (on a macroscale), visibility of the growth region, low consumable material costs, and high growth rates. Although the cold crucible method combines many of the best features of the FZ and Cz techniques, the molten Si must be superheated and volatile dopants such as In, Ga, and Tl are unfortunately evaporated.

Because most industrial advances in the FZ growth technologies have come about empirically, detailed analysis of the growth process has not kept pace with presently used FZ methods. Theoretical modeling of the melt dynamics has led to some understanding of the growth process, but it is very incomplete. The characteristics of the FZ melt must be more accurately modeled if an understanding of the heat balance and flow, isotherm shapes, density (including inversion) and surface tension variations is to lead to better methods of controlling the growth conditions. Moreover, such studies should contribute to the design and execution of FZ experiments in low-gravity (g) environments. In addition, knowledge gained by studying silicon FZ methods should be applicable to other FZ processed materials.

As noted by E. Kern [10], the main goal of the FZ growth project of NASA’s Material Processing in Space program is to thoroughly understand the molten zone/freezing crystal system and all the mechanisms that govern this system. In addition, more optimal crystal growth conditions at g=1 and possible improvements made by processing in near zero-g environments need to be investigated. To accomplish this, the melt and interface properties, the heat and mass flows, and the dependencies of these on each other and on the growth rate and g levels must be studied.

To transform a polycrystalline material into a single crystal, it is not always necessary to melt the entire sample or charge before growing the desired monocrystal. In some cases, it is possible to melt a small portion of the original charge, translate this molten zone through the charge, and hopefully leave a monocrystal behind the translating molten zone. The actual heating sources for this type zone melting process are varied and include induction, resistance, electron beam, and laser beam. The molten zone itself can be moved through the charge by either moving the heating source over the charge or by moving the charge through the heating source. The actual charge may, but need not be, contained in some type of crucible or ampoule. If no container is involved, the technique is called a float zone method and is used for reactive or high melting point materials. For most float zone applications,
the molten zone is held intact by surface tension with the occasional aid of a magnetic field [6]. A simple illustration of the float zone technique is given in Figure 1-1.

In order to reduce nonuniformities in such things as resistivities and defect distributions, for example, the entire solid-melt system must be characterized. Since the float zone process involves two solid-melt interfaces, possible gas interfaces, heat and mass transfers, various driving forces and complex heating sources, an analysis of the entire float zone process would be very complex. A more feasible approach (at least for an initial investigation) is to examine each component of the system separately, particularly if the mathematical models are to be manageable. Three principal system components are: (1) the shapes of the melt and solid-melt interfaces, (2) the heat and mass transfers, and (3) the heating and cooling sources. This study combines facets of all three components.

While many investigators, e.g., R. Brown [2], R. Naumann [14], and W. Wilcox [20], are making significant progress studying the solid-melt interface shapes and the thermal gradients at the solid-melt interfaces for float zone and analogous systems subject to specified surface boundary conditions, the principal thrust of this effort was to study and compute the surface boundary conditions required to give flat FZ solid-melt interfaces.

The completion of this study hopefully results in a better understanding of the FZ diffusion and growth mechanisms and should provide FZ furnace designers with better methods for controlling solid-melt interface shapes and for computing thermal profiles and gradients. Moreover, the methods developed in this study should aid in the design of FZ heaters that achieve the required melt fluxes with minimal energy expenditures and, hence, perhaps reduce the system power requirements (a natural concern for any long-term, low-g FZ experiment). In particular, if radio frequency heating is used, the methodology developed in this study should be useful for computing the performance requirements and position of auxiliary heating and insulation required for the proper thermal profiles. In addition, the methodology developed in this effort might provide, for future studies, a starting point for the more complex and realistic case of a slightly concave solid-melt interface.

This study was performed in two phases. The first phase analyzed the solid zones' surface boundary conditions required for flat solid-melt interfaces when given (a priori) the melt zone surface boundary conditions. The second phase complemented the first and analyzed the melt zone surface boundary conditions required for flat solid-melt interfaces when given (a priori) the surface boundary conditions for the solid zones. Dual integral transform methods were used in both phases; in addition, both phases required the use of various numerical methods for boundary value problems. Although such a study has apparently never before been undertaken, analogous studies for the Bridgman-Stockbarger method have been completed by L. Foster [7], [8].

Mathematical descriptions of the problems posed above are stated in Section 1-2. In Chapter 2, various mathematical tools are developed followed by some rather interesting examples. The methodologies used to compute the surface
Figure 1-1 Illustration of the Float Zone Process
boundary conditions for the solid regions and melt zone surface boundary conditions required for FLAC interface shapes are developed in Chapters 3 and 4. The results of various test cases using NASA supplied data are presented in Chapter 5 and recommendations for future efforts are given in Chapter 6. A simple user's guide to various computer codes (listed in Appendix C) implementing the methods described in Chapters 2, 3, and 4 is presented in of Appendix A. Appendix B contains the proof of a claim made in Section 2.3.

1.2 MATHEMATICAL STATEMENT OF THE CONTROL PROBLEMS

Concise mathematical statements of the problems described in the previous section are given next using the numbered equations in the FZ model shown in Figure 1-2. First suppose that the melt zone surface temperature is some a priori known (by design or happenstance) distribution \( h(x) \) (Figure 1-2, Equation (FZ8)). Since the temperature at both of the assumed flat solid-melt interfaces is the material melting point (Equations (FZ1) and (FZ3)), the temperature distribution in the melt zone is known and may be computed by the method described in Section 2.2. Hence, the axial thermal gradients in the melt zone at both of the solid-melt interfaces are known (see Section 2.2). Invoking Equations (FZ2) and (FZ4), the solid regions' axial thermal gradients at the interfaces are also known.†

For the moment, consider the lower solid region \((x<0\) in Figure 1-2) and let \( B(r) \) denote the known required thermal gradient* in the solid region at the interface \((x=0)\), i.e.,

\[
T_x(0,r) = B(r), \quad 0 < r < 1
\]  

(1.2.1)

The basic idea is to compute a temperature distribution \( f(x) \), \( x \geq 0 \), (henceforth called a surface control function), to be maintained on the surface of the lower solid region such that the resulting temperature distribution, \( T(x,r) \), for the lower solid region satisfies Equation (1.2.1). This is concisely stated in Problem P1-1.

† Equations (FZ2) and (FZ4) of Figure 1-2 guarantee the conservation of energy at the solid-melt interfaces \((x=0\) and \( x=Q)\). \( k_s \) and \( k_l \) are the solid and liquid thermal conductivities while \( \Lambda \) is product of the growth rate, solid density and latent heat of fusion [15].

* Standard mathematical nomenclature is used in this report. Both the operator and subscript notation are used for partial derivatives, e.g., \( \frac{\partial T}{\partial x} \) and \( T_x \) both denote the partial derivative of \( T(x,r) \) with respect to \( x \). For functions of one variable, the 'prime' convention for derivatives is observed, e.g., \( h''(x) \) denotes the second derivative of \( h(x) \). The Laplacian operator is denoted by \( \Delta \) and is, in cylindrical coordinates,

\[
\Delta T = T_{xx} + T_{rr} + \frac{1}{r} T_r.
\]
\( \Delta T = \rho_s \frac{\partial T}{\partial x} \)
\( T(x,1) = g(x) \) \hspace{1cm} (FZ5)

\( \Delta T = \rho_s \frac{\partial T}{\partial x} \)
\( T(x,1) = h(x) \) \hspace{1cm} (FZ7)

\( \Delta T = \rho_s \frac{\partial T}{\partial x} \)
\( T(x,1) = f(x) \) \hspace{1cm} (FZ9)

\( (FZ1) \quad T_{x=Q^-} = T_{x=Q^+} = \) melt pt.

\( (FZ2) \quad -k_L \frac{\partial T}{\partial x} \bigg|_{x=Q^-} + k_s \frac{\partial T}{\partial x} \bigg|_{x=Q^+} + \Delta = 0 \)

\( (FZ3) \quad T_{x=0^+} = T_{x=0^-} = \) melt pt.

\( (FZ4) \quad -k_L \frac{\partial T}{\partial x} \bigg|_{x=0^+} + k_s \frac{\partial T}{\partial x} \bigg|_{x=0^-} + \Delta = 0 \)

Figure 1-2 FZ Model
Problem PI-1. Compute \( f(x) \) such that the solution \( T(x,r) \) of

\[
\begin{align*}
\Delta T &= P_s \frac{\partial T}{\partial x}, \quad x < 0, \ 0 < r < 1 \\
T(x,1) &= f(x), \quad x < 0, \\
T(0,r) &= A(r), \quad 0 < r < 1
\end{align*}
\] (1.2)

also satisfies the boundary condition (1.2.1).

The constant \( P_s \) is the solid Peclet number \([20]\) and from a practical viewpoint, the function \( A(r) \) in Equations (1.2.1) is the material melting point. Moreover, since numerical methods will be employed, Condition (1.2.1) will only be satisfied approximately in practice.

Having stated the question for the lower solid region, the corresponding question for the upper solid region is analogous. Namely, let \( B(r) \) be the required thermal gradient in the upper solid region at the upper solid-melt interface \( (x=Q) \), i.e.,

\[
\frac{\partial T}{\partial x}(x,r) = B(r), \quad x = Q, \ 0 < r < 1.
\] (1.2.3)

Then find a surface temperature distribution \( g(x), \ x>Q \) (henceforth also called a surface control function), to be maintained such that the resulting temperature distribution, \( T(x,r) \), for the upper solid region satisfies (1.2.3). This is concisely stated in Problem PI-2.

Problem PI-2. Determine \( g(x) \) such that the solution \( T(x,r) \) of

\[
\begin{align*}
\Delta T &= P_s \frac{\partial T}{\partial x}, \quad x > Q, \ 0 < r < 1 \\
T(x,1) &= g(x), \quad x > Q, \\
T(Q,r) &= A(r), \quad 0 < r < 1
\end{align*}
\] (1.2.4)

also satisfies the boundary condition (1.2.3).

† To help make various computer codes listed in Appendix C easier to follow, the thermal gradients in Problem PI-1 and PI-2 are both represented by the same symbol, \( B(r) \); however, these gradients are not necessarily the same. For generality, a similar remark holds for the symbol \( A(r) \).
As before, in practice $A(r)$ is set to the melting temperature and Equation (1.2.3) will only be satisfied approximately due to the numerical solution of the problem.

Problems P1-1 and P1-2 stated above belong to the class of so called ill-posed or over-under posed problems. Unlike most classical second order boundary value problems where each portion of the boundary surface is assigned a boundary condition, Problems P1-1 and P1-2 have two boundary conditions (over-posed) assigned to each of their respective solid-melt interfaces (for example, in Problem P1-1, $T(0,r)=A(r)$ and $T_x(0,r)=B(r)$) and no boundary condition (under-posed) assigned to the lateral surfaces of either of the solid regions. Indeed, part of the problem is to determine the proper missing boundary condition (for example, $T(x,1)=f(x)$ for Problem P1-1) so as to relax the overposing of boundary conditions at the solid-melt interfaces. The solutions of Problem P1-1 and P1-2 are the subject of Chapter 3.

Next suppose that the solid regions' surface temperature distributions $f(x)$ and $g(x)$ (see Figure 1-2, Equations (FZ6) and (FZ10)) are fixed (by design or happenstance). Since the temperature at both of the solid-melt interfaces is assumed to be the melting temperature for FZ applications, the temperature distributions in both of the solid regions are computable (see Section 2.3). Hence the axial thermal gradients in the solid regions at the solid-melt interfaces are computable. Thus, the axial thermal gradients in the melt zone at the solid-melt interfaces ($x=0$ and $x=Q$) are known after invoking Equations (FZ2) and (FZ4) of Figure 1-2 and are denoted by

\[
\begin{align*}
T_x(0,r) &= A(r), \quad 0 < r < 1 \\
T_x(Q,r) &= B(r), \quad 0 < r < 1 
\end{align*}
\]  

(1.2.5)

The problem is to determine a surface temperature $h(x)$, $0 < x < Q$ (henceforth called the melt zone surface control function), to be maintained on the melt zone surface such that the resulting temperature distribution, $T(x,r)$, for the melt zone satisfies (1.2.5). This is concisely stated in Problem P1-3.

**Problem P1-3** Determine $h(x)$ such that the solution $T(x,r)$ of

\[
\begin{align*}
\Delta T &= P \frac{\partial T}{\partial x}, \quad 0 < x < Q, \quad 0 < r < 1 \\
T(x,1) &= h(x), \quad 0 < x < Q \\
T(Q,r) &= C(r), \quad 0 < r < 1 \\
T(Q,r) &= D(r), \quad 0 < r < 1 
\end{align*}
\]  

(1.2.6)

also satisfies the boundary conditions (1.2.5)
The constant $P_l$ is the liquid Peclet number and from a FZ point of view, $C(r)$ and $D(r)$ equal the material melting point. As with Problems P1-1 and P1-2, the numerical nature of the proposed solution method (the subject of Section 4.0) means that Conditions (1.2.5) will only be approximately satisfied.
2.0 TWO CLASSICAL PROBLEMS

Before turning to those moral and mental aspects of the matter which present the greatest difficulties, let the inquirer begin by mastering more elementary problems.

--Sherlock Holmes, "A Study in Scarlet"

2.1 DESCRIPTION OF THE CLASSICAL PROBLEMS AND CHAPTER OUTLINE

Before developing methods to compute the melt zone and solid regions' surface control functions which will yield the desired flat solid-melt interfaces, two more elementary problems must be dispatched. These are:

Problem P2-1: Given a surface temperature distribution for the melt zone, compute the resulting interior temperature distribution of the melt zone.

Problem P2-2: Given a surface temperature distribution for one of the semi-infinite solid regions, compute the resulting interior temperature distribution for that region.

In addition to solving Problems P2-1 and P2-2, methods for approximating the interface gradients are presented in this chapter. The techniques developed to solve Problems P2-1 and P2-2 will have three important functions in this study. First, they will be used to generate the solid and melt zone gradients required at the interfaces. Second, and probably most important, the solution techniques for Problems P2-1 and P2-2 will introduce the essential definitions and dual integral transforms which will be used later to compute the desired surface control functions (Chapters 3 and 4). Third, these techniques will be used to study how well (or poorly) the computed melt zone (or solid region) surface control function performs.

Problems P2-1 and P2-2 are resolved in Sections 2.2 and 2.3 respectively. Some numerical test cases are discussed in Section 2.3 along with two examples with correspondingly important remarks.

2.2 SOLUTION OF PROBLEM P2-1

Suppose the melt zone of Figure 1-2 is isolated (and perhaps translated) as displayed in Figure 2-1.

To reduce the terminology, the solid-melt interfaces will henceforth be referred to merely as the interfaces. The axial thermal gradient in a solid region (or melt zone) at an interface will be referred to as a solid (region) (a melt zone) interface gradient.
Figure 2-1  Generalized Melt Zone

Realistically, the melt zone end temperatures $A(r)$ and $B(r)$ are both the material melting temperature; however, for sake of illustration, we require only that $A(r)$ and $B(r)$ be sufficiently smooth. Problem P2-1 can then be mathematically stated as:

Problem P2-3: Determine $T(x,r)$ such that

\[
\Delta T = \frac{\partial^2 T}{\partial x^2}, \quad 0 < r < 1 \text{ and } x_o < x < x_N
\]  

(2.2.1)

\[
T(x_o,r) = A(r), \quad 0 < r < 1
\]  

(2.2.2)

\[
T(x_N,r) = B(r), \quad 0 < r < 1
\]  

(2.2.3)

\[
T(x,1) = h(x), \quad x_o < x < x_N
\]  

(2.2.4)

and

\[
T_r(x,0) = 0, \quad x_o < x < x_N
\]  

(2.2.5)

where $A(r)$, $B(r)$ and $h(x)$ are sufficiently smooth, $A(1)=h(x_o)$ and $B(1)=h(x_N)$, and $P$ (the Peclet number with the subscript "I" suppressed for convenience) is a positive constant.

Before solving Problem P2-3, some notation is in order:

Notation N2-1:

(1) $\mathcal{A}(r) = A(r)-A(1)$
(ii) \( \mathfrak{R}(r) = B(r) - B(1) \)

(iii) \( \psi(r) = J_0(\lambda_n r) \) where \( \lambda_1 < \lambda_2 < \lambda_3 < \ldots \) is the increasing sequence of real roots of the Bessel function \( J_0 \).

