LAMINAR FREE CONVECTIVE HEAT TRANSFER IN THE SUPERCRITICAL REGION WITH VARIABLE FLUID PROPERTIES

By

AFSHIN JAHANSHAHI GHAJAR

Bachelor of Science
Oklahoma State University
Stillwater, Oklahoma
1974

Master of Science
Oklahoma State University
Stillwater, Oklahoma
1975

Submitted to the Faculty of the Graduate College
of the Oklahoma State University
in partial fulfillment of the requirements
for the Degree of
DOCTOR OF PHILOSOPHY
July, 1979

LAMINAR FREE CONVECTIVE HEAT TRANSFER IN THE SUPERCRITICAL REGION WITH VARIABLE FLUID PROPERTIES

Thesis Approved:

Thesis Adviser

Dean of the Graduate College

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

NOMENCLATURE

English Letter Symbols

A,B,C,D	constants used in Equations (4.15) and (4.26)
A1	defined by Equation (3.16)
A2	defined by Equation (3.17)
A3	defined by Equation (3.18)
A_2,A_3,A_4,A_5,B_2,B_3,B_5 C_2,C_3,C_5,b,K	constants used in the equation of state, see Equation (4.1)
A_{11}-A_{55}, B_{11}-B_{55}, C_{11}-C_{55}, D_{11}-D_{55}	constants used in Equations (5.1) through (5.5)
a,b,c,d	constants used in Equations (7.5), (7.10), and (7.18)
a_1,b_1,c_1	constants used in Equations (7.8) and (7.17)
a_o through a_5	defined by Equation (4.2)
B1	defined by Equation (3.19)
B2	defined by Equation (3.20)
C	defined by Equation (2.4)
C_P	specific heat at constant pressure
C_V	specific heat at constant volume
$\overline{C}$	reference temperature constant
CA through CD	constants used in Equation (5.7)
CE through CH	constants used in Equation (5.8)
CI through CL	constants used in Equation (5.9)
CM,CN.CP,CR, CS,CT,CU,CW	constants used in Equation (5.10)

CAA through CDD	constants used in Equation (5.11)
CEE through CHH	constants used in Equation (5.12)
c,n	constants used in Equation (7.4)
$F(\eta)$	similarity variable, defined by Equation (3.7)
Gr	Grashof number, $Gr = [gL^3/\nu^2 (\rho_\infty - \rho_w/\rho_w)]$
g	acceleration due to gravity
h	heat transfer coefficient
K	thermal conductivity
L	overall length of plate
M	molecular weight
Nu	Nusselt number, $Nu = hL/K$
P	pressure
P_c	critical pressure
P_r	reduced pressure, $P_r = P/P_c$
Pr	Prandtl number
$\dot{q}"$	heat flux
R	gas constant
Ra	Rayleigh number, Ra = (Pr Gr)
s	specific entropy
T	temperature
T_w	wall or plate temperature
T_∞	free stream temperature
T_m	mean temperature
T_{ref} or T_x	reference temperature
T_c	critical temperature
T_r	reduced temperature, $T_r = T/T_c$
T_M	pseudocritical or transposed critical temperature

u	velocity component along the plate, or specific internal energy
v	velocity component perpendicular to the plate, or specific volume
X	defined by Equation (4.14)
x	coordinate along the plate, or the parameter in the reference temperature equation, Equation (7.3)
x_{13}, x_{14}, x_{15}	constants used in Equation (4.9)
y	coordinate perpendicular to the plate
z	defined by Equation (4.2)
z_c	critical compressibility factor, see Equation (4.25)

Greek Letter Symbols

ΔT	temperature difference, $\Delta T = T_w - T_\infty$
δ	boundary layer thickness
η	similarity coordinate as defined by Equation (3.6)
λ	defined by Equation (4.25)
μ	absolute viscosity
ν	kinematic viscosity, $\nu = \mu/\rho$
ρ	density
ρ_c	critical density
ρ_r	reduced density, $\rho_r = \rho/\rho_c$
ψ	stream function, defined by Equation (3.5)
$\theta(\eta)$	dimensionless temperature, defined by Equation (3.8)

Subscripts

c	refers to critical property

c.p.	refers to constant property solution
m	evaluated at the mean temperature
o	refers to zero or low pressure property
P	refers to constant pressure
r	refers to reduced property
ref	evaluated at the reference temperature
T_x	evaluated at the temperature T_x
v	refers to constant volume
v.p.	refers to variable property solution
w	evaluated at the wall temperature
x	evaluated at a particular point along a surface, a local parameter as opposed to a mean parameter
∞	evaluated at the free stream temperature

Symbols for Units

atm	atmosphere
Btu	British thermal units
°C	degree Celsius
cal	calorie
cm	centimeter
ft	foot
°F	degree Fahrenheit
g	gram
hr	hour
in.	inch
°K	degree Kelvin
lbf	pound force
lbm	pound mass

mm	millimeter
psia	pound force per square inch absolute
°R	degree Rankine
sec	second

CHAPTER I

INTRODUCTION

Usually, rates of heat transfer by free convection are low compared to forced convection. However, several investigators (1) have observed high rates of free convection heat transfer in the critical and supercritical regions. This is primarily due to the fact that specific heat and compressibility for fluids in the critical and supercritical regions increase significantly when compared to the variation of these properties in the subcritical region.

High rates of convective heat transfer to fluids in the supercritical region has become increasingly important in connection with such applications as the use of helium to cool the coils of superconducting electromagnets and superconducting electronic or power-transmission equipment, the use of hydrogen as a working fluid or fuel for both chemical and nuclear rockets; the use of water in electricity generating plants, and the use of methane as a coolant and fuel for the supersonic transport (1). Free convection is also important in the efficient cooling of turbine blades in high speed and high temperature gas turbines, in cooling of rocket motors by hydrocarbons, in the refrigeration problems, and in cyrogenic computer design (2).

The growing applications for high rates of free convection heat transfer to fluids in the supercritical region necessitates the accurate analytical prediction of heat transfer in this region. The analysis for

heat transfer to fluids in the supercritical region is fairly difficult because of the peculiar and relatively large variation of the physical properties of fluids in the supercritical region with temperature, particularly because of the inversion of dependence of constant pressure specific heat upon temperature (see Figures 2 through 6, Appendix B).

In the present study the isothermal, vertical flat plate is chosen as the mathematical model and a steady, two-dimensional laminar boundary layer type of flow is assumed. Many investigators have made analytical investigations of this problem (3) through (7), but the existing prediction schemes are generally inefficient, complicated, and difficult to use. The available prediction methods are not general (i.e., they are dependent on the fluid of interest), and require extensive computer programming. They also require information on the development of analytical procedures for determining the thermodynamic and transport properties including their derivatives for the fluids of interest. A simple, accurate, and practical method for predicting heat transfer to fluids in the supercritical region appeared to be desirable.

The primary objectives of this study were: (1) to develop an accurate, efficient, and general scheme to predict heat transfer to fluids in the supercritical region under variable property conditions in laminar free convection on an isothermal vertical flat plate, and (2) to extend this prediction scheme to develop an accurate, shorthand method for calculating heat transfer to fluids in the supercritical region. This shorthand method involves the use of the results for constant property fluids, and of a rule (reference temperature) for extending these constant property results to variable property situations in the supercritical region.

The problem analysis included the following phases:

1. The application of the equations expressing conservation of mass, momentum, and energy to the problem of free convection under steady state conditions on a vertical flat plate with constant wall and free stream temperatures. These sets of nonlinear partial differential equations with variable coefficients were reduced to a set of two nonlinear ordinary differential equations with variable coefficients by means of the compressible flow stream function and a similarity variable. The variable coefficients were expressed in dimensionless form in terms of the thermodynamic and transport properties (see Chapter III).

2. The development of analytical procedures for determining the thermodynamic and transport properties including their derivatives for Refrigerant-114, water, and carbon dioxide in the supercritical region. The variation with temperature at constant pressure of density, specific heat at constant pressure, absolute viscosity, and thermal conductivity were determined. For most of the fluids encountered in the supercritical region, expressions for physical properties did not exist and were developed from the available experimental data (see Chapter IV).

3. The numerical solution of the reduced ordinary nonlinear differential momentum and energy equations by the method of Lentini-Pereyra (8). This is a variable order finite difference method for nonlinear multipoint boundary value problems. The method is extremely accurate and performs very efficiently. The numerical solution to these differential equations is complicated by the fact that two of the boundary conditions are unknown, namely, $F''(0)$ and $\theta'(0)$ (see Chapter V).

4. The development of computer programs to: (a) evaluate transport and thermodynamic properties of Refrigerant-114, water, and carbon

dioxide in the supercritical region; (b) solve the reduced ordinary nonlinear differential equations; and (c) calculate heat transfer coefficients for different combinations of wall and free stream temperatures (see Chapter V and Appendix D).

5. The demonstration of the validity and universality of the present variable free convective heat transfer model with the available experimental results and theoretical models for Refrigerant-114, water, and carbon dioxide (see Chapter VI).

6. The development of a correlation for the calculated heat transfer data employing conventional dimensionless groups. The correlation developed was for Refrigerant-114, water, and carbon dioxide applicable in the range of supercritical temperatures and pressures considered in this study (see Chapter VII).

7. The development of an optimization technique that would predict the reference temperature. This was achieved by evaluating the physical properties in the dimensionless groups of the correlation developed for the calculated heat transfer data (see part 6) at an optimum temperature which resulted in the best agreement between the correlated and calculated heat transfer coefficients for certain values of wall and free stream temperatures at different specified constant pressures (see Chapter VII).

8. The development of general plots for reference temperature for different fluids in the supercritical region and wide ranges of temperature differences between the wall and free stream at different constant pressures. These generalized plots were developed for Refrigerant-114, water, and carbon dioxide (see Chapter VII).

9. The demonstration of the validity and universality of the reference temperature method. This was achieved by comparing the constant property heat transfer results obtained using the reference temperature scheme with the available experimental and theoretical property heat transfer results for the three fluids (see Chapter VII).

CHAPTER II

LITERATURE SURVEY

In this chapter a brief review of the literature describing the analytical and experimental studies related to this work are presented.

The survey of the work reported in the literature was classified into three different categories. They are:

1. Free Convection
   (a) Analytical
   (b) Experimental
2. Physical Properties
3. Reference Temperature.

Each one of the above categories will be presented separately.

Free Convection

Analytical Work

The first analytical solution to the problem of free convection was obtained by Pohlhausen (9). He was able to show how the partial differential equations of momentum and energy could be transformed into ordinary differential equations. He obtained a solution for the case of air with Prandtl number of 0.73, assuming that the fluid properties are constant with the exception of the role that density variations play in developing buoyancy forces.

Ostrach (10) presented a detailed treatment of the constant property problem for free convection along a vertical flat plate. His paper contained a thorough coverage of the derivations of the equations and presents numerically computed cases for the range of Prandtl numbers from 0.01 to 1000.

The first variable property analytical study was done by Sparrow and Gregg (11). Their work was concerned with free convection along a vertical flat plate. They considered several ideal gases in which a perfect gas type variation was used for density and the constant pressure specific heat; absolute viscosity and thermal conductivity were considered as functions of temperature alone. For absolute viscosity and thermal conductivity a Sutherland type of relation was used. Therefore, their results are for a limited class of fluid behavior.

Fritsch and Grosh's (3) analytical work relates to the problem of free convection heat transfer to supercritical water along a vertical flat plate. In their investigation variations of density and constant pressure specific heat were only considered, while the absolute viscosity and thermal conductivity were assumed to be constant and were evaluated at the mean temperature. The results from their partial variable property analysis for water were as much as 22 percent lower than the experimental results (12).

Fritsch and Grosh concluded that better agreement would have been obtained if account were taken of the variations with temperature of the transport properties as well as the thermodynamic properties. They utilized the assumption of constant absolute viscosity and thermal conductivity to reduce the set of nonlinear partial differential equations to a set of nonlinear differential equations. This procedure has the

effect of eliminating the role that the derivatives of these properties play in their analysis; however, in certain portions of the supercritical region these properties do have strong variations. Because of the restrictions placed on their work, the applicability of their solution in the region above the thermodynamic critical point is not always sufficiently accurate.

Hasegawa and Yoshioka (4) analyzed laminar free convection heat transfer from an isothermal vertical plate for fluids close to the transposed critical point. They calculated the first perturbation solution to the non-perturbated solution for the fluid having physical properties at transposed critical point, expressing all the relevant physical properties as some power functions of enthalpy. Their expressions for physical properties as a function of enthalpy is applicable only for a narrow temperature range. Their results seem to be in qualitative agreement with the experimental results only in case of small temperature differences (12). For large temperature differences, however, the heat transfer coefficient estimated from their analysis was much smaller than those measured. Therefore, the results of their work may be reasonable in a narrow range of temperature difference between wall or free stream and transposed critical point.

The theoretical study of Nishikawa and Ito (6) was made on the laminar free convective heat transfer from an isothermal plate to fluids at supercritical pressures by taking into account the temperature dependence of all relevant physical properties. In their studies the heat transfer characteristics of water and carbon dioxide were clarified by integrating the similarity transformed differential equations numerically. The results of their analysis agrees favorably with the analysis of Fritsch and

Grosh (3), but fails to show good agreement with experimental data of water (12) and carbon dioxide (29) (30).

Parker and Mullin (5) extended the work of Sparrow and Gregg (11) to obtain a method which is more generally useful in all regions of fluid behavior and for all types of fluid in which proper thermodynamic relationships are available. Their technique allows for variation in density, specific heat, absolute viscosity, and thermal conductivity. An important step in the development of their method is the use of thermodynamic relationships to obtain derivatives of properties with respect to temperature. Their method was demonstrated for Refrigerant-114 using a Martin (13) equation of state along with Sutherland and Bromley (14) (15) equations for absolute viscosity and thermal conductivity, corrected for pressure.

An analytical investigation of free convection heat transfer to supercritical water was made by Nowak and Konanur (7). They calculated heat transfer to supercritical water (at 3400 psia in the transposed critical region) by stable laminar free convection from an isothermal, vertical flat plate. In their investigation the actual variations with temperature of all or some of the thermophysical properties of supercritical water were taken into consideration. Fair agreement was found between their analytical values and existing experimental data (12). They concluded that the reasons for lack of complete agreement between their analytical results and the experimental data are due to deficiencies in the existing experimental heat transfer data, and errors in the thermophysical properties of water in the supercritical region.

Experimental Work

The free convection experimental work related to this study is summarized in Table I of Appendix A. The experimental work presented in this section deals primarily with heat transfer from small test sections, such as wires, filaments, and plates. Table I lists source references for the experimental work, the type of fluid used, the geometry of the apparatus, and the range of conditions at which the experiments were conducted (1).

Physical Properties

Table II of Appendix A summarized the physical properties of the dominant working fluids in the supercritical region. This table lists source references for equations of state and transport properties. It also includes an indication as to whether the particular authors suggest computation of other useful thermodynamic and transport properties (1).

Reference Temperature

There are no reported analytical investigations to calculate reference temperature in the supercritical region under variable property conditions in laminar free convection on a vertical flat plate. The utility of the reference temperature lies in the fact that it allows constant property results to be used to compute variable property results.

Parker et al. (5) and Nowak et al. (7) in their work have pointed out that the use of typical reference temperatures in the supercritical region would not produce consistent results. The most widely used reference temperatures are: the surface temperature, the free stream temperature, or a temperature partway between the surface and the free stream.

These reference temperatures produce good results for the problems with moderate property variations. In the supercritical region where there exist large variations in the physical property values compared to the fluids in the subcritical region a more sophisticated method is required.

The only reported analysis in the literature that calculates reference temperature under variable property conditions in laminar free convection on an isothermal vertical flat plate is the work of Sparrow and Gregg (11). Because of the restrictions placed on their work, their suggested reference temperature is not applicable in the entire region above the thermodynamic critical point. They propose that for gases the variable property problem of free convection can be computed using constant property results by evaluating the properties at a reference temperature defined by

$$T_{ref} = T_w - 0.38\,(T_w - T_\infty) \tag{2.1}$$

Since the different gases considered by Sparrow and Gregg did not have large variations in property values compared to the fluids in the critical and supercritical regions, the constant property results for which the property values were evaluated at the reference temperature defined by Equation (2.1) agreed within one percent of the variable property results.

Equation (2.1) is essentially based on a perfect gas state with a power-law and Sutherland-type formula to describe absolute viscosity and thermal conductivity variations. In effect, Sparrow and Gregg found a reference temperature for evaluating thermal conductivity and absolute viscosity such that the constant property solution would yield the correct value for heat transfer coefficient. The constant 0.38 of the

reference temperature Equation (2.1) can be computed using the information in their paper and the following reasoning:

1. The heat transfer for the variable property case is equal to the heat transfer for the constant property case using the reference temperature; that is:

$$\dot{q}''_w = \dot{q}''_{ref} \tag{2.2}$$

2. Continuing

$$K_w C_w \theta'(0)_w = K_{ref} C_{ref} \theta'(0)_{ref} \tag{2.3}$$

where

$\theta'(0)_w$ = the dimensionless temperature slope at the wall for the variable property case; and

$\theta'(0)_{ref}$ = the dimensionless temperature slope at the wall for the constant property case.

3. The term C is defined as

$$C = \left[\frac{g}{4\nu^2}\left(\frac{\rho_\infty - \rho_w}{\rho_w}\right)\right]^{1/4} \tag{2.4}$$

and therefore,

$$\frac{C_{ref}}{C_w} = \left(\frac{\nu_w}{\nu_{ref}}\right)^{1/2} \tag{2.5}$$

4. The substitution of Equation (2.5) into Equation (2.3) gives the necessary equation to be satisfied if Equation (2.2) is to be true; that is:

$$\left(\frac{K_{ref}}{K_w}\right)\left(\frac{\mu_w}{\mu_{ref}}\right)^{1/2}\left(\frac{\rho_{ref}}{\rho_w}\right)^{1/2} = \frac{\theta'(0)_w}{\theta'(0)_{ref}} \tag{2.6}$$

Using Equation (2.6) and the expressions for the physical properties of a given fluid including the values of $\theta'(0)_w$ and $\theta'(0)_{ref}$, one can find an expression to compute the reference temperature constant directly.

Sparrow and Gregg considered the fluid (Gas A)

$$P = \rho R T \tag{2.7a}$$

$$K \sim T^{3/4} \tag{2.7b}$$

$$\mu \sim T^{3/4} \tag{2.7c}$$

$$C_p = \text{constant} \tag{2.7d}$$

$$Pr = \text{constant} \tag{2.7e}$$

The value of $\bar{C}$, the constant of the reference temperature Equation (2.1), will now be evaluated for this type of fluid. The various ratios that are required by Equation (2.6) are formed as follows:

$$\frac{\rho_{ref}}{\rho_w} = \frac{T_w}{T_{ref}} \tag{2.8}$$

$$\frac{\mu_w}{\mu_{ref}} = \frac{K_w}{K_{ref}} = \left(\frac{T_w}{T_{ref}}\right)^{3/4} \tag{2.9}$$

Substituting Equations (2.8) and (2.9) into Equation (2.6) yields

$$\frac{T_{ref}}{T_w} = \left(\frac{\theta'(0)_{ref}}{\theta'(0)_w}\right)^{8} \tag{2.10}$$

The reference temperature Equation (2.1) may be rearranged as follows:

$$\frac{T_{ref}}{T_w} = 1 - \bar{C}\left(1 - \frac{T_\infty}{T_w}\right) \tag{2.11}$$

The combining of Equations (2.10) and (2.11) and then solving for $\bar{C}$ gives

$$\bar{C} = \frac{1 - (\theta'(0)_{ref}/\theta'(0)_w)^8}{1 - (T_\infty/T_w)} \tag{2.12}$$

The results obtained from application of Equation (2.12) to Gas A with Pr = 0.7 are tabulated in Table III (Appendix A). These results were used to evaluate the constant of the reference temperature equation.

Another fluid considered by Sparrow and Gregg was (Gas B)

$$P = \rho R T \tag{2.13a}$$

$$K \sim T^{2/3} \tag{2.13b}$$

$$\mu \sim T^{2/3} \tag{2.13c}$$

$$C_p = \text{constant} \tag{2.13d}$$

$$Pr = \text{constant} \tag{2.13e}$$

If the same procedure that was used for Gas A is applied to Gas B, the following is obtained:

$$\bar{C} = \frac{1 - (\theta'(0)_{ref}/\theta'(0)_w)^6}{1 - (T_\infty/T_w)} \tag{2.14}$$

The application of Equation (2.14) to Gas B with Pr = 0.7 produces results similar to those of Gas A tabulated in Table III (Appendix A).

Sparrow and Gregg plotted T_{ref}/T_w versus $(T_w - T_\infty)/T_w$ for various values of T_w/T_∞, that is, Equation (2.11). The slope of these lines represents the reference temperature constant $\bar{C}$. The table of $\bar{C}$ values for Gas A given in Appendix A confirms the choice of $\bar{C}$ = 0.38 as a good estimate of the constant used for the reference temperature equation.

In view of the types of fluid states assumed by Sparrow and Gregg, they have restricted their results to that portion of the property diagram where their physical property assumptions are valid which is primarily in the perfect gas region. Therefore, the reference temperature equation suggested by Sparrow and Gregg is not valid throughout the supercritical region. This is primarily due to the fact that the parameter $\overline{C}$ of Equation (2.11) is no longer a constant in this region.

Since heat transfer calculation for the problems relating to free convection along a vertical flat plate in the supercritical region is complicated, tedious, and time-consuming, it would be desirable to use the simple constant property equations to produce the variable property results. The important question is what reference temperature should be used for evaluation of the properties of the constant property problem in order to produce the variable property results. In view of lack of such investigations in the literature, it would be the primary objective of this study to answer this question.

In order to answer the question on the choice of a reference temperature, it is necessary to employ an analytical procedure that predicts heat transfer to fluids in the supercritical region under variable property conditions. The literature search shows that very little analytical work has been done in predicting heat transfer to fluids in the supercritical region considering all the physical property variations. The few existing prediction schemes are inefficient, complicated, difficult to use, and in most cases do not compare well with the experimental data. Therefore, in order to complete the reference temperature analysis, it is also necessary to develop a general, accurate, and reliable method to

predict heat transfer to fluids in the supercritical region under variable property conditions.