(iv) \( G(x) = P'\bar{h}(x) - h''(x) \)

Solution Technique: The basic idea is to assume the solution \( T(x,r) \) is the sum of the lateral surface temperature \( h(x) \) plus some unknown function \( \Theta(x,r) \), i.e.,

\[
T(x,r) = \Theta(x,r) + h(x)
\]

Problem P2-3 can then be recast as:

\[
\Delta \theta = P \frac{\partial \Theta}{\partial x} + \mathcal{G}, \quad 0 < r < 1 \quad \text{and} \quad x_0 < x < x_N \quad (2.2.6)
\]

\[
\Theta(x_0,r) = \mathcal{A}(r), \quad 0 < r < 1 \quad (2.2.7)
\]

\[
\Theta(x_N,r) = \mathfrak{B}(r), \quad 0 < r < 1 \quad (2.2.8)
\]

\[
\Theta(x,1) = 0, \quad x_0 < x < x_N \quad (2.2.9)
\]

and

\[
\theta_r(x,0) = 0, \quad x_0 < x < x_N \quad (2.2.10)
\]

Although Equation (2.2.6) is more complex than Equation (2.2.1), the corresponding boundary conditions are greatly simplified. First, the Dirichlet condition (2.2.4) is replaced by a simple homogenous boundary condition (2.2.9). In addition, because \( \mathcal{A}(1) = \mathfrak{B}(1) = 0 \), the boundary conditions (2.2.7) and (2.2.8) can be further simplified by various Bessel series expansions. For the moment, assume \( \Theta(x,r) \) is expanded as

\[
\Theta(x,r) = \sum_{n=1}^{\infty} C_n(x) \psi_n(r) \quad (2.2.11)
\]

Then using the following well known property of Bessel functions [18],

\[
\int_0^1 \psi_n(r) \psi_m(r) r dr = \begin{cases} 0 & \text{if } n \neq m \\ \frac{1}{2} J_1^2(\lambda_n) & \text{if } n = m \end{cases} \quad (2.2.12)
\]

the functions \( C_n(x) \) of Equation (2.2.11) are computed to be

2-3
If the integral portion of Equation (2.2.13) is denoted by \( \tilde{\Theta}_n(x) \), then Equations (2.2.11) and (2.2.13) may be combined to form a dual integral transform pair:

\[
\Theta(x,r) = \sum_{n=1}^{\infty} \frac{2\gamma_n(r)\phi_n(x)}{J_1(\lambda_n)} \quad \text{and} \quad \tilde{\Theta}_n(x) = \int_0^1 \Theta(x,r)\psi_n(r)rdr.
\]  

Unfortunately, the desired \( \Theta(x,r) \) of Equation (2.2.14) involves \( \tilde{\Theta}_n(x) \) which in turn requires knowing \( \Theta(x,r) \); fortunately, this rather circular problem may be resolved by invoking Green's theorem. If both sides of the partial differential equation (2.2.6) are multiplied by \( \psi_n(r)rdr \) and the resulting terms integrated from \( r=0 \) to \( r=1 \), a application of Green's theorem combined with the fact that

\[
\begin{align*}
\psi_n' & - \Theta \frac{\partial}{\partial r} \psi_n |_{r=1} = 0 \\
\psi_n' & - \Theta \frac{\partial}{\partial r} \psi_n |_{r=0} = 0
\end{align*}
\]

implies

\[
\tilde{\Theta}_n(x) - \frac{2}{r} \tilde{\Theta}_n(x) - \frac{2}{\lambda_n} \tilde{\Theta}_n(x) = \tilde{G}_n(x), \quad x_0 < x < x_N
\]  

where

\[
\tilde{G}_n(x) = \int_0^1 G(x)\psi_n(r)rdr = G(x) \frac{J_1(\lambda_n)}{\lambda_n}
\]  

Since \( \mathcal{A}(1) = \mathcal{B}(1) = 0 \), the smooth functions \( \mathcal{A}(r) \) and \( \mathcal{B}(r) \) may be represented by the following Bessel expansions:

\[
\mathcal{A}(r) = \sum_{n=1}^{\infty} A_n J_0(\lambda_n r)
\]  

\[
\mathcal{B}(r) = \sum_{n=1}^{\infty} B_n J_0(\lambda_n r)
\]  

The above coefficients \( A_n \) and \( B_n \) could be computed using integral
environments [18], for example,

\[ A_n = \frac{2}{J_1^2(\lambda_n)} \int_0^1 A(r) J_0(\lambda_n r) r \, dr \]

However, to avoid the eventually required numerical integration of such representations, the coefficients \( A_n \) and \( B_n \) may be approximated using a least squares method as described at the end of this section. Combining Equations (2.2.7), (2.2.8), (2.2.12), and (2.2.14)-(2.2.18), \( \Theta_n(x) \) may be uncoupled from \( \Theta(x,r) \) as the solution of the following two point boundary value problem:

\[
\begin{align*}
\frac{d^2 \Theta_n}{dx^2} - \frac{\lambda_n^2}{n^2} \Theta_n &= \Theta_n \quad x_0 < x < x_N \\
\Theta_n(x_0) &= \frac{J_1(\lambda_n)}{\lambda_n} \\
\Theta_n(x_N) &= \frac{J_1(\lambda_n)}{\lambda_n}
\end{align*}
\]

(2.2.19)

Since \( \lambda_n > 0 \), it is well known [5] that Problem (2.2.19) has a unique solution. Although the solution of (2.2.19) could be determined by a variation of parameters method [1], such a technique inevitably requires numerical integration. A more straightforward method is to discretize (2.2.19) in the following fashion. First, the interval from \( x_0 \) to \( x_N \) is partitioned by the grid points:

\[ x_j = x_0 + j \Delta x \quad ; \quad j = 0, \ldots, M \]

where \( M \Delta x = x_N - x_0 \). Then solve the following finite difference analog of the boundary value problem (2.2.19):

\[
\frac{\mu_{j+1} - 2\mu_j + \mu_{j-1}}{(\Delta x)^2} - \frac{\mu_{j+1} - \mu_{j-1}}{2\Delta x} - \lambda_n^2 \mu_j \Theta_n(t_j), \quad j = 1, \ldots, M-1
\]

[2.2.20]
The linear system (2.2.20) is tridiagonal and guaranteed to have a solution [9] if \( P \Delta x \leq 2 \). Moreover, the solution vector \( \{ \mu_0, \cdots, \mu_N \} \) provides a second order approximation of \( u_n(x) \), i.e.,

\[
| \mu_j - \tilde{u}_n(t_j) | = O((\Delta x)^2)
\]

In addition, the boundary derivatives of \( \tilde{u}_n \) may be accurately approximated [3] by the following unbalanced finite differences:

\[
\tilde{u}_n'(x_0) = \frac{-3\mu_4 + 16\mu_3 - 36\mu_2 + 48\mu_1 - 25\mu_0}{12\Delta x}
\]

\[
\tilde{u}_n'(x_N) = \frac{3\mu_4 - 16\mu_3 + 36\mu_2 - 48\mu_1 + 25\mu_0}{12\Delta x}
\]

(2.2.21)

Since

\[
T_n(x, r) = h^*(x) + 2\sum_{n=1}^{\infty} \frac{\psi_n(r)}{J_1(\lambda_n)} \tilde{u}_n'(x)
\]

(2.2.22)

Equations (2.2.21) and (2.2.22) may be combined to approximate the axial gradients at \( x = x_0 \) and \( x = x_N \) (a very important requirement in Chapters 3 and 4).

To finish this section, a short description is given of how the coefficients \( A_n \) of Equation (2.2.17) are approximated (the same technique applies to Equation (2.2.18)). First, denote \( t_i = (i-1)/M \), \( i = 1, \ldots, M + 1 \) and select \( N << M \) (typically \( N = 20 \) and \( M = 100 \)). Define an \((M+1)\) by \( N \) array \( L \) and \((M+1)\) by \( (M+1) \) dimension vector \( b \) by the respective elements:

\[
L_{i,j} = J_0(\lambda_j t_i) \quad \text{and} \quad b_i = \Lambda(t_i)
\]

Let \( \bar{a} \) be the solution of the linear least squares problem [17, Chapter 5]:

\[
L\bar{a} = b
\]

(2.2.23)

Then the first \( N \) coefficients, \( \Lambda_n \), of (2.2.17) are approximated by

\[
\Lambda_n \approx \bar{a}_n, \quad n = 1, \cdots, N
\]
2.3 SOLUTION OF PROBLEM P2-2

Analogous to the solution technique of Problem P2-1 in Section 2.2, suppose the lower solid region of Figure 1-2 is isolated as shown in Figure 2-2.

\[ T(x, r) = A(r) \]

\[ T(x, 1) = f(x) \]

**STATE EQUATION**

\[ \Delta T = P_s \frac{\partial T}{\partial x} \]

**REMARK:** FOR FZ PROBLEM

\[ A(r) = \text{MELT PT.} \]

Figure 2-2 Generalized Lower Solid Region
Realistically, the upper end temperature \( A(r) \) of the lower solid region is the material melting temperature; however, for the sake of illustration, it is only required that \( A(r) \) be sufficiently smooth. In addition, it is assumed that the lateral surface temperature \( f(x) \) is smooth, asymptotically constant as \( x \to -\infty \) and is such that \( f' \) and \( f'' \) approach zero as \( x \to -\infty \) (loosely, this means \( f(x) \) resembles a horizontal line as \( x \) approaches \(-\infty\)). Mathematically, the lower solid region case of Problem P2-2 may be stated as:

**Problem P2-4:** Determine \( T(x,r) \) such that

\[
\Delta T = PT_x', \quad 0 < r < 1 \text{ and } x < 0 \tag{2.3.1}
\]

\[
T(0,r) = A(r), \quad 0 < r < 1 \tag{2.3.2}
\]

\[
T(x,1) = f(x), \quad x < 0 \tag{2.3.3}
\]

\[
T_r(x,0) = 0, \quad x < 0 \tag{2.3.4}
\]

and

\[
\lim_{x \to -\infty} \max_{0 < r < 1} |f(x) - T(x,r)| = 0 \tag{2.3.5}
\]

The functions \( A(r) \) and \( f(x) \) are assumed sufficiently smooth, and for compatibility, \( A(1) = f(0) \). In addition, \( \lim f(x) \) exists and is finite and both \( f' \) and \( f'' \) approach zero as \( x \to -\infty \). The constant \( P \) is assumed to be positive (the subscript "s" is suppressed for convenience).

The notation established in Section 2.2 will be retained with the exception of \( G(x) \) which now represents \( G(x) = Pf'(x) - f''(x) \). The solution technique is very similar to that used in Section 2.2 First, \( T(x,r) \) is expressed as

\[
T(x,r) = \theta(x,r) + f(x)
\]

and Equations (2.3.1) - (2.3.5) are recast as:

\[
\Delta \theta = P \theta_x + G, \quad 0 < r < 1 \text{ and } x < 0 \tag{2.3.6}
\]

\[
\theta(0,r) = A(r), \quad 0 < r < 1 \tag{2.3.7}
\]

\[
\theta(x,1) = 0, \quad x < 0 \tag{2.3.8}
\]

\[
\theta_r(x,0) = 0, \quad x < 0 \tag{2.3.9}
\]

and

\[
\lim_{x \to -\infty} \max_{0 < r < 1} |\theta(x,r)| = 0 \tag{2.3.10}
\]
The boundary value problem (BVP) given by the Equations (2.3.6) - (2.3.10) is solved in a manner similar to the solution of the BVP (2.2.6) - (2.2.10). If \( f(x) \) is represented as in Equation (2.2.17), then the BVP (2.3.6) - (2.3.10) may be transformed by the dual integral transform pair (2.2.14) into the following boundary value problem:

\[
\bar{\sigma}'' - p \bar{\sigma}' + \lambda \bar{\sigma} = \bar{G}, \quad x < 0
\]

\[
\bar{\sigma}(0) = -\frac{\lambda}{2}
\]

and

\[
\lim_{x \to \infty} \bar{\sigma}(x) = 0
\]

where \( \bar{G} \) is still defined as in (2.2.16). Using a variation of parameters technique, the solution of this BVP is given by

\[
\bar{\sigma}(x) = \left[ A_n + \frac{1}{S_n} \int_{0}^{x} \bar{G}_n e^{-\alpha_n t} dt \right] e^\alpha x
\]

\[
+ \left[ B_n - \frac{1}{S_n} \int_{0}^{x} \bar{G}_n e^{-\beta_n t} dt \right] e^\beta x
\]

where

\[
S_n = \sqrt{p^2 + 4\lambda^2}
\]

\[
\alpha_n = \frac{p + S_n}{2n}
\]

\[
\beta_n = \frac{p - S_n}{2n}
\]

\[
B_n = -\frac{1}{S_n} \int_{-\infty}^{0} \bar{G}_n e^{-\beta_n t} dt
\]

and

\[
A_n = -B_n + \frac{\phi_n J_n^2(\lambda_n)}{2}
\]

Since each summand in (2.3.11) is the product of an exponentially exploding and exponentially decaying term, the proof that \( \lim_{x \to \infty} \bar{\sigma}(x) = 0 \) is rather delicate and is reserved for Appendix B. The solution of Problem P2-4 is
Since the float zone process also involves the upper solid region of Figure 1-2, an upper region analog of Problem P2-4 must be solved. After translating the upper interface to \( x = 0 \) for convenience, the mathematical statement of such a problem is:

**Problem P2-5:** Determine \( T(x,r) \) such that

\[
\Delta T = \mathcal{P}_x, \quad 0 < r < 1, \quad x > 0
\]

\[
T(0,r) = A(r), \quad 0 < r < 1
\]

\[
T(x,1) = g(x), \quad x > 0
\]

\[
T_r(x,0) = 0, \quad x > 0
\]

and

\[
\lim_{x \to \infty} \max_{0 < r < 1} |g(x) - T(x,r)| = 0
\]

As before, \( A(r) \) and \( g(x) \) are assumed sufficiently smooth and, for compatibility, \( A(1) = g(0) \). In addition, \( \lim g(x) \) exists and is finite, both \( g' \) and \( g'' \) approach zero as \( x \to \infty \), and \( P > 0 \).

Without belaboring the details, the solution of Problem P2-5 is given by Equation (2.3.12) \( g(x) \) obviously replaces \( f(x) \) where \( \mathcal{F}_n(x) \) is still represented by Equation (2.3.11) with the \( \mathcal{G}_n, \mathcal{S}_n, \alpha_n \) and \( B_n \) unchanged but with new \( A_n \) and \( B_n \), namely

\[
A_n = -\frac{1}{S_n} \int_0^\infty \mathcal{G}_n e^{-\alpha_n t} e^{-\ln^2 t} \, dt
\]

and

\[
B_n = -A_n + A_n \frac{J_n^2(\lambda_n)}{2}
\]

As a computational aside, the numerical method described in Section 2.2 can be used to approximate the analytically defined solutions of this section. For example, in the upper solid region case, if the surface temperature \( g(x) \) is rather constant for \( x \), say, greater than some \( L \), then the solution of Problem P2-5 may be approximated for \( 0 < x < x_N \) by the solution of Problem P2-3 with \( x_N \) set to, say 3L, and \( B(r) = g(x_N) \) and \( h(x) = g(x) \). Moreover, the gradient at the translated bottom, \( x = 0 \), of the upper solid region may be accurately estimated by the approximate gradient generated by Equations (2.2.21) and (2.2.22).
Numerous test cases to numerically verify the above remarks were generated with $A(r) = 0$ (to simulate a solid-melt interface), $0.1 \leq P \leq 1.0$, and $x_N = 2L$ and $3L$ (I typically on the order of 10). The approximate temperatures (and gradients) so obtained were quite accurate.

2.4 AN UNSETTLING FACT WITH AN ADDED NICE SURPRISE

Consider for the moment Problem P2-4. If $A(r) = 0$ (to simulate a solid-melt interface) and two temperature distributions $T_1$ and $T_2$ are generated corresponding to two surface temperature conditions $f_1(x)$ and $f_2(x)$, then if $f_1 \approx f_2$, it is reasonable to expect $T_1 \approx T_2$. In addition, if $f_1 \approx f_2$ near $x = 0$, then it is also reasonable to expect that the corresponding thermal gradients of $T_1$ and $T_2$ will be close at $x = 0$. These intuitive observations are indeed true and may be rigorously proven after such concepts as "close" are precisely defined. All of this, however, might lead to the assumption that if $f_1$ and $f_2$ are not close, then the corresponding thermal gradients and temperature distributions near the simulated solid-melt interface ($x = 0$) are probably not close. This, of course, is not always true, and will be illustrated in this section by two examples. In fact, the second example will demonstrate the somewhat unsettling fact that it is quite possible for $f_1(x)$ to exponentially explode while $f_2(x)$ remains nicely bounded with the corresponding thermal gradients at $x = 0$ virtually indistinguishable. In light of the development presented in Section 1.2, this implies there might exist many varied surface control functions, all of which provide the required (or nearly so) thermal gradient at the desired interface. If this is the case, then the float zone furnace designer may have at his disposal many different prospective surface control functions to choose from (a nice surprise). For example, the designer might select a surface control function that requires a minimum of power.

The two examples in this section clearly demonstrate that small changes in the thermal gradient at the end boundary ($x = 0$) can result in a rather large change in the resulting surface control function. This, as noted before, can provide an entire family of useful surface control functions if the FZ designer is willing to permit a slight "misfit" (albeit small) between the desired and obtained thermal gradients at the end boundary ($x = 0$ for the following examples). Unfortunately, this also means that an attempt to measure the sensitivity of the required surface control functions to changes in the material or system parameters (which obviously produce changes in the desired interface thermal gradient) can be quite misleading and should probably not be attempted.

Example E2-1: In this example, $P = 0.1, A(r) = 0$ and the lower solid region case ($x < 0$) is selected. The nominal surface temperature $f_1$ is illustrated in Figure 2-3; the surface temperatures $f_2, \ldots, f_5$ (also illustrated in Figure 2-3) are perturbations of the nominal $f_1$.

Letting $T_i$ denote the thermal distributions corresponding to the surface temperatures $f_i$, the relative difference (measured in both $L^2$ and $L^\infty$ norms\(^{+}\)) between the nominal gradient of $T_i$ and each of the gradients of $T_2, i = 2, \ldots, 5,$ at the end boundary, $x = 0$, is illustrated in Figure 2-4.

\[^{+}\] For a function $h(r)$, $0 \leq r \leq 1$, the $L^2$ and $L^\infty$ norms are (respectively)

$$||h||_2 = \left[ \int_0^1 h^2(r)dr \right]^{1/2} \text{ and } ||h||_\infty = \max_{0 \leq r \leq 1} |h(r)|.$$
Figure 2-3 Nominal and Perturbed Surface Temperatures

Figure 2-4 Influence of Perturbations of the Surface Temperature on the Thermal Gradient
Note that even for the cases where $f_1$, the nominal surface temperature, is perturbed relatively close to the end boundary ($x = 0$), the corresponding perturbations of the thermal gradients at the $x = 0$ boundary are still remarkably close to that of the nominal gradient.

Example E2-2: In this example, $P = 0.1$, $A(r) = 0$, and the upper solid region (translated to $x > 0$) is selected. Suppose it is required that the thermal gradient at $x = 0$ be identically 1, i.e., $T_x(0,r) = 1$. A particular surface control function $g_{1}(x)$ which will give the desired result is the exponentially growing surface temperature:

$$g_1(x) = -10 + 10\exp(x/10), \; x>0$$

In fact, the corresponding thermal distribution $T_1$ is identical to $g_1$. Suppose $g_2(x), \ldots, g_5(x)$ are surface control functions that equal $g_1(x)$ on an interval $[0,z_i]$, $i = 2, \ldots, 5$ but are asymptotically constant as $x \to \infty$ (see Figure 2-5). The relative $L^2$ and $L^\infty$ differences between the thermal gradients at $x = 0$ of the corresponding temperature distributions $T_2, \ldots, T_5$ and the thermal gradient of $T_1$ is illustrated in Figure 2-6. Note that even when $g_2$ (a bounded surface temperature) separates from $g_1$ (an unbounded surface temperature) rather close to the end boundary ($x = 0$), the two corresponding thermal gradients at $x = 0$ are remarkably close (see Figure 2-3, $z = 0.5$).
Figure 2-5 Nominal and Perturbed Surface Temperatures

\[ s_1 = -10 + 10 \exp(x/10) \]

Figure 2-6 Influence of Perturbations of the Surface Temperature on the Thermal Gradients

\[ \text{Relative } L^1 \text{ Difference} \]

\[ \text{Relative } L^2 \text{ Difference} \]

\[ x, \text{ the distance from } x=0 \text{ where } s_1(x) \text{ is perturbed} \]
3.0 THE COOLING CONTROL FUNCTIONS

It is an old maxim of mine that when you have excluded the impossible, whatever remains, however improbable, must be the truth.

―Sherlock Holmes, "The Adventure of the Beryl Coronet"

The main thrust of this chapter is to provide solution methods for Problems Pl-1 and Pl-2. Beginning with Problem Pl-1, suppose a temperature distribution \( T(x,r) \) is required to satisfy two known boundary conditions

\[
T(0,r) = A(r) \quad (3.0.1)
\]
\[
T_x(0,r) = B(r) \quad (3.0.2)
\]

at the lower solid region's end boundary (the melt-solid interface in practice) as depicted in Figure 3-1.

\[ \Delta T = \int_1^y \frac{3T}{3x} \]  
(State Equation)

\[
T(0,r) = A(r) \quad \text{KNOWN}
\]
\[
\frac{3}{3x}T(0,r) = B(r) \quad \text{KNOWN}
\]
\[
T(x,1) = f(x) \quad \text{UNKNOWN}
\]

Remark: For FZ problems,
\( A(r) = \) Melting Temperature

Figure 3-1 FZ Lower Solid Region Problem
Recall from Section 1.2 that the problem is to find some (at this point unknown) cooling control function \( f(x), \ x < 0 \), such that the solution of the well-posed boundary value problem:

\[
\Delta T = P \frac{T}{s_x}, \ x < 0, \ 0 < r < 1 \tag{3.0.3}
\]

\[
T(0,r) = A(r), \ 0 < r < 1 \tag{3.0.4}
\]

and

\[
T(x,1) = f(x), \ x < 0 \tag{3.0.5}
\]

also satisfies the addition boundary condition (3.0.2). The basic idea of the proposed method is to solve the boundary value problem (3.0.3)-(3.0.5) by the method described in Section 2.3 and, in the process find a sufficient number of conditions to allow the calculation of the desired, but unknown, \( f(x) \). First, from a practical point of view, any viable control function \( f(x) \) should become rather constant as the distance from the lower interface increases. Thus it is expected that:

\[
\lim_{x \to -\infty} f(x) \text{ exists and is finite} \tag{3.0.6}
\]

and

\[
\lim_{x \to -\infty} f'(x) = \lim_{x \to -\infty} f''(x) = 0 \tag{3.0.7}
\]

Proceeding as in Section 2.3, denote

\[
T(x,r) = \Theta(x,r) + f(x)
\]

\[
G(x) = Pf' - f''
\]

\[
A(r) = A(r) - f(0)
\]

\[
B(r) = B(r) - f'(0)
\]

For \( f(x) \) to be compatible with \( A(r) \) and \( B(r) \),

\[
A(1) = f(0) \tag{3.0.8}
\]

and

\[
B(1) = f'(0) \tag{3.0.9}
\]

Then Equations (3.0.2)-(3.0.5) reduce to

\[\text{For the moment, suppress the solid subscript "s", i.e., } P_s = P.\]
\[ \Delta \theta = P \theta_{x} + G(x) \]
\[ \theta(0, r) = \mathcal{A}(r) \]
\[ \theta_x(0, r) = \mathcal{B}(r) \]
\[ \theta(x, 1) = 0 \]

(3.0.10)

Denote \( \psi_{n}(r) = J_{0}(\lambda_{n}r) \) where \( \lambda_{1} < \lambda_{2} < \lambda_{3} < ... \) are the real roots of the zero order Bessel function. As in Chapter 2, let

\[ \theta(x, r) = \sum_{M=1}^{\infty} \frac{2 \psi_{M}(r)}{J_{1}(\lambda_{M})} \overline{\theta}_{M}(x) \]

(3.0.11)

and

\[ \overline{\theta}_{M}(x) = \int_{0}^{1} \theta(x, r) \psi_{M}(r) r dr \]

(3.0.12)

form a dual integral transform pair. If \( \mathcal{A}(r) \) and \( \mathcal{B}(r) \) are expanded in the Bessel series:

\[ \mathcal{A}(r) = \sum_{n=1}^{\infty} \mathcal{A}_{n} J_{0}(\lambda_{n}r) \]

(3.0.13)

\[ \mathcal{B}(r) = \sum_{n=1}^{\infty} \mathcal{B}_{n} J_{0}(\lambda_{n}r) \]

(3.0.14)

and denoting

\[ \overline{G}_{M}(x) = \int_{0}^{1} G(x) \psi_{M}(r) r dr = G(x) \frac{J_{1}(\lambda_{M})}{\lambda_{M}} \]

(3.0.15)

then operating on (3.0.10) by the integral transform (3.0.12) yields

\[ -\lambda_{M} \overline{\theta}_{M} + \overline{\theta}'_{M} = p \overline{\theta}'_{M} + \overline{G}_{M} \] \( x < 0 \)

(3.0.16)

\[ \overline{\theta}_{M}(0) = A_{M} \frac{J^{2}_{1}(\lambda_{M})}{2} \]

(3.0.17)

3-3
In practice, the Bessel coefficients, \(\mathcal{A}_n\) and \(\mathcal{B}_n\) in (3.0.13) and (3.0.14) are approximated by least square methods as described in Section 2.2. In addition, for FZ applications \(\mathcal{A}_n = 0\) because \(A(r)\) is constant (the material melting temperature). Denoting

\[
S_M = \sqrt{P^2 + 4\lambda_M^2} \quad (3.0.13)
\]

\[
\alpha_M = \left( P + S_M \right) / 2
\]

and

\[
\beta_M = \left( P - S_M \right) / 2,
\]

the solution of (3.0.16)-(3.0.18) is then:

\[
\bar{\mathcal{B}}_M(x) = \frac{J_1^2(\lambda_M)}{2S_M} \left( \begin{array}{c} \mathcal{A}_M - \mathcal{B}_M \\ \mathcal{B}_M \end{array} \right) e^{\alpha_M x} + e^{\alpha_M x} \int_0^x \frac{\bar{G}_M(t)}{S_M} e^{-\alpha_M t} dt
\]

\[
+ \frac{J_1^2(\lambda_M)}{2S_M} \left( \begin{array}{c} \mathcal{A}_M \mathcal{A}_M - \mathcal{B}_M \\ \mathcal{B}_M \end{array} \right) e^{\beta_M x} + e^{\beta_M x} \int_0^x \frac{\bar{G}_M(t)}{S_M} e^{-\beta_M t} dt
\]

Since \(\bar{G}_M(x) = (Pf'(x) - f''(x))J_1^2(\lambda_M) \sqrt{2}\) approaches zero as \(x\) proceeds toward negative infinity (see (3.0.7)), an argument similar to that found in Appendix B will show that the second summand in (3.0.19) approaches zero as \(x\) approaches negative infinity; since \(\alpha_M\) is positive, the first summand of (3.0.19) shares a similar fate. In light of (3.0.6), it is reasonable to assume (or require depending on the point of view) that

\[
\lim_{x \to -\infty} \max_{0 < r < 1} |T(x,r) - f(x)| = 0
\]
and hence \( \lim_{x \to \infty} \theta_n(x) = 0 \). Combining these observations with (3.0.19), the remaining conditions to be used in determining \( f(x) \) are easily discerned, namely:

\[
\lim_{x \to \infty} \left[ \frac{j_1^2(\lambda_M)}{2 \bar{s}_M} (\alpha_M A_M - B_M) - \int_0^x \frac{\bar{G}_M(t)}{s_M} e^{-\beta_M t} \, dt \right] e^{\beta_M x} = 0 \quad (3.0.20)
\]

Since \( \beta_M < 0 \), an analysis similar to that of Appendix B will show that (3.0.20) will be satisfied if

\[
\frac{j_1^2(\lambda_M)}{2} (\alpha_M A_M - B_M) = -\int_{-\infty}^0 \bar{G}_M(t)e^{-\beta_M t} \, dt \quad (3.0.21)
\]

Since

\[
\bar{G}_M(t) = G(t)j_1(\lambda_M)/\lambda_M = (Pf'(t) - f'(t))j_1(\lambda_M)/\lambda_M,
\]

combining (3.0.6)-(3.0.9), and (3.0.21) with two applications of integration by parts yields:

\[
\lambda_M j_1(\lambda_M) (\alpha_M A_M - B_M)/2
\]

\[
= \beta_M (A_M - P) \int_{-\infty}^0 f(t)e^{-\beta_M t} \, dt + (\beta_M - P) A(1) + B(1)
\]

Denoting

\[
R_M = \frac{1}{2} \frac{\lambda_M j_1(\lambda_M) (\alpha_M A_M - B_M) + (P - \beta_M) A(1) - B(1)}{\beta_M (P - \beta_M)}
\]

the desired properties of the surface function \( f(x) \) may be summarized as:
\( f(0) = A(1) \) and \( f'(0) = B(1) \)

\[ \lim_{x \to -\infty} f(x) \text{ exists and is finite} \]

and \( f'(x) \) and \( f''(x) \to 0 \) as \( x \to -\infty \)

\[
R_M = \int_{-\infty}^{0} f(t)e^{-\beta_M t} \, dt, \quad M = 1, 2, \ldots.
\]

To numerically approximate such a surface control function as \( f(x) \), let

\[
f(x) = \sum_{k=1}^{NSYS} c_k e^{(k-1)\beta_M t}
\]

Then, in light of (3.0.22), set

\[
c_1 + \cdots + c_{NSYS} = A(1)
\]

\[
c_2 + 2c_3 + \cdots + (NSYS-1)c_{NSYS} = B(1)
\]

and

\[
\sum_{k=1}^{NSYS} c_k \int_{-\infty}^{0} \exp((k-1-\beta_M)\beta_M t) \, dt = R_M, \quad M = 1, 2, \ldots, \text{MTERM}
\]

If the \((\text{MTERM}+2)\) by \(NSYS\) matrix \(L\) and \((\text{MTERM}+2)\) dimension vector \(\vec{b}\) are defined, for \(j = 1, 2, \ldots, NSYS\), by

\[
\begin{align*}
\beta_{1j} &= 1 \quad \text{and} \quad b_1 = A(1) \\
\beta_{2j} &= j-1 \quad \text{and} \quad b_2 = B(1) \\
\beta_{ij} &= \frac{1}{j-\beta_{i-2}-1} \\
\end{align*}
\]

\[
b_i = \tilde{R}_{i-2}, \quad i = 3, 4, \ldots, \text{MTERM} + 2
\]

† The index in the expansion of \( f(x) \) starts at \( k=1 \) instead of \( k=0 \) to make referencing this section from the accompanying FORTRAN documentation easier (Appendix A). The index limits \(NSYS\) and \(\text{MTERM}\) noted here will be used in the same role in the accompanying FORTRAN codes (Appendix C).
then, provided $MTERM+2 > NSYS$, the coefficients $c_k$ of (3.0.23) may be set to the least squares solution of

$$\mathbf{L} \mathbf{c} = \mathbf{b}$$

(3.0.25)

that is,

$$
\begin{pmatrix}
  c_1 \\
  c_2 \\
  \vdots \\
  c_{NSYS}
\end{pmatrix}
= \mathbf{c}
$$

The solution method for Problem P1-2 is similar to the above and hence most of the details are left to the reader. Using the notation established for Problem P1-2 denote

$$G(x) = P(x) - g''(x)$$

and

$$A(r) = A(r) - A(1)$$

and

$$B(r) = B(r) - B(1)$$

As before, expand $A(x)$ and $B(x)$ as

$$A(x) = \sum_{n=1}^{\infty} A_n J_0(\lambda_n r)$$

and

$$B(x) = \sum_{n=1}^{\infty} B_n J_0(\lambda_n r)$$

Then using the above established notation for $S_n$, $\alpha_n$, and $\beta_n$, the desirable properties of an upper solid region surface control function $g(x)$ are summarized as:

$$g(Q) = A(1)$$  \hspace{1cm} (3.0.26)

$$g'(Q) = B(1)$$  \hspace{1cm} (3.0.27)

$$\frac{\lambda_n J_0(\lambda_n)}{2} \left(\beta_n A_n - B_n\right) = \int_Q^{\infty} G(r) e^{-\alpha_n(Q-t)} \, dt$$  \hspace{1cm} (3.0.28)

$$\lim_{x \to \infty} g(x) \text{ exists and is finite}$$  \hspace{1cm} (3.0.29)
and

\[ \lim_{x \to c} g'(x) = \lim_{x \to c} g''(x) = 0 \] (3.0.30)

Mimicking the previous analysis, (3.0.28) is simplified by two applications of integration by parts. Thereafter, an approximation of \( g(x) \) given by:

\[ g(x) = \sum_{k=1}^{\text{NSYS}} c_k e^{(k-1)(Q-x)} \] (3.0.31)

is substituted into Equations (3.0.26)-(3.0.28) and the desired coefficients determined by a least squares method.
4.0 THE HEATING CONTROL FUNCTION

In five minutes you will say that it is all so absurdly simple. --Sherlock Holmes, "The Adventure of the Dancing Man"

The ultimate goal of this chapter is the solution of Problem P1-3. Suppose that the temperature distribution $T(x,r)$ for the melt zone is required to not only satisfy the state equation

$$\Delta T = P_2 T, \quad 0 < x < Q, \quad 0 < r < 1$$

(4.0.1)

but also must satisfy for $0 < r < 1$ the four boundary conditions:

$$T(0, r) = C(r) \quad (4.0.2)$$
$$T(Q, r) = D(r) \quad (4.0.3)$$
$$T_x(0, r) = A(r) \quad (4.0.4)$$

and

$$T_x(Q, r) = B(r) \quad (4.0.5)$$

Unfortunately, not only is too much information supplied for the two end boundaries ($x=0$ and $x=Q$; see Figure 4-1), no information whatsoever is supplied for the remaining boundary, $r=1$ (again see Figure 4-1).

**Figure 4-1** FZ Melt Zone Problem
The problem, therefore, is to find some heating control function \( h(x) \), \( 0 \leq x \leq Q \), such that the solution \( T(x, r) \) of the well-posed problem defined by the boundary condition \( T(x, 1) = h(x) \), the boundary conditions (4.0.2) and (4.0.3) and the state equation (4.0.1) also satisfies (or nearly so) the additional conditions (4.0.4) and (4.0.5).

For simplicity, the functions \( C(r) \) and \( D(r) \) are both assumed to be zero. The generalization for nonconstant \( C(r) \) or \( D(r) \) is similar to the following analysis and is left to the interested reader. As in Chapters 2 and 3, define

\[
G(x) = P_2 h'(x) - h''(x) \quad (4.0.6)
\]

\[
\mathcal{A}(r) = A(r) - A(1) = \sum_{n=1}^{\infty} A_n J_0(\lambda_n r) \quad (4.0.7)
\]

and

\[
\mathcal{B}(r) = B(r) - B(1) = \sum_{n=1}^{\infty} B_n J_0(\lambda_n r) \quad (4.0.8)
\]

If \( T(x, r) \) is decomposed into

\[
T(x, r) = \theta(x, r) + h(x) \quad (4.0.9)
\]

then Equations (4.0.1)-(4.0.3) imply

\[
\begin{aligned}
\Delta \theta &= P_2 x + G, \quad 0 < x < Q, \quad 0 < r < 1 \\
\theta(0, r) &= \theta(Q, r) = 0, \quad 0 < r < 1
\end{aligned} \quad (4.0.10)
\]

Denoting \( \overline{G}(x) = G(x) \cdot J_1(\lambda) / \lambda \) and transforming (4.0.10) by the integral transform (3.0.12),

\[
\begin{aligned}
\overline{\theta}' &= P_2 \overline{\theta}' - \lambda \overline{\theta} = \overline{G} , \quad 0 < x < Q \\
\overline{\theta}(0) &= \overline{\theta}(Q) = 0
\end{aligned} \quad (4.0.11)
\]

\[ \text{For FZ work, both } C(r) \text{ and } D(r) \text{ are set to the material melting temperature which can itself always be assigned to be zero on some translated temperature scale.} \]

\[ \text{For convenience, suppress the } \ell \text{ (liquid) subscript, i.e., } P_\ell = P \]
For convenience, let

\[ s_n = \sqrt{P^2 + 4\lambda_n^2} \]

and

\[ \alpha_n = (P + S_n)/2 \]
\[ \beta_n = (P - S_n)/2, \]

By a variation of parameters method, the solution of (4.0.11) is

\[ \overline{\Theta}_n(x) = \left[ K_n - \frac{1}{S_n} \int_0^x \overline{\Gamma}_n(t)e^{-\beta_n t} dt \right] \beta_n e^{\alpha_n x} \]
\[ + \left[ -K_n + \frac{1}{S_n} \int_0^x \overline{\Gamma}_n(t)e^{-\alpha_n t} dt \right] \alpha_n e^{\beta_n x} \]

(4.0.12)

where

\[ K_n = \frac{\int_0^Q \overline{\Theta}_n(t) \left[ e^{n(Q-t)} - e^{nQ} \right] dt}{\int_0^Q \left[ e^{nQ} - e^{nQ} \right]} \]

For the desired \( h(x) \) to be compatible with Equations (4.0.2)-(4.0.5) (recall \( C(r) \) and \( D(r) \) are set to zero), \( h(0) = 0 = h(Q) \), \( h'(0) = A(1) \) and \( h'(Q) = B(1) \). In addition, since \( \theta_x(0, r) = \Phi(r) \) and \( \theta_x(Q, r) = \Psi(r) \),

\[ \begin{align*}
\overline{\Theta}_n'(0) &= \Phi_n J_1^2(\lambda_n)/2 \\
\overline{\Theta}_n'(Q) &= \Psi_n J_1^2(\lambda_n)/2
\end{align*} \]

(4.0.13)
Combining Equations (4.0.13) with the derivative of \( \bar{\sigma}_n(x) \) (obtaining by differentiating \(^\dagger\) (4.0.12)) yields

\[
-\frac{\mathcal{H}_n J_1(\lambda_n)\lambda_n}{2} A(1) = \int_0^Q K_1(n,t) h(t) \, dt
\]

(4.0.14)

and

\[
-\frac{\mathcal{H}_n J_1(\lambda_n)}{2} - B(1) = \int_0^Q K_2(n,t) h(t) \, dt
\]

(4.0.15)

where, if \( C_n \) denotes,

\[
C_n = \frac{1}{4} \left( P^2 - S_n^2 \right) / \left[ 1 - \exp(-S_n Q) \right]
\]

then kernels \( K_1 \) and \( K_2 \) in Equations (4.0.14) and (4.0.15) are defined by

\[
K_1(n,t) = C_n \begin{bmatrix} -\alpha & -\beta \bar{t} + S_n Q \\ \alpha & \beta \bar{t} + S_n Q \end{bmatrix}
\]

(4.0.16)

and

\[
K_2(n,t) = C_n \left( 1 - \bar{e}^{-S_n t} \right) \bar{e}^{S_n Q}
\]

(4.0.17)