CHAPTER III

THE FUNDAMENTAL EQUATIONS FOR LAMINAR FREE CONVECTION

It is desired to calculate the nondimensionalized heat transfer coefficient for constant and variable fluid property problems. This chapter outlines the basic equations that were used to solve the constant and variable fluid property problems for laminar free convection on an isothermal vertical flat plate. The necessary mathematical developments that needed to be undertaken in order to generate the reduced set of equations to be handled by the computer program are introduced. Finally, the expressions needed for heat transfer calculations are presented.

Physical Model

The physical model and the coordinate system of the present model are shown in Figure 1 (Appendix B). Since it seems easier to visualize occurrences associated with the case where the wall temperature exceeds the free stream temperature, the analysis will be directed toward this situation. However, for the reverse case of the free convection heat transfer problem, a slight change in the coordinate system is necessary, so that the results of the above case are applicable.

In the present model, the lower edge of the plate is chosen as the origin. The distances X and Y are measured along and normal to the plate, respectively. The plate extends from X equal to zero to X equal

to infinity (i.e., a semi-infinite vertical plate). The direction of the gravity force is taken as pointing downward. This leads to the free convection motion to be upward and away from the leading edge, and the boundary layer thickness at the leading edge to be zero. The velocity components along X and Y directions are u and v, respectively.

Fundamental Equations

The fundamental equations of conservation of mass, momentum, and energy for laminar free convection in a boundary layer may be written in the following form (see Reference (51)):

Mass

$$\frac{\partial}{\partial x}(\rho u) + \frac{\partial}{\partial y}(\rho v) = 0 \tag{3.1}$$

Momentum

$$\rho\left(u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y}\right) = g(\rho_\infty - \rho) + \frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y}\right) \tag{3.2}$$

Energy

$$\rho C_p\left(u\frac{\partial T}{\partial x} + v\frac{\partial T}{\partial y}\right) = \frac{\partial}{\partial y}\left(K\frac{\partial T}{\partial y}\right) \tag{3.3}$$

In accordance with the usual practice in free convection, viscous dissipation and work against the gravity field were neglected. The deriving force for the free convection motion is the buoyancy term $g(\rho_\infty - \rho)$; it is formed by combining the pressure gradient $\partial P/\partial x$ with the body force ρg.

The problem will be completely defined by assigning the boundary conditions. The boundary conditions appropriate to the problem are

$$\left.\begin{aligned} v &= 0 \\ u &= 0 \\ T &= T_w \end{aligned}\right\} y = 0 \qquad \left.\begin{aligned} u &= 0 \\ T &= T_\infty \end{aligned}\right\} y = \infty \tag{3.4}$$

In writing the conservation equations and their boundary conditions the following assumptions were made:

(1) The flow is laminar.

(2) The flow is steady.

(3) The flow is two-dimensional.

(4) The fluid properties are variable.

(5) The wall temperature is constant.

(6) The free stream temperature is constant.

(7) The mechanism of heat transfer is only by conduction and convection.

Reduction to Ordinary Differential Equations

Equations (3.1) through (3.3) form a set of nonlinear, nonhomogeneous, simultaneous partial differential equations with variable coefficients, whose exact analytical solutions are unknown. These equations can be reduced to a set of two nonlinear ordinary differential equations by means of the compressible stream function and a similarity variable. The stream function reduces the number of dependent variables, the conservation of mass equation is eliminated. However, it has the disadvantage that the order of the momentum and energy equations are increased. The similarity variable transforms the partial differential equations to ordinary differential equations. The stream function and similarity variable are defined as follows:

Stream Function

$$u = \frac{\rho_w}{\rho} \left(\frac{\partial \psi}{\partial y}\right) \tag{3.5a}$$

$$v = -\frac{\rho_w}{\rho} \left(\frac{\partial \psi}{\partial x}\right) \tag{3.5b}$$

Similarity Variable

$$\eta = C_w x^{-1/4} \int_0^y \frac{\rho}{\rho_w} dy \tag{3.6}$$

where

$$C_w = \left[\frac{g}{4\nu_w^2} \left(\frac{\rho_\infty - \rho_w}{\rho_w}\right)\right]^{1/4}$$

According to the definition of similarity, the new dependent velocity and temperature functions, F and θ, respectively, are assumed to be dependent upon η alone. The new dependent variables F and θ are given by

$$F(\eta) = \left(\frac{1}{4\nu_w C_w}\right) \left(\frac{\psi}{x^{3/4}}\right) \tag{3.7}$$

$$\theta(\eta) = \frac{T - T_\infty}{T_w - T_\infty} \tag{3.8}$$

The function F is related to the tangential and normal velocities u and v and the function θ is related to the temperature distribution.

Using the transformation equations, Equations (3.5) through (3.8), we will obtain the set of two nonlinear ordinary differential equations, previously published by Sparrow and Gregg (11).

$$\frac{d}{d\eta} \left(\frac{\rho\mu}{\rho_w \mu_w} F''\right) + 3FF'' - 2(F')^2 + \frac{(\rho_\infty - \rho)\rho_w}{(\rho_\infty - \rho_w)\rho} = 0 \tag{3.9}$$

$$\frac{d}{d\eta}\left(\frac{\rho K}{\rho_w K_w}\theta'\right) + 3Pr_w\left(\frac{C_P}{C_{P_w}}\right)F\theta' = 0 \tag{3.10}$$

The boundary conditions, Equation (3.4), transform to

$$\left.\begin{aligned} F &= 0 \\ F' &= 0 \\ \theta &= 1 \end{aligned}\right\} \eta = 0 \qquad \left.\begin{aligned} F' &= 0 \\ \theta &= 0 \end{aligned}\right\} \eta = \infty \tag{3.11}$$

It may be noted that the transformation relations, as well as the resulting ordinary differential equations, can be reduced directly to the well-known equations for the constant property fluid by making the usual approximations. The resulting equations for momentum and energy, respectively, are

$$F''' + 3FF'' - 2(F')^2 + \theta = 0 \tag{3.12}$$

$$\theta'' + 3Pr F\theta' = 0 \tag{3.13}$$

The boundary conditions are identical to those given by Equation (3.11). In Equations (3.12) and (3.13), Prandtl number is the only parameter upon which the solution of the equations depends.

The form in which Equations (3.9) and (3.10) exists requires for the physical properties to be expressed as functions of the independent variable η or as functions of the dependent variables F and θ. The mathematical procedures necessary for solving these equations are long and tedious. However, at a particular pressure, the physical properties can be expressed as functions of temperature alone. Since temperature is a function of $\theta(\eta)$ (see Equation (3.8)), it is an easy matter to obtain the physical properties as functions of $\theta(\eta)$ once the state equations for the physical properties are known. Manipulation of the

coefficients in Equations (3.9) and (3.10) by the use of certain well-known thermodynamic relationships results in the following equations as previously suggested by Parker and Mullin (5). A sample detailed derivation is given in Appendix C.

Momentum Equation

$$F''' = -(A1)\theta' F'' - 3(A2)FF'' + 2(A2)(F')^2 - (A3) \tag{3.14}$$

Energy Equation

$$\theta'' = -(B1)(\theta')^2 - 3(B2)F\theta' \tag{3.15}$$

Dimensionless Coefficients

$$A1 = \left[\frac{1}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_P + \frac{1}{\mu}\left(\frac{\partial \mu}{\partial T}\right)_P\right](\Delta T) \tag{3.16}$$

$$A2 = \frac{\rho_w \mu_w}{\rho \mu} \tag{3.17}$$

$$A3 = (A2)\left[\frac{\rho_\infty - \rho}{\rho_\infty - \rho_w}\right]\frac{\rho_w}{\rho} \tag{3.18}$$

$$B1 = \left[\frac{1}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_P + \frac{1}{K}\left(\frac{\partial K}{\partial T}\right)_P\right](\Delta T) \tag{3.19}$$

$$B2 = (A2)(Pr) \tag{3.20}$$

The five coefficients A1, A2, A3, B1, and B2 can be determined once the equations of state for the physical properties C_p, μ, K, and ρ are known as functions of temperature and pressure. The analytical solutions to the reduced differential Equations (3.14) and (3.15) which satisfy the boundary conditions given by Equation (3.11) are not known and were obtained by numerical methods on a computer (see Chapter V). These differential equations were solved simultaneously due to the appearance of common terms in the two differential equations (i.e., the differential

equations are coupled). Physically, this is due to the fact that the fluid motions in free convection are caused by density variations which in turn are due to the temperature gradients in the fluids.

Heat Transfer at the Wall

The local heat flux from the wall (plate) to the fluid may be calculated using Fourier's law:

$$\dot{q}''_{x,w} = -K_w\left(\frac{\partial T}{\partial y}\right)_{y=0} \tag{3.21}$$

Introducing the dimensionless variables from Equations (3.6) through (3.8) in Equation (3.21), the local heat transfer rate in terms of the solutions of the differential equations becomes

$$\dot{q}''_{x,w} = -K_w(T_w - T_\infty)\, C_w x^{-1/4}\, \theta'(0) \tag{3.22}$$

where $\theta'(0)$, the slope of the dimensionless temperature profile at the wall, is found from the solutions of Equations (3.14) and (3.15) for the variable property case, or Equations (3.12) and (3.13) for the constant property case, along with the boundary conditions given by Equation (3.11).

The local heat transfer coefficient is defined by the expression

$$h_x = \frac{\dot{q}''_{x,w}}{(T_w - T_\infty)} \tag{3.23}$$

A dimensionless representation of the local heat transfer coefficient in terms of the solution of the differential equations is achieved by defining a local Nusselt number and a local Grashof number. The

defined Grashof number is applicable to both constant and variable property case.

$$Nu_{x,w} = \frac{h_x \, x}{K_w} \tag{3.24}$$

$$Gr_{x,w} = \left[\frac{gx^3}{\nu_w^2} \left(\frac{\rho_\infty - \rho_w}{\rho_w}\right)\right] \tag{3.25}$$

Using these definitions of h_x, $Nu_{x,w}$, and $Gr_{x,w}$, the local heat flux given by Equation (3.22) becomes

$$Nu_{x,w} = - \frac{Gr_{x,w}^{1/4}}{\sqrt{2}} \theta'(0) \tag{3.26}$$

Dimensionless heat transfer coefficients on a vertical flat plate in laminar free convection are usually presented in the form of Equation (3.26).

CHAPTER IV

PHYSICAL PROPERTIES

In this chapter the need for the expressions for physical properties as functions of pressure and temperature is established. The availability of these expressions is reviewed. The analytical procedures for determining these expressions are explained. Finally, the expressions developed for physical properties based on the available experimental data are presented.

In order to solve the free convection heat transfer problem (either contant or variable property), the accurate knowledge of both the thermodynamic and transport properties is necessary. That is, before the solutions to the reduced form of the momentum and energy equations, Equations (3.14) and (3.15) for the variable property case or Equations (3.12) and (3.13) for the constant property case, can be obtained it is necessary to determine the five dimensionless coefficients A1, A2, A3, B1, and B2 expressed by Equations (3.16) through (3.20). These coefficients require the knowledge of the following thermodynamic and transport properties:

Thermodynamic Properties	Symbols
1. Density	ρ
2. Coefficient of volume expansion	$\frac{1}{\rho}\left(\partial\rho/\partial T\right)_p$
3. Specific heat at constant pressure	C_p
Transport Properties	**Symbols**
1. Absolute viscosity	μ

2. The term $\frac{1}{\mu}(\partial\mu/\partial T)_P$

3. Thermal conductivity K

4. The term $\frac{1}{K}(\partial K/\partial T)_P$

The values of the above properties can be determined for any temperature at a particular pressure if the following information is known:

1. An equation of state for the fluid of interest.

2. An expression for the specific heat at constant pressure as a function of temperature.

3. An expression for the absolute viscosity as a function of temperature.

4. An expression for the thermal conductivity as a function of temperature.

The most widely used working fluids in the supercritical region are halocarbon compounds (Refrigerants), water, carbon dioxide, helium, hydrogen, nitrogen, oxygen, propane, and ammonia (see Table II, Appendix A). Refrigerant-114 was the only working fluid for which all of the above information was available in the region where both pressures and temperatures were above critical. On the other hand, for most of the widely used fluids in the supercritical region, there is enough experimental thermodynamic and transport property data available in the literature (see Table II, Appendi A). Therefore, based on the available experimental data, it was possible to develop the analytical procedures for determining the required expressions for the physical properties.

Determination of the four required expressions for physical properties to start the heat transfer calculations is a very difficult task in itself. This complication is due to the peculiar variations of the physical properties of fluids in the supercritical region. For

illustrative purposes Figures 2 through 5 (Appendix B) show variations in the thermodynamic and transport properties of supercritical water with respect to temperature at a constant pressure of 3400 psia.

Once the types of equations that would represent the physical properties were recognized, it was possible to develop the analytical procedures necessary for determination of the physical property expressions based on the experimental data. A computer program called MARQ developed by Jackson and Chandler (52) was adopted for this purpose. This program performs a nonlinear least squares fit of a user-supplied function to a given set of data, using Marquardt-S method, or the Gauss-Newton method, or a modified Gauss-Newton method. A complete documented listing of the MARQ program is given in Appendix D.

To demonstrate the universality of the present model, it was necessary to test the method with different fluids in the supercritical region. The fluids chosen for this study were Refrigerant-114, water, and carbon dioxide. These fluids were chosen due to the availability of reliable physical property expressions and experimental data (see Table II, Appendix A), and also existence of theoretical and experimental heat transfer data (see Table I, Appendix A).

To start the solutions of the reduced differential equations, physical property expressions were developed for water and carbon dioxide. These expressions along with the thermodynamic and transport properties of Refrigerant-114, compiled and developed by Mullin (50), are the equation of state, coefficient of volume expansion, specific heat at constant pressure, absolute viscosity, the term $[1/\mu(\partial\mu/\partial T)_p]$, thermal conductivity, and the term $[1/K\ (\partial K/\partial T)_p]$.

Equation of State

Martin (13) developed a twelve parameter equation of state for Refrigerant-114. This equation, which is valid in the supercritical region, was used to determine the density as a function of temperature at constant pressure. The same type of equation was used in the MARQ program to fit the pressure-volume-temperature experimental data of supercritical water (40) (41) (53) and supercritical carbon dioxide (54) (55) (56). The fit to the given set of data was excellent. The average absolute errors between the fitted results and the experimental data for both fluids were less than 2 percent.

The equation of state developed by Martin (13) is as follows:

$$P = \frac{RT}{(V-b)} + \frac{A_2 + B_2T + C_2\ EXP(-K\ T_r)}{(V-b)^2} + \frac{A_3 + B_3T + C_3\ EXP(-K\ T_r)}{(V-b)^3} + \frac{A_4}{(V-b)^4} + \frac{A_5 + B_5T + C_5\ EXP(-K\ T_r)}{(V-b)^5} \qquad (4.1)$$

where

$$T_r = \frac{T}{T_c}$$

The units and constants to be used in Equation (4.1) for all three fluids (Refrigerant-114, water, and carbon dioxide) are tabulated in Table IV (Appendix A).

To obtain a value of $\rho = \rho(T)$ at constant pressure it was necessary to use numerical techniques due to the fact that the equation of state is explicit only in pressure. The procedure is as follows:

1. Define the following terms:

$$z = (V - b) \tag{4.2a}$$

$$a_o = P \tag{4.2b}$$

$$a_1 = RT \tag{4.2c}$$

$$a_2 = A_2 + B_2T + C_2 \, EXP(-K \, T_r) \tag{4.2d}$$

$$a_3 = A_3 + B_3T + C_3 \, EXP(-K \, T_r) \tag{4.2e}$$

$$a_4 = A_4 \tag{4.2f}$$

$$a_5 = A_5 + B_5T + C_5 \, EXP(-K \, T_r) \tag{4.2g}$$

2. The equation of state, Equation (4.1), was rewritten as follows:

$$f(z) = a_oz^5 - a_1z^4 - a_2z^3 - a_3z^2 - a_4z - a_5 = 0 \tag{4.3}$$

3. The first derivative of the 5th degree polynomial, Equation (4.3), is:

$$f'(z) = 5a_oz^4 - 4a_1z^3 - 3a_2z^2 - 2a_3z - a_4 \tag{4.4}$$

4. The problem was then reduced to finding the proper root of the polynomial for a specified pressure and temperature. The Newton-Raphson (57) iteration method was used to obtain the roots. This equation has the form:

$$z_{n+1} = z_n - \frac{f(z)}{f'(z)} \tag{4.5}$$

5. After the proper root for Equation (4.3) was obtained, the density was computed from the following expression:

$$\rho = \frac{1}{(z + b)} \tag{4.6}$$

The entire procedure was programmed for the IBM 370 computer. The procedure is part of the physical property subroutines developed for Refrigerant-114, water, and carbon dioxide (see Appendix D).

Having established the equations of state for the fluids of interest in the supercritical region, the next step in evaluation of the dimensionless coefficients of the reduced differential equations, expressed by Equations (3.16) through (3.20), was determination of the coefficient of volume expansion.

Coefficient of Volume Expansion, $1/\rho\ (\partial\rho/\partial T)_P$

An expression for the coefficient of volume expansion was obtained as follows:

1. Equation of state, Equation (4.1), is implicit in volume and temperature. Therefore, it was helpful to start with the techniques of implicit differentiation given by the following expression:

$$\left(\frac{\partial V}{\partial T}\right)_P \left(\frac{\partial T}{\partial P}\right)_V \left(\frac{\partial P}{\partial V}\right)_T = -1 \tag{4.7}$$

2. An expression for the coefficient of volume expansion in terms of the derivatives of the equation of state was obtained by rearrangement of Equation (4.7). This equation had the advantage that the dependent variable of the derivatives was pressure. The resulting equation was

$$\frac{1}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_P = \rho\,\frac{(\partial P/\partial T)_V}{(\partial P/\partial V)_T} \tag{4.8}$$

where

$$V = \frac{1}{\rho}$$

and

$$\left(\frac{\partial P}{\partial T}\right)_V = \frac{R}{(V-b)} + \frac{B_2 - (K/T_c)\, C_2\, EXP(-K\, T_r)}{(V-b)^2}$$

$$+ \frac{B_3 - (K/T_c)\, C_3\, EXP(-K\, T_r)}{(V-b)^3}$$

$$+ \frac{B_5 - (K/T_c)\, C_5\, EXP(-K\, T_r)}{(V-b)^5}$$

and

$$\left(\frac{\partial P}{\partial V}\right)_T = -\frac{RT}{(V-b)^2} - \frac{2[A_2 + B_2T + C_2\, EXP(-K\, T_r)]}{(V-b)^3}$$

$$- \frac{3[A_3 + B_3T + C_3\, EXP(-K\, T_r)]}{(V-b)^4}$$

$$- \frac{4A_4}{(V-b)^5} - \frac{5[A_5 + B_5T + C_5\, EXP(-K\, T_r)]}{(V-b)^6}$$

These equations were programmed for the IBM 370 computer. The above equations are part of the physical property subroutines given in Appendix D. The last thermodynamic expression to be developed was the specific heat at constant pressure.

Specific Heat at Constant Pressure

Along with an equation of state for Refrigerant-114, Martin (13) also developed the followin expression for the specific heat at constant volume and zero pressure for Refrigerant-114.

$$C_{V_o} = X_{13} + X_{14}T + X_{15}T^2 \qquad (4.9)$$

The units and constants to be used in Equation (4.9) are tabulated in Table V (Appendix A).

Reynolds and Perkins (58) indicate that the same type of equation as given above would also express the specific heat at constant volume and zero pressure for other supercritical fluids, namely water and carbon dioxide. With this information and the use of the equation of state and the thermodynamic potential functions, it was possible to derive an equation for the specific heat at constant pressure. An examination of the thermodynamic relation

$$C_P - C_V = T \left(\frac{\partial P}{\partial T}\right)_V \left(\frac{\partial V}{\partial T}\right)_P \tag{4.10}$$

shows that expressions for the right-hand side may be obtained from the equation of state. Since an expression for the specific heat at constant volume and zero pressure was known, the problem was then reudced to obtaining an expression for the effects of pressure on the value of the specific heat at constant volume. This was accomplished by the use of thermodynamic potential functions and Maxwell equations. The end result is given by Equation (4.11).

$$C_V = C_{V_o} - T\left(\frac{K}{T_c}\right)^2 EXP(-K\ T_r) \left[\frac{C_2}{(V-b)} + \frac{C_3}{2(V-b)^2} + \frac{C_5}{4(V-b)^4}\right] \tag{4.11}$$

For a detailed derivation of Equation (4.11) see Appendix C. Equation (4.11) was used to calculate the effect of pressure change in the specific heat at constant volume for a given temperature.

The specific heat at constant pressure was then obtained by substituting Equations (4.9) and (4.11) into Equation (4.10), and then solving Equation (4.10) for C_P, the specific heat at constant pressure. The resulting equation for specific heat calculations was

$$C_P = X_{13} + X_{14}T + X_{15}T^2$$

$$- T\left(\frac{K}{T_c}\right) \text{EXP}(-K\ T_r) \left[\frac{C_2}{(V - b)} + \frac{C_3}{2(V - b)^2} + \frac{C_5}{4(V - b)^4}\right]$$

$$+ T\left(\frac{\partial P}{\partial T}\right)_V \left(\frac{\partial V}{\partial T}\right)_P \qquad (4.12)$$

The above equation was employed in the MARQ program to fit the specific heat at constant pressure experimental data of supercritical water (37) (38) (40) (41) (59) and supercritical carbon dioxide (46) through (49). At this stage it was found that simultaneous fit of experimental specific volume and specific heat data would result in a better fit, when compared to the fit with experimental specific heat data alone. This could be primarily due to the errors involved in the experimental thermodynamic property measurements of different investigators. Considering the peculiar variation of specific heat at constant pressure with respect to temperature in the supercritical region (see Figure 6, Appendix B), the fit to the given set of data was considered to be very good. The average absolute errors between the fitted results and the experimental data for both fluids were less than 4 percent. Equation (4.12) was also used by Mullin (50) to calculate specific heat at constant pressure for Refrigerant-114.

The units and constant to be used in Equation (4.12) for all three fluids (Refrigerant-114, water, and carbon dioxide) are tabulated in Table V (Appendix A). Equation (4.12) was programmed for the IBM 370 computer and is part of the physical property subroutines given in Appendix D.

Having established the required expressions for the thermodynamic properties, the next step was development of the necessary equations for

the transport properties. The first transport property considered was the absolute viscosity.