\(^\dagger\) The actual process of differentiating (4.0.12) is routine but laborious and is left to the industrious reader. However, this is not to imply that great care should not be taken; several of the integrands are the difference of large functions (a numerically delicate situation). For the industrious reader willing to check these results, the removal of derivatives from integrands by integration by parts is necessary.
The desired $h(x)$ is numerically approximated by some expansion of the form:

$$h(x) \approx \sum_{k=1}^{\text{NSYS}} c_k h_k(x) \quad (4.0.18)$$

(for example, let $h_k(x) = x^{k-1}$). To finish this development, solve for the coefficients $c_k$ in Equation (4.0.18) by solving (in a least squares sense) the System (4.0.19)-(4.0.23) given below.

\[
\begin{align*}
\sum_{k=1}^{\text{NSYS}} c_k h_k(0) &= 0 \\
\sum_{k=1}^{\text{NSYS}} c_k h_k(Q) &= 0 \\
\sum_{k=1}^{\text{NSYS}} c_k h'_k(0) &= A(1) \\
\sum_{k=1}^{\text{NSYS}} c_k h'_k(Q) &= B(1) \\
\sum_{k=1}^{\text{NSYS}} a_{jnk} c_k &= b_{jn} \quad , n = 1, 2, \ldots , \text{MTERM}, \text{ and } j = 1, 2 \\
\end{align*}
\]

where

\[
\begin{align*}
a_{jnk} &= \int_0^Q K_j(n,t) h_k(t) \, dt \\
b_{jn} &= -\frac{1}{2} A_n J_1(\lambda_n) \lambda_n - A(1) \\
\end{align*}
\]

and

\[
\begin{align*}
b_{2n} &= \frac{1}{2} B_n J_1(\lambda_n) \lambda_n + B(1) \\
\end{align*}
\]

\[\dagger\] In order to use a least squares method, the index limits \text{NSYS} and \text{MTERM} should be selected such that \text{NSYS} / 2 < \text{MTERM} + 2.
5.0 TEST CASES

5.1 SOLID REGION SURFACE CONTROL FUNCTIONS

In this section, the methodology developed in Chapter 3 is illustrated using material and system data provided by NASA. Recall that the problem is to find, after being given the melt zone surface temperature distribution, the surface control functions for the solid regions which are compatible with flat interfaces. The material and system parameters used are listed in Table 5-1 and were provided (and in some cases appropriately modified) by E. Kern (NASA contractor, [11]) and E. Cothran (NASA, [4]). The material selected was silicon.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radius</td>
<td>0.2413 cm</td>
</tr>
<tr>
<td>Melt Length</td>
<td>1.1684 cm</td>
</tr>
<tr>
<td>Conductivity</td>
<td></td>
</tr>
<tr>
<td>Solid</td>
<td>7.5 cal/°K m sec</td>
</tr>
<tr>
<td>Melt</td>
<td>16 cal/°K m sec</td>
</tr>
<tr>
<td>Density</td>
<td></td>
</tr>
<tr>
<td>Solid</td>
<td>2.28 gm/cm³</td>
</tr>
<tr>
<td>Melt</td>
<td>2.53 gm/cm³</td>
</tr>
<tr>
<td>Heat Capacity</td>
<td></td>
</tr>
<tr>
<td>Solid</td>
<td>0.241 cal/°K gm</td>
</tr>
<tr>
<td>Melt</td>
<td>0.265 cal/°K gm</td>
</tr>
<tr>
<td>Latent Heat</td>
<td>431 cal/gm</td>
</tr>
<tr>
<td>Growth Rate</td>
<td>2.5 mm/min</td>
</tr>
<tr>
<td>Peclet Number</td>
<td></td>
</tr>
<tr>
<td>Solid</td>
<td>0.00734</td>
</tr>
<tr>
<td>Melt</td>
<td>0.00421</td>
</tr>
<tr>
<td>Melting Temperature</td>
<td>1693° K</td>
</tr>
</tbody>
</table>

The melt zone surface temperature distribution used was suggested by E. Kern [11] and is illustrated in Figure 5-1.

The surface control functions for various combinations of MTERM and NSYS (see Equations (3.0.23) and (3.0.24)) obtained by the methods of Chapter 3 are illustrated in Figure 5-2.
Figure 5-1 Melt Zone Surface Temperature Distribution

Figure 5-2 Solid Regions' Surface Control Functions

<table>
<thead>
<tr>
<th>Curve</th>
<th>MTERM</th>
<th>MSTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>9</td>
</tr>
</tbody>
</table>
In light of Section 2.4, the variety of control functions depicted in Figure 5-2 is expected. In addition, the results of Section 2.4 suggest that the two lower solid surface control functions that eventually are above the melting temperature may be modified as illustrated in Figure 5-3 without substantially changing the thermal gradients at \( x=0 \).

The relative differences between the thermal gradients (in the solid regions) required\(^{\dagger}\) at the interfaces (\( x=0.0 \) cm and 1.1684 cm) and those obtained using the surface control functions defined in Figures 5-2 and 5-3 are listed in Table 5-2.

**TABLE 5-2**

<table>
<thead>
<tr>
<th>MTERM</th>
<th>MSYS</th>
<th>Solid Region</th>
<th>Surface Control Function Definition</th>
<th>Relative Difference (in ( L^2 ) norm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>3</td>
<td>Upper</td>
<td>Figure 5-2</td>
<td>0.0175</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lower</td>
<td>Figure 5-2</td>
<td>0.17</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>Upper</td>
<td>Figure 5-2</td>
<td>0.00012</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lower</td>
<td>Figure 5-2</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lower</td>
<td>Figure 5-3</td>
<td>0.056</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>Upper</td>
<td>Figure 5-2</td>
<td>0.000027</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lower</td>
<td>Figure 5-2</td>
<td>0.0013</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lower</td>
<td>Figure 5-3</td>
<td>0.061</td>
</tr>
</tbody>
</table>

As an aside, in a series of test cases over a range of values for MTERM and NSYS, the relative differences between the required interface gradients and those obtained using surface control functions first decreased and then increased as NSYS (or MTERM) was increased while holding fixed the value of MTERM (or NSYS). Naturally, this is to be expected since an approximate solution of an ill-posed problem is attempted by employing an overposed system. This, of course, reinforces the old maxim of always examining a computed solution for "reasonableness." In fact, the computer software developed (see Appendix A) to determine the solutions of Problems P1-1 and P1-2 automatically computes the relative-errors between the required interface gradients and those resulting from the use of the surface control functions. It cannot be overstated: always examine these relative errors before accepting a computed solution as reasonable.

\(^{\dagger}\) See the Boundary Conditions (1.2.1) and (1.2.3).

5-3
Figure 5-3 Modification of Lower Solid Region's Surface Control Functions
5.2 MELT ZONE SURFACE CONTROL FUNCTIONS

The techniques developed in Chapter 4 are illustrated in this section using material and system data as provided by NASA. The problem is to determine a melt zone surface control function compatible with some solid regions' surface temperature distributions (provided a priori) such that flat melt-solid interfaces are achieved. The material and system parameters used are listed in Table 5-3 and were provided by E. Kern (NASA contractor [1]) and E. Cothran (NASA [4]). The material selected was silicon.

TABLE 5-3 MATERIAL AND SYSTEM PARAMETERS

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crystal Radius</td>
<td>0.69 cm</td>
</tr>
<tr>
<td>Melt Length</td>
<td>1.43 cm</td>
</tr>
<tr>
<td>Conductivity</td>
<td></td>
</tr>
<tr>
<td>Solid</td>
<td>7.5 cal/°K m sec</td>
</tr>
<tr>
<td>Melt</td>
<td>16 cal/°K m sec</td>
</tr>
<tr>
<td>Density</td>
<td></td>
</tr>
<tr>
<td>Solid</td>
<td>2.28 gm/cm³</td>
</tr>
<tr>
<td>Melt</td>
<td>2.53 gm/cm³</td>
</tr>
<tr>
<td>Heat Capacity</td>
<td></td>
</tr>
<tr>
<td>Solid</td>
<td>0.241 cal/°K gm</td>
</tr>
<tr>
<td>Melt</td>
<td>0.265 cal/°K gm</td>
</tr>
<tr>
<td>Latent Heat</td>
<td>431 cal/gm</td>
</tr>
<tr>
<td>Growth Rate</td>
<td>2 mm/min</td>
</tr>
<tr>
<td>Peclet Number</td>
<td></td>
</tr>
<tr>
<td>Solid</td>
<td>0.01685</td>
</tr>
<tr>
<td>Melt</td>
<td>0.009638</td>
</tr>
<tr>
<td>Melting Temperature</td>
<td>1693° K</td>
</tr>
</tbody>
</table>

The lower and upper solid regions’ surface temperature distributions used were suggested by E. Kern [1] and are illustrated in Figure 5-4.

The melt zone surface control functions for various combinations of MTERM and NSYS (see Equations (4.0.18)-(4.0.23)) obtained by the methods of Chapter 4 are illustrated in Figure 5-5.

Because of the ill-posed nature of the problem, a variety of surface control functions is expected. The relative differences between the required melt zone interface gradients and those obtained using the surface control functions defined in Figure 5-5 are listed in Table 5-4.

† See Boundary Condition (1.2.5)
Figure 5-4 Solid Region's Surface Temperature
Figure 5-5  Melt Zone Surface Control Function
In a series of test cases over a range of values for $MTERM$ and $NSYS$, the relative differences between the required interface gradients and those obtained using the surface control functions first decreased and then increased as $NSYS$ (or $MTERM$) was increased while fixing the value of $MTERM$ (or $NSYS$). As in the previous section, this is to be expected because the solution technique employed uses over-posed systems to approximately solve an ill-posed problem. As before, the computed melt zone surface control function should be checked for reasonableness. For example, it is quite possible that the surface control function could be less than the melting temperature on part of the melt surface in which case the control function should be modified or rejected. In addition, the relative differences between the required melt zone interface gradients and those obtained using a candidate melt zone surface control function should be examined before accepting the control function as an approximate solution of Problem P1-3. Incidentally, these relative differences are approximated and displayed by the software developed for Problem P1-3.
6.0 FUTURE WORK AND UNRESOLVED ISSUES

Although the work presented in this report is, in itself, rather complete, several side issues remain unresolved and should be included in any continuation of this type of research. In this chapter, some of these issues are addressed.

6.1 VERIFICATION USING FREE BOUNDARY ALGORITHMS

The basic idea of the three methods described in Chapters 3 and 4 was to determine the properties a surface control function must have if a flat interface was to be maintained. Unfortunately, both methods involved many numerical approximations and some simplifying assumptions. As an example, for the method described in Chapter 4, the thermal distributions in the assumed infinitely long solid regions were approximated by numerical methods designed for finite length regions. In addition, the interface gradients were approximated by finite differences (a second source of error) followed by least squares Bessel function fits of these approximate interface gradients (a third possible source of error.) Thereafter, the surface control function was approximated by solving an overposed system of equations using only a finite number of terms in the control function (another source of error). Given these several possible sources of error, the actual interface shapes maintained using the computed surface control function should be constructed using some multiphase free boundary algorithm (for a survey, see [19]). The results of such numerical experiments should hopefully further verify the methods discussed in this report and should indicate some future areas to be studied with error reduction in mind.

6.2 MAINTAINING CURVED INTERFACES

Although thermal stresses, which can generate defects in the crystal, are generally minimal for a planar interface [2], a slightly curved interface shape is also quite desirable in some cases. Specifying the desired shapes, the required surface control functions could probably (with sufficient investigation) be constructed using methods similar to those in Chapters 3 and 4 after the introduction of transformations similar to those described in [12].

6.3 NON-DIRICHLET BOUNDARY CONDITIONS

Boundary conditions other than the Dirichlet type (Equations (FZ6), (FZ8), and (FZ10) of Figure 1-2) should be investigated. Fortunately, much of the work for this type of problem will probably be straightforward. For example, suppose a question like Problem P1-3 is to be solved where the Dirichlet boundary condition (see Equation (1.2.6))

\[ T(x,1) = h(x), \quad 0 < x < Q \]

is replaced by a boundary condition of the type

\[ T_c(x,1) = k \left[ T(x,1) - S^\alpha(x) \right] \]  \hspace{1cm} (6.3.1)

6-1
where $S(x)$ is the desired surface control function (for example, if $a = 4$, then $S(x)$ might be the temperature of a furnace wall providing radiant heating). To solve this problem, first solve Problem P1-3 as stated in Section 1.2. Using the computed Dirichlet type surface control function $h(x)$, next solve Problem P2-3 (let $x_0 = 0$ and $x_N = Q$). Then place the resulting temperature distribution $T(x,r)$ into the boundary condition (6.3.1) and solve for the desired surface control function $S(x)$.

6.4 Basis Functions Used to Expand the Control Functions

The exponentially decaying functions used to expand the solid regions' surface control functions ($f(x)$ and $g(x)$ of Equations (3.0.23) and 3.0.31, respectively) were selected because they represented what a typical control function would be intuitively expected to resemble and because they allowed for simple integrations in Equations (3.0.22) and (3.0.28). However, from a computational point of view, these basis functions are not the best. For example, some preliminary experiments indicate that replacing Equations (3.0.23) and (3.0.31) by

$$f(x) = c_1 + \sum_{k=2}^{\text{NSYS}} c_k \exp \left(-\frac{(t-k-2)^2}{\text{E}}\right)$$

and

$$g(x) = c_1 + \sum_{k=2}^{\text{NSYS}} c_k \exp \left(-\frac{(t-k+2+Q)^2}{\text{E}}\right)$$

respectively can significantly reduce the least squares residuals of overposed systems like Equation (3.0.25). More study is needed to find basis functions that both further reduce the least squares residuals and are not too difficult to integrate in equations like (3.0.22) and (3.0.28).

6.5 Applications of Linear Programming

The linear system of equations (3.0.25) will be underposed if $\text{MTEP}^{\text{+2}} < \text{NSYS}$. However, the desired coefficient vector $\bar{C}$ may still be determined as follows. Let $\bar{F}$ be the residual vector of Equation (3.0.25), i.e., $\bar{F} = \bar{b} - \bar{L}\bar{C}$.

It is sometimes dangerous to approximate a function $f(x)$ by a sum:

$$f(x) = \sum_{k=1}^{N} c_k f_k(x)$$

where all or most of the functions $f_k(x)$ "resemble" each other, e.g., $f_k(x) = \exp(kx)$. This, for example, is why Chebyshev polynomials are preferred over the so-called standard basis, $f_k(x) = x^k$, for polynomial approximation on certain domains.
Then solve the linear programming problem [13 pg 15]

$$\min \sum_{1}^{\infty} |x_i|$$

subject to \( L C + r = b \) \( (6.5.1) \)

In fact, such a technique might be used to reduce the chance of say, \( f(x) \) of (3.0.23), becoming positive† (recall that the melting temperature was translated to zero) as \( x \) approaches negative infinity. To accomplish this, first select a grid,

\[ x_1 < x_2 < \ldots < x_N < 0 \]

partitioning a portion of the lower solid region, and then adjoin to (6.5.1) the additional \( N \) constraints (see Equation (3.0.23)):

$$\sum_{k=1}^{NSYS} c_k \exp ((k-1)x_i) < 0, \ i=1, \ldots, N$$

Some preliminary numerical experiments suggest this idea has sufficient potential to warrant further investigation. Although this discussion has centered on the lower solid region, these ideas are applicable to either of the solid regions or to the melt zone.

6.6 MODELS AND REALITY

The FZ process was modeled in this effort as a steady state process on an infinitely long boule. Unfortunately for the modeler (but fortunately for the commercial FZ operator), the boule has finite length*. For finite length boules, the problem of finding the proper surface control functions to maintain flat interfaces would now involve end effects and various time transients.

† The partial differential equations used to establish the desired surface control functions are quite ignorant of the fact that surface control functions for solid regions should always be below the material melting point. In fact, in some numerical experiments where \( MTERM \) and \( MSYS \) were large, the computed surface control functions for one of the solid regions became greater than the melting temperature. This is only one of the dangers in trying to solve an overposed problem.

* Fortunately, the assumptions and results of this effort are still quite reasonable for long boules with slow growth rates.
However, the basic ideas discussed in this report could probably be extended to cover such difficulties. The resulting partial differential equations would involve the additional term

\[ \frac{\partial}{\partial t} T(\xi, r, t) \]

(where \( t \) represents time) and hence would be parabolic instead of elliptic. The boundary conditions would also be time dependent. However, the dual integral transform pairs used in this report should still provide enough information concerning the surface control functions (required for flat interfaces) to allow for their construction.

In addition, the fluid dynamics of the melt zone should be incorporated in the computation of the surface control functions. Of the topics discussed in this chapter, this is undoubtedly the most difficult one to model and resolve.
BIBLIOGRAPHY


APPENDIX A.0 THE OWNERS MANUAL
(or a user guide to developed software)

A.1 INTRODUCTION

The data input procedures and output interpretations for the software developed to approximate the solutions of Problems P1-1, P1-2, P1-3 and P2-1 is the subject of this appendix. To begin, Problem P2-1 is covered in Appendix A.2 and is followed by Problems P1-1 and P1-2 in Appendix A.3. To finish, Problem P1-3 is the subject of Appendix A.4.

A.2 USER CONSIDERATIONS FOR PROBLEM P2-1 SOFTWARE

The data input procedure and output interpretation for the software developed for Problem P2-1 is the subject of this section. To begin, all the required data are input in the form of punched cards. The definitions and formats of this input data are summarized in Table A-1.

**TABLE A-1 PROBLEM P2-1 SOFTWARE INPUT**

<table>
<thead>
<tr>
<th>PROGRAM SYMBOL</th>
<th>VARIABLE DEFINITION</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>READ(5,16)IHFC,M</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16 FORMAT(2F10)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IHFC</td>
<td>= 1 if a cubic spline will be used to approximate the boundary function ( h(x) ) in Condition (2.2.4).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>= 0 if the user will supply a functional form of ( h(x) ) (see Condition (2.2.4)). In this case, the user must insert this functional form of ( h(x) ) in the sub-routine HFC (see the software list in Appendix C.1).</td>
<td></td>
</tr>
<tr>
<td>M</td>
<td>Number of knots used to approximate ( h(x) ) (see Condition (2.2.4)) by a cubic spline (IHFC=1). If IHFC=0, set M=0.</td>
<td></td>
</tr>
<tr>
<td>DO 32 I = 1,M</td>
<td></td>
<td></td>
</tr>
<tr>
<td>READ(5,22)XD(I),YD(I)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>32 CONTINUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22 FORMAT(4E20.10)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XD(I)</td>
<td>The dimensionless axial position of the Ith knot used to approximate ( h(x) ) (see Condition (2.2.4)). Ignore if IHFC=1. XD(I)&lt;XD(I+1).</td>
<td></td>
</tr>
</tbody>
</table>

\( x, r \) and \( R \) will denote the axial distance, the radial position and the rod radius respectively. \( T \) will denote whatever temperature scale the user prefers.
An input sample is illustrated next in Figure A-1.