Absolute Viscosity

For Refrigerant-114, Parker and Mullin (5) developed an expression for viscosity at low pressure by the use of a Sutherland-type (60) viscosity equation and low pressure viscosity experimental data of Downing (61):

$$\mu_o = 1.0167 \times 10^{-3} \left[\frac{T^{3/2}}{T + 586.55}\right] \tag{4.13}$$

The viscosity data from the above equation was then corrected for pressure based on the equation developed by Jossi et al. (14):

$$(\mu - \mu_o)X = [23.12 \, EXP(1.079 \, \rho_r) - 25.0] \times 10^{-5} \tag{4.14}$$

where

$$X = \frac{T_c^{1/6}}{M^{1/2} \, P_c^{2/3}}$$

and

$$\rho_r = \frac{\rho}{\rho_c}$$

The units and constants to be used in Equations (4.13) and (4.14) are given in Table VI (Appendix A).

For water and carbon dioxide, due to the availability of good density and viscosity experimental data, it was possible to develop the following simple correlation:

$$\mu = A + BT + C\rho^D \tag{4.15}$$

The above equation was developed based on the correlations reported by Fritsch et al. (3) (39) and Nowak et al. (59) for water. Equation (4.15) was used in the MARQ program to fit the absolute viscosity experimental data of supercritical water (39) (40) (41) and supercritical carbon dioxide (42) (43). The fit to the given set of data was excellent. The average absolute errors between the fitted results and the experimental data for both fluids were less than 1.5 percent.

The units and constants to be used in the viscosity equations for all three fluids (Refrigerant-114, water, and carbon dioxide) are tabulated in Table VI (Appendix A). These equations were programmed for the IBM 370 computer and are part of the physical property subroutines given in Appendix D.

Having established the absolute viscosity expressions for the fluids of interest in the supercritical region, the next step in evaluation of the dimensionless coefficients of the reduced differential equations, expressed by Equations (3.16) through (3.20), was determination of the term $1/\mu\ (\partial\mu/\partial T)_P$.

The Term $1/\mu\ (\partial\mu/\partial T)_P$

An expression for the term $1/\mu\ (\partial\mu/\partial T)_P$ was obtained employing the following equations:

$$\mu = \mu(\rho, T) \tag{4.16}$$

$$\frac{1}{\mu}\left(\frac{\partial\mu}{\partial T}\right)_P = \frac{1}{\mu}\left[\left(\frac{\partial\mu}{\partial\rho}\right)_T \left(\frac{\partial\rho}{\partial T}\right)_P + \left(\frac{\partial\mu}{\partial T}\right)_\rho\right] \tag{4.17}$$

Performing the operations indicated by Equation (4.17) on Equation (4.15) and the equation of state, Equation (4.1), resulted in the following expressions for water and carbon dioxide:

$$\left(\frac{\partial \mu}{\partial \rho}\right)_T = (C)(D)(\rho)^{(D-1)} \tag{4.18}$$

$$\left(\frac{\partial \rho}{\partial T}\right)_P = \rho^2 \frac{(\partial P/\partial T)_V}{(\partial P/\partial V)_T} \tag{4.19}$$

$$\left(\frac{\partial \mu}{\partial T}\right)_\rho = B \tag{4.20}$$

Substitution of Equations (4.15), (4.18), (4.19), and (4.20) into Equation (4.17) resulted in the desired transport property expressions for water and carbon dioxide.

The same procedure was used to evaluate the term $1/\mu\ (\partial\mu/\partial T)_P$ for Refrigerant-114. In this case Equations (4.13) and (4.14), along with the equation of state were used. The following expressions for Refrigerant-114 were obtained:

$$\left(\frac{\partial \mu}{\partial \rho}\right)_T = \frac{1.079}{\rho_c}\left[(\mu - \mu_o) + \frac{2.5 \times 10^{-4}}{X}\right] \tag{4.21}$$

$$\left(\frac{\partial \rho}{\partial T}\right)_P = \rho^2 \frac{(\partial P/\partial T)_V}{(\partial P/\partial V)_T} \tag{4.22}$$

$$\left(\frac{\partial \mu}{\partial T}\right)_\rho = \left(\frac{3}{2}\right)\frac{\mu_o}{T} - \left(\frac{\mu_o}{T + 586.55}\right) \tag{4.23}$$

Substitution of Equations (4.13), (4.14), (4.21), (4.22), and (4.23) in Equation (4.17) resulted in the desired transport property expression for Refrigerant-114.

These equations were programmed for the IBM 370 computer and are part of the physical property subroutines given in Appendix D. The last transport property expression to be developed was the thermal conductivity.

Thermal Conductivity

For Refrigerant-114, Parker and Mullin (5) developed an expression for thermal conductivity at low pressure based on the equation recommended by Downing (61):

$$\frac{MK_o}{\mu_o} = 0.0132\ C_{V_o} + 0.034 - \frac{0.007}{T_r} \qquad (4.24)$$

The thermal conductivity data from the above equation was then corrected for pressure based on the equation developed by Stiel and Thodos (37):

$$(K - K_o)\ \lambda Z_c^5 = 13.10 \times 10^{-8}\ [EXP(0.67\ \rho_r) - 1.069] \qquad (4.25)$$

where

$$\lambda = \frac{M^{1/2}\ T_c^{1/6}}{P_c^{2/3}}$$

and

$$Z_c = \frac{P_c}{RT_c\ \rho_c}$$

The units and constants to be used in Equations (4.24) and (4.25) are given in Table VII (Appendix A).

For water and carbon dioxide, due to the availability of good density and thermal conductivity experimental data, it was possible to develop the following simple correlation:

$$K = A + BT + C\rho^D \qquad (4.26)$$

The above equation was developed based on the correlations reported by Fritsch et al. (3) (39) and Nowak et al. (59) for water. Equation (4.26)

was used in the MARQ program to fit the thermal conductivity experimental data of supercritical water (39) (40) (41) and supercritical carbon dioxide (44) (45). The fit to the given set of data was very good. The average absolute errors between the fitted results and the experimental data for both fluids was less than 3 percent.

The units and constants to be used in the thermal conductivity equations for all three fluids (Refrigerant-114, water, and carbon dioxide) are tabulated in Table VII (Appendix A). These equations were programmed for the IBM 370 computer and are part of the physical property subroutines given in Appendix D.

The last transport property to be determined for the evaluation of the dimensionless coefficients A1, A2, A3, B1, and B2, expressed by Equations (3.16) through (3.20), was the term $1/K\ (\partial K/\partial T)_P$.

The Term $1/K\ (\partial K/\partial T)_P$

An expression for the term $1/K\ (\partial K/\partial T)_P$ was obtained in a manner similar to that used to obtain $1/\mu\ (\partial \mu/\partial T)_P$. The following equations were employed:

$$K = K(\rho, T) \tag{4.27}$$

$$\frac{1}{K}\left(\frac{\partial K}{\partial T}\right)_P = \frac{1}{K}\left[\left(\frac{\partial K}{\partial \rho}\right)_T \left(\frac{\partial \rho}{\partial T}\right)_P + \left(\frac{\partial K}{\partial T}\right)_\rho\right] \tag{4.28}$$

Performing the operations indicated by Equation (4.28) on Equation (4.26) and the equation of state, Equation (4.1), resulted in the following expressions for water and carbon dioxide:

$$\left(\frac{\partial K}{\partial \rho}\right)_T = (C)(D)(\rho)^{(D-1)} \tag{4.29}$$

$$\left(\frac{\partial \rho}{\partial T}\right)_P = \rho^2 \frac{(\partial P/\partial T)_V}{(\partial P/\partial V)_T} \tag{4.30}$$

$$\left(\frac{\partial K}{\partial T}\right)_\rho = B \tag{4.31}$$

Substitution of the Equations (4.26) and (4.29) through (4.31) in Equation (4.28) resulted in the desired transport property expressions for water and carbon dioxide.

The same procedure was used to evaluate the term $1/K\ (\partial K/\partial T)_P$ for Refrigerant-114. In this case Equations (4.24) and (4.25), along with the equation of state were used. The following expressions for Refrigerant-114 were obtained:

$$\left(\frac{\partial K}{\partial \rho}\right)_T = \frac{0.67}{\rho_c}\left[(K - K_o) + \frac{14.003 \times 10^{-8}}{\lambda Z_c^5}\right] \tag{4.32}$$

$$\left(\frac{\partial \rho}{\partial T}\right)_P = \rho^2 \frac{(\partial P/\partial T)_V}{(\partial P/\partial V)_T} \tag{4.33}$$

$$\left(\frac{\partial K}{\partial T}\right)_\rho = 0.0132\ C_{V_o}\left(\frac{\partial \mu_o}{\partial T}\right)_\rho + 0.0132\ \mu_o\left(\frac{\partial C_{V_o}}{\partial T}\right)_\rho + \frac{0.034}{M}\left(\frac{\partial \mu_o}{\partial T}\right)_\rho - \frac{0.007}{M\ T_r}\left(\frac{\partial \mu_o}{\partial T}\right)_\rho + \frac{0.007}{M\ T_r}\left(\frac{\mu_o}{T}\right) \tag{4.34}$$

where

$$\left(\frac{\partial \mu_o}{\partial T}\right) = \frac{3}{2}\left(\frac{\mu_o}{T}\right) - \left(\frac{\mu_o}{T + 586.55}\right)$$

and

$$\left(\frac{\partial C_{V_o}}{\partial T}\right)_\rho = 3.49 \times 10^{-4} - 3.34 \times 10^{-7}\ T$$

Substitution of Equations (4.24), (4.25), and (4.32) through (4.34) into Equation (4.28) resulted in the desired transport property expression for

Refrigerant-114. These equations were programmed for the IBM 370 computer and are part of the physical property subroutines given in Appendix D.

The thermodynamic and transport property expressions developed for water and carbon dioxide in this chapter showed to be in very good agreement with the available physical property experimental data. These equations appear to represent the data within their experimental accuracy. In obtaining these expressions, as the situation permitted, a special effort was made to keep these equations as simple as possible.

The major difficulty encountered in obtaining these expressions was the discontinuity in the experimental data. In most cases the supercritical experimental physical properties reported in the literature did not cover a wide range of temperature and pressure discretely. This was especially true for the absolute viscosity, thermal conductivity, and specific heat data. Due to the peculiar variations of the physical properties of the fluids in the supercritical region (see Figures 2 through 6, Appendix B), this lack of continuity in the data made the fitting process more complicated.

In general, as one would expect, for a wide range of temperatures and pressures, the accuracy of the physical property equations would increase as one moves away from the critical point. The greatest deviation, of course, occurs in the neighborhood of the critical point, where the physical properties change is extremely rapid with temperature.

The physical property equations developed and presented in this chapter for water, carbon dioxide, and Refrigerant-114 are summarized in Table VIII (Appendix A), along with the information relative to the development of these expressions. This table lists source references

for the experimental data on the thermodynamic and transport properties. It also includes an indication as to the number of data points used in the development of the equations, the percent deviation between the fitted results and the experimental data, and the range of application of these equations. Some of this information was not available for Refrigerant-114.

It should be noted that the units used in the equations for the physical properties of Refrigerant-114 are inconsistent in the sense that different units were used to express the same property. This is due to the different choices made by different investigators. Since the coefficients of the reduced differential equations, Equations (3.14) and (3.15), are dimensionless, the equations were developed in the same system of units as the work reported in the literature and the necessary unit changes were made in the computer program. A complete documented listing of the physical property subroutines for Refrigerant-114, water, and carbon dioxide are given in Appendix D. These programs compute the transport and thermodynamic properties for the indicated fluids in the supercritical region. The programs also compute the five dimensionless coefficients A1, A2, A3, B1, and B2, expressed by Equations (3.16) through (3.20), needed for the solutions of the reduced differential Equations (3.14) and (3.15), for the variable property case.

CHAPTER V

NUMERICAL METHOD FOR THE SOLUTION OF THE REDUCED EQUATIONS

It will be the purpose of this chapter to explain how the set of reduced ordinary nonlinear differential equations introduced in Chapter III were solved.

The analytical investigations reported in the literature for the problem of free convection heat transfer to fluids in the supercritical region employ the classical fourth order Runge-Kutta numerical integration technique (2) (3) (5) (6) (7). This method was used by the investigators to solve the set of reduced ordinary nonlinear differential Equations (3.14) and (3.15) for the variable property case, along with their boundary conditions, Equation (3.11). Konanur (2) in his work points out that the accuracy of the Runge-Kutta method of integration depends upon the step size, $\Delta\eta$. The smaller values of $\Delta\eta$ introduce round-off errors and large values of $\Delta\eta$ give rise to truncation errors. Hence, for any problem the step size should be estimated by trial integrations. In general, the primary disadvantages of the Runge-Kutta integration method are (62):

1. The method requires significantly more computer time than other methods of comparable accuracy.
2. Local error estimates are somewhat difficult to obtain.
3. The method is not very accurate.

Solution by the Runge-Kutta method is possible if all the necessary boundary values are known. In the present problem, at the first boundary (η equal to zero) out of five required initial boundary values (F, F', F", θ, and θ') only three of them (F, F', and θ) are known. Hence, the other two unknown boundary values (F" and θ') have to be estimated. Now the solution of the present problem is one of finding the accurate unknown values. Once the initial values are known, one can proceed with the integration process. For the problem under study the necessary five starting boundary conditions are (see Equation (3.11)):

$$F(0) = 0$$

$$F'(0) = 0$$

$$F''(0) = \text{Unknown}$$

$$\theta(0) = 1$$

$$\theta'(0) = \text{Unknown}.$$

The unknown conditions F" and θ' at the boundary are very important because they fix the shear stress and heat transfer at the plate, respectively. Therefore, they must be determined very accurately. In the works reported in the literature, different iterative schemes were employed to overcome this difficulty. Fritsch et al. (3) and Konanur (2) in their work carry out integration of the reduced equations; they first estimate values of F" and θ' at the first boundary (η equal to zero) and then proceed to a large value of η to check the results with the desired boundary conditions. Then, using an iterative technique, the estimated values were then adjusted until suitable agreement between the calculated and the stipulated values were noted. In order for their method to converge at a reasonable amount of computer time and with reasonable accuracy, good estimates of the unknown boundary conditions are

required. In the absence of good estimates the method is very inefficient. Konanur (2) also points out that the success of his model depends upon the case under study and upon how close the guessed values are to required values. Parker and Mullin (5) used the iterative method of Newton-Raphson to predict the unknown boundary conditions. Their method requires linearization of the reduced ordinary nonlinear differential equations and fairly good estimates for the starting boundary conditions. The above mentioned numerical procedures used by different investigators to solve the type of problem under consideration in this study produce fairly reasonable results, but their drawbacks are:

1. The numerical procedures are inefficient.
2. The numerical methods are not very accurate.
3. The numerical programs are unadaptable. These programs were exclusively written for one type of problem, and usage of these programs for similar type problems requires major modifications.

In pursuit of the primary objectives of this study it was necessary to adopt a numerical procedure which is easy to use, efficient, very accurate, and general. The numerical technique developed by Lentini and Pereyra (8) was used for this purpose. This is an adaptive finite difference method for first order nonlinear systems of ordinary differential equations subject to multipoint nonlinear boundary conditions. The method is based on a discretization studied earlier by H. B. Keller (63). In this method variable order is provided through deferred corrections, while a built-in natural asymptotic estimator is used to automatically refine the mesh in order to achieve a required tolerance. This method is easy to use, extremely accurate, and it performs very efficiently in comparison with the results published in the literature for multipoint,

nonlinear boundary value problems (8). Moreover, the solutions of the reduced differential equations based on this method do not depend upon an initial estimate of the unknown boundary conditions. A thorough and complete discussion of this method is given in References (8) and (64).

Evaluation of the Coefficients for the Reduced Differential Equations

At this point it is necessary to explain the method used to evaluate the coefficients of the reduced equations. This is a necessary step before the procedure used to solve the set of reduced ordinary nonlinear equations can be outlined. Since it was desired to express the coefficients of the reduced differential equations as a function of $\theta(\eta)$ (see Chapter III), it was necessary to make a decision as to what method should be used to evaluate the dimensionless coefficients given by Equations (3.16) through (3.20). It appeared that the following two possible methods were available:

1. The most desirable method would be the evaluation of the dimensionless coefficients of the reduced differential equations by an exact method. This method required writing of the numerical method in such a way that for a given η it permitted the integration method to evaluate $\theta(\eta)$ at an η which in turn allowed the evaluation of the temperature. Then, with this information, evaluation of the dimensionless coefficients given by Equations (3.16) through (3.20) was made possible. However, it was found that the computer time required for this procedure was totally impractical. Due to the iterative nature of the problem under study, to perform a few iterations without achieving convergence, the procedure required several hours of computer time.

2. The second possibility would be to first compute the values of the dimensionless coefficients, Equations (3.16) through (3.20), for a given temperature range. Then, using a polynomial, curve fit the data by the method of least squares. It is then possible to obtain an expression for the dimensionless coefficients as a function of $\theta(\eta)$. This method will introduce some error due to the imperfection of the least squares fit; however, it has the advantage of converging rapidly.

In view of the above experience the second method was used. This method was implemented by first expressing the physical properties shown below as follows:

$$\left[\frac{1}{\rho}\left(\frac{\partial\rho}{\partial T}\right)_P + \frac{1}{\mu}\left(\frac{\partial\mu}{\partial T}\right)_P\right] = A_{11} + B_{11}T + C_{11}T^2 + D_{11}T^3 \qquad (5.1)$$

$$\left[\frac{1}{\rho\mu}\right] = A_{22} + B_{22}T + C_{22}T^2 + D_{22}T^3 \qquad (5.2)$$

$$\left[\frac{1}{\rho^2\mu}\right] = A_{33} + B_{33}T + C_{33}T^2 + D_{33}T^3 \qquad (5.3)$$

$$\left[\frac{1}{\rho}\left(\frac{\partial\rho}{\partial T}\right)_P + \frac{1}{K}\left(\frac{\partial K}{\partial T}\right)_P\right] = A_{44} + B_{44}T + C_{44}T^2 + D_{44}T^3 \qquad (5.4)$$

$$\left[\frac{C_P}{\rho K}\right] = A_{55} + B_{55}T + C_{55}T^2 + D_{55}T^3 \qquad (5.5)$$

The coefficients for the above polynomials were determined by the method of least squares. A complete documented listing of the least squares program is given in Appendix D. For Refrigerant-114, water, and carbon dioxide used in this study, a third degree polynomial was the highest degree of polynomial required to result in the best possible fit. As mentioned earlier, the method introduced small errors in the calculations due to the imperfection of the least squares fit. At times in the neighborhood of the pseudocritical point this error for the term $[C_p/\rho K]$,

given by Equation (5.5), became significant. This could be mainly contributed to the peculiar variation of the specific heat for fluids in the supercritical region with respect to temperature in that region (see Figure 6, Appendix B). However, it was possible to minimize the error by using smaller temperature intervals for this case. For illustrative purposes, Figures 7, 8, and 9 (Appendix B) show plots of Equations (5.1) through (5.5). The substitution of the expression

$$T = T_{\infty} + \Delta T\ \theta(\eta) \tag{5.6}$$

into Equations (5.1) through (5.5), and the substitution of Equations (5.1) through (5.5) into Equations (3.16) through (3.20) will give the desired equations for the dimensionless coefficients of the reduced differential Equations (3.14) and (3.15) as functions of $\theta(\eta)$. As a result the following expressions were obtained:

$$A1 = CA + CB\ \theta + CC\ \theta^2 + CD\ \theta^3 \tag{5.7}$$

$$3(A2) = CE + CF\ \theta + CG\ \theta^2 + CH\ \theta^3 \tag{5.8}$$

$$2(A2) = CI + CJ\ \theta + CK\ \theta^2 + CL\ \theta^3 \tag{5.9}$$

$$A3 = (CM + CN\ \theta + CP\ \theta^2 + CR\ \theta^3) - (CS + CT\ \theta + CU\ \theta^2 + CW\ \theta^3) \tag{5.10}$$

$$B1 = CAA + CBB\ \theta + CCC\ \theta^2 + CDD\ \theta^3 \tag{5.11}$$

$$3(B2) = CEE + CFF\ \theta + CGG\ \theta^2 + CHH\ \theta^3 \tag{5.12}$$

The coefficients for Equations (5.7) through (5.12) are given in Appendix C. Equations (5.7) through (5.10) given above represent the dimensionless coefficients of the momentum equation, Equation (3.14), and the Equations (5.11) and (5.12) express the dimensionless coefficients of the energy equation, Equation (3.15). The advantage of the method used is that no derivatives of the polynomials are necessary in order to evaluate

the dimensionless property coefficients of the reduced differential equations.

The substitution of Equations (5.7) through (5.12) into Equations (3.14) and (3.15) permit the reduced differential equations to be expressed as follows:

Momentum Equation

$$F''' = -(CA + CB\,\theta + CC\,\theta^2 + CD\,\theta^3)\,\theta'\,F'' - (CE + CF\,\theta + CG\,\theta^2 + CH\,\theta^3)\,FF'' + (CI + CJ\,\theta + CK\,\theta^2 + CL\,\theta^3)\,(F')^2 - (CM + CN\,\theta + CP\,\theta^2 + CR\,\theta^3) + (CS + CT\,\theta + CU\,\theta^2 + CW\,\theta^3) \tag{5.13}$$

Energy Equation

$$\theta'' = -(CAA + CBB\,\theta + CCC\,\theta^2 + CDD\,\theta^3)\,(\theta')^2 - (CEE + CFF\,\theta + CGG\,\theta^2 + CHH\,\theta^3)\,F\,\theta' \tag{5.14}$$

The next step in the development of the reduced differential equations for the computer program was to form a set of first order differential equations to represent the third order momentum differential equation, Equation (5.13), and the reduced second order energy equation, Equation (5.14). The following procedures were used:

1. Define:

$$G_1 = F \tag{5.15a}$$

$$G_2 = F' \tag{5.15b}$$

$$G_3 = F'' \tag{5.15c}$$

$$G_4 = \theta \tag{5.15d}$$

$$G_5 = \theta' \tag{5.15e}$$

2. The set of differential equations, Equations (5.13) and (5.14), become:

$$Y_1 = \frac{dG_1}{d\eta} = G_2 \tag{5.16a}$$

$$Y_2 = \frac{dG_2}{d\eta} = G_3 \tag{5.16b}$$

$$\begin{aligned} Y_3 = \frac{dG_3}{d\eta} = &-(CA + CB\ G_4 + CC\ G_4^2 + CD\ G_4^3)\ G_5 G_3 \\ &- (CE + CF\ G_4 + CG\ G_4^2 + CH\ G_4^3)\ G_1 G_3 \\ &+ (CI + CJ\ G_4 + CK\ G_4^2 + CL\ G_4^3)\ (G_2)^2 \\ &- (CM + CN\ G_4 + CP\ G_4^2 + CR\ G_4^3) \\ &+ (CS + CT\ G_4 + CU\ G_4^2 + CW\ G_4^3) \end{aligned} \tag{5.16c}$$

$$Y_4 = \frac{dG_4}{d\eta} = G_5 \tag{5.16d}$$

$$\begin{aligned} Y_5 = \frac{dG_5}{d\eta} = &-(CAA + CBB\ G_4 + CCC\ G_4^2 + CDD\ G_4^3)\ (G_5)^2 \\ &- (CEE + CFF\ G_4 + CGG\ G_4^2 + CHH\ G_4^3)\ G_1 G_5 \end{aligned} \tag{5.16e}$$

The boundary conditions, Equation (3.11), transform to

$$\left.\begin{aligned} G_1 &= 0 \\ G_2 &= 0 \\ G_3 &= \text{Unknown} \\ G_4 &= 1 \\ G_5 &= \text{Unknown} \end{aligned}\right\} \eta = 0 \qquad \left.\begin{aligned} G_1 &= \text{Unknown} \\ G_2 &= 0 \\ G_3 &= \text{Unknown} \\ G_4 &= 0 \\ G_5 &= \text{Unknown} \end{aligned}\right\} \eta = \infty \tag{5.17}$$

Equation (5.16) along with the boundary condition, Equation (5.17), are the set of first order nonlinear differential equations of the

reduced momentum and energy differential equations to be solved on the computer.

Procedures for Solving the Differential Equations

The procedures and various programs used in solving the set of reduced ordinary nonlinear differential equations are outlined here. The nondimensionalized heat transfer coefficients were then obtained using the solutions to these equations. The procedures used consisted of five different computer programs. They are defined as follows:

1. The Physical Property Program - Subroutine PROP
2. The Least Squares Program - Subroutine POLY
3. The Coefficient Program - Subroutine COEFF
4. The Lentini-Pereyra Program - Subroutine SYSSOL
5. The Heat Transfer Program - Subroutine HEAT.

A complete listing of these programs with some sample calculations are given in Appendix D. These programs are well documented; therefore, a brief discussion of each program will be given here.

The Physical Property Program (Subroutine PROP)

This program is a FORTRAN representation of the thermodynamic and transport property equations developed in Chapter IV. The purpose of this program was to compute physical properties and their derivatives for the fluids in the supercritical region at a constant pressure, and a given temperature range. The program was written very generally and requires only four pieces of information. They are:

1. Pressure (PSIA).
2. Initial Temperature (°R).

3. Final Temperature (°R).

4. An initial estimate for the specific volume (ft^3/lbm).

The initial value of the specific volume needs only to be an estimated value, since its only purpose was to start the iteration process of Newton-Raphson on the equations of state (see Chapter IV).

For Refrigerant-114, water, and carbon dioxide used in this study three independent Physical Property Programs were developed. A complete documented listing of these programs are given in Appendix D.

The Physical Property Program also computes the following information:

$$\left[\frac{1}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_P + \frac{1}{\mu}\left(\frac{\partial \mu}{\partial T}\right)_P\right] \quad \text{versus} \quad T$$

$$\left[\frac{1}{\rho\mu}\right] \quad \text{versus} \quad T$$

$$\left[\frac{1}{\rho^2\mu}\right] \quad \text{versus} \quad T$$

$$\left[\frac{1}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_P + \frac{1}{K}\left(\frac{\partial K}{\partial T}\right)_P\right] \quad \text{versus} \quad T$$

$$\left[\frac{C_P}{\rho K}\right] \quad \text{versus} \quad T$$

The above data obtained from the Physical Property Program provides the necessary information to compute the coefficients for Equations (5.1) through (5.5) with the help of the Least Squares Program.

The Least Squares Program (Subroutine POLY)

The purpose of the Least Squares Program was to determine the regression coefficients for Equations (5.1) through (5.5), based on the information obtained from the Physical Property Program. The program calculates the regression coefficients and analysis of variance table

for polynomials of successively increasing degrees. The program also automatically terminates when there is no reduction in the residual sum of squares between two successive degrees of the polynomials; and calculates a residual table for the given and fitted data. This termination occurs before the analysis for the highest degree of polynomial specified is completed. The following information is required to start the Least Squares Program:

1. Independent and dependent variables--for the problem under study these were the physical property versus temperature information obtained from the Physical Property Program.

2. Total number of observations.

3. The highest degree of polynomial desired--for the problem under study third degree or lower.

The following coefficients for Equations (5.1) through (5.5) were computed by the Least Squares Program:

$$\begin{bmatrix} A_{11}, B_{11}, C_{11}, D_{11} \\ A_{22}, B_{22}, C_{22}, D_{22} \\ A_{33}, B_{33}, C_{33}, D_{33} \\ A_{44}, B_{44}, C_{44}, D_{44} \\ A_{55}, B_{55}, C_{55}, D_{55} \end{bmatrix}$$

These coefficients provide the necessary information to compute the coefficients of Equations (5.7) through (5.12) given in Appendix C. These coefficients are computed in the Coefficient Program described next.

The Coefficient Program (Subroutine COEFF)

The purpose of this program was to determine the coefficients for

Equations (5.7) through (5.12). These equations represent the property coefficients of the reduced differential equations as functions of $\theta(n)$. These coefficients were evaluated from equations presented in Appendix C for the terms A1, 3(A2), 2(A2), A3, B1, and 3(B2). In order to evaluate the desired coefficients the following information is required:

1. The polynomial coefficients for Equations (5.1) through (5.5), which is obtained from the Least Squares Program.

2. Wall and free stream temperatures, which is obtained from the statement of the problem.

3. The wall viscosity, the wall and free stream density, which is obtained from the Physical Property Program.

The following coefficients for Equations (5.7) through (5.12) were computed by the Coefficient Program.

$$\begin{bmatrix} CA, & CB, & CC, & CD \\ CE, & CF, & CG, & CH \\ CI, & CJ, & CK, & CL \\ CM, & CN, & CP, & CR \\ CS, & CT, & CV, & CW \\ CAA, & CBB, & CCC, & CDD \\ CEE, & CFF, & CGG, & CHH \end{bmatrix}$$

These coefficients provide the necessary information for the solution of the reduced differential equations by the Lentini-Pereyra Program.

The Lentini-Pereyra Program (Subroutine SYSSOL)

The purpose of this program was to solve Equation (5.16) along with the boundary conditions, Equation (5.17). These are the set of first

order nonlinear differential equations of the reduced momentum and energy equations. In order to solve the differential equations by the Lentini-Pereyra Program (Subroutine SYSSOL), the following information is required:

1. The coefficients of Equation (5.16) or Equations (5.7) through (5.12), which is obtained from the Coefficient Program.

2. The set of first order differential equations, Equation (5.16), which is represented by Subroutine FF2.

3. The Jacobian of these first order differential equations, which is represented by Subroutine JACOB2 (for equations, see Appendix C).

In addition to the above mentioned subroutines, the deriver program for the Lentini-Pereyra Program (Subroutine SYSSOL), requires the following information:

1. The boundary conditions, Equation (5.17), a value of zero was assumed for the unknown boundary conditions.

2. Number of points in the initial mesh, counting the end points the value should be greater than 3.

3. User's desired accuracy, this specifies the final absolute error of the estimated values.

This program is very general, accurate, easy to use, efficient, and only requires information which is related to the problem statement. Moreover, a good estimate of the unknown boundary conditions is not essential for rapid convergence of the iterative scheme built in the program. A complete documented listing of the program is given in Appendix D. The solutions of the differential equations obtained by the Lentini-Pereyra Program were designated as

$F(\eta)$ $\theta(\eta)$

$F'(\eta)$ $\theta'(\eta)$

$F''(\eta)$ $\theta''(\eta)$

$F'''(\eta)$

The Lentini-Pereyra Program was also used to obtain the solutions to the constant property momentum and energy Equations (3.12) and (3.13) and their boundary conditions Equation (3.11). Since the coefficients of the differential equations are all constants, the numerical solution for this case was much simpler in comparison to the variable property case. For this case the Lentini-Pereyra Program requires the same type of information as the ones outlined for the variable property case, with the exception of the Coefficient Program. The only parameter upon which the solution of the constant property case depends is Prandtl number, which is obtained through the Physical Property Program. The constant property results were used in the development of the reference temperature method.

For the constant property case, the set of first order differential equations and their Jacobians used in the Subroutine FF1 and Subroutine JACOB1 are presented in Appendix C.

The solutions of the differential equations obtained from the Lentini-Pereyra Program were used to calculate heat transfer coefficients through the Heat Transfer Program.

The Heat Transfer Program (Subroutine HEAT)

This program was used to evaluate the local heat flux, the local heat transfer coefficient, the local Grashof number, the local dimensionless heat transfer coefficient (local Nusselt number), and the wall and

free stream Prandtl numbers. The necessary information to perform these calculations were obtained from the Physical Property Program, the Lentini-Pereyra Program, and the problem statement. The following equations introduced earlier in Chapter III were the necessary equations to perform these calculations:

1. Local Heat Flux

$$\dot{q}''_{x,w} = -K_w (T_w - T_\infty) C_w x^{-1/4} \theta'(0)$$

where

$$C_w = \left[\frac{g\rho_w^2}{4\mu_w^2} \left(\frac{\rho_\infty - \rho_w}{\rho_w}\right)\right]^{1/4}$$

In the above equation, $\theta'(0)$ is the slope of the dimensionless temperature profile at the wall. This information was provided by the Lentini-Pereyra Program from the solutions of the reduced differential equations.

2. Local Heat Transfer Coefficient

$$h_x = \frac{\dot{q}''_{x,w}}{(T_w - T_\infty)}$$

3. Local Wall Grashof Number

$$Gr_{x,w} = \left[\frac{gx^3\rho_w^2}{\mu_w^2} \left(\frac{\rho_\infty - \rho_w}{\rho_w}\right)\right]$$

4. Local Wall Nusselt Number

$$Nu_{x,w} = -\frac{Gr_{x,w}^{1/4}}{\sqrt{2}} \theta'(0)$$

5. Wall and Free Stream Prandtl Numbers

$$Pr_w = \frac{\mu_w C_{P_w}}{K_w}$$

$$Pr_\infty = \frac{\mu_\infty C_{P_\infty}}{K_\infty}$$

All computer programs described in this chapter are located in Appendix D.

Several different problems relating to free convection along a vertical flat plate in the supercritical region for Refrigerant-114, water, and carbon dioxide were solved using the equations and procedures presented in this work. The results will be used in the next chapters to develop the reference temperature method proposed in this study.

CHAPTER VI

HEAT TRANSFER RESULTS

In this chapter the validity and universality of the present variable property heat transfer model is established. The present model is compared with several theoretical and experimental investigations for Refrigerant-114, water, and carbon dioxide, and the results are discussed. Finally, the numerical solutions of the present model for the constant property case are compared with the well-known constant property analysis of Ostrach (10).

In order to complete the analysis of the reference temperature method, it was necessary to establish the validity and universality of the present variable property heat transfer model for fluids in the supercritical region. This was accomplished by comparing the solutions of the present model with the former theoretical and experimental studies. Comparisons were made with the theoretical investigation of Parker and Mullin (5) for Refrigerant-114, experimental investigation of Fritsch and Grosh (12) for water, theoretical investigation of Nowak and Konanur (7) for water, experimental investigations of Nishikawa et al. (30) and Kato et al. (29) for carbon dioxide, and theoretical investigation of Nishikawa and Ito (6) for carbon dioxide.

Parker and Mullin (5) solved four different problems relating to free convection along a vertical flat plate in the supercritical region. These problems were solved for Refrigerant-114 at a pressure of 540 psia and a

temperature difference ($\Delta T = T_w - T_\infty$) of 5°F for four values of the wall temperature, 305, 310, 315, and 320°F. Values of the local heat flux, $\dot{q}''_x$, versus wall temperature, T_w, for the analysis of Parker and Mullin are presented in Figure 10 (Appendix B). This figure also shows the results of comparison between the present analytical model and the analytical investigation of Parker and Mullin (5). As shown in Figure 10, the heat transfer results of this model are in excellent agreement with the theoretical results of Parker and Mullin. In this analysis the same physical property expressions as the ones used by Parker and Mullin for Refrigerant-114 were used. However, the numerical methods for the solutions of the differential equations were completely different (see Chapter V). In view of the above experience, it could be concluded that the excellent agreement between the two models is a good indication as to the validity of the present numerical technique. In comparison to the model of Parker and Mullin, the present model proved to be more general, less complicated, and easier to use.

In conjunction with Figure 10, it is important to note that the maximum value of local heat flux occurs at a wall temperature of 310°F, and a free stream temperature of 305°F. These temperatures are very close to the temperature at which specific heat assumes its maximum, that is, transposed critical temperature (T_M = 306°F). This increase in the rate of heat flux is mainly due to the direct and significant influence of the specific heat maximum on heat transfer calculations for the cases where the wall and free stream temperatures are close to T_M, the transposed critical temperature. This temperature also had a direct and significant effect in the development of the reference temperature method. The reference temperature method will be discussed in the next chapter.

Fritsch and Grosh (12) measured heat transfer data to supercritical water. Their experimental measurements were made in laminar free convection from a vertical flat plate, for water close to its critical point. The point-by-point comparison of the predicted heat transfer fluxes with the experimental data of Fritsch and Grosh is shown in Figure 11. The agreement with the experimental data is considered to be very good.

The lack of complete agreement between the present model and the experimental heat transfer data of Fritsch and Grosh (12) for supercritical water may, in part, be attributable to: (a) the fact that the Fritsch and Grosh experimental heat transfer data were obtained from a surface more closely resembling constant heat flux conditions than one at a uniform temperature. A constant heat flux surface gives higher heat fluxes and heat transfer coefficients than those from a corresponding isothermal heat transfer surface; (b) the fact that the chosen height of the vertical heating wall (1/2 in.) was so small that Rayleigh numbers in their experiments were sufficiently small, but their results could not be said to be an experiment for a vertical wall; and (c) the errors in the physical properties of water in the supercritical region. Figure 12 (Appendix B) compares the predicted values of the Prandtl numbers with the experimental data of Fritsch and Grosh (12). Prandtl number was the only reported experimental physical property data. This figure shows that the predicted Prandtl numbers are higher than the experimental data, but since Prandtl number by definition is the ratio of the product of absolute viscosity and specific heat at constant pressure over thermal conductivity ($Pr = \mu C_p/K$), it is not clear which one of these physical properties is in disagreement with the experimental data.

In reference to the experimental heat transfer data of Fritsch and Grosh (12) there was no indication as to the effect of radiation on their reported heat transfer results for the cases where the difference between the wall and the free stream temperatures were large. In view of lack of such knowledge, calculations were made to include the effect of radiation heat transfer. Based on the calculations made for the experimental data of Fritsch and Grosh, the rate of heat transfer due to radiation proved to be negligible.

The local heat fluxes predicted by the present model are compared to the predictions given by the theoretical model of Nowak and Konanur (7), and the experimental data of Fritsch and Grosh (12) for supercritical water in Figure 13 (Appendix B). It seems from Figure 13 that the present model predicts the experimental data consistently better than the analytical model of Nowak and Konanur. The superiority of the present model becomes more significant for the cases where the difference between the wall and the free stream temperatures become large.

Nishikawa et al. (30) and Kato et al. (29) measured heat transfer data to supercritical carbon dioxide. Their experimental measurements were made in laminar free convection from a vertical flat plate, for carbon dioxide close to its critical point. The comparison between the predicted local heat fluxes and the experimental data is shown in Figure 14 (Appendix B). The predictions of the present model show excellent agreement with the experimental data of Nishikawa et al. (30) for carbon dioxide. However, the present prediction scheme did not predict the experimental data of Kato et al. (29) for carbon dioxide very well. This lack of agreement is due to the fact that in the cases of Kato et al. Rayleigh numbers in their experiments were too large (10^9 ~ 10^{12}) to

regard their results as experiments in the laminar regime. However, the height of the vertical walls in their apparatus was fairly large (20 mm) and their results could be considered to be an experiment for a vertical wall.

The results of comparison between the present analytical model and the theoretical model of Nishikawa and Ito (6) for supercritical carbon dioxide is also shown in Figure 14. This figure shows that the present model predicts the experimental data of Nishikawa et al. (30) far better than the analytical model of Nishikawa and Ito (6). The superiority of the present model becomes more pronounced for the cases where the difference between the wall and the free stream temperatures become large.

In general, the present prediction model appears to be in very good agreement with the available experimental data of fluids in the supercritical region. Aside from the specific points that were mentioned in the discussion of the results, in general the lack of complete agreement between the existing model and the experimental data might be attributable to (a) errors in the thermodynamic and transport properties of fluids in the supercritical region, and to (b) the deficiencies in the existing experimental heat transfer data. Among many other items which must be examined, the technique to measure the temperature of the heating wall were not suitable in some experiments, and pressure vessels in some apparatus were too small to realize the condition of stagnant fluid in infinitely large spaces. Accordingly, in some cases, it would be premature to discuss the validity of the analysis by comparison with the available experimental data. The present model in all cases proved to be better than the existing theoretical models in predicting the experimental heat transfer data for fluids in the supercritical region.

The superiority of the present model was far more pronounced for cases where the difference between the wall and the free stream temperatures become large. In contrast to the theoretical models of Nowak and Konanur (7) for water, and Nishikawa and Ito (6) for carbon dioxide, the present model does not specifically require a knowledge of the experimental values of the physical properties of fluids in the supercritical region and their derivatives for the problem under study. The present model also proved to be more general (i.e., it is independent of the fluid of interest), less complicated, easier to use, and more accurate in comparison to the specific models of Nowak and Kananur (7) for water, Nishikawa and Ito (6) for carbon dioxide, and Parker and Mullin (5) for Refrigerant-114.

Figure 15 compares the theoretical values of local heat fluxes predicted by the present model with the experimental values of supercritical water and carbon dioxide. The correlation between the theoretical and experimental values is reasonable and all of the supercritical free convection heat transfer data are predicted by the theory within an error of $\pm$20 percent. This shows that the present model is reasonably good at predicting heat transfer to fluids in the supercritical region under variable property conditions in laminar free convection on a vertical flat plate.

The utility of the reference temperature lies in the fact that it allows constant property results to be used for variable properties. Therefore, before introducing the development of the reference temperature method (see Chapter VII), it is proper to prove the validity of the present model in predicting the constant property results. This was achieved by comparing the results of the present prediction technique with the well-known constant property analysis of Ostrach (10). The most

important results to be considered in the constant property analysis for the problem under study are the velocity and temperature distributions.

By means of the various transformations made in the analysis as presented in Chapter III, it can easily be verified that for the constant property case

$$F'(\eta) = \frac{ux/\nu_w}{2\sqrt{Gr_{x,w}}} \tag{6.1}$$

and

$$\theta(\eta) = \frac{T - T_\infty}{T_w - T_\infty} \tag{6.2}$$

where

$$\eta = \left(\frac{Gr_{x,w}}{4}\right)^{1/4} \frac{y}{x} \tag{6.3}$$

The dimensionless velocity and temperature distributions as given by Equations (6.1) and (6.2) are presented in Figures 16 and 17 (Appendix B), respectively, as functions of η. In these figures the velocity and temperature profiles predicted by the present model are compared with the analysis of Ostrach (10) for Pr = 10. The computations made with the present model agree with the results of Ostrach up to the fourth significant figure. The excellent agreement between the two analyses proves the validity of the present model in predicting the constant property results.

In reference to the above calculations, it should be noted that in computation of the velocity and temperature profiles, for both constant and variable fluid properties, the value of η equal to 10 was used to satisfy the second boundary conditions at infinity (see Equation (3.11)).

In all the cases considered in this study the established value of infinity never exceeded $\eta = 8$; however, to be on the safe side infinity was defined as $\eta = 10$.

CHAPTER VII

REFERENCE TEMPERATURE METHOD

This chapter will outline the development of a shorthand method by which heat transfer can be evaluated. In the latter part of this chapter the reference temperature results for three fluids will be presented and discussed.

From Chapters III, IV, and V of this study it can be readily seen that solution to the problems relating to variable property conditions is a long, complicated, and tedious process. It is hoped to eliminate most of this tedious work by using an accurate shorthand method for heat transfer calculations, a technique referred to as the reference temperature method.

For engineering applications it has been found convenient to employ the constant property analytic solutions, or the experimental data obtained with small temperature differences, and then to apply some kind of correction to account for property variations. Most of the variable property results indicate that fairly simple corrections will generally suffice over a moderate range of properties. For the problems in which the absolute value of the properties varies markedly through the boundary layer (see Figures 2 through 6, Appendix B), no simple correction scheme is available and thus a complete numerical integration of the applicable differential equations for each such application will be required. The

problem of free convection along a vertical flat plate in the supercritical region under variable property conditions falls into this category.

The most extensively used scheme for correction of the constant property results is the reference temperature method. In this method a characteristic temperature is chosen at which the properties appearing in the nondimensional groups (Nu, Gr, Pr, etc.) may be evaluated. The constant property results at that temperature may be used to evaluate variable property behavior. Typically this temperature may be the surface temperature, the free stream temperature, or a temperature partway between the surface and the free stream; there is no general rule.

On the basis of this conception, reference temperature can be expressed as follows:

$$T_{ref} = T_w - \bar{C} (T_w - T_\infty) \tag{7.1}$$

In the analytical investigations reported in the literature for reference temperature, the parameter $\bar{C}$ of Equation (7.1) has been assumed to have a constant value. This assumption is valid only when the thermodynamic and transport properties of the problem do not vary markedly through the boundary layer, that is, when variations in the physical properties at constant pressure can be assumed linear for small change in temperature. In this case determination of the "reference temperature constant," parameter $\bar{C}$, is a fairly simple matter (see Chapter II).