The output is labeled clearly for ease of use. The input data are first viewed followed by the thermal distribution (the approximate solution of Problem P2-3) given in table format (see Figure A-2). Incidentally, the axial and radial positions in Figure A-2 are given in dimensionless form (x/R and r/R). The thermal gradients at X0 and XN are given last in a table format (again, see Figure A-2).
INPUT DATA

The surface temperature distribution is approximated by the cubic spline through the following 5 (x, temp) data points:

<table>
<thead>
<tr>
<th>X</th>
<th>Surface Temp</th>
</tr>
</thead>
<tbody>
<tr>
<td>+1.00E+00</td>
<td>+2.00E+00</td>
</tr>
<tr>
<td>+5.00E-01</td>
<td>+3.75E-01</td>
</tr>
<tr>
<td>+0.00E+00</td>
<td>+0.00E+00</td>
</tr>
<tr>
<td>+5.00E-01</td>
<td>+1.25E-01</td>
</tr>
<tr>
<td>+1.00E+00</td>
<td>+0.00E+00</td>
</tr>
</tbody>
</table>

P X0 XN HSUN GRID WN
+7.34E-02 +1.00E+01 +1.00E+01 20 500 8

TEMPERATURE DISTRIBUTION

<table>
<thead>
<tr>
<th>X</th>
<th>Rn</th>
<th>Rm</th>
<th>Rn</th>
<th>Rm</th>
<th>Rn</th>
<th>Rm</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00E+00</td>
<td>-1.450691E-13</td>
<td>-1.450691E-13</td>
<td>-1.450691E-13</td>
<td>-1.450691E-13</td>
<td>-1.450691E-13</td>
<td>-1.450691E-13</td>
</tr>
<tr>
<td>9.99E-01</td>
<td>-1.225947E-02</td>
<td>-1.225947E-02</td>
<td>-1.225947E-02</td>
<td>-1.225947E-02</td>
<td>-1.225947E-02</td>
<td>-1.225947E-02</td>
</tr>
<tr>
<td>9.98E-01</td>
<td>-1.174003E-02</td>
<td>-1.174003E-02</td>
<td>-1.174003E-02</td>
<td>-1.174003E-02</td>
<td>-1.174003E-02</td>
<td>-1.174003E-02</td>
</tr>
<tr>
<td>9.94E-01</td>
<td>-1.490455E-02</td>
<td>-1.490455E-02</td>
<td>-1.490455E-02</td>
<td>-1.490455E-02</td>
<td>-1.490455E-02</td>
<td>-1.490455E-02</td>
</tr>
</tbody>
</table>

Figure A-2 Sample Output For Problem P2-1 Software
Figure A-2  Sample Output For Problem P2-1 Software (Cont)
| 0.030000 | 0.100000 | 0.030000 | 0.276005 | 0.050000 | 0.425000
| 0.040000 | 0.100000 | 0.050000 | 0.100000 | 0.050000 | 0.425000
| 0.050000 | 0.100000 | 0.060000 | 0.100000 | 0.060000 | 0.100000
| 0.070000 | 0.100000 | 0.080000 | 0.100000 | 0.080000 | 0.100000
| 0.090000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.110000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.120000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.130000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.140000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.150000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.160000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.170000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.180000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.190000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.200000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.210000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.220000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.230000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.240000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.250000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000
| 0.260000 | 0.100000 | 0.100000 | 0.100000 | 0.100000 | 0.100000

| 0.870000 | 0.100000 | 0.404317E01 | 0.508000E00 | 0.32011E00
| 0.880000 | 0.100000 | 0.394368E01 | 0.5511E00 | 0.32011E00
| 0.890000 | 0.100000 | 0.384393E01 | 0.5730E00 | 0.32011E00
| 0.900000 | 0.100000 | 0.374362E01 | 0.5942E00 | 0.32011E00
| 0.910000 | 0.100000 | 0.364328E01 | 0.6152E00 | 0.32011E00
| 0.920000 | 0.100000 | 0.354314E01 | 0.6356E00 | 0.32011E00
| 0.930000 | 0.100000 | 0.344350E01 | 0.6548E00 | 0.32011E00
| 0.940000 | 0.100000 | 0.334333E01 | 0.6735E00 | 0.32011E00
| 0.950000 | 0.100000 | 0.324412E01 | 0.6917E00 | 0.32011E00
| 0.960000 | 0.100000 | 0.314254E01 | 0.7088E00 | 0.32011E00
| 0.970000 | 0.100000 | 0.304761E01 | 0.7247E00 | 0.32011E00
| 0.980000 | 0.100000 | 0.295313E01 | 0.7400E00 | 0.32011E00
| 0.990000 | 0.100000 | 0.285913E01 | 0.7542E00 | 0.32011E00
| 1.000000 | 0.100000 | 0.276558E01 | 0.7670E00 | 0.32011E00

Figure A-2 Sample Output For Problem P2-1 Software (Cont)
To finish, the user must supply an algorithm to fit (in the least squares sense) a linear combination of functions to a given set of data points (see the end of Section 2.2 for a short discussion). This algorithm is required in the subroutine COEFS. In addition, an algorithm to evaluate the \( J_0 \) Bessel function (required in the subroutine JO) must be provided. These required algorithms are generally available from the host computer library or may be obtained from various software packages such as the IMSL and FUNPACK.

---

\[ ^7 \text{The user is warned, however, that many host computer mathematics libraries (with the general exception of IBM) still contain numerous faux pas that were well known years ago and still remain uncorrected.} \]
A.3 USER CONSIDERATIONS FOR PROBLEMS PI-1 and PI-2 SOFTWARE

The data input procedure and output interpretation for the software developed for Problems PI-1 and PI-2 are the subjects of this section. Recall that the general problem is to find the solid regions' surface control functions (f(x) and g(x) in Problems PI-1 and PI-2 respectively) which, for the sake of flat interfaces, are compatible with the a priori given melt zone surface temperature distribution (h(x)). At present, all the required data are input in the form of punched cards. The definitions and formats of the input data are summarized in Table A-2.
### TABLE A-2 PROBLEMS P1-1 AND P1-2 SOFTWARE INPUT

<table>
<thead>
<tr>
<th>PROGRAM SYMBOL</th>
<th>VARIABLE DEFINITION</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>READ(5,80)IHFC,M</td>
<td></td>
<td></td>
</tr>
<tr>
<td>80 FORMAT(ZI10)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IHFC</td>
<td>1 if a cubic spline will be used to approximate the melt zone surface temperature distribution, h(x). 0 if the user will supply a functional form of h(x). In this case, the user must insert this functional form of h(x) in the subroutine HFC (see the software list in Appendix C.3)</td>
<td></td>
</tr>
<tr>
<td>M</td>
<td>Number of knots used to approximate h(x) by a cubic spline (IHFC=1). If IHFC=0, set M=0.</td>
<td></td>
</tr>
<tr>
<td>DO 32 I = 1,M</td>
<td></td>
<td></td>
</tr>
<tr>
<td>READ(5,22)XD(I),YD(I)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>32 CONTINUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22 FORMAT (2E20.10)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XD(I)</td>
<td>The axial position of the Ith knot used to approximate h(x). Ignore if IHFC=0. In addition, XD(I)=XD(I+1) and it is recommended that the axial position be measured from the lower melt-solid interface, i.e., XD(1)=0.0.</td>
<td>rad</td>
</tr>
<tr>
<td>YD(I)</td>
<td>The surface temperature represented by the Ith knot used to approximate h(x). Ignore if IHFC=0.</td>
<td>°K above melting temp.</td>
</tr>
<tr>
<td>READ(5,10)P,XO,XN,XTERM,MSUM,NGRID,NR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10 FORMAT(3F10.5,4I10)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>Melt Zone Peclet Number</td>
<td>dimensionless</td>
</tr>
<tr>
<td>XO</td>
<td>Axial position of lower melt-solid interface; XO=0.0 is recommended</td>
<td>rad</td>
</tr>
<tr>
<td>XN</td>
<td>Axial position of upper melt-solid interface</td>
<td>rad</td>
</tr>
<tr>
<td>XTERM</td>
<td>Set to Zero</td>
<td></td>
</tr>
<tr>
<td>MSUM</td>
<td>The first MSUM terms of the expansion in Equations (2.2.14) are used to approximate the melt zone temperature distribution, T(x,r). MSUM must be less than 31.</td>
<td></td>
</tr>
<tr>
<td>NGRID</td>
<td>(XN-XO)/NGRID is the grid size employed in System (2.2.20). In addition, the melt zone temperature distribution is output for NGRID/10+1 axial values from XO to XN. NGRID may not exceed 500 and must be divisible by 10.</td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td>The melt zone temperature distribution is output for NR+1 radial values (rad) from 0 to 1. NR may not exceed 100.</td>
<td></td>
</tr>
</tbody>
</table>

† All lengths will be measured in radius (rad) units, e.g., 2 rad is as long as the rod is wide. The temperature scale will be °K above or below the material melting point.
### TABLE A-2 PROBLEMS P1-1 AND P1-2 SOFTWARE INPUT (CONT)

<table>
<thead>
<tr>
<th>PROGRAM SYMBOL</th>
<th>VARIABLE DEFINITION</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>Lower solid region Peclet number</td>
<td>dimension-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>less</td>
</tr>
<tr>
<td>XO</td>
<td>The semi-infinite lower solid region is, for computational Purposes, truncated to</td>
<td>rad</td>
</tr>
<tr>
<td></td>
<td>a finite length. XO is the axial position of lower end of this truncated region.</td>
<td></td>
</tr>
<tr>
<td>XN</td>
<td>Review the end of Section 2.3 for details.</td>
<td></td>
</tr>
<tr>
<td>MTERM</td>
<td>Axial position of lower melt-solid interface.</td>
<td>rad</td>
</tr>
<tr>
<td>MSUM</td>
<td>Integer parameter determining the size of system used to compute the lower solid</td>
<td></td>
</tr>
<tr>
<td></td>
<td>region's surface control function. See Equation (3.0.24).</td>
<td></td>
</tr>
<tr>
<td>NGRID</td>
<td>The first MSUM terms of the expansion in Equations (2.2.14) are used to approximate</td>
<td></td>
</tr>
<tr>
<td></td>
<td>the lower solid temperature distribution, T(x,r). MSUM must be less than 21.</td>
<td></td>
</tr>
<tr>
<td>(XN-XO)/NGRID</td>
<td>(XN-XO)/NGRID is the grid size employed in System (2.2.20). In addition, the lower</td>
<td></td>
</tr>
<tr>
<td></td>
<td>solid temperature distribution is output for NGRID/10+1 axial values from X0 to XN.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NGRID may not exceed 500 and must be divisible by 10.</td>
<td></td>
</tr>
<tr>
<td>NR</td>
<td>The lower solid temperature distribution is output for NR+1 radial values (rad) from</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 to 1. NR may not exceed 100.</td>
<td></td>
</tr>
<tr>
<td>RKS</td>
<td>Conductivity of material in lower solid region</td>
<td>cal</td>
</tr>
<tr>
<td>RKL</td>
<td>Conductivity of material in melt zone</td>
<td>cal</td>
</tr>
<tr>
<td>RL</td>
<td>$A$ of Equation P24, Figure 1-2. RL is the product of the growth rate, the solid</td>
<td>cal</td>
</tr>
<tr>
<td></td>
<td>materials density and the latent heat.</td>
<td>sec rad</td>
</tr>
<tr>
<td>NSYS</td>
<td>Number of terms in expansion of lower solid region's surface control function</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(see Equation (3.0.23)). NSYS may not exceed 20.</td>
<td></td>
</tr>
</tbody>
</table>
### TABLE A-2 PROBLEMS P1-1 AND P1-2 SOFTWARE INPUT (CONT)

<table>
<thead>
<tr>
<th>PROGRAM SYMBOL</th>
<th>VARIABLE DEFINITION</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>READ(5,21)MINTERM,MAXTERM,DELTTERM,MINNSYS,MAXNSYS,DELNSYS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21 FORMAT(8110)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The software is designed to compute the lower solid region's surface control function for many combinations of MTERM and NSYS. To do this, the user must supply the bounds and increments for the cases desired.

- MINTERM: Lower bound on MTERM.
- MAXTERM: Upper bound on MTERM.
- DELTTERM: Integer increment for MTERM.
- MINNSYS: Lower bound on NSYS.
- MAXNSYS: Upper bound on NSYS.
- DELNSYS: Integer increment for NSYS.

After the lower solid region's surface control function, \( f(x) \), is determined, the user may specify certain modifications of the surface control function. These options are principally of use when the surface control function is above the material's melting point on portions of the lower solid region's surface. If \( f(x) \) is interpreted as the \( pK \) above or below the melting point, then let \( A \) be the lower endpoint of the largest subinterval of the form \([A,XN]\) on which \( f(x) \) is not positive. If \( A < X0 \), reassign \( A \) to be \( X0 \). Let \((XMIN,FMIN)\) be the minimum point of \( f(x) \) on the interval \([A,XN]\).

<table>
<thead>
<tr>
<th>IOPTION</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>( f(x) ) is not to be modified. In this case, CLIP may be assigned any value, e.g., zero.</td>
</tr>
<tr>
<td>1</td>
<td>( f(x) ) is to be redefined as:</td>
</tr>
<tr>
<td></td>
<td>[ f(x) ] if ( x &gt; XMIN )</td>
</tr>
<tr>
<td></td>
<td>( FMIN ) otherwise</td>
</tr>
<tr>
<td>2</td>
<td>( f(x) ) is to be redefined as:</td>
</tr>
<tr>
<td></td>
<td>( f(x) ) if ( x &gt; XMIN )</td>
</tr>
<tr>
<td></td>
<td>( \min {f(x), CLIP} ) otherwise</td>
</tr>
</tbody>
</table>

A-11
TABLE A-2 PROBLEMS P1-1 AND P1-2 SOFTWARE INPUT (CONT)

<table>
<thead>
<tr>
<th>PROGRAM SYMBOL</th>
<th>VARIABLE DEFINITION</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>Peclat number for upper solid region</td>
<td>Dimensionless</td>
</tr>
<tr>
<td>XO</td>
<td>Axial position of upper interface</td>
<td>rad</td>
</tr>
<tr>
<td>XN</td>
<td>The semi-infinite upper solid region is, for computational purposes, truncated to a finite length. XN is the axial position of the upper end of this truncated region.</td>
<td></td>
</tr>
<tr>
<td>XMIN</td>
<td>The first XMIN terms of the expansion in Equations (2.2.14) are used to approximate the upper solid regions' temperature distribution, T(x,r). XSUM must be less than 21.</td>
<td></td>
</tr>
<tr>
<td>NTERM</td>
<td>Number of equations (n=1,2,...,NTERM) of the type given in Equation (3.0.28) used in least squares generation of upper solid region's surface control function.</td>
<td></td>
</tr>
<tr>
<td>MSUM</td>
<td>The upper solid temp. distribution is output for XR+1 radial values (rad) from 0 to 1. NR may not exceed 100.</td>
<td></td>
</tr>
<tr>
<td>NGRID</td>
<td>(XN-XO)/NGRID is the grid size employed in System (2.2.20). In addition, the upper solid region's temperature distribution is output for NGRID/10+1 axial values from XO to XN. NGRID may not exceed 500 and must be divisible by 10.</td>
<td></td>
</tr>
<tr>
<td>NR</td>
<td>Axial position of upper interface</td>
<td></td>
</tr>
<tr>
<td>RKS</td>
<td>Conductivity of material in upper solid region</td>
<td>cal</td>
</tr>
<tr>
<td>RKL</td>
<td>Conductivity of material in melt zone</td>
<td>cal</td>
</tr>
<tr>
<td>RL</td>
<td>A of Equation F22, Figure 1-2. RL is the product of the growth rate, the solid material's density and the latent heat</td>
<td>cal</td>
</tr>
</tbody>
</table>

READ(5,10)P,XO,XN,TERM,MSUM,NGRID,NR
10 FORMAT(3F10.5,4I10)

90 READ(5,90)RKS,RKL,RL,NSYS
FORMAT(3E20.10,110)
### Table A-2: Problems P1-1 and P1-2 Software Input (Cont)

<table>
<thead>
<tr>
<th>PROGRAM SYMBOL</th>
<th>VARIABLE DEFINITION</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSYS</td>
<td>Number of terms in exponential expansion of the upper solid region's surface control function (see Equation (3.0.31))</td>
<td></td>
</tr>
<tr>
<td>MINTERM, MAXTERM, DELTERM, MNNSYS, MAXNSYS, DELNSYS</td>
<td>( \frac{\text{IOPTION}}{\text{CLIP}} )</td>
<td></td>
</tr>
<tr>
<td>MINTERM, MAXTERM, DELTERM, MNNSYS, MAXNSYS, DELNSYS</td>
<td>As previously defined above but applied to the upper solid region.</td>
<td></td>
</tr>
</tbody>
</table>

**Program Symbol Definitions:**

- **NSYS**: Number of terms in exponential expansion of the upper solid region's surface control function.
- **MINTERM**, **MAXTERM**, **DELTTERM**, **MNNSYS**, **MAXNSYS**, **DELNNSYS**: As previously defined above but applied to the upper solid region.
- **IOPTION**, **CLIP**: After the upper solid region's surface control function, \( g(x) \), is determined, the user may specify certain modifications of this surface control function. The options are principally of use when the surface control function is above the material's melting point on portions of the upper solid region's surface. The definitions of **IOPTION** and **CLIP** are similar to their previous definitions above and are illustrated below.

**Diagram:**

- **IOPTION**
  - \( 3^\circ \) above the melting temperature
  - \( X_0 \)
  - \( X_{MIN} \)
  - \( Y_0 \)
  - **CLIP**
  - **IOPTION=1**
  - **IOPTION=2**
  - **IOPTION=3**
An input sample is illustrated in Figure A-3.

The output is clearly labeled for ease of use. The melt zone input data is first displayed followed by the melt zone temperature distribution (given in table format) and interface gradients (see Figure A-4). The lower solid region's material and system parameters and the values of MINTERM, ..., DELNSYS are next displayed followed by the required lower solid region interface gradient, B(r) of Equation (1.2.1) (again see Figure A-4).