For the present study the above assumption is not valid. Since the absolute value of the thermodynamic and transport properties varies peculiarly through the boundary layer (see Figures 2 through 6, Appendix B). The parameter $\bar{C}$ is no longer a constant and may be a complicated function of a dimensionless temperature group expressed as follows:

$$x = f(T_M - T_\infty / T_w - T_\infty) \tag{7.2}$$

T_M is the pseudocritical or transposed critical temperature, normally defined as the temperature where the specific heat at constant pressure attains a maximum for a given supercritical pressure (see Figure 3, Appendix B). Then, the reference temperature expression given by Equation (7.1) becomes

$$T_x = T_w - x(T_w - T_\infty) \tag{7.3}$$

An expression similar to Equation (7.2) was suggested by Eckert (65) in an attempt to correlate the ear critical, forced convection heat transfer data obtained in heated-tube experiments.

Prior to the present study there appeared to be no analysis available to predict the reference temperature for the problem of free convection along a vertical flat plate in the supercritical region under variable property conditions. It was the primary objective of this study to make this prediction possible. To achieve this goal five different steps had to be accomplished. In order to aid better understanding of the steps taken in the development of the reference temperature method, Refrigerant-114 at a single supercritical pressure of 540 psia was chosen for illustrative purposes. In the latter part of this chapter the results of the following ste were applied to several fluids at different supercritical pressures in order to prove the validity and universality of the resent reference temperature method. The steps taken in the development of the reference temperature method were:

1. Calculation of heat transfer coefficients. With the use of the equations, procedures, and computer programs introduced and developed in Chapters III through V, the local dimensionless heat transfer coefficients

(local Nusselt number) were calculated for several cases. For illustrative purposes Refrigerant-114 was chosen as the working fluid and heat transfer data were obtained for 123 different cases, covering a wide range of temperature differences between the wall and the free stream. The range of temperatures covered in these calculations were from 300°F to 350°F at a constant pressure of 540 psia.

2. Development of a correlation for the calculated dimensionless heat transfer data, using data obtained from step 1. The conventional constant property correlation of laminar free convective heat transfer such as

$$Nu = c(Gr\ Pr)^n = c(Ra)^n \tag{7.4}$$

where Rayleigh number Ra is taken as the independent variable and Nusselt number Nu as the dependent variable fails to correlate the heat transfer data in the supercritical region. This is mainly due to the strong temperature dependency of the physical properties.

The following basic form was employed to correlate the heat transfer data for the variable property conditions:

$$Nu_{x,w} = a(Gr_{x,w})^b\ (Pr_\infty)^c\ \left(\frac{T_\infty}{T_w - T_\infty}\right)^d \tag{7.5}$$

where the constants a, b, , and d of Equation (7.5) were obtained from the MARQ program. For the 123 physical cases considered for Refrigerant-114 at 540 psia pressure, the following values were obtained for the contants a, b, c, and d of Equation (7.5):

a = 0.40248195

b = 0.25

$c = 0.31833951$

$d = 0.00169742.$

The constants given above fit the 123 data points with an average absolute error of less than 1 percent.

3. Development of a correlation for constant property results. The utility of the reference temperature method lies in the fact that it allows use of constant property results to compute variable property values. Therefore, it was essential to include the constant property analysis in determination of the reference temperature. The slope of the temperature profile at the wall $\theta'(0)$ used in these analyses was obtained from the solution of the following constant property momentum and energy equations:

$$F''' + 3FF'' - 2(F')^2 + \theta = 0 \tag{7.6}$$

$$\theta'' + 3PrF\theta' = 0 \tag{7.7}$$

In Equations (7.6) and (7.7), Prandtl number is the single parameter of the equations. For the pressures and temperatures in this study, Prandtl number covered a range from 0 to 40. For this range of Prandtl numbers the following correlation was obtained for $\theta'(0)$ as a function of Prandtl number:

$$[-\theta'(0)]_{c.p.} = \quad + b_1(Pr)^{c_1} \tag{7.8}$$

where

$a_1 = 0.10;$

$b_1 = 0.48615754;$ and

$c_1 = 0.33141328.$

These constants were obtained using the MARQ program. Equation (7.8)

with the constant given above predicts the slope of the temperature profile at the wall for the constant property cases within an average absolute error of 1.3 percent.

Based on the equations presented in Chapter III, the slope of the temperature profile at wall, $\theta'(0)$, can be expressed in terms of the dimensionless groups Nusselt number and Grashof number as follows:

$$[-\theta'(0)]_{c.p.} = \frac{Nu_{x,w}}{(Gr_{x,w}/4)^{1/4}} \tag{7.9}$$

4. Development of an optimization procedure to predict reference temperature. For this purpose Equation (7.5) was represented as

$$(Nu_{x,w})_{T_x} = a(Gr_{x,w})^{b}_{T_x} (Pr)^{c}_{T_x} \left(\frac{T_\infty}{T_w - T_\infty}\right)^{d} \tag{7.10}$$

where the properties in the Nusselt number, Grashof number, and Prandtl number were evaluated at a characteristic temperature T_x, which is a complicated function of wall, free stream, and transposed critical temperatures. The value of T_x was then optimized between the free stream and wall temperatures until the best agreement between the correlated values, $(Nu_{x,w})_{T_x}$ and $[Nu_{x,w}/(Gr_{x,w}/4)^{1/4}]_{T_x}$, obtained from Equation (7.10) at temperature T_x, and the calculated values of $(Nu_{x,w})$ and $[-\theta'(0)_{c.p.} = Nu_{x,w}/(Gr_{x,w}/4)^{1/4}]$ were reached. This optimum temperature is the reference temperature for that case. In this comparison the calculated Nusselt numbers $(Nu_{x,w})$ were obtained from the variable property analysis for the cases considered (i.e., step 1), and the values of $\theta'(0)_{c.p.}$ were calculated from Equation (7.8).

5. Development of a general plot for reference temperature. The reference temperature for fluids in the supercritical region is a

complicated function of wall, free stream, and transposed critical temperatures. Therefore, it was not possible to express this temperature with a single equation that would represent the fluids in the supercritical region under the physical conditions considered. The next best possible approach in presentation of reference temperature was to express the variation of reference temperature T_x with wall temperature and ratio of wall to free stream temperature. The plot of $x = (T_w - T_x)/(T_w - T_\infty)$ versus T_w/T_∞ for various wall temperatures is shown in Figure 18 (Appendix B). This plot was generated for Refrigerant-114 at a constant pressure of 540 psia, and three different wall temperatures.

In an attempt to obtain a presentation for reference temperature which is more easily interpretable, the parameter x (expressing the ratio of the wall temperature T_w minus reference temperature T_x to wall temperature minus free stream temperature T_∞) was plotted over the parameter $(T_M - T_\infty)/(T_w - T_\infty)$ in Figure 19 (Appendix B). In this parameter, T_M is that temperature at which specific heat assumes its maximum value. It may be observed that in this presentation the influence of wall temperature T_w has been eliminated, since all the curves for different T_w values align quite well along a single curve.

Steps 1 through 5 presented above complete the analysis and development of the reference temperature method. The results obtained based on these steps were then applied to several fluids for wide ranges of supercritical pressures and temperatures. These results will be presented and discussed next.

As presented in Figure 19, the influence of wall temperature can be eliminated if $x = (T_w - T_x)/(T_w - T_\infty)$ is represented versus $(T_M - T_\infty)/(T_w - T_\infty)$ rather than T_w/T_∞ (see Figure 18). The next step in the

development of the reference temperature analysis was to generalize the single curve of Figure 19 to cover a wide range of supercritical temperatures and pressures for several fluids. For this purpose the three fluids, Refrigerant-114, water, and carbon dioxide, were chosen. The ranges of supercritical temperatures and pressures used to develop the reference temperature plots for the three fluids were:

1. Refrigerant-114
    a. Pressure range, P = 500, 540, 600 psia
    b. Temperature range, T = 300 to 350°F
2. Water
    a. Pressure range, P = 3400, 3600, 3800 psia
    b. Temperature range, T = 715 to 750°F
3. Carbon Dioxide
    a. Pressure range, P = 1200, 1400, 1600 psia
    b. Temperature range, T = 90 to 140°F.

The ranges of pressures and temperatures chosen for the above fluids cover the experimental pressure and temperature ranges reported in the literature (see Table I, Appendix A).

Considering the five major steps outlined for the development of the reference temperature, the first step in generalization of Figure 19 was to obtain the dimensionless heat transfer coefficients. For each one of the pressures assumed for the three fluids, as given above, ten different wall temperatures were chosen. These temperatures were selected in such a way that they would cover the wide range of temperatures encountered for each fluid, as given above. For each case the wall temperature was kept constant and the free stream temperature was varied to within one degree of the specified wall temperature. The

dimensionless heat transfer coefficients for these problems relating to the supercritical region were then calculated based on the equations, procedures, and computer programs introduced in Chapters III, IV, and V. The variable property heat transfer results for each fluid for the range of temperatures and pressures employed were then correlated with the same type of correlation given by Equation (7.5). The constants a, b, c, and d of Equation (7.5) obtained for the three fluids are tabulated in Table IX (Appendix A). This table also lists the percent deviation between the correlated and calculated results, and the range of pressures and temperatures used for these correlations. The constants of Equation (7.5), tabulated in Table IX were obtained from the MARQ program.

The reference temperature results for Refrigerant-114, water, and carbon dioxide are presented in Figures 20, 21, and 22 (Appendix B), respectively. These figures cover a wide range of supercritical temperatures and pressures for each fluid. Figures 20, 21, and 22 represent the variation of $x = (T_w - T_x)/(T_w - T_\infty)$ with $(T_M - T_\infty)/(T_w - T_\infty)$ for various wall and free stream temperatures at three specific pressures for each fluid. These figures were produced following the five-step procedure outlined earlier for the development of reference temperature.

Before discussing the results of Figures 20, 21, and 22, it would be important to first concentrate on an isolated case, and then consider the more generalized results. For this purpose the reference temperature results of Refrigerant-114 at a constant pressure of 540 psia were chosen (see Figure 20). These results are represented in Figure 23 (Appendix B) in terms of variation of $x = (T_w - T_x)/(T_w - T_\infty)$ with $(T_M - T_\infty)/(T_w - T_\infty)$ at various wall temperatures for a constant pressure.

As shown in Figure 23 the data points align quite well along a single curve.

There are several interesting points to be noted about the single curve of Figure 23. The curve of $x = (T_w - T_x)/(T_w - T_\infty)$ behaves very nicely and smoothly up to certain values of $(T_M - T_\infty)/(T_w - T_\infty)$ and then there is a sudden abrupt change in the direction and behavior of the curve. This abrupt change in the direction of the curve is primarily due to the large influence of T_M, the temperature at which specific heat assumes its maximum, on the reference temperature. This change occurs for the values of $(T_M - T_\infty)/(T_w - T_\infty)$ in the range of -0.25 to 2.00. These values of $(T_M - T_\infty)/(T_w - T_\infty)$ occur where the wall temperature T_w or the free stream temperature T_∞ are close to T_M (T_M = 306°F for Refrigerant-114 at a pressure of 540 psia), that is, when T_w or T_∞ are within about five degrees F of T_M on both sides of the specific heat peak. Figures 24 and 25 (Appendix B) show the variations in the thermodynamic and transport properties of Refrigerant-114 with respect to temperature at a constant pressure of 540 psia. From these figures it can be seen that around the temperature of T_M = 306°F the physical properties that go through an abrupt change are the specific heat and Prandtl number. This abrupt change in the specific heat has a direct and strong influence on the reference temperature.

For the cases where both the wall temperature and free stream temperature are away from the T_M (for the results of Figure 23 this would be on the right side of the specific heat maximum) the direct and strong influence of T_M on the reference temperature vanishes and the variation of $x = (T_w - T_x)/(T_w - T_\infty)$ with $(T_M - T_\infty)/(T_w - T_\infty)$ can be represented by a simple curve. For the results presented in Figure 23, this simple

curve covers the values of $(T_M - T_\infty)/(T_w - T_\infty)$ in the range of -43.0 to -0.25.

As shown in Figures 20, 21, and 22 the variation of $x = (T_w - T_x)/(T_w - T_\infty)$ versus $(T_M - T_\infty)/(T_w - T_\infty)$ for Refrigerant-114, water, and carbon dioxide at different supercritical pressures and wide range of temperatures show results similar to the ones presented in Figure 23 for Refrigerant-114 at a constant pressure of 540 psia. Therefore, the very same observations and conclusions made about the results of Figure 23, which were for a specific fluid at a single pressure, can be employed for the results of Figures 20, 21, and 22, which are for several fluids at several different pressures and temperatures.

Figures 20, 21, and 22 indicate that it is possible to obtain a single curve for different pressures up to a point where the direct influence of T_M on the reference temperature is not significant, that is, for the cases where the wall and free stream temperatures were more than five degrees to the right of the specific heat peak. For the cases where T_w or T_∞ are in the vicinity of T_M, a single curve was not obtainable. This was primarily due to the fact that for each pressure the specific heat assumes a different maximum value, since in these cases T_M has a direct and significant influence on the prediction of the reference temperature.

However, it is interesting to note that in Figures 20, 21, and 22, for the cases where T_w or T_∞ were close to T_M, the reference temperature curves for all three pressures considered for each fluid branched off from the same line. The amount of departure of these curves from the single curve increases as the pressure is increased, since the higher the pressure the further is the maximum peak of specific heat removed

from the critical point. For the cases where the free stream temperature was on the left of the specific heat peak, that is where the ratio $(T_M - T_\infty)/(T_w - T_\infty)$ was positive; as soon as the direct influence of the specific heat maximum on the reference temperature was diminished, the reference temperature curves behaved smoothly again. This phenomenon can best be observed by considering those reference temperature curves presented in Figures 20, 21, and 22, which were obtained for the highest pressures utilized.

The solid lines drawn through the data points given in Figures 20, 21, and 22 present the best fit obtained for the reference temperature data generated for the fluids under study. Plots similar to the one given in Figure 23 for Refrigerant-114 at a constant pressure of 540 psia were generated for the three fluids at every one of the pressures considered. Then, the best fit through the data points in these plots was demonstrated by a solid curve which expresses the variation of $x = (T_w - T_x)/(T_w - T_\infty)$ versus $(T_M - T_\infty)/(T_w - T_\infty)$. These results, as mentioned earlier, are presented in Figures 20, 21, and 22.

In order to complete the analysis of the reference temperature method, it was necessary to establish the validity and universality of this method. This was accomplished by comparing the constant property heat transfer results calculated with the predicted reference temperatures obtained from Figures 20, 21, and 22 with the experimental and theoretical variable property heat transfer results presented in Chapter VI for the three fluids under study. In addition, some of the variable property heat transfer results obtained with the use of the equations, procedures, and computer programs introduced in Chapters III, IV,

and V were compared with the constant property heat transfer results obtained with the predicted reference temperatures.

In order to calculate the constant property heat transfer results, for the free convective heat transfer problems considered in these comparisons, the dimensionless reference temperatures $x = (T_w - T_x)/(T_w - T_\infty)$, were obtained from Figures 20, 21, or 22 for different values of $(T_M - T_\infty)/(T_w - T_\infty)$ for the fluids of interest. The reference temperatures were then calculated using the following equation:

$$T_{ref} = T_x = T_w - x(T_w - T_\infty) \tag{7.11}$$

These reference temperatures were used to evaluate the value of Prandtl number in the constant property equations. The following constant property equations for momentum and energy were used:

$$F''' = -3FF'' + 2(F')^2 - \theta \tag{7.12}$$

$$\theta'' = -3Pr_{ref}F\theta' \tag{7.13}$$

where Pr_{ref} is Prandtl number evaluated at a reference temperature.

The boundary conditions for Equations (7.12) and (7.13) are:

$$\left.\begin{aligned} F &= 0 \\ F' &= 0 \\ \theta &= 1 \end{aligned}\right\} \quad \eta = 0 \qquad\qquad \left.\begin{aligned} F' &= 0 \\ \theta &= 0 \end{aligned}\right\} \quad \eta = \infty \tag{7.14}$$

Equations (7.12) and (7.13) along with the boundary conditions given by Equation (7.14) were solved on the computer using the Lentini-Pereyra Program (see Chapter V). The slope of the dimensionless temperature profile at the wall, $\theta'(0)$ obtained from the solution of the conservation equations was used in the following equations to obtain the local heat flux and the local dimensionless heat transfer coefficient

(local Nusselt number) for the constant property problem:

$$(\dot{q}''_{x,w})_{c.p.} = -K_{ref}(T_w - T_\infty)\, C_{ref} x^{-1/4}\, [\theta'(0)]_{ref} \qquad (7.15)$$

where

$$C_{ref} = \left[\frac{g\,\rho_{ref}^2}{4\,\mu_{ref}^2}\left(\frac{\rho_\infty - \rho_w}{\rho_w}\right)\right]^{1/4}$$

and

$$(Nu_{x,w})_{c.p.} = -\frac{Gr_{x,ref}^{1/4}}{\sqrt{2}}\,[\theta'(0)]_{ref} \qquad (7.16)$$

where

$$Gr_{x,ref} = \left[\frac{gx^3\,\rho_{ref}^2}{\mu_{ref}^2}\left(\frac{\rho_\infty - \rho_w}{\rho_w}\right)\right]$$

In Equations (7.15) and (7.16) the subscript (c.p.) designates the constant property results.

The constant property heat flux results obtained from Equation (7.15) were compared with the theoretical and experimental results presented in Chapter V. This comparison is shown in Figure 26 (Appendix B). From this figure it can be seen that the constant property heat flux results predicted by the reference temperature method are within an error of $\pm$20 percent of the variable property experimental and theoretical results.

In addition to the above comparison, the variable property heat transfer results of Refrigerant-114 at 540 psia, based on the equations, procedures, and computer programs introduced in Chapters III, IV, and V were compared with the constant property heat transfer results obtained

with the predicted reference temperatures. This was achieved by choosing 37 points on the 540 psia curve of Figure 20 for Refrigerant-114 that would best reproduce the curve. Most of the data points were chosen for the wall and free stream temperatures that best represented the abrupt change in the curve. The heat transfer results obtained for these data points cover a wide temperature range of 300 to 350°F. The comparison between the constant property Nusselt numbers obtained from Equation (7.16) and the variable property Nusselt numbers obtained from Equation (3.26) are presented in Figure 27 (Appendix B). From this figure it can be seen that the constant property Nusselt numbers predicted by the reference temperature method are within an error of $\pm$10 percent of the variable property results.

In general, based on the results presented in this chapter for the reference temperature method, it can be concluded that one single curve for prediction of reference temperature in certain regions can be established. That is, the region where the influence of T_M, the temperature at which specific heat assumes its maximum value, is not directly significant on the reference temperature. The heat transfer results obtained using the predicted reference temperatures in conjunction with the constant property equations when compared wiht the experimental and theoretical variable property heat transfer results of Refrigerant-114, water, and carbon dioxide, produced very good results. For the regions where the influence of T_M on the reference temperature was significant, it was still possible to predict reference temperature and in turn heat transfer coefficients with good accuracy, but the curves used for these predictions were pressure dependent.

As pointed out earlier the utility of the reference temperature lies in the fact that it allows constant property results to be used to compute variable property results. Once the reference temperature for a particular problem under study is specified, it is an easy matter to obtain the variable property heat transfer results. This can be accomplished by obtaining the local heat flux or local Nusselt number from Equations (7.15) or (7.16), respectively. The requirements for evaluation of these equations are the physical property information, and the constant property solutions, that is, the slope of the temperature profile at the wall as a function of the Prandtl number evaluated at the specified reference temperature. The physical property information can be obtained from the equations and computer programs developed in Chapters IV and V, respectively. The solution to the constant property problem can be acquired from the solution of the conservation of momentum and energy equations, Equations (7.12) and (7.13), along with their boundary conditions, Equation (7.14), with the use of the Lentini-Pereyra Program (see Chapter V). To simplify the reference temperature method one step further, the solution to the constant property equations for the range of Prandtl numbers considered in this study are presented in Figure 28 (Appendix B). These constant property results were also correlated as a function of Prandtl number according to the following equation:

$$-\theta'(0) = a_1 + b_1 (Pr)^{c_1} \tag{7.17}$$

where

$a_1 = 0.10$;

$b_1 = 0.48615754$; and

$c_1 = 0.33141328$.

The constants a_1, b_1, and c_1 of Equation (7.17) were obtained using the MARQ program. Equation (7.17) with the constants given above predicts the solution to the constant property problems within an average absolute error of 1.3 percent. The above equation covers a range of Prandtl number from 0 to 40.

It should also be pointed out that for a specified problem the free convective heat transfer coefficients to fluids in the supercritical region with the exception of liquid metals, can be directly calculated from the following equation:

$$Nu_{x,w} = \frac{h_x\, x}{K_w} = a(Gr_{x,w})^b\ (Pr_\infty)^c\ \left(\frac{T_\infty}{T_w - T_\infty}\right)^d \qquad (7.18)$$

The constants a, b, c, and d of Equation (7.18) obtained for Refrigerant-114, water, and carbon dioxide along with its range of application are tabulated in Table IX (Appendix A). Equation (7.18) requires information on physical properties, which can be obtained from the equations and computer programs developed in Chapters IV and V, respectively.

CHAPTER VIII

SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

Summary and Conclusions

The accomplishments of this investigation may be summarized as follows:

1. The fundamental partial differential equations of conservation of mass, momentum, and energy for laminar free convection in a boundary layer along a vertical flat plate, Equations (3.1) through (3.3), were reduced to a set of two ordinary differential equations, Equations (3.14) and (3.15), by means of the compressible stream function and a similarity variable, Equations (3.5) and (3.6). In the development of these reduced differential equations no simplifying assumptions were made regarding the physical properties. The property coefficients, Equations (3.16) through (3.20), were expressed as functions of $\theta(\eta)$, once expressions for the physical properties ρ, C_p, μ, and K as a function of temperature at a particular pressure are known.

2. Expressions for thermodynamic and transport properties in the supercritical region as functions of pressure and temperature were developed for Refrigerant-114, water, and carbon dioxide (see Chapter IV). Development of these expressions was necessary in order to solve the reduced ordinary differential momentum and energy equations, Equations (3.14) and (3.15), for the variable property case.

3. The numerical solution of the reduced differential momentum and energy equations is complicated by the fact that two of the five wall boundary conditions are unknown. These two unknown boundary conditions are $F''(0)$ and $\theta'(0)$ and their values must be obtained in order to evaluate the heat transfer and shearing stress. The iterative finite difference numerical technique of Lentini and Pereyra (8) which was described in Chapter V was demonstrated to be a very effective and practical numerical method for overcoming this difficulty. This numerical procedure is easy to use, efficient, accurate, and general.

4. The validity and universality of the present variable property free convective heat transfer model has been demonstrated in Chapter VI for three fluids at supercritical conditions. The present prediction scheme showed very good agreement with the theoretical heat transfer results of Parker and Mullin (5) for Refrigerant-114, and experimental heat transfer results of Fritsch and Grosh (12) and Nishikawa et al. (30) for water and carbon dioxide, respectively (see Figures 10, 11, 14, and 15).

5. The present model also proved to be more general (i.e., it is independent of the fluid of interest), less complicated, easier to use, and more accurate in comparison to the specific models of Nowak and Konanur (7) for water, Nishikawa et al. (6) for carbon dioxide, and Parker and Mullin (5) for Refrigerant-114 (see Figures 10, 13, and 14).

6. The validity of the present model in predicting the constant property results was also presented in Chapter VI. The prediction scheme showed excellent agreement with the well-known constant property analysis of Ostrach (10) (see Figures 16 and 17).

7. In Chapter VII a shorthand method was developed by which heat transfer to fluids in the supercritical region under variable property conditions in laminar free convection on a vertical flat plate can be evaluated. This technique is referred to as the reference temperature method. Based on this method three generalized plots for Refrigerant-114, water, and carbon dioxide were developed. These plots cover a wide range of supercritical temperatures and pressures (see Figures 20, 21, and 22). Based on the results of these plots, it can be concluded that one single curve for prediction of reference temperature in certain regions can be established. That is, the region where the influence of T_M, the temperature at which specific heat assumes its maximum value, is not directly significant on the reference temperature. For the regions where the influence of T_M on the reference temperature was significant it was still possible to predict reference temperature, but the curves used for these predictions were pressure dependent.

8. The utility of the reference temperature lies in the fact that it allows constant property results to be used to compute variable property results. Once the reference temperature for a particular problem under study is specified, it is an easy matter to obtain the variable property heat transfer results. This can be accomplished by obtaining the local heat flux or local Nusselt number from Equations (7.15) or (7.16), respectively. The requirements for evaluation of these equations are the physical property information, and the constant property solutions, that is, the slope of the temperature profile at the wall as a function of the Prandtl number evaluated at the specified reference temperature.

9. The validity and universality of the reference temperature method has been demonstrated in Chapter VII for the three fluids. The heat transfer results obtained with the reference temperature scheme showed very good agreement with the theoretical heat transfer results of Parker and Mullin (5) for Refrigerant-114, and experimental heat transfer results of Fritsch and Grosh (12) and Nishikawa et al. (30) for water and carbon dioxide, respectively (see Figure 26).

10. The validity of the reference temperature method was also checked against the theoretical heat transfer results of Refrigerant-114 at 540 psia obtained from the variable property analysis and Lentini-Pereyra Program (see Chapters III, IV, and V). These heat transfer data covered a wide range of temperatures which were not encountered with the available experimental data. The comparison between the theoretical variable property results and the constant property results obtained in conjunction with the predicted reference temperatures showed excellent agreement (see Figure 27).

11. In general, based on the results presented in Chapter VII for the reference temperature method, it can be concluded that the heat transfer results obtained using the predicted reference temperature in conjunction with the constant property equations when compared with the experimental and theoretical variable property heat transfer results of Refrigerant-114, water, and carbon dioxide produced very good results.

12. For the three fluids encountered in this study a correlation was also developed which would predict heat transfer once the physical properties for the problem under study are known (see Equation (7.18)). This correlation is applicable to Refrigerant-114, water, and carbon dioxide in the range of temperatures and pressures considered (see Table IX).

Recommendations

Based on the observations made during this study, the following recommendations are made:

1. The variable property prediction scheme should be investigated with more heat transfer experimental results which cover wide ranges of supercritical pressures and temperatures. Such experimental results are not available.

2. Accurate data on density, specific heat, absolute viscosity, and thermal conductivity in the critical and supercritical regions are needed for different fluids.

3. Since the physical properties in the supercritical region vary peculiarly with respect to temperature at a constant pressure, it is essential that the experimental investigations for the physical properties should be carried out at small temperature increments to insure accurate determination of these variations.

BIBLIOGRAPHY

(1) Hendrick, R. C., R. J. Simoneau, and R. V. Smith. "Survey of Heat Transfer to Near-Critical Fluids." NASA Technical Note D-5886 (1970).

(2) Konanur, A. K. "An Analytical Investigation of Free Convection Heat Transfer to a Supercritical Fluid." (Unpublished M.E.Sc. thesis, The University of Western Ontario, 1967.)

(3) Fritsch, C. A., and R. J. Grosh. "Free Convective Heat Transfer to a Supercritical Fluid." Proceedings of the 1961 International Heat Transfer Conference, ASME, Part V (1961), pp. 1010-1016.

(4) Hasegawa, S., and K. Yoshioka. "An Analysis for Free Convective Heat Transfer to Supercritical Fluids." Proceedings of the Third International Heat Transfer Conference, ASME, Vol. 2 (1966), pp. 214-222.

(5) Parker, J. D., and T. E. Mullin. "Natural Convection in the Supercritical Region." Presented at the Institution of Mechanical Engineers Symposium on Heat Transfer and Fluid Dynamics of Near Critical Fluids, Paper 1, March 27-29, Bristol, England, 1968.

(6) Nishikawa, K., and T. Ito. "An Analysis of Free Convective Heat Transfer from an Isothermal Vertical Plate to Supercritical Fluids." J. Heat Mass Transfer, Vol. 12 (1969), pp. 1449-1463.

(7) Nowak, E. S., and A. K. Konanur. "An Analytical Investigation of Free Convection Heat Transfer to Supercritical Water." ASME, Paper No. 70-WA/HT-6 (March, 1970).

(8) Lentini, M., and V. Pereyra. "A Variable Order Finite Difference Method for Nonlinear Multipoint Boundary Value Problems." Mathematics of Computation, Vol. 28, No. 128 (October, 1974), pp. 981-1003.

(9) Pohlhausen, E. "The Heat Exchange Between Solid Bodies and Fluids With Small Friction and Small Conduction of Heat." ZAMM, Vol. 1 (1921), p. 115.

(10) Ostrach, S. "An Analysis of Laminar Free Convection Flow and Heat Transfer About a Flat Plate Parallel to the Direction of the Generating Body Force." NASA Technical Note 2635 (February, 1952).

(11) Sparrow, E. M., and J. L. Gregg. "The Variable Fluid Property Problem in Free Convection." Trans. ASME, Vol. 80 (May, 1958), pp. 879-886.

(12) Fritsch, C. A., and R. J. Grosh. "Free Convection Heat Transfer to Supercritical Water Experimental Measurements." J. Heat Transfer, Trans. ASME, Vol. 85, No. 4 (November, 1963), pp. 289-294.

(13) Martin, J. J. "Thermodynamic Properties of Dichlortetra-fluoro-ethane." J. Chem. Engng. Data, Vol. 5, No. 3 (July, 1960), pp. 334-336.

(14) Jossi, J. A., L. I. Stiel, and G. Thodus. "The Viscosity of Pure Substances in the Dense Gaseous and Liquid Phase." AICHE J., Vol. 8, No. 1 (March, 1962), pp. 11-15.

(15) Stiel, L. I., and G. Thodus. "The Thermal Conductivity of Non-polar Substances in the Dense Gaseous and Liquid Region." AICHE J., Vol. 10, No. 1 (January, 1964), pp. 26-64.

(16) Doughty, D. L., and R. M. Drake. "Free Convection Heat Transfer From a Horizontal Right Circular Cylinder to Freon 12 Near the Critical State." Trans. ASME, Vol. 78, No. 8 (November, 1956), pp. 1843-1850.

(17) Bonilla, C. F., and L. A. Sigel. "High Intensity Natural Convection Heat Transfer Near the Critical Point." Chem. Eng. Prog. Ser., Vol. 57, No. 32 (1961), pp. 87-95.

(18) Griffith, J. D., and R. H. Sabersky. "Convection in a Fluid at Supercritical Pressures." ARS J., Vol. 30, No. 3 (March, 1960), pp. 289-291.

(19) Skirpov, V. P., and P. I. Potashev. "Heat Exchange With Carbon Dioxide Along Supercritical Isotherms Under Free Convection Conditions." NASA TT F-11333 (1967).

(20) Brodowicz, K., and J. Bialokoz. "Free Convection Heat Transfer From a Vertical Plate to Freon-12 Near the Critical State." Archiwum Budowy Maszyn, Vol. 10, No. 4 (1963), pp. 289-303.

(21) Holt, V. E., and R. J. Grosh. "Free Convection Heat Transfer Up to Near Critical Conditions." Nucleonics, Vol. 21, No. 8 (August, 1963), pp. 122-125.

(22) Dubrovina, E. N., and V. P. Skripov. "Convective Heat Transfer to Carbon Dioxide in the Supercritical Region." Proceedings of the Second All-Soviet Union Conference on Heat and Mass Transfer. Vol. 1. C. Gazely, Jr., J. P. Hartnett, and E. R. G. Eckert, eds. Berkeley: University of California Press, 1966, pp. 36-45.

(23) Knapp, K. K., and R. H. Sabersky. "Free Convection Heat Transfer to Carbon Dioxide Near the Critical Point." Int. J. Heat Mass Transfer, Vol. 9, No. 1 (January, 1966), pp. 41-51.

(24) Nishikawa, K., and K. Miyabe. "On the Boiling-Like Phenomena at Supercritical Pressures." Mem. Fac. Eng. Kyushu Univ., Vol. 25, No. 1 (December, 1965), pp. 1-25.

(25) Daniels, T. C., and J. W. Bramall. "An Experimental Investigation of the Heating Mechanism of Carbon Dioxide Above and Below the Critical Point." Proc. Inst. Mech. Eng., Vol. 180, Part 3C (1965), pp. 247-259.

(26) Graham, R. W., R. C. Hendricks, and R. C. Ehlers. "Analytical and Experimental Study of Pool Heating of Liquid Hydrogen Over a Range of Acceleration." NASA Technical Note D-1883 (1964).

(27) Larson, J. R., and R. J. Schoenhals. "Turbulent Free Convection in Near Critical Water." J. Heat Transfer, Vol. 88, No. 4 (November, 1966), pp. 407-414.

(28) Goldstein, U., and W. Aung. "Heat Transfer to Free Convection From a Horizontal Wire to Carbon Dioxide in the Critical Region." ASME, Paper No. 67-WA/HT-2 (November, 1967).

(29) Kato, Y., N. Nishiwaki, and M. Hirata. Trans. Japan Soc. Mech. Engrs., Vol. 33 (1967), p. 1816.

(30) Nishikawa, K., T. Ito, M. Kinoshita, M. Hatano, and H. Kanei. Tech. Rep. Kyushu Univ., Vol. 40 (1967), p. 986.

(31) Mann, D. B. "The Thermodynamic Properties of Helium From 3° to 300°K Between 0.5 and 100 Atmospheres." National Bureau of Standards, Technical Note 154 (January, 1962).

(32) Roder, H. M., and R. D. Goodwin. "Provisional Thermodynamic Functions for Parahydrogen." National Bureau of Standards, Technical Note 130 (December, 1961).

(33) Strobridge, T. R. "The Thermodynamic Properties of Nitrogen From 64° to 300°K Between 0.1 to 200 Atmospheres." National Bureau of Standards, Technical Note 129 (January, 1962).

(34) Weber, L. A. "Thermodynamic and Related Properties of Oxygen From the Triple Point to 300°K at Pressures to 330 Atmospheres." National Bureau of Standards, Rep. 9710 (June, 1968).

(35) Lyman, R. S. "Propane Thermodynamic Property Equations." ASME, Paper No. 68-WA/PID-7 (December, 1968).

(36) Martin, J. J. "Correlations and Equations Used in Calculating the Thermodynamic Properties of Freon Refrigerants." Thermodynamic Transport Properties of Gases, Liquids, and Solids. New York: ASME and McGraw-Hill Book Company, 1959.

(37) Stiel, L. I., and G. Thodos. "The Prediction of the Transport Properties of Pure Gaseous and Liquid Substances." Progress in International Research on Thermodynamic and Transport Properties, ASME (1962), pp. 352-365.

(38) Brokaw, R. S. "Statistical Mechanical Theories of Transport Properties." Presented at the International Conference on the Properties of Steam, September 9-13, 1968, Japan.

(39) Fritsch, C. A., and R. J. Grosh. "A Description of Thermal Conductivity and Dynamic Viscosity of Water in the Critical Region." ANL-6238 (October, 1960).

(40) Meyer, C. A., R. B. McClintock, G. J. Silvestri, and R. C. Spencer. "1967 ASME Steam Tables." Thermodynamic and Transport Properties of Steam. New York: ASME, 1967.

(41) McClintock, R. B., and G. J. Silvestri. Formulations and Iterative Procedures for the Calculation of Properties of Steam. New York: ASME, 1968.

(42) Michels, A., A. Botzen, and W. Schuurman. "The Viscosity of Carbon Dioxide Between 0°C and 75°C and at Pressures Up to 2000 Atmospheres." Physica, Vol. 23 (1957), p. 95.

(43) Kestin, J., J. H. Whitelaw, and T. F. Zien. "The Viscosity of Carbon Dioxide in the Neighborhood of the Critical Point." Physica, Vol. 30 (1964), pp. 161-181.

(44) Guildner, L. A. "Thermal Conductivity of Gases. II. Thermal Conductivity of Carbon Dioxide Near the Critical Point." J. of Research of the National Bureau of Standards, Vol. 66A, No. 4 (1962), pp. 341-348.

(45) Sengers, J. V., and A. Michels. "The Thermal Conductivity of Carbon Dioxide in the Region of the Critical Point." Second Symposium on Thermophysical Properties, ASME. New York: Academic Press, January 24-26, 1962, p. 434.

(46) Koppel, L. B., and J. M. Smith. "Thermal Properties of Carbon Dioxide in the Critical Region." J. Chem. Engng. Data, Vol. 5, No. 4 (1960), pp. 437-440.

(47) Michels, A., and S. R. DeGroot. "Thermodynamical Properties of Carbon Dioxide as a Function of Density and Temperature." Applied Science Research, Vol. A1 (1948), p. 94.

(48) DeGroot, S. R., and A. Michels. "The Joule-Thomason Effect and the Specific Heat at Constant Pressure of Carbon Dioxide." Physica, Vol. 14, No. 4 (1948), pp. 218-222.

(49) Michels, A., and S. R. DeGroot. "Thermodynamical Properties of Carbon Dioxide as a Function of Pressure and Temperature." Applied Science Research, Vol. A1 (1948), p. 103.

(50) Mullin, T. E. "Laminar Natural Convection of a Fluid in the Supercritical Region." (Unpublished Ph.D. thesis, Oklahoma State University, 1968.)

(51) Parker, J. D., J. H. Boggs, and E. F. Blick. Introduction to Fluid Mechanics and Heat Transfer. 3rd ed. New York: Addison-Wesley Publishing Company, Inc., 1974.

(52) Jackson, L. W., and J. P. Chandler. Personal communcation. Department of Computing and Information Sciences, Oklahoma State University, Stillwater, Oklahoma, 1978.

(53) Nowak, E. S., R. J. Grosh, and P. E. Liley. "Smoothed Pressure-Volume-Temperature Data for Water in the Critical Region Derived From Experimental Measurements." J. Heat Transfer, Trans. ASME, Vol. 83, No. 2 (February, 1961), pp. 14-26.

(54) Michels, A., and C. Michels. "Isotherms of CO_2 Between 0° and 150° and Pressures from 16 to 250 Atmospheres (Amagat Densities Between 18 and 206)." Proceedings of the Royal Society of London, Vol. A153 (1936), pp. 201-214.

(55) Michels, A., C. Michels, and H. Wouters. "Isotherms of CO_2 Between 70 and 3000 Atmospheres (Amagat Densities Between 200 and 600)." Proceedings of the Royal Society of London, Vol. A153 (1936), pp. 214-224.

(56) Michels, A., B. Blaisse, and C. Michels. "The Isotherms of CO_2 in Neighbourhood of the Critical Point and Round the Coexistence Line." Proceedings of the Royal Society of London, Vol. A160 (1937), pp. 358-375.

(57) Hamming, R. W. Numerical Methods for Scientists and Engineers. 2nd ed. New York: McGraw-Hill Book Company, 1973.

(58) Reynolds, W. C., and H. C. Perkins. Engineering Thermodynamics. 1st ed. New York: McGraw-Hill Book Company, 1970.

(59) Nowak, E. S., and R. J. Grosh. "An Investigation of Certain Thermodynamic and Transport Properties of Water and Water Vapor in the Critical Region." Argonne National Laboratory--6064 (October, 1959).

(60) Reid, R. C., and T. K. Sherwood. The Properties of Gases and Liquids. 2nd ed. New York: McGraw-Hill Book Company, 1968.

(61) Downing, R. C. "Transport Properties of Freon Fluorocarbons." Dupont Tech. Bull. C-30, 1965.

(62) Hornbeck, R. W. Numerical Methods. 1st ed. New York: Quantum Publishers, Inc., 1975.

(63) Keller, H. B. "Accurate Difference Methods for Nonlinear Two-Point Boundary Value Problems." SIAM J. Numer. Anal., Vol. 6 (1969), pp. 8-30.

(64) Pereyra, V. "Variable Order Variable Step Finite Difference Methods for Nonlinear Boundary Value Problems." Proceedings, Conference on the Numerical Solution of the Differential Equations, Dundee, Scotland. Berlin: Springer-Verlag, 1973.

(65) Eckert, E. R. G. "Discussion of Heat Transfer and Fluid Friction for Fully Developed Turbulent Flow of Air and Supercritical Water With Variable Fluid Properties." Trans. ASME, Vol. 76, No. 1 (January, 1954), pp. 83-84.

APPENDIX A

TABLES

TABLE I

SUMMARY OF FREE CONVECTION EXPERIMENTAL WORK

Investigator	Date of Work	Reference	Fluid	Geometry	Range of Conditions: Ratio of Pressure to Critical Pressure P/P_c	Range of Conditions: Ratio of Temperature to Critical Temperature T/T_c
Doughty and Drake	1956	16	R-12 (ccl_2F_2)	Horizontal Wire	0.28 to 1.70	0.77 to 1.04
Bonilla and Sigel	1960	17	n-pentane (C_5H_{12})	Horizontal Plate	0.72 to 1.50	0.65 to 0.74 (fluid) 0.71 to 1.06 (surface)
Griffith and Sabersky	1960	18	R-114 ($C_2cl_2F_4$)	Horizontal Wire	0.42 to 1.15	0.725 to 1.080
Skripov and Potashev	1962	19	CO_2	Vertical Wire	0.82 to 1.25	1.00 to 1.02
Brodowicz and Bialokoz	1963	20	R-12 (ccl_2F_2)	Vertical Plate	1	0.98 to 1.07
Holt and Grosh	1963	21	H_2O	Horizontal Wire and Vertical Ribbon	1.5×10^{-4} to 1.25	0.75 to 1.00 (fluid) 0.75 to 2.00 (surface)

TABLE I (Continued)

Investigator	Date of Work	Reference	Fluid	Geometry	Range of Conditions: Ratio of Pressure to Critical Pressure P/P_c	Range of Conditions: Ratio of Temperature to Critical Temperature T/T_c
Fritsch and Grosh	1963	12	H_2O	Vertical Ribbon	1.03 to 1.06	1.0036 to 1.0086
Dubrovina and Skripov	1964	22	CO_2	Horizontal Wire, Horizontal and Vertical Ribbon	0.99 to 1.30	1.00 to 1.03
Knapp and Sabersky	1965	23	CO_2	Horizontal and Vertical Wire	1.03 to 1.40	0.93 to 1.09
Nishikawa and Miyabe	1965	24	CO_2	Horizontal Wire	1.00 to 1.33	0.98 to 1.63
Daniels and Bramall	1965	25	CO_2	Horizontal Monel Wire	0.79 to 1.16	0.86 to 1.03
Graham et al.	1965	26	H_2	Horizontal Ribbon	0.32 to 1.38	0.76 to 1.19
Larson and Schoenhals	1966	27	H_2O	Vertical Ribbon	1.01 to 1.03	0.98 to 1.04
Goldstein and Aung	1967	28	CO_2	Horizontal Wire	1.03 to 1.21	0.94 to 1.09

TABLE I (Continued)

Investigator	Date of Work	Reference	Fluid	Geometry	Range of Conditions: Ratio of Pressure to Critical Pressure P/P_c	Range of Conditions: Ratio of Temperature to Critical Temperature T/T_c
Kato et al.	1967	29	CO_2	Vertical Wire	1.097	0.96 to 1.54
Nishikawa et al.	1967	30	CO_2	Vertical Wire	1.097	0.96 to 1.74

TABLE II

SOURCES OF PHYSICAL PROPERTIES FOR FLUIDS IN THE SUPERCRITICAL REGION

	Critical Constants			Source Reference		Properties Recommended as Computable by Source Authors							
Fluid	Pressure, P_c, atm	Temperature, T_c, °K	Density, ρ_c, g/cm^3	Equation of State	Transport Properties	Pressure-Density-Temperature P-ρ-T	Enthalpy H	Specific Heat at Constant Pressure C_p	Specific Heat at Constant Volume C_V	Expansion Coefficient $\beta = -\frac{1}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_P$	Entropy S	Absolute Viscosity μ	Thermal Conductivity K
Helium-4	2.26	5.30	0.0692	31	37,38	x	x				x	x	x
Hydrogen	12.80	32.99	0.0314	32	37,38	x	x	x	x	x	x	x	x
Nitrogen	33.50	126.20	0.3110	33	37,38	x	x					x	x
Oxygen	50.14	154.80	0.4358	34	37,38	x	x	x	x	x	x	x	x
Propane	42.10	370.00	0.2100	35	---	x	x	x	x	x	x		
Water	218.20	647.20	0.3220	*	37-41	x	x	x	x	x	x	x	x
Carbon Dioxide	72.90	304.20	0.4680	*	42-49	x	x	x	x	x		x	x
R-12	40.69	385.10	0.5570	36	---	x	x		x		x		
R-114	31.45	418.70	0.5820	13	50	x		x	x	x		x	x
Ammonia	112.30	406.10	0.2360	---	37,38	x				x		x	x

*Present study.