For each of various acceptable combinations of MTERM and NSYS (recall the definitions of MINTERM, ..., DELNSYS), a lower solid region surface control function is computed. For each of these cases, the values of MTERM and NSYS are first displayed followed by the coefficients (see Equation (3.0.23)) used to determine the surface control function. Using the surface control function, the temperature distribution in the lower solid region is next displayed (in table form†). The lower solid region's interface gradient is then given followed last by the relative difference (in the L² norm) between the required lower solid region interface gradient and the interface gradient obtained by use of the surface control function (see Figure A-5).

After all the lower solid region cases (various combinations of MTERM and NSYS) are given, the results for the upper solid region are displayed in a similar fashion (see Figure A-6).

To finish, the user must supply the two algorithms described at the end of Appendix A.2. In addition, an algorithm to solve (in a least squares sense) an overposed system of linear equations must also be provided for use in the subroutines SOLID2 and SOLID3 (see Appendix C.3).

† The surface control function values can be read from this table in the R=1 column.
Figure A-3  Sample Input For Problems P1-1 And P1-2 Software
HELM ZONE

INPUT DATA

The surface temperature distribution is approximated by the cubic spline through the following 24 \((x, \text{temp})\) data points.

<table>
<thead>
<tr>
<th>X</th>
<th>Surface Temp</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00000000E+00</td>
<td>9.99999999E+00</td>
</tr>
<tr>
<td>1.00000000E+00</td>
<td>8.99999999E+00</td>
</tr>
<tr>
<td>2.00000000E+00</td>
<td>7.99999999E+00</td>
</tr>
<tr>
<td>3.00000000E+00</td>
<td>6.99999999E+00</td>
</tr>
<tr>
<td>4.00000000E+00</td>
<td>5.99999999E+00</td>
</tr>
<tr>
<td>5.00000000E+00</td>
<td>4.99999999E+00</td>
</tr>
<tr>
<td>6.00000000E+00</td>
<td>3.99999999E+00</td>
</tr>
<tr>
<td>7.00000000E+00</td>
<td>2.99999999E+00</td>
</tr>
<tr>
<td>8.00000000E+00</td>
<td>1.99999999E+00</td>
</tr>
<tr>
<td>9.00000000E+00</td>
<td>0.99999999E+00</td>
</tr>
</tbody>
</table>

Figure A-4 Sample Melt Zone Output For Problems Pl-1 And Pl-2 Software
Figure A-4 Sample Melt Zone Output For Problems P1-1 And P1-2 Software (Cont)
FOR ITTERM = 4 AND NSYS = 3

**LOWER SOLID SURFACE CONTROL COEFFICIENTS**

<table>
<thead>
<tr>
<th>k</th>
<th>C(k)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.977395e+01</td>
</tr>
<tr>
<td>2</td>
<td>-0.159636e+02</td>
</tr>
<tr>
<td>3</td>
<td>-0.618907e+01</td>
</tr>
</tbody>
</table>

**LOWER SOLID TEMPERATURE DISTRIBUTION**

<table>
<thead>
<tr>
<th>x</th>
<th>C1</th>
</tr>
</thead>
<tbody>
<tr>
<td>-.00000</td>
<td>-.000000000</td>
</tr>
<tr>
<td>-.20000</td>
<td>-.0955197e+12</td>
</tr>
<tr>
<td>-.40000</td>
<td>-.0852675e+00</td>
</tr>
<tr>
<td>-.60000</td>
<td>-.2461796e+00</td>
</tr>
<tr>
<td>-.80000</td>
<td>-.3502535e+00</td>
</tr>
<tr>
<td>-1.00000</td>
<td>-.3523191e+00</td>
</tr>
<tr>
<td>-1.20000</td>
<td>-.2631601e+00</td>
</tr>
<tr>
<td>-1.40000</td>
<td>-.1970096e+00</td>
</tr>
<tr>
<td>-1.60000</td>
<td>-.1013147e+00</td>
</tr>
<tr>
<td>-1.80000</td>
<td>-.6633624e+00</td>
</tr>
<tr>
<td>-2.00000</td>
<td>-.5462630e+00</td>
</tr>
<tr>
<td>-2.20000</td>
<td>-.7723228e+00</td>
</tr>
<tr>
<td>-2.40000</td>
<td>-.7777970e+00</td>
</tr>
<tr>
<td>-2.60000</td>
<td>-.8053922e+00</td>
</tr>
<tr>
<td>-2.80000</td>
<td>-.8347108e+00</td>
</tr>
<tr>
<td>-3.00000</td>
<td>-.8596510e+00</td>
</tr>
<tr>
<td>-3.20000</td>
<td>-.9195256e+00</td>
</tr>
<tr>
<td>-3.40000</td>
<td>-.9105727e+00</td>
</tr>
</tbody>
</table>

Figure A-5 Sample Lower Solid Zone Output For Problems PI-1 And PI-2 Software (Cont)
Figure A-5 Sample Lower Solid Zone Output For Problems P1-1 And P1-2 Software (Cont)
Figure A-6 Sample Upper Solid Zone Output For Problems P1-1 And P1-2 Software
<table>
<thead>
<tr>
<th>X</th>
<th>C(K)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>415996163E+02</td>
</tr>
<tr>
<td>2</td>
<td>6114089295E+02</td>
</tr>
<tr>
<td>3</td>
<td>1954103132E+02</td>
</tr>
</tbody>
</table>

**Upper Solid Temperature Distribution**

**Figure A-6** Sample Upper Solid Zone Output For Problems P1-1 And P1-2 Software (Cont)
UPPER SOLID THERMAL GRADIENTS

\[
\begin{array}{ccc}
R & \text{GRAD. AT } x = 4.84200 & \text{GRAD. AT } x = 14.84200 \\
0.00000 & -2.1042864E+02 & -5.5041805E+02 \\
0.10000 & -2.1063253E+02 & -5.4856793E+02 \\
0.20000 & -2.1141142E+02 & -5.4163330E+02 \\
0.30000 & -2.1288494E+02 & -5.3000701E+02 \\
0.40000 & -2.1446163E+02 & -5.1362838E+02 \\
0.50000 & -2.1671998E+02 & -4.9236875E+02 \\
0.60000 & -2.1941159E+02 & -4.6590301E+02 \\
0.70000 & -2.2240841E+02 & -4.3901752E+02 \\
0.80000 & -2.2555483E+02 & -4.0752557E+02 \\
0.90000 & -2.2915628E+02 & -3.7757116E+02 \\
1.00000 & -2.2058471E+02 & -3.4775711E+02 \\
\end{array}
\]

RELATIVE DIFFERENCES BETWEEN REQUIRED AND OBTAINED GRADIENTS

L-2 ERROR

0.01752

FOR NTERM = 4 AND MBSY = 3

Figure A-6  Sample Upper Solid Zone Output For Problems PL-1 And PL-2 Software (Cont)
USER CONSIDERATIONS FOR PROBLEM P1-3 SOFTWARE

The data input procedure and output interpretation for the software developed for Problem P1-3 are the subjects of this section. Recall that the problem is to find a melt zone surface control function \( h(x) \) in Problem P1-3 which, for the sake of flat interfaces, is compatible with the \( a \) priori given surface temperature distributions of the two solid regions. At present, all the required data are input in the form of punched cards. The definitions and formats of the input data are summarized in Table A-3.
### TABLE A-3  PROBLEM P1-3 SOFTWARE INPUT

<table>
<thead>
<tr>
<th>PROGRAM SYMBOL</th>
<th>VARIABLE DEFINITION</th>
<th>UNITS†</th>
</tr>
</thead>
<tbody>
<tr>
<td>READ(5,12)P,MSUM,NGRID,NR</td>
<td>Paclee number of upper solid region</td>
<td>Dimensionless</td>
</tr>
<tr>
<td>12 FORMAT(E20.10,6I10)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>The first MSUM terms of the expansion in Equations (2.2.14) are used to approximate the upper solid region's temperature distribution, T(x,r). MSUM must be less than 21.</td>
<td></td>
</tr>
<tr>
<td>MSUM</td>
<td>For computational purposes, the semi-infinite upper solid region is truncated to a finite length (SLENGTH). SLENGTH/NGRID is the grid size employed in System (2.2.20) which is used to approximate the temperature distribution in this truncated upper solid region. In addition, the upper solid region's temperature distribution is displayed for NGRID/10+1 uniformly spaced axial values. NGRID may not exceed 500 and must be divisible by 10.</td>
<td></td>
</tr>
<tr>
<td>NGRID</td>
<td>The upper solid region's temperature distribution is output for NR+1 radial values (rad) from 0 to 1. NR may not exceed 100.</td>
<td></td>
</tr>
<tr>
<td>NR</td>
<td>READ(5,22)RKS,RKL,RL,SLENGTH</td>
<td></td>
</tr>
<tr>
<td>22 FORMAT(4E20.10)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RKS</td>
<td>Conductivity of material in upper solid region</td>
<td>cal sec rad⁻²</td>
</tr>
<tr>
<td>RKL</td>
<td>Conductivity of material in melt zone</td>
<td>cal sec rad⁻²</td>
</tr>
<tr>
<td>RL</td>
<td>$\Lambda$ of Equation FZ2, Figure 1-2. RL is the product of the growth rate, the solid material's density and the latent heat.</td>
<td></td>
</tr>
<tr>
<td>SLENGTH</td>
<td>For computational purposes, the semi-infinite upper solid region is truncated to a finite length, SLENGTH. For details, review the end of Section 2.3.</td>
<td>rad</td>
</tr>
<tr>
<td>READ(5,22)Q</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22 FORMAT(4E20.10)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Q</td>
<td>Q is the length of the melt zone</td>
<td>rad</td>
</tr>
<tr>
<td>READ(5,16)INFC,M</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16 FORMAT(10I5)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

† All lengths will be measured in radius (rad) units, e.g., 2 rad is as long as the rod is wide. The temperature scale will be °K above or below the material melting point.
TABLE A-3 PROBLEM P1-3 SOFTWARE INPUT (CONT)

<table>
<thead>
<tr>
<th>PROGRAM SYMBOL</th>
<th>VARIABLE DEFINITION</th>
</tr>
</thead>
</table>
| IHFC           | = 1 if a cubic spline will be used to approximate the surface temperature distribution, g(x), of the upper solid region.  
= 0 if the user will supply a functional form of g(x). In this case, the user must insert this functional form of g(x) in the subroutine IHFC (see the software list in Appendix C.4). |
| M             | Number of knots used to approximate g(x) by a cubic spline. (IHFC=1). If IHFC=0, set M=0. |

```fortran
DO 32 I = 1, M
READ(5,22)XD(I),YD(I)
32 CONTINUE
22 FORMAT (4E20.10)
```

<table>
<thead>
<tr>
<th>XD(I)</th>
<th>The axial position of the Ith knot used to approximate g(x). Ignore if IHFC=0. In addition, XD(I)&lt;XD(I+1), XD(1)=0 and XD(M)=Q+SLENGTH</th>
</tr>
</thead>
<tbody>
<tr>
<td>YD(I)</td>
<td>The surface temperature represented by the Ith knot used to approximate g(x). Ignore if IHFC=0.</td>
</tr>
</tbody>
</table>

```fortran
READ(5,12)P,MSUM,NGRID,NR
12 FORMAT(E20.10,6I10)
```

<table>
<thead>
<tr>
<th>P</th>
<th>Same definitions as above but applied to the lower solid region</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSUM</td>
<td></td>
</tr>
<tr>
<td>NGRID</td>
<td></td>
</tr>
<tr>
<td>NR</td>
<td></td>
</tr>
</tbody>
</table>

```fortran
READ(5,22)RKS,RKL,RL,SLENGTH
22 FORMAT(4E20.10)
```

<table>
<thead>
<tr>
<th>RKS</th>
<th>Conductivity of material in lower solid region</th>
</tr>
</thead>
<tbody>
<tr>
<td>RKL</td>
<td>Conductivity of material in melt zone</td>
</tr>
<tr>
<td>RL</td>
<td>$R$ of Equation F24, Figure 1-2. RL is the product of the growth rate, solid material's density, and latent heat</td>
</tr>
<tr>
<td>SLENGTH</td>
<td>For computational purposes, the semi-infinite lower solid region is truncated to a finite length, SLENGTH. For details, review the end of Section 2.3.</td>
</tr>
</tbody>
</table>

UNITS

- IHFC: No units
- M: Number
- XD(I), YD(I): Radians
- P, MSUM, NGRID, NR: Scalars
- RKS, RKL, RL, SLENGTH: Calories per square meter per second (cal/s/m²)
### TABLE A-3 PROBLEM P1-3 SOFTWARE INPUT (CONT)

<table>
<thead>
<tr>
<th>PROGRAM SYMBOL</th>
<th>VARIABLE DEFINITION</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>READ(5,16)IHFC,M</strong></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td><strong>FORMAT(10I5)</strong></td>
<td></td>
</tr>
<tr>
<td>IHFC</td>
<td>Same definitions as previously given but applied to the surface temperature distribution, ( f(x) ), of the lower solid region.</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>DO 32 I = 1,M</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>READ(5,22)XD(I),YD(I)</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>CONTINUE</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>FORMAT(4E20.10)</strong></td>
<td></td>
</tr>
<tr>
<td>XD(I)</td>
<td>The axial position of the ( I )th knot used to approximate ( f(x) ). Ignore if IHFC=0. In addition, ( XD(I)&lt;XD(I+1) ), ( XD(1)=-SLENGTH ), and ( XD(M)=0.0 ).</td>
<td><strong>rad</strong></td>
</tr>
<tr>
<td>YD(I)</td>
<td>The surface temperature represented by the ( I )th knot used to approximate ( f(x) ). Ignore if IHFC=0.</td>
<td><strong>°K below melting temp.</strong></td>
</tr>
<tr>
<td></td>
<td><strong>READ(5,12)P,MSUM,NGRID,NS</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>FORMAT(E20.10,6E10)</strong></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MSUM</td>
<td>Same definitions as previously given but applied to the melt zone of length ( Q )</td>
<td></td>
</tr>
<tr>
<td>NGRID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NR</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>READ(5,16)MAXTERM,MINTERM,MAXNSYS,MINNSYS,DELTERM,DELNSYS</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>FORMAT(10I5)</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>The software is designed to compute the melt zone surface control function for various combinations of MAXTERM and NSYS. To do this, the user must apply the bounds and increments for the cases desired.</td>
<td></td>
</tr>
<tr>
<td>MAXTERM</td>
<td>Upper bound on MAXTERM</td>
<td></td>
</tr>
<tr>
<td>MINTERM</td>
<td>Lower bound on MINTERM</td>
<td></td>
</tr>
<tr>
<td>MAXNSYS</td>
<td>Upper bound on NSYS</td>
<td></td>
</tr>
<tr>
<td>MINNSYS</td>
<td>Lower bound on NSYS</td>
<td></td>
</tr>
<tr>
<td>DELNSYS</td>
<td>Integer increment for NSYS</td>
<td></td>
</tr>
</tbody>
</table>

A-27
An input sample is illustrated in Figure A-7.

The output is clearly labeled for ease of use. The input data for the upper solid region is output first followed by displays of the upper solid region’s temperature distribution (given in table format) and interface gradient (see Figure A-8). The lower solid region follows in a similar fashion (Figure A-9). Using Equations FZ2 and FZ4 of Figure 1-2, the required melt zone interface gradients are computed and then displayed along with the melt zone input data (see Figure A-10). For each of various combinations of MTERM and NSYS (recall the definition of MINTERM, ..., DELNSYS), the expansion coefficients of the melt zone surface control function (see Equation (4.0.18)) are output. Using the computed surface control function, the melt zone temperature distribution (given in table† form) and interface gradients are displayed next followed last by the relative difference (in the L2 norm) between the required melt zone interface gradients and those obtained by use of the surface control function (see Figure A-11).

To finish, the user must supply the two algorithms described at the end of Appendix A.2. Also, an algorithm to solve (in a least squares sense) an overposed system of linear equations (for example, see [17, Chapter 5]) must be provided for use in the subroutine MELTI (see Appendix C.4 for code listing). In addition, a numerical integration routine is required for use in the subroutines INTEGL1 and INTEGL2 (during software verification, the numerical quadrature code CADRE was used and is available in the EIMSL package or from the open literature [16, Chapter 7]).

†The melt zone surface control function can be read from this table in the R=1 column.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th>20</th>
<th>400</th>
<th>2</th>
<th>1.53951318</th>
<th>10.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0165</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.0574</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.072</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure A-7 Sample Input For Problem P1-3 Sample**

A-29
Figure A-8 Sample Upper Solid Region Output For Problem P1-3 Software (Cont)
The surface temperature distribution is approximated by the cubic spline through the following 32 \((x, \text{temp})\) data points:

- \(x\):
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)
  - \(=1000000000\cdot 10^{-1}\)

- \(\text{Surface Temp}\):
  - \(=1140000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)
  - \(=1130000000\cdot 10^{-1}\)

Figure A-9 Sample Lower Solid Region Output For Problem P1-3 Software
### Lower Solid Temperature Distribution

#### Table

<table>
<thead>
<tr>
<th>X</th>
<th>A12</th>
<th>B12</th>
<th>R12</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00000</td>
<td>1.000000000000</td>
<td>0.000000000000</td>
<td>0.000000000000</td>
</tr>
<tr>
<td>-0.20000</td>
<td>-1.2021367E+10</td>
<td>-5.96879E+02</td>
<td>-5.000000000000</td>
</tr>
<tr>
<td>-0.40000</td>
<td>-1.3480294E+02</td>
<td>-1.2153169E+03</td>
<td>-1.000000000000</td>
</tr>
</tbody>
</table>

### Lower Solid Thermal Gradients

**Grad. at X = -10.00000**

<table>
<thead>
<tr>
<th>X</th>
<th>A11</th>
<th>B11</th>
<th>R11</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00000</td>
<td>1.377422E+01</td>
<td>1.3501772E+01</td>
<td>1.3501772E+01</td>
</tr>
<tr>
<td>1.00000</td>
<td>1.377422E+01</td>
<td>1.3501772E+01</td>
<td>1.3501772E+01</td>
</tr>
<tr>
<td>2.00000</td>
<td>1.377422E+01</td>
<td>1.3501772E+01</td>
<td>1.3501772E+01</td>
</tr>
<tr>
<td>3.00000</td>
<td>1.377422E+01</td>
<td>1.3501772E+01</td>
<td>1.3501772E+01</td>
</tr>
<tr>
<td>4.00000</td>
<td>1.377422E+01</td>
<td>1.3501772E+01</td>
<td>1.3501772E+01</td>
</tr>
<tr>
<td>5.00000</td>
<td>1.377422E+01</td>
<td>1.3501772E+01</td>
<td>1.3501772E+01</td>
</tr>
<tr>
<td>6.00000</td>
<td>1.377422E+01</td>
<td>1.3501772E+01</td>
<td>1.3501772E+01</td>
</tr>
<tr>
<td>7.00000</td>
<td>1.377422E+01</td>
<td>1.3501772E+01</td>
<td>1.3501772E+01</td>
</tr>
<tr>
<td>8.00000</td>
<td>1.377422E+01</td>
<td>1.3501772E+01</td>
<td>1.3501772E+01</td>
</tr>
<tr>
<td>9.00000</td>
<td>1.377422E+01</td>
<td>1.3501772E+01</td>
<td>1.3501772E+01</td>
</tr>
<tr>
<td>10.00000</td>
<td>1.377422E+01</td>
<td>1.3501772E+01</td>
<td>1.3501772E+01</td>
</tr>
</tbody>
</table>

Figure A-9 Sample Lower Solid Region Output For Problem P1-3 Software (Cont)
Figure A-10 Sample Output Of Required Melt Zone Interface Gradients For Problem Pl-3 Software
### Melt Zone Surface Control Coefficients

For $\text{ITERM} = 10$ and $\text{HSYS} = 12$

<table>
<thead>
<tr>
<th>$K$</th>
<th>$C(K)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2480592493E+03</td>
</tr>
<tr>
<td>2</td>
<td>922902871E+03</td>
</tr>
<tr>
<td>3</td>
<td>1759588526E+04</td>
</tr>
<tr>
<td>4</td>
<td>7261786648E+04</td>
</tr>
<tr>
<td>5</td>
<td>5399299555E+04</td>
</tr>
<tr>
<td>6</td>
<td>2114058954E+05</td>
</tr>
<tr>
<td>7</td>
<td>7817962690E+05</td>
</tr>
<tr>
<td>8</td>
<td>2858342107E+05</td>
</tr>
<tr>
<td>9</td>
<td>5394050399E+05</td>
</tr>
<tr>
<td>10</td>
<td>1882092097E+05</td>
</tr>
<tr>
<td>11</td>
<td>1410888937E+04</td>
</tr>
<tr>
<td>12</td>
<td>4754649004E+04</td>
</tr>
</tbody>
</table>

### Melt Zone Temperature Distribution

Center: $R = 0.00000000$, $N = 0.50000000$, $R = 1.00000000$

<table>
<thead>
<tr>
<th>$x$</th>
<th>$C(x)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.072000</td>
<td>-1.4060626E+00</td>
</tr>
<tr>
<td>2.030560</td>
<td>3.466312E+01</td>
</tr>
</tbody>
</table>

Figure A-11 Sample Output Of Melt Zone Surface Control Function, Temperature Distribution, and Interface Gradients For Problem Pl-1 Software
### Melt Zone Thermal Gradients

<table>
<thead>
<tr>
<th>x</th>
<th>(1.989120)</th>
<th>(1.947680)</th>
<th>(1.906240)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>(2782676)E+01</td>
<td>(11860446)E+02</td>
<td>(1857631)E+02</td>
</tr>
<tr>
<td>T</td>
<td>(2346746)E+01</td>
<td>(1107672)E+02</td>
<td>(1482080)E+02</td>
</tr>
<tr>
<td>T</td>
<td>(447623)E+01</td>
<td>(175027)E+01</td>
<td>(117576)E+02</td>
</tr>
</tbody>
</table>

#### Relative Differences Between Required and Obtained Gradients

<table>
<thead>
<tr>
<th>x</th>
<th>(0.041440)</th>
<th>(0.00000)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>(5522753)E+01</td>
<td>(5748838)E+00</td>
</tr>
<tr>
<td>T</td>
<td>(666884)E+01</td>
<td>(3740089)E+00</td>
</tr>
</tbody>
</table>

**Figure A-11** Sample Output Of Melt Zone Surface Control Function, Temperature Distribution, And Interface Gradients For Problem PI-3 Software (Cont)
APPENDIX B. CONVERGENCE OF EQUATION (2.3.11)

Recall, from Section 2.3, Equation (2.3.11)

\[
\bar{\sigma}_n(x) = \left[ A_n + \frac{1}{S_n} \int_0^x \bar{G}_n e^{-\alpha_n t} \, dt \right] e^{\alpha_n x} \\
+ \left[ B_n - \frac{1}{S_n} \int_0^x \bar{G}_n e^{-\beta_n t} \, dt \right] e^{\beta_n x}
\]  

(2.3.11)

where

\[
S_n = \sqrt{P^2 + 4\lambda_n^2}
\]

\[
\alpha_n = \frac{(P + S_n)}{2}
\]

\[
\beta_n = \frac{(P - S_n)}{2}
\]

\[
B_n = -\frac{1}{S_n} \int_{-\infty}^0 \bar{G}_n e^{-\beta_n t} \, dt
\]

and

\[
A_n = -B_n + \frac{J_1^2(\lambda_n)}{n^2}
\]

In this appendix, under the assumptions of Section 2.3, it will be shown that

\[
\lim_{x \to \infty} \bar{\sigma}_n(x) = 0
\]

For convenience, suspend the use of the "n" subscript and define

\[
\| \bar{\sigma} \|_{(a, b)} = \max_{a \leq x \leq b} | \bar{\sigma}(x) |
\]

Since \( \bar{G} \) approaches zero as \( x \) proceeds to negative infinity, if given some \( C > 0 \), then there exists some \( N < J \) such that \( | \bar{G}(t) | < \epsilon \) if \( t < N \). Then since \( \alpha > 0 \),

\[
\lim_{x \to -\infty} e^{\alpha x} \left| \int_0^x \bar{G}(t) e^{-\alpha t} \, dt \right|
\]

\[
< \lim_{x \to -\infty} e^{\alpha x} \left| \int_0^N \bar{G}(t) e^{-\alpha t} \, dt + \| \bar{G}(N, 0) \|_{(a, b)} \int_0^{N} e^{-\alpha t} \, dt \right|
\]

\[
= \frac{\epsilon}{\alpha}
\]

B-1
Hence, since the above $\varepsilon$ is arbitrary and $\alpha > 0$, the first summand of (2.3.11) converges to zero as $x$ proceeds to negative infinity. Next, for the second summand of (2.3.11),

\[
\lim_{x \to -\infty} \left| B - \frac{1}{S} \int_{0}^{x} \frac{e^{-\beta t}}{G} \, dt \right| e^{\beta x}
\]

\[
< \lim_{x \to -\infty} \left| \frac{1}{S} \int_{-\infty}^{y} \frac{e^{-\beta t}}{G} \, dt \right| e^{\beta x}
\]

\[
< \lim_{x \to -\infty} \frac{e^{-\beta x} - e^{-\infty}}{\beta} \frac{e^{\beta x}}{S} \left\| \bar{G} \right\| (-\infty, x) = 0
\]

because $\lim_{x \to -\infty} \bar{G} = 0$. Hence $\lim_{x \to -\infty} \bar{G}(x) = 0$.

A clever man understands the need for proof. 

--Proverbs 14:15
C.0 INTRODUCTION

The computer codes developed to solve Problems P1-1, P1-2, P1-3, and P2-1 are given in this appendix. The codes themselves contain numerous comments correlating portions of the codes with sections and equations in this report. The code for Problem P2-1 is listed in Appendix C.2 and is followed by the code for Problems P1-1 and P1-2 in Appendix C.3. To finish, the code for Problem P1-3 is listed in Appendix C.4.

C.2 COMPUTER CODE LIST FOR PROBLEM P2-1

The computer code for Problem P2-1 is listed in Figure C-1. Before using this code, the user should review the remarks given at the end of Appendix A.2.
**Program Purpose:**

Compute steady-state temperature distribution and thermal gradients for finite length translating cylinder. This problem is described in detail in Section 2.2 of final report (D-148A).

**Boundary Conditions:**

- The control of float zone interfaces by the use of selected boundary conditions.

**Source:**

Science Applications, Inc.

**Authors:**

Larry M. Foster

John Mcintosh

**Reference:**

- The control of float zone interfaces by the use of selected boundary conditions.

**Remarks:**

- Software developed and tested on CDC 7600/4400 and UNIVAC 1108.

**Input Variables and Functions:**

- P = Peclet number
- nsum = number of terms in series expansion of temperature distribution (the desired solution)
- X0 = axial position of lower end of cylinder
- XN = axial position of upper end of cylinder
- NR = number of divisions of cylinder radius used in solution of Q, G, F, boundary value problem resulting from transformation of the PBE modeling the temperature.
- NR = number of divisions of cylinder radius used in output of temperature distribution
- IHFC = 1 if a discrete data point form of the surface temperature is used provided
- 0 if a user defined functional form of the surface temperature is provided
- (X0, Y0) = user provided data pts for the axial distance (X0) and corresponding surface temperature (Y0)
- M = number of data pts, input if IHFC = 1
- SET TO IF IHFC = 0
- HFC = USER PROVIDED (IF IHFC = 0) SURFACE TEMPERATURE DISTRIBUTION
- APC = user provided temperature distribution on the lower (axial position = X0) end of the cylinder
- BFC = user provided temperature distribution on the upper (axial position = XN) end of the cylinder

**Output Variables:**

- THOLD = temperature distribution array in the cylinder (from X0 to XN axially, with NR divisions of the radius)
- GRANX0 = axial thermal gradient array at X0

---

Figure C-1. Computer Code List for Problem P2-1
CC GRADXN = AXIAL THERMAL GRADIENT ARRAY AT Y1
CC USER SUPPLIED MATHEMATICAL SOFTWARE
CC = A LEAST SQUARES ALGORITHM TO SOLVE OVER POSES SYSTEMS OF
CC LINEAR EQUATIONS (REQUIRED IN SUBROUTINE COEFS)
CC = AN ALGORITHM TO EVALUATE BESSEL FUNCTIONS (REQUIRED IN
CC SUBROUTINE J0)
CC CC

CCFCCFCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCF

RAFL J1,J1LAM
COMMON/C,HJLAMO(20),J1O(20),J1LAMO(20)
COMMON/C10,ASCRIP(20),BSCRIP(20)
COMMON/RFAD/P,MSH,X0,XN,IGRID,HR
COMMON/C,P(101),R51(20,101),SN51(20)
COMMON/C20/4(505),8(505),C(505),0(505),Y(505),OPTA(505),GAMA(505)
COMMON/C21/THE TAB(20,505),THMLO(101),Y(10),GRADY(101),GRADX(101)
CHARACTER*1 R15,ALPHA(6)
CHARACTER*1 STARS,STAR(6)
DATA R13/*R*
ALPHA(L)=R1
STAR(L)=STARS

P1 CONTINUE

CCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCF

RLAMD(M)=ROOT OF JN BESSK FCN
J1(M)=J1(RLAMD(M)) WHERE J1 IS BESSL FCN
J1LAM(M)=J1LAM(J1(M))

RLAMD( 1)=0.0000000000
RLAMD( 2)=2.4555869654
RLAMD( 3)=4.9099884333
RLAMD( 4)=7.3643904860
RLAMD( 5)=9.8188341983
RLAMD( 6)=12.2732787688
RLAMD( 7)=14.7277252483
RLAMD( 8)=17.1821707225
RLAMD( 9)=19.6366162002
RLAMD(10)=22.0909596697
RLAMD(11)=24.5453031391
RLAMD(12)=26.9996466084
RLAMD(13)=29.4539900778
RLAMD(14)=31.9083335472
RLAMD(15)=34.3626769166
RLAMD(16)=36.8169192859
RLAMD(17)=39.2711616553
RLAMD(18)=41.7254030246
RLAMD(19)=44.1796443939
RLAMD(20)=46.6338857632
J1( 1)=0.510175973
J1( 2)=0.100268055
J1( 3)=0.271292629
J1( 4)=0.214638831
J1( 5)=0.205624433
J1( 6)=0.187728030
J1( 7)=0.171265902
J1( 8)=0.157175307
J1( 9)=0.151563908
J1(10)=0.143659777
J1(11)=0.139294230
J1(12)=0.13426267
J1(13)=0.128069471
J1(14)=0.121380268
J1(15)=0.117211108
J1(16)=0.113291926

Figure C-1. Computer Code List for Problem P2-1 (Cont)
Figure C-1. Computer Code List for Problem P2-1 (Cont)
Figure C-1. Computer Code List for Problem P2-1 (Cont)
Figure C-1. Computer Code List for Problem P2-1 (Cont)
WRITE(6,10)
30 FORMAT(///,4(F4.2,2X,T9H),/)
DATA POINTS ////.37H
 X SURFACE TEMP.
 DO 32 J=1,N
 READ(5,22)X(1),Y(J)
 22 FORMAT(5E20.10)
 10 FORMAT(2I10)
 WRITE(6,30)X(1),Y(J)
 30 CONTINUE
 CALL CDFRCN
 A0 CONTINUE
 READ(5,10)P,X0,XN,MSUM,NGRID,NS
 10 FORMAT(3F10.5,4I10)
 WRITE(6,20)X0,XN,MSUM,NGRID,NS
 20 FORMAT(///,4H1,2X,3E12.4,15.2I7)
 RETURN
 END

This subroutine supplies lateral surface temperature

SEE EQUATION (2,2.4) OF FINAL REPORT

SUBROUTINE HFC(X,ANS)
COMM/N/O/1/HFC
DIMENSION C(7)
IF(HFC.EQ.1) GOTO 0
SUBROUTINE HFC(X0,ANS)
COMMON/C5/HFC
RETURN
END

LATERAL SURFACE TEMPERATURE PROVIDED BY SPLINE
FIT OF USER SUPPLIED DATA (X0,YM)

CALL SPLINE(X,ANS)
RETURN
END

THIS subroutine supplies radial temperature distribution
ON LOWER END OF CYLINDER - SEE EQUATION (2,2.2) OF FINAL REPORT
SUBROUTINE HFC(R,ANS)
COMMON/READ1/P,MSUM,X0,XN,NSUM,NR
SUBROUTINE HFC(X0,ANS)
CALL HFC(X0,ANS)
RETURN
END

THIS subroutine supplies radial temperature distribution
ON UPPER END OF CYLINDER - SEE EQUATION (2,2.3) OF FINAL REPORT

CALL HFC(R,ANS)
RETURN
END

Figure C-1. Computer Code List for Problem P2-1 (Cont)
Figure C-1. Computer Code List for Problem P2-1 (Cont)
Figure C-1. Computer Code List for Problem P2-1 (Cont)
C
EQUIVALENCE(C1(1,1)*C1(1,1))

M=1
ND=1

NO 2 K=1,H

\( n(k) = x_0(k+1) = x_0(k) \)
\[ p(k) = d(k)/a. \]

2 E(K) = (Y(K+1) - YD(K))/D(K)

NO 3 K=2,H

3 R(K)=E(K)=P(K-1)

**A** (1,2) = 1.0 - O(1)/O(2)

**A** (1,3) = N(1)/O(2)

**A** (2,2) = **R** + N(1)/O(2)

**A** (2,3) = N(1)/A(1,3) + A(2,2)

R(K)=O(2)/A(2,2)

NO 4 K=3,H

**A** (K,2) = **R** * (P(K-1) + P(K)) = P(K-1) * A(K-1,2)

R(K)=B(K) = P(K-1) * H(K-2)

**A** (K,3) = **R** / A(K,2)

4 R(K) = O(K)/A(K,2)

G=O(M+1)/O(M)

**A** (N0,1) = 1.0 - A(M-1,3)

**A** (N0,2) = 1.0 - A(M,1) * A(M,3)

R(N0) = B(M+1) = A(N0,3) * B(M)

Z(N0) = B(N0) * A(N0,2)

NO 6 M=1, N=2

K=M=1

6 Z(K)=B(K) * A(K,1) + Z(K+1)

**Z** (1) = A(1,2) * Z(2) = A(1,3) * Z(3)

NO 7 K=1,H

G=1.0 / N (D(K))

C (1,K) = Z(K) = 0

C (2,K) = Z(K+1) = 0

C (3,K) = (YD(K)/D(K)) - Z(K) * P(K)

7 C (4,K) = YD(K+1)/D(K) = Z(K+1) * P(K)

M=1

RETURN

C
END COFFEN

END

Figure C-1. Computer Code List for Problem P2-1 (Cont)
C.3 COMPUTER CODE LIST FOR PROBLEMS P1-1 and P1-2

The computer code for Problems P1-1 and P1-2 is listed in Figure C-2. Before using this code, the user should review the remarks made at the end of Appendix A.3.
Figure C-2. Computer Code List for Problems P1-1 and P1-2
Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)
Figure C-2. Computer Code List for Problems
P1-1 and P1-2 (Cont)

C-14
Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)

C-15
Figure C-2. Computer Code List for Problems
PI-1 and PI-2 (Cont)
Figure C-2. Computer Code List for Problems P1-1 and P1-2 (Cont)
Figure C-2. Computer Code List for Problems
P1-1 and P1-2 (Cont.)
Figure C-2. Computer Code List for Problems

PI-1 and PI-2 (Cont)

C-19
Figure C-2. Computer Code List for Problems
Pl-1 and Pl-2 (Cont)
Figure C-2. Computer Code List for Problems
P1-1 and P1-2 (Cont)
CC THIS SUBROUTINE SUPPLIES RADIAL TEMPERATURE DISTRIBUTION
CC ON LOWER END OF CYLINDER - SEE EQUATION (2.2.2) OF FINAL REPORT

SUBROUTINE HFCXO(ANS)
COMMON/READ/P,MTERM,SN,NC,GRID,MR
CC USER SUPPLIED LOWER END TEMPERATURE A(R)
CALL HFCXO(ANS)
RETURN
END

CC THIS SUBROUTINE SUPPLIES RADIAL TEMPERATURE DISTRIBUTION
CC ON UPPER END OF CYLINDER - SEE EQUATION (2.2.2) OF FINAL REPORT

SUBROUTINE HFCXN(ANS)
COMMON/READ/P,MTERM,SN,NC,GRID,MR
CC USER SUPPLIED UPPER END TEMPERATURE B(R)
CALL HFCXN(ANS)
RETURN
END

CC THIS SUBROUTINE FITS RESELL SURFACES TO DATA BY LEAST SQUARES
CC METHOD - SEE EQUATIONS (2.2.17), (2.2.18) AND (2.2.23)
CC OF FINAL REPORT