TABLE III

TABLE OF $\bar{C}$ VALUES FOR GAS A

$\frac{T_w}{T_\infty}$	$-\frac{\theta'(0)_w}{\sqrt{2}}$	$\frac{\theta'(0)_{ref}}{\theta'(0)_w}$*	$(\frac{\theta'(0)_{ref}}{\theta'(0)_w})^8$	$1-(\frac{\theta'(0)_{ref}}{\theta'(0)_w})^8$	$\frac{T_\infty}{T_w}$	$1-(\frac{T_\infty}{T_w})$	$\bar{C}$
4	0.371	0.95148	0.672	0.328	1/4	3/4	0.438
3	0.368	0.95920	0.716	0.284	1/3	2/3	0.427
5/2	0.366	0.96448	0.749	0.251	2/5	3/5	0.418
2	0.363	0.97245	0.799	0.201	1/2	1/2	0.402
3/4	0.348	1.01436	1.122	-0.122	4/3	-1/3	0.366
1/2	0.339	1.04129	1.383	-0.383	2/1	-1	0.383
1/3	0.330	1.06969	1.714	-0.714	3/1	-2	0.357
1/4	0.323	1.09287	2.035	-1.035	4/1	-3	0.345

*Where $\theta'(0)_{ref}$ was obtained from constant property solution with Pr = 0.7.

$Nu_{x,w}/Gr_{x,w}^{1/4} = -(\theta'(0)_{ref}/\sqrt{2}) = 0.353.$

TABLE IV

UNITS AND CONSTANTS FOR EQUATION OF STATE

Parameter	Description	Units	Refrigerant-114	Water	Carbon Dioxide
P	Pressure	$lbf/in.^2$	---	---	---
T	Temperature	°R	---	---	---
V	Specific Volume	ft^3/lbm	---	---	---
T_c	Critical Temperature	°R	753.95	1165.30	547.56
R	Gas Constant	$ft^3/°R\text{-}in.^2$	0.0628018	0.59569248	0.24384004
K	Constant	---	3.00	0.86919801	0.12018118×10^{1}
b	Constant	---	5.9149070×10^{-3}	$-0.24670554 \times 10^{-1}$	$-0.18588948 \times 10^{-1}$
A_2	Constant	---	-2.3856704	$-0.22524809 \times 10^{2}$	$-0.13326704 \times 10^{2}$
B_2	Constant	---	1.0801207×10^{-3}	$0.11652976 \times 10^{-1}$	$-0.83298325 \times 10^{-2}$
C_2	Constant	---	-6.5643648	$-0.70259944 \times 10^{2}$	0.62417313×10^{2}
A_3	Constant	---	3.4055687×10^{-2}	$-0.87423115 \times 10^{1}$	$-0.73090008 \times 10^{1}$
B_3	Constant	---	$-5.3336494 \times 10^{-6}$	$0.70670168 \times 10^{-2}$	$0.90982283 \times 10^{-2}$
C_3	Constant	---	1.6366057×10^{-1}	-0.19906674	0.57368866×10^{1}
A_4	Constant	---	$-3.8574810 \times 10^{-4}$	$0.71943958 \times 10^{-1}$	$0.18744415 \times 10^{-1}$
A_5	Constant	---	1.6017659×10^{-6}	$0.12179538 \times 10^{-1}$	$0.10958040 \times 10^{-1}$
B_5	Constant	---	$6.2632340 \times 10^{-10}$	$-0.15583440 \times 10^{-4}$	$-0.12655010 \times 10^{-4}$
C_5	Constant	---	$-1.0165314 \times 10^{-5}$	$0.13679639 \times 10^{-1}$	$-0.13531694 \times 10^{-1}$

TABLE V

UNITS AND CONSTANTS FOR SPECIFIC HEAT AT CONSTANT PRESSURE

Parameter	Description	Units	Refrigerant-114	Water	Carbon Dioxide
C_P or C_V	Specific Heat	Btu/lbm-°R	---	---	---
T	Temperature	°R	---	---	---
V	Specific Volume	ft^3/lbm	---	---	---
T_c	Critical Temperature	°R	753.95	1165.30	547.56
R	Gas Constant	ft^3/°R-$in.^2$	0.0628018	0.59569248	0.24384004
K	Constant	---	3.0	0.84750477	0.12092998×10^{1}
b	Constant	---	5.9149070×10^{-3}	$-0.23690403 \times 10^{-1}$	$-0.30146832 \times 10^{-1}$
A_2	Constant	---	-2.3856704	$-0.22981253 \times 10^{2}$	$-0.10972492 \times 10^{2}$
B_2	Constant	---	1.0801207×10^{-3}	$0.11977436 \times 10^{-1}$	$-0.56307166 \times 10^{-2}$
C_2	Constant	---	-6.5643648	$-0.72422920 \times 10^{2}$	0.47571869×10^{2}
A_3	Constant	---	3.4055687×10^{-2}	$-0.87176941 \times 10^{1}$	$-0.72139233 \times 10^{1}$
B_3	Constant	---	$-5.3336494 \times 10^{-6}$	$0.71550030 \times 10^{-2}$	$0.90332088 \times 10^{-2}$
C_3	Constant	---	1.6366057×10^{-1}	-0.19803821	0.63052626×10^{1}
A_4	Constant	---	$-3.8574810 \times 10^{-4}$	$0.77529884 \times 10^{-1}$	$0.13941827 \times 10^{-2}$
A_5	Constant	---	1.6017659×10^{-6}	$0.11963517 \times 10^{-1}$	$0.11295915 \times 10^{-1}$
B_5	Constant	---	$6.2632340 \times 10^{-10}$	$-0.15597259 \times 10^{-4}$	$-0.12727667 \times 10^{-4}$
C_5	Constant	---	$-1.0165314 \times 10^{-5}$	$0.12926977 \times 10^{-1}$	$-0.13171462 \times 10^{-1}$

TABLE V (Continued)

Parameter	Description	Units	Refrigerant-114	Water	Carbon Dioxide
X_{13}	Constant	---	0.0175	0.11488819×10^{1}	0.70600271×10^{1}
X_{14}	Constant	---	3.49×10^{-4}	$0.44963023 \times 10^{-4}$	$-0.17494759 \times 10^{-1}$
X_{15}	Constant	---	-1.67×10^{-7}	$0.28090134 \times 10^{-8}$	$0.10779554 \times 10^{-4}$

TABLE VI

UNITS AND CONSTANTS FOR ABSOLUTE VISCOSITY

Parameter	Description	Units	Refrigerant-114	Water	Carbon Dioxide
μ	Viscosity	Centipoises* or lbm/ft-sec	---	---	---
μ_o	Low Pressure Viscosity	Centipoises	---	---	---
M	Molecular Weight	---	170.936	---	---
P_c	Critical Pressure	Atmosphere	32.190	---	---
T_c	Critical Temperature	°K	418.880	---	---
T	Temperature	°R	---	---	---
ρ_c	Critical Density	g/cm^3	0.582	---	---
ρ	Density	g/cm^{3*} or lbm/ft^3	---	---	---
A	Constant	---	---	0.9717059×10^{-5}	$-0.2003388 \times 10^{-5}$
B	Constant	---	---	0.6335240×10^{-8}	0.2267791×10^{-7}
C	Constant	---	---	0.1024267×10^{-6}	0.1030594×10^{-7}
D	Constant	---	---	1.57	0.2089048×10^{1}

Note: The units designated by (*) exclusively refer to Refrigerant-114.

TABLE VII

UNITS AND CONSTANTS FOR THERMAL CONDUCTIVITY

Parameter	Description	Units	Refrigerant-114	Water	Carbon Dioxide
K	Thermal Conductivity	$\frac{\text{cal}^*}{\text{sec-cm-°K}}$ or $\frac{\text{Btu}}{\text{hr-ft-°F}}$	---	---	---
K_o	Low Pressure Thermal Conductivity	cal/sec-cm-°K	---	---	---
μ_o	Low Pressure Viscosity	centipoises	---	---	---
C_{V_o}	Low Pressure Specific Heat	cal/g-mole-°C	---	---	---
M	Molecular Weight	---	170.936	---	---
R	Gas Constant	cal/g-mole-°K	1.987	---	---
P_c	Critical Pressure	atmosphere	32.190	---	---
T_c	Critical Temperature	°K	418.880	---	---
T	Temperature	°R or °F$^+$	---	---	---
ρ_c	Critical Density	g-mole/cm^3	99.485	---	---
ρ	Density	$\frac{\text{g-mole}^*}{\text{cm}^3}$ or $\frac{\text{lbm}}{\text{ft}^3}$	---	---	---
A	Constant	---	---	3.6200×10^{-3}	$0.84746444 \times 10^{-2}$
B	Constant	---	---	0.0380×10^{-3}	$-0.44669005 \times 10^{-4}$
C	Constant	---	---	2.6670×10^{-3}	$0.11056622 \times 10^{-1}$
D	Constant	---	---	1.24	0.46278907

Note: The units designated by (*) or (+) exclusively refer to Refrigerant-114 and water, respectively.

TABLE VIII

SUMMARY OF PHYSICAL PROPERTY EQUATIONS

Fluid	Eq. No. (See Ch. IV)	Source Ref.	No. of Points	Percent Deviation	Eq. No. (See Ch. IV)	Source Ref.	No. of Points	Percent Deviation	$T_r = T/T_c$	$P_r = P/P_c$
	Equation of State				Specific Heat at Constant Pressure				Range of Application	
Water	(4.1)	40, 41, 53	64	±1.38	(4.12)	37,38, 40,41, 59	64	±3.90	1.0080 to 1.0400	1.0440 to 1.2470
Carbon Dioxide	(4.1)	54-56	52	±1.80	(4.12)	46-49	34	±3.80	1.0028 to 1.1455	1.0560 to 1.9550
Refrigerant -114	(4.1)	13	---	---	(4.12)	50	---	---	1.0076 to 1.1461	1.0355 to 2.3754
Fluid	Absolute Viscosity				Thermal Conductivity				Range of Application	
Water	(4.15)	39-41	64	±0.53	(4.26)	39-41	64	±2.10	1.0080 to 1.0400	1.0440 to 1.2470
Carbon Dioxide	(4.15)	42,43	72	±1.40	(4.26)	44,45	39	±2.50	1.0028 to 1.1455	1.0560 to 1.9550
Refrigerant -114	(4.13) and (4.14)	50	---	---	(4.24) and (4.25)	50	---	---	1.0076 to 1.1461	1.0355 to 2.3754

TABLE IX

CONSTANTS FOR HEAT TRANSFER CORRELATION

Fluid	Constants				Percent Deviation	Range of Application	
	a	b	c	d		$T_r = T/T_c$	$P_r = P/P_c$
Refrigerant -114	0.39992773	0.25	0.30848627	0.0049719059	1.5	1.0076 to 1.0740	1.0638 to 1.2766
Water	0.31530544	0.25	0.39214071	-0.022304567	2.5	1.0080 to 1.0380	1.0598 to 1.1845
Carbon Dioxide	0.32646432	0.25	0.40539598	-0.027499479	2.4	1.0038 to 1.0950	1.1194 to 1.4925