SUBROUTINE COEFS(NY,MR,COEF,COEFL)
INTEGER NR,NC,A
REAL F(N101),Y(101),Z(20),X(N400)
EXTERNAL F

CALL IFLSQ(F,R,Y,MR,COEF,NC,A,IER)
IF (IER.EQ.1) PRINT(40)
RETURN
END

FUNCTION P USERED IN SUBROUTINE COEFS F(N+R)*JO(LINDA;N+R)
REAL FUNCTION F(N,R)
INTEGER N
REAL R, P, LAMO(20)
LAMO(1) = 2.4042559777
LAMO(2) = 15.5200781103
LAMO(3) = 0.4537279129
LAMO(4) = 11.7915835291
LAMO(5) = 14.9301178886
LAMO(6) = 14.0710636079
LAMO(7) = 21.2183662989
LAMO(8) = 24.359715308
LAMO(9) = 27.49879152
LAMO(10) = 30.636000458
LAMO(11) = 33.7798202136
LAMO(12) = 36.917083537
LAMO(13) = 39.054297646
LAMO(14) = 42.19971732
LAMO(15) = 44.3411683717
LAMO(16) = 47.4826068974
LAMO(17) = 50.6290518411
LAMO(18) = 53.7655107590
LAMO(19) = 56.9040839261
LAMO(20) = 62.0080019602
CALL J0(X,Y)
FNY
RETURN
END

THIS SUBROUTINE COMPUTES THE JA BESSEL FCN. Y=JN(X)
SUBROUTINE J0(X,Y)
USER SUPPLIED JA FCN. PLACED HERE. THE INSL ROUTINE MHSJO
IS ILLUSTRATED BELOW
REAL MHSJO(Y, JER)
RETURN
SUBROUTINE FOR SOLVING A SYSTEM OF LINEAR SIMULTANEOUS
EQUATIONS HAVING A TRIDIAGONAL COEFFICIENT MATRIX. DIAGONALS
ARE STORED IN THE ARRAYS A, B, AND C. THE COMPUTED
SOLUTION VECTOR V(1) ... V(L) IS STORED IN THE ARRAY Y.
SUBROUTINE TRIDAG(L)
COMMON/C20/A(500), B(500), C(500), D(500), E(500), F(500), GAMMA(500)
COMPUTE INTERMEDIATE ARRAYS BETA AND GAMMA
BETA (1) = 1
GAMMA (1) = D (1) / BETA (1)
IFP = 2
DO 1 = IFP
BETA (1) = D (1) / GAMMA (1) + (1) / BETA (1)
GAMMA (1) = D (1) / GAMMA (1) + (1) / BETA (1)
1 CONTINUE

Figure C-2. Computer Code List for Problems
P1-1 and P1-2 (Cont)

C-23
1 CONTINUE
CC
CC COMPUTE FINAL SOLN. VECTORE V
CC
CC
Y(L) = GAMMA(I)
LASTL = LAST
INL = K
V(I) = GAMMA(I) = C(I) + V(I+1)/NFT(I)
2 CONTINUE
RETURN
END

CC
CC THIS SUBROUTINE DETERMINES THE LOWER SOLID REGIONS SURFACE
CC CONTROL FUNCTION AS OUTLINED IN CHAPTER 3 OF FINAL REPORT.
CC
CC SUBROUTINE SOLID2
REAL J1,J1LAM,H203J0
COMMON/C1/R AND(20),J1(20),J1LAM(20)
COMMON/C9/V(1),PSI(20,101),SA(I)(20)
COMMON/C9/CNEF(20),RH
COMMON/C2/TCAJ,THETL(3)
COMMON/C23/GRAD2(101),GRAD3(101)
COMMON/C24/GRAD3(20),AKL,AL,NSYS
COMMON/READ/P,NTERM,MSUM,X0,INN,GRID,TP
COMMON/C3/F(4)
COMMON/C7/IFLAG2,IFLAG3
COMMON/C77/MAT(20,20),RHS(20)
COMMON/C87/AMAT(20,20),RHS(20)
DIMENSION Y(101),C(A),IMK(20),W(950)
DIMENSION ALPHA(20)
IF(FLAG2,ER,0) GOTO120
WRITE(6,933)

DO 30 JJ=1,101
J1=JJ
WRITE(6,933)

DO 30 JJ=1,101
J1=JJ
WRITE(6,933)

CC
CC DETERMINE LOWER SOLID REGION INTERFACE GRADIENT SEE EQUATION
CC (P2A), FIGURE 1-2 OF FINAL REPORT
CC
CC GRAD2(J)=R(KL)GRAD2(J)=RL)/RNS
WRITE(6,555)R(J),GRAD2(J)

555 FORMAT(1H,9X,FAIL0,10X,E20,1A)

Y(J)=GRAD2(J)=GRAD2(101)
40 CONTINUE
CC
CC DETERMINE COEFFICIENTS IN BESSEL EXPANSION (3.0.14)
CC
CC CALL COEFS(R,Y,101,20,COEF)
CC
CC DETERMINE MATRIX AND VECTOR ELEMENTS AS DEFINED IN
CC EQUATION (3.0.24)
CC
CC
Figure C-2. Computer Code List for Problems
P1-1 and P1-2 (Cont)
Figure C-2. Computer Code List for Problems Pi-1 and Pi-2 (Cont)

C-25
THIS SUBROUTINE DETERMINES THE UPPER SOLID REGIONS SURFACE CONTROL FUNCTION AS OUTLINED IN CHAPTER 3 OF FINAL REPORT.

SUBROUTINE SOLR3
REAL JLJILAM,MMBSJ0
COMMON/C1/RAND(20),J1(20),J1LAM(20)
COMMON/C5/R(101),PSI(PG,101),SOI(101)
COMMON/CW/CCEF(20),RH
COMMON/C2Z/TCASE,THEL(3)
COMMON/C3Z/GRAD2(101),GRAD3(101)
COMMON/CZ/RKS,RKL,R,L,NST3
COMMON/READ/R,P,MTN,MUM,XO,XN,HGRID,NN
COMMON/C3Z/F(2)
COMMON/C76/IFLAG,IVLS,IFLAG3
COMMON/C77/AMAT(20,20),RMS(20)
DIMENSION Y(201),ENK(201),EN(950)
DIMENSION ALPHA(20)
IF(IFLAG,.EQ.0) GOTO10

WRITE_br333>
733
CRMAT///,SH
101XRT7X,4X,RO)
DO 50 JJ=1,101
J=JJ-1

GADS(JI) (RUL) &D (J) - RLI/5S
WRITE(4,555) R(J),GRAD3(J)
55 FORMAT(1H1,1X,82X,1HT,1X,27X,8NHGRAD)

DO 30 J=1,101
J=JJ-1

DETERMINE MATRIX AND VECTOR ELEMENTS AS DEFINED IN EQUATIONS
C (PZ), FIGURE 1-2 OF FINAL REPORT

CALL COEFS(3,Y,101,20,COEF)

DETERMINE UPPER SOLID REGION INTERFACE GRADIENT SEE EQUATION
C (PZ), Figure 1-2 of Final Report

ON 80 M=1,INTER
L=2
RHS(L)=RLAND(M)+J1(M)*COEF(M)/(+2,0)

CONTINUE
B=GRAD3(101)
A=THEL(TCASE)

DO 103 M=1,INTER

ALPHA(M)=A+P*RLAND(M)
ALPHA/M=1P*BSOAR(ALPHA(M))/(2,0)
L=2
RHS(L)=RHS (L)/(ALPHA(M)=A+P*ALPHA (M))
103 CONTINUE

DO 106 M=1,INTER

AMAT(L,K)=1,0
106 CONTINUE

AMAT(L,K)=1,0

Figure C-2. Computer Code List for Problems
PI-1 and PI-2 (Cont)

C-26
08 CONTINUE
RHS(1)=AF1
DO 109 K=1,NSYS
AMAT(2,K)=1.K
109 CONTINUE
RHS(2)=BF1
120 CONTINUE
IFLAGSA=N
MHEMTM=2
DO 111 I=1,NSYS
E(I)=0.0
111 CONTINUE
CC
SOLVE THE OVER POSED LINEAR SYSTEM OF EQUATIONS (3.0,26) =
CC
(3.0,28) FOR THE COEFFICIENTS TO BE USED IN (3.6,31). THE
CC
INSL ROUTING LLBOF IS ILLUSTRATED BELOW
CC
CALL LLBOF(AMAT,20,NSYS,RHS,20,1,0,E,COEF,70,THK,WK,IER)
RHOLO=0.0
DO 113 K=1,NSYS
RHOLO=RHOLO+COEF(K)
113 CONTINUE
COEF(1)=AOF1-RHOLO
CC
DISPLAY COEFFICIENTS USED IN THE EXPANSION OF THE UPPER SOLID
CC
REGION SURFACE CONTROL FUNCTION (SEE EQUATION (4.0,31))
CC
WRITE(6,1400)
140 FORMAT(111)
WRITE(6,777)
777 FORMAT(11H3,1X,A34,HU,PE R , SOL I D, SUR F A C E , CO EF F IC I N T S , I I)
WRITE(6,790)
90 FORMAT(11H3,1X,A34,1H9X,22X,1H4C(V))
DO 778 I=1,NSYS
WRITE(6,6466) I, COEF(I)
778 CONTINUE
GOTO180
180 CONTINUE
RETURN
END
CC
PURPOSE
CC
= PERFORM LINE SEARCH TO DETERMINE MIN PT. ON THE SURFACE
CC
CONTROL FUNCTION. USED IF IOPT = 1 OR 2.
CC
SUBROUTINE LINSCH(XMIN)
COMMON/READ1/P,TERM,HSUM,XO,XN,NGRID,NS
DIMENSION F18(105)
AM=0.0
BN=M=10
STORE=1.0
DELM=0.200
DO 90 IM=1,200
XM=IM*DELX
CALL FNC(X,Y)
RHOLO=STORE
IF(RHOLO.LE.0.0) GO TO 100
90 CONTINUE

Figure C-2. Computer Code List for Problems
P1-1 and P1-2 (Cont)
Figure C-2. Computer Code List for Problems
   P1-1 and P1-2 (Cont)
Figure C-2. Computer Code List for Problems
P1-1 and Pl-2 (Cont)

C-29
Figure C-2. Computer Code List for Problems
P1-1 and P1-2 (Cont)
Figure C-2. Computer Code List for Problems
P1-1 and P1-2 (Cont)
C.4 COMPUTER CODE LIST FOR PROBLEM P1-3

The computer code for Problem P1-3 is listed in Figure C-3. Before using this code, the user should review the remarks made at the end of Appendix A.4.
Program Purpose:

Compute the melt zone surface control function required for flat solid-melt interfaces as described in Problem P1-2 and solved in Chapter 4 of the final report (to NASA) entitled the control of float zone interfaces by the use of selected boundary conditions by Science Applications, Inc.

Source:

Authors:
LA: M. Foster
John McIntosh

Reference:
The control of float zone interfaces by the use of selected boundary conditions - (Final report - SAI-83/503404U) - Science Applications, Inc.

Remarks:
Software developed and tested on UNIVAC 1108.

All equations referenced in code below are contained in the final report - The control of float zone interfaces by the use of selected boundary conditions.

Input variables and functions:
P = Peclet number
MSUM = Number of terms in series expansion of temperature distribution (the desired solution)
BASIS = User provided functions to expand the melt zone surface control function - See Eq. 14.0.18 of final report and subroutine basis of this code.
XA = Axial position of lower end of cylinder
XN = Axial position of upper end of cylinder
MGRID = Number of grid points used in numerical solution of O. D. F. boundary value problem resulting from transformation of Fre the modeling temperature.
NR = Number divisions of cylinder radius used in output of temperature distribution
RKS = Solid thermal conductivity
RKL = Liquid thermal conductivity
RL = Product of crystal growth rate, solid density, and latent heat of fusion
SL = Length of solid regions to be considered
Q = Length of melt zone
IMFC = 1 if discrete data point form of the surface temperature is used provided
0 if a user defined functional form of the surface temperature is provided
(XD, YO) = User provided data pts for the axial distance (XD) and corresponding surface temperature (YO)
M = Number of data pts, input if IMFC = 1

Figure C-3. Compute code list for Problem P1-3

C-33
Figure C-3. Computer Code List For
Problem Pl-3 (Cont)
Figure C-3. Computer Code List For Problem PI-3 (Cont)
Figure C-3. Computer Code List For Problem P1-3 (Cont)

C-36
Figure C-3. Computer Code List For Problem P1-3 (Cont)
Figure C-3. Computer Code List For Problem P1-3 (Cont)
Figure C-3. Computer Code List For
Problem Pl-3 (Cont)
CALL J~(VAR~Y)
P%I(n,r)sv
24n
CONTINUE
23n
CONTINUE
Dn=0x
22n
DO 750 I=1,10
21n
T(J)=0.0
20n
DO 270 H=1,MSUM
19n
T(J)=T(J)+2.0*PSI(I,J)*THFTAB(M,J)/SQJ1(M)
18n
CONTINUE
17n
X=X0+(J-1)*DX
16n
CALL HF (X, ANS)
15n
T(J)=Y(J)+ANS
14n
CONTINUE
13n
DO 280 J=1,10
12n
T(J)=0.0
11n
JHOLD=NSTOP+10+J
10n
DO 290 H=1,MSUM
9n
T(J)=T(J)+2.0*PSI(I,J)*THFTAB(M,JHOLD)/SQJ1(M)
8n
CONTINUE
7n
X=X0+(JHOLD-1)*DX
6n
CALL HF (X, ANS)
5n
T(J)=Y(J)+ANS
4n
CONTINUE
3n
CC Approximate Thermal Gradients At Cylinder Ends
2n
= See Equations (2.2.21) And (2.2.22) Of Final Report
1n
GRANX(I) = 3*IF(6)+18*IF(7)-36*I*8*IF(9)-25*IF(10)/(12*DX)
0n
GRANY(I) = 3*IF(5)+18*IF(4)-3*I*3*IF(3)+8*IF(2)-25*IF(11)/(12*DX)
-1n
IF(SKIP=1=11)
0n
XHOLD=ISKIP=0.00000001/10.0
-2n
IF(XHOLD,GT,0.005) GOTO 250
-3n
WRITE(*,3001)R(I),GRANX(I),GRANY(I)
-4n
FORMAT(1,M,49X,F8.6,3X,E17.9,7X,E17.9)
-5n
CONTINUE
-6n
RETURN
-7n
END

CC This Subroutine Approximates (by finite difference) General Equation Of Final Report

SUBROUTINE QBAR(M,X,ANS)
REAL JJ,JILAM
COMMON/C1/RLAM(20),J1(20),JILAM(20)
COMMON/FRAD/PI,*TERM,MSUM,X0,XN,HGRID,NR
EPSLON=0.01
X1=X+EPSLON
X2=X-EPSLON
CALL HF (X1, ANS)
CALL HF (X2, ANS)
G0=P(ANS2-ANS1)/(2.0*EPSLON)
G0=(ANS2-ANS1)/(EPSLON+EPSLON)
ANS=G0*J1(1,M)
RETURN
END

Figure C-3. Computer Code List For Problem Pl-3 (Cont)
SUBROUTINE DELTERM, DELNSYS
COMMON/READ/F, TERM, S, SUM, X, XN, GRID, NR
COMMON/CASE, I, CASE, T
COMMON/GRID/GRID, NR, R, R, R
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMMON/GRID/GRID, GRID, GRID, GRID, GRID
COMMON/CASE/CASE, CASE, CASE, CASE, CASE
COMM
Figure C-3. Computer Code List For Problem P1-3 (Cont)
SUBROUTINE AFC(R,ANS)
COMMON/RAD1/P,TERM,HSUM,X0,XN,GRID,NR
CALL HFC(XN,ANS)
RETURN
END

SUBROUTINE HFC(X0,ANS)
COMMON/RAD1/P,TERM,HSUM,X0,XN,GRID,NR
CALL HFC(XN,ANS)
RETURN
END

SUBROUTINE IFLSQ Below is the IMSL LEAST SQUARES FUNCTION
CALL IFLSQ(F,R,Y,NR,COEF,IER), Y(10), COEF(20), IER(460)
EXTERNAL P

SUBROUTINE COEFS(r,y,NR,NCOF,CQCF,IER)
EXTERNAL P

FUNCTION F(R,N)
COMMON/CI/R,LAMBDA(20),J1(20),JI(20)
x=LAMBDA(N)*R
CALL JO(X,Y)
F= Y
RETURN
END

FUNCTION JO(X,Y)
RETURN
END

SUBROUTINE JOI(X,Y)
RETURN
END

Figure C-3. Computer Code List For Problem Pl-3 (Cont)
Figure C-3. Computer Code List For
Problem P1-3 (Cont)
Figure C-3. Computer Code List For Problem Pl-3 (Cont)

C-45
CALL DBARIS(K,0,ANS)
AL2(3,K)=ANS
CALL DBARIS(K,G,ANS)
AL2(4,K)=ANS
CONTINUE
DO 510 WYS=MINNSY,MAXNSY,CFLNSY
DO 500 ITERM=INTERM,MAXTERM,OFTERM
NNM=2*TERM
IF(INN.LE.NSYS) GOM500
DO 520 I=1,G
RHS(I)=RHS(I)
DO 550 J=1,J0
AL(I,J)=AL(I,J)
CONTINUE
DO 570 I=1,TERMF
JMN=TERM+1
TERM=TERM+1
DO 580 J=1,J0
AL(I,J)=AL(I,J)
CONTINUE
RHS(I)=RHS(I)
CONTINUE
E(1)=0.0
E(2)=0.0
E(3)=0.0
E(4)=0.0

CONTINUE
50 CONTINUE
CALL LLRRF(AL,NN,NN,NN,NN,RHS,TERM,E,CMPOLY,20,INK,VRK,IER)
CONTINUE
52 CONTINUE
WRITE(6,90)
90 FORMAT(1H1,10X,7HMM E L T Z E N E S U R F A C E C O N T R O L I N G C O E Q F I C E N T S )
WRITE(6,789) ITERM,HNTS
789 FORMAT(1H1,10X,7HMM E L T Z E N E S U R F A C E C O N T R O L I N G C O E Q F I C E N T S )
WRITE(6,90)
90 FORMAT(1H1,10X,7HMM E L T Z E N E S U R F A C E C O N T R O L I N G C O E Q F I C E N T S )
CONTINUE
50A CONTINUE
RETURN
END

SUBROUTINE INTEGRAL(M,N,ANS)
EXTERNAL G
COMMON/C2/GRADAT0(101),GRADATQ(101),RHS1(20),RHS2(20),S(20),G,
MT1(20,101),MT2(20,101),AL2(46,20),RHS(44),MHR(1901),IMK(20)
COMMON/C27/KERNEL,MMKERNEL,MMKERNEL
KKERNEL1
MKERNEL2
MKERNEL3

Figure C-3. Computer Code List For Problem P1-3 (Cont)
Figure C-3. Computer Code List For Problem PI-3 (Cont)

C-47
Figure C-3. Computer Code List For Problem P1-3 (Cont)
Figure C-3. Computer Code List For Problem P1-3 (Cont)

U.S. GOVERNMENT PRINTING OFFICE 1984-746-070/113

C-49