APPENDIX B

FIGURES

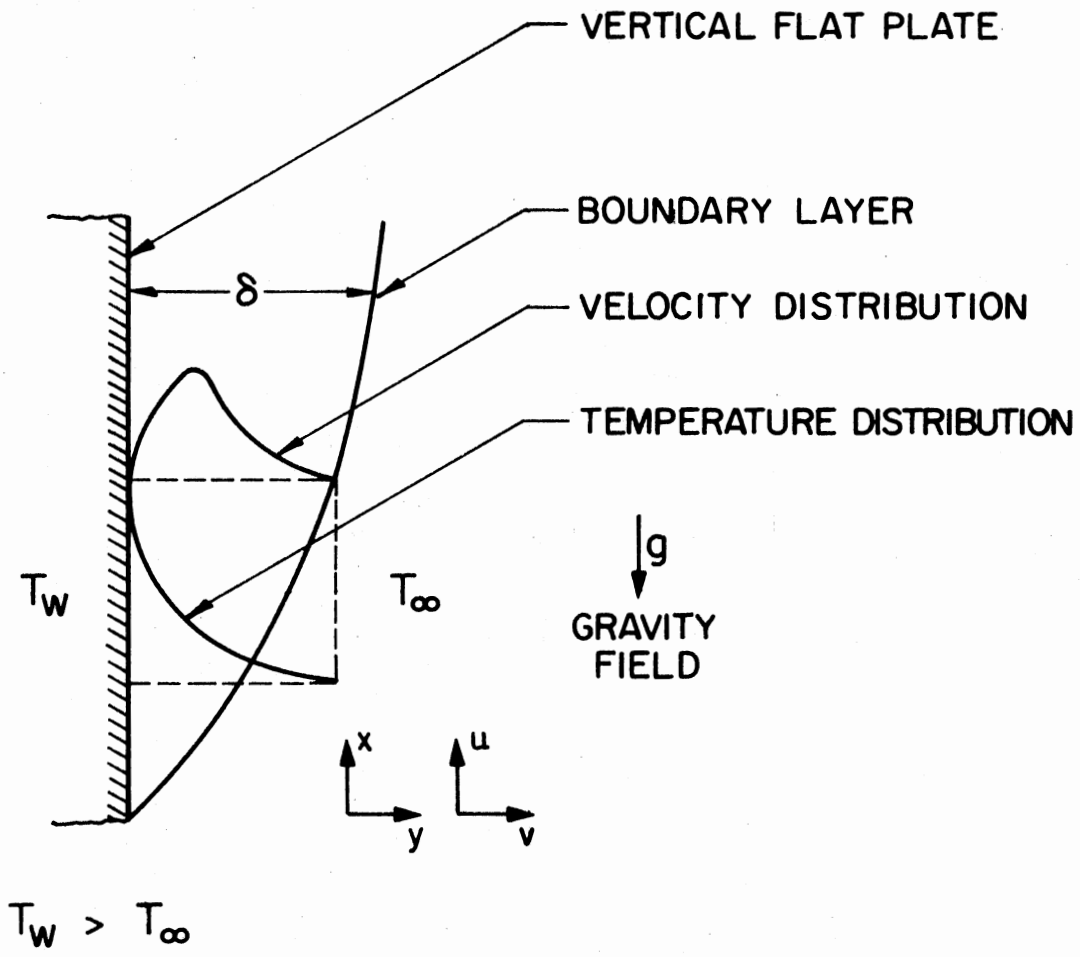

Figure 1. Physical Model and Coordinate System

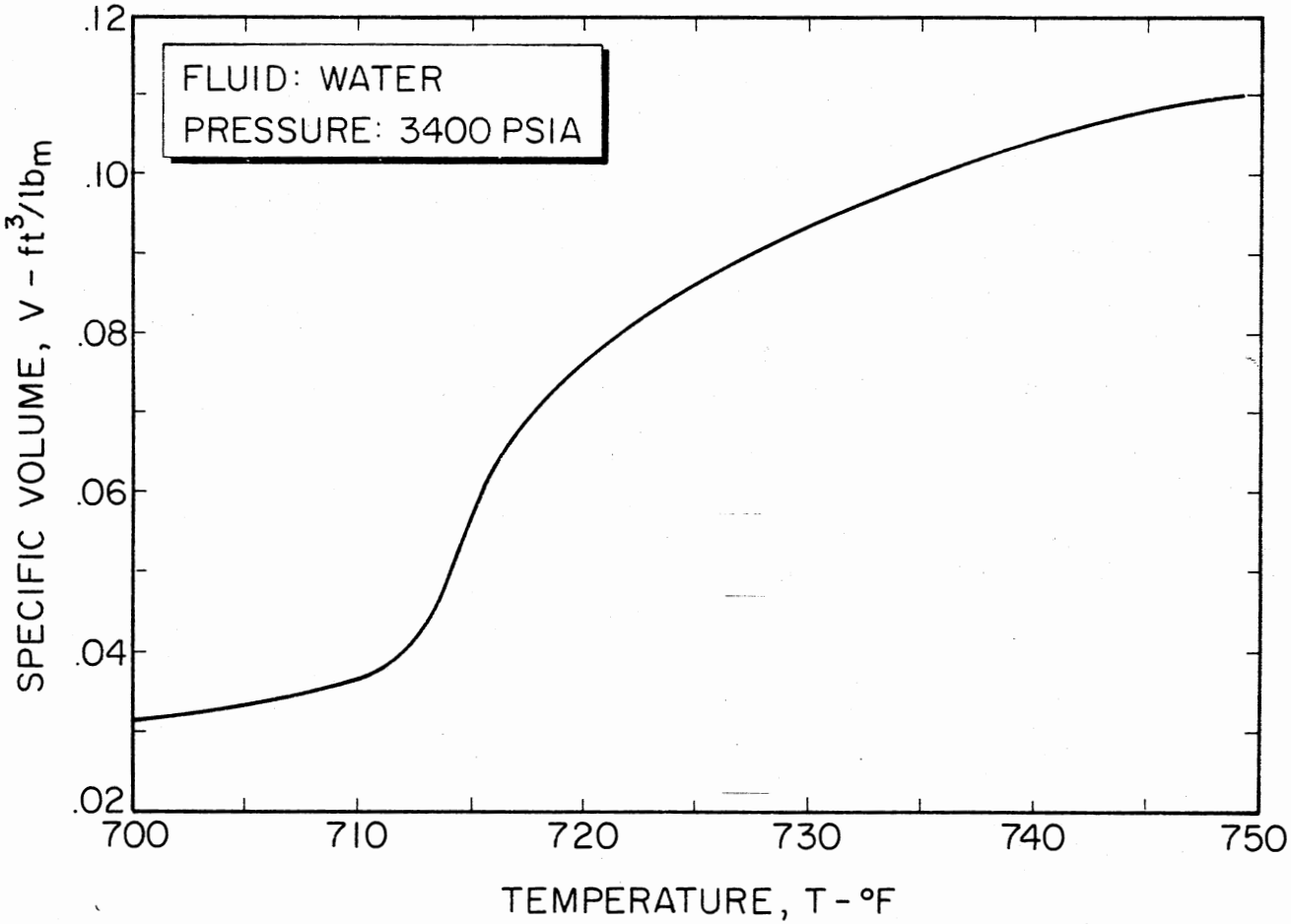

Figure 2. Specific Volume in the Supercritical Region

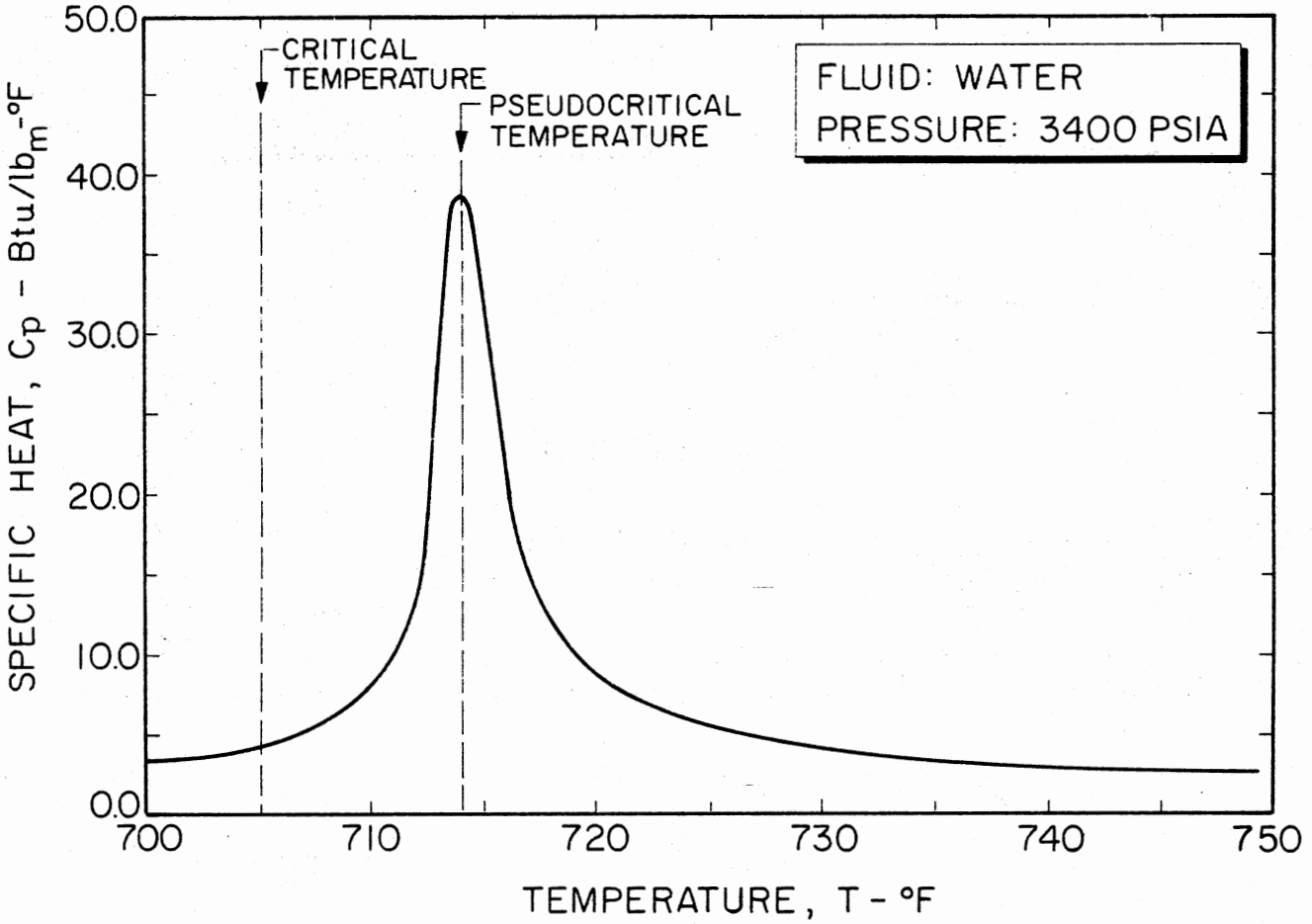

Figure 3. Specific Heat in the Supercritical Region

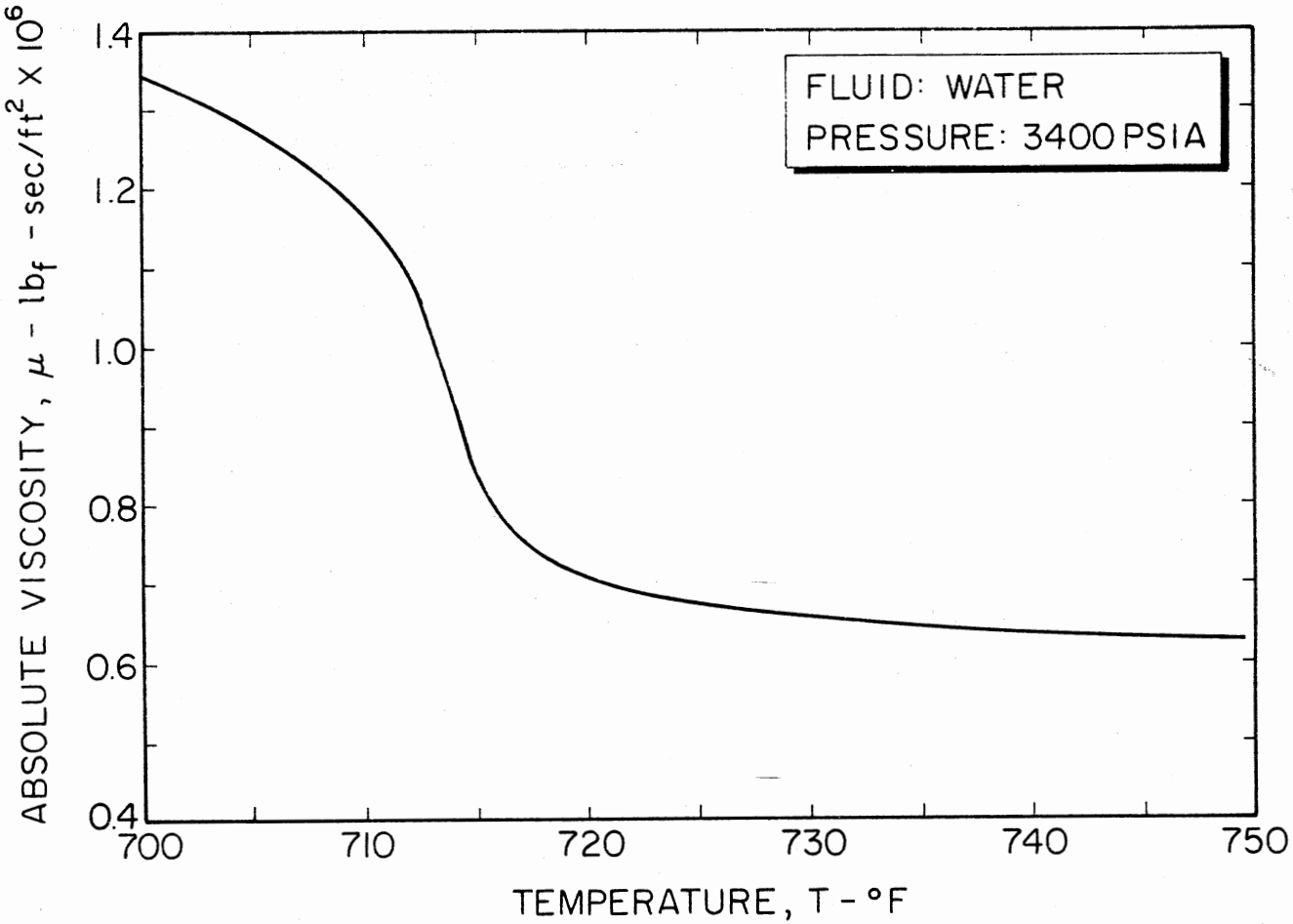

Figure 4. Absolute Viscosity in the Supercritical Region

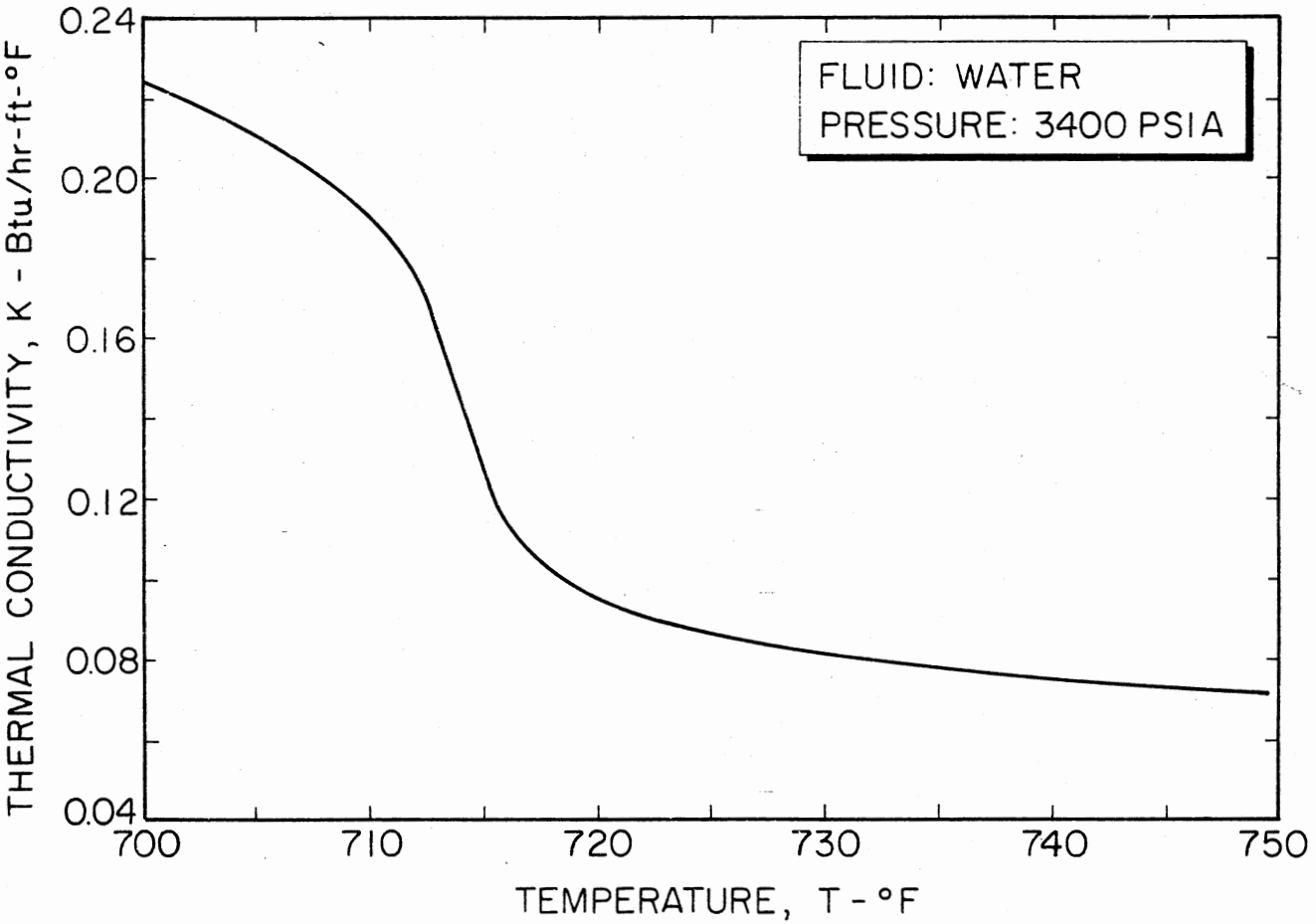

Figure 5. Thermal Conductivity in the Supercritical Region

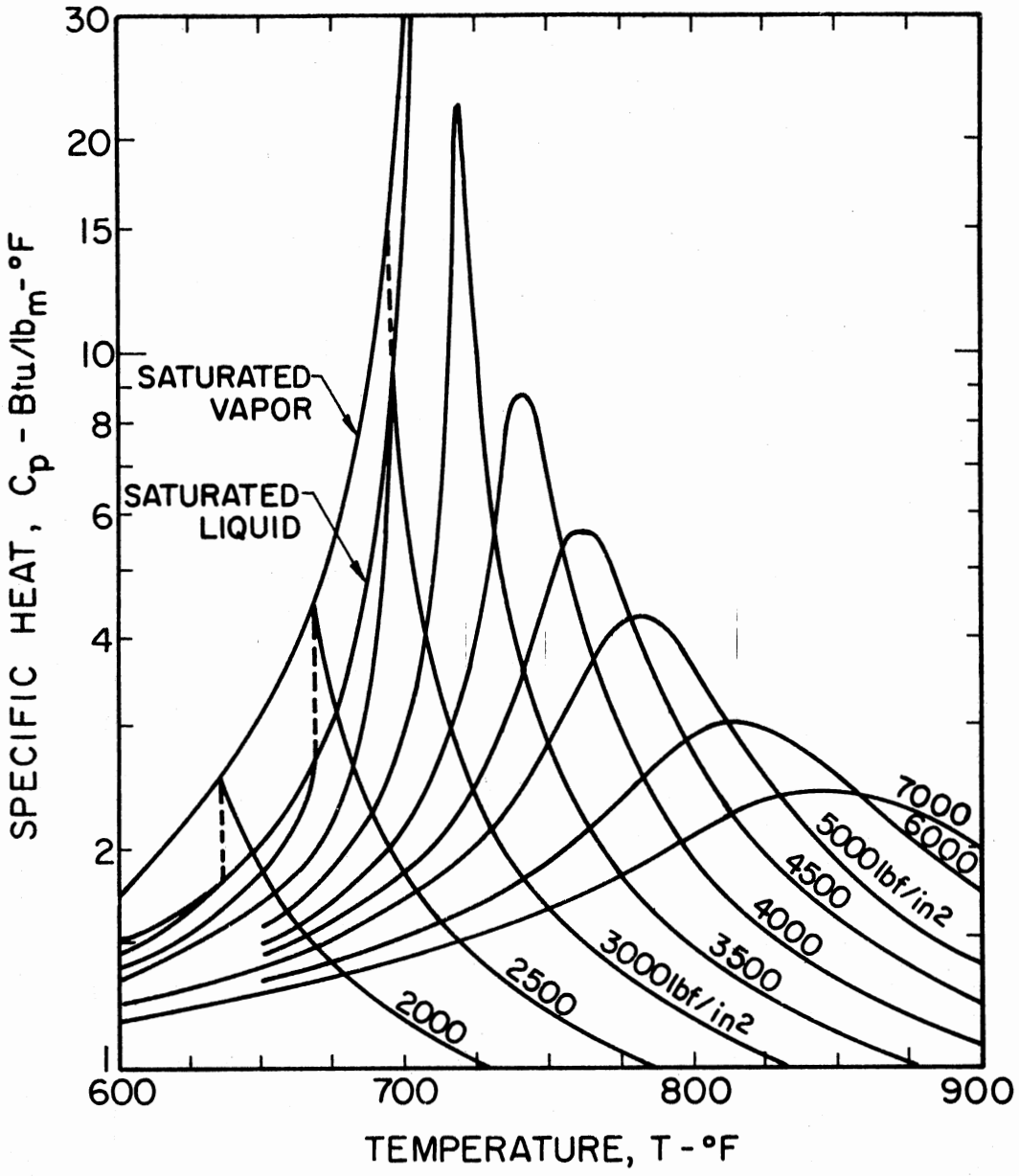

Figure 6. Specific Heat at Constant Pressure Above and Below the Critical Point for Water

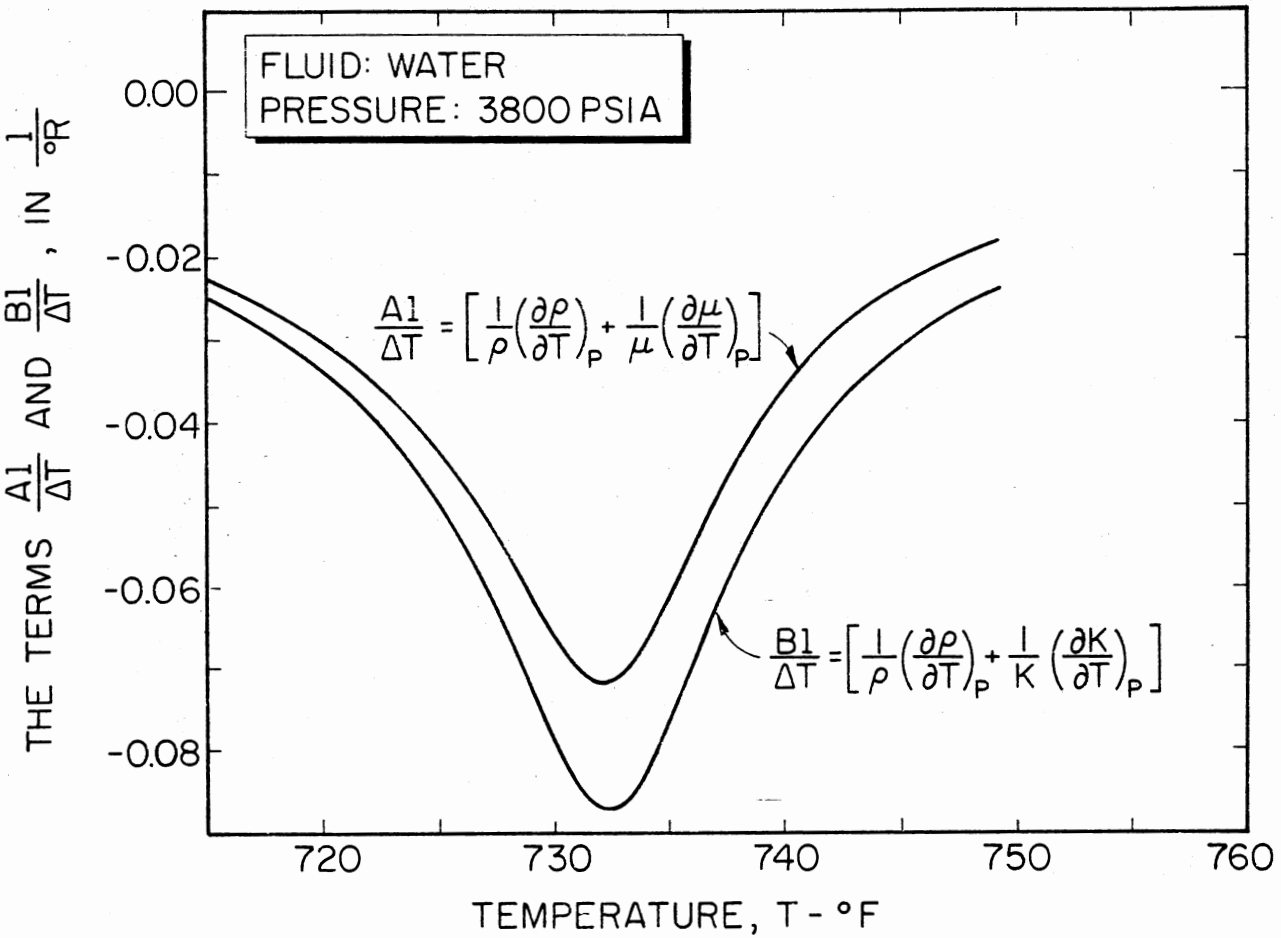

Figure 7. The Terms A1/ΔT and B1/ΔT (See Equations (3.16) and (3.19))

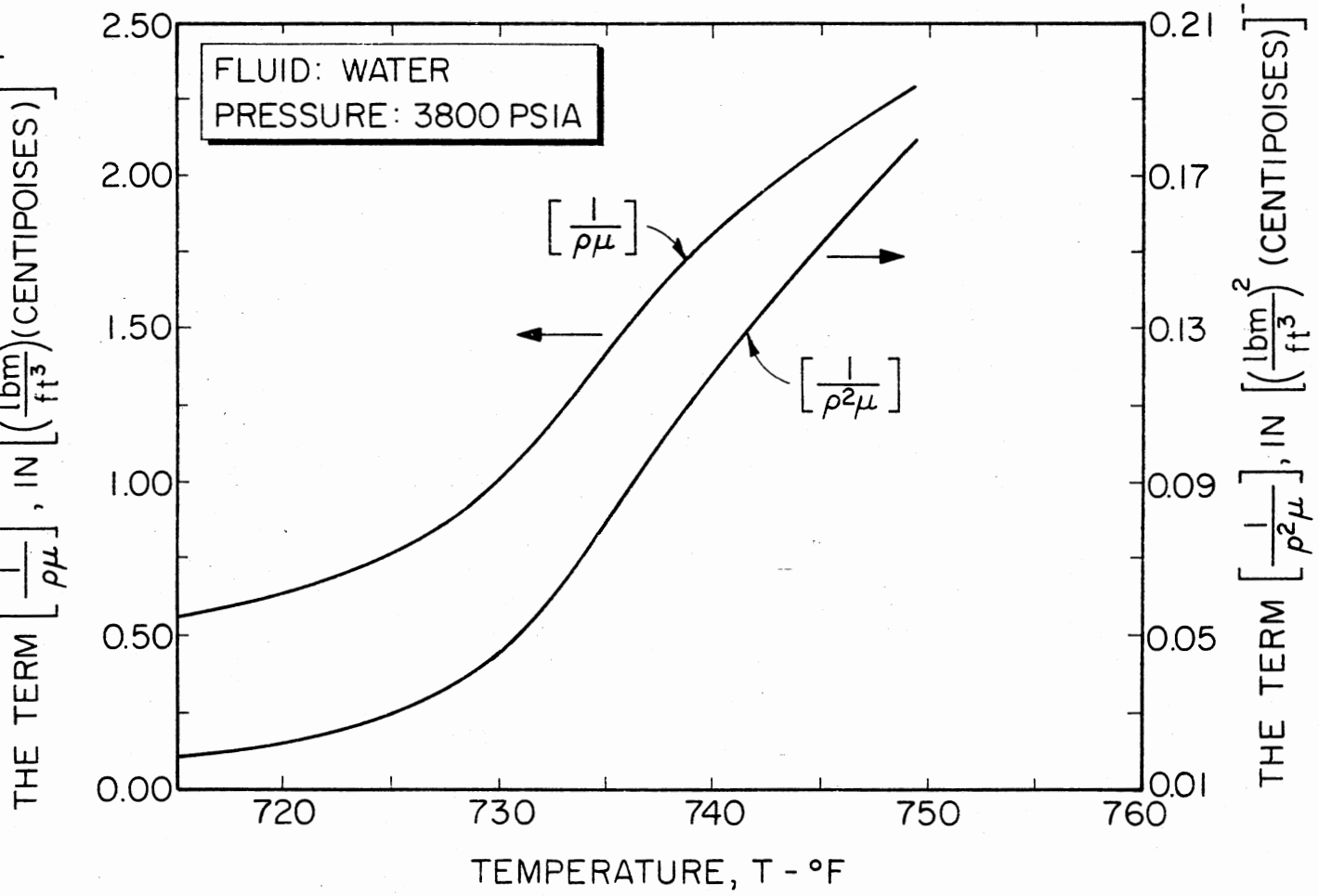

Figure 8. The Terms $[1/\rho\mu]$ and $[1/\rho^2\mu]$

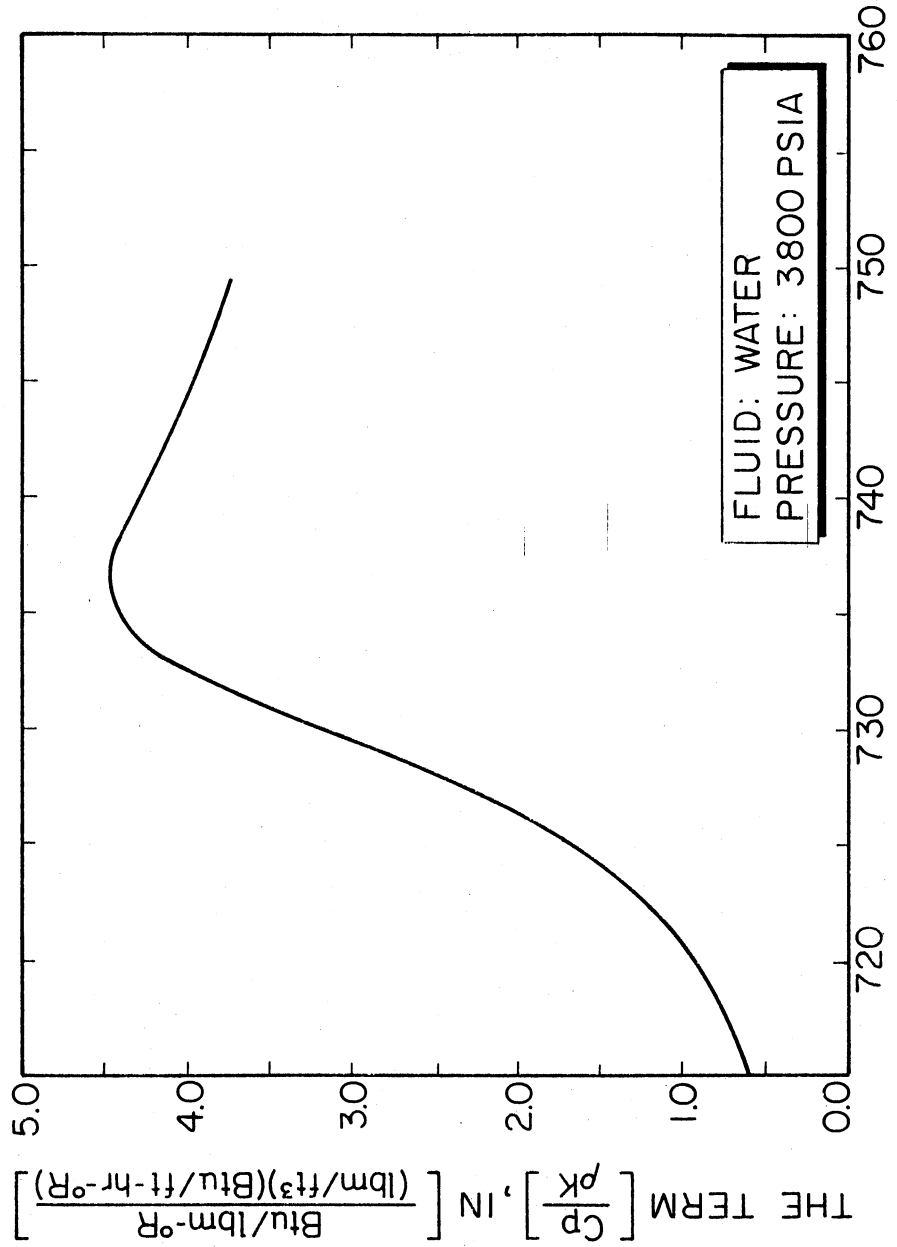

Figure 9. The Term $[C_p/\rho K]$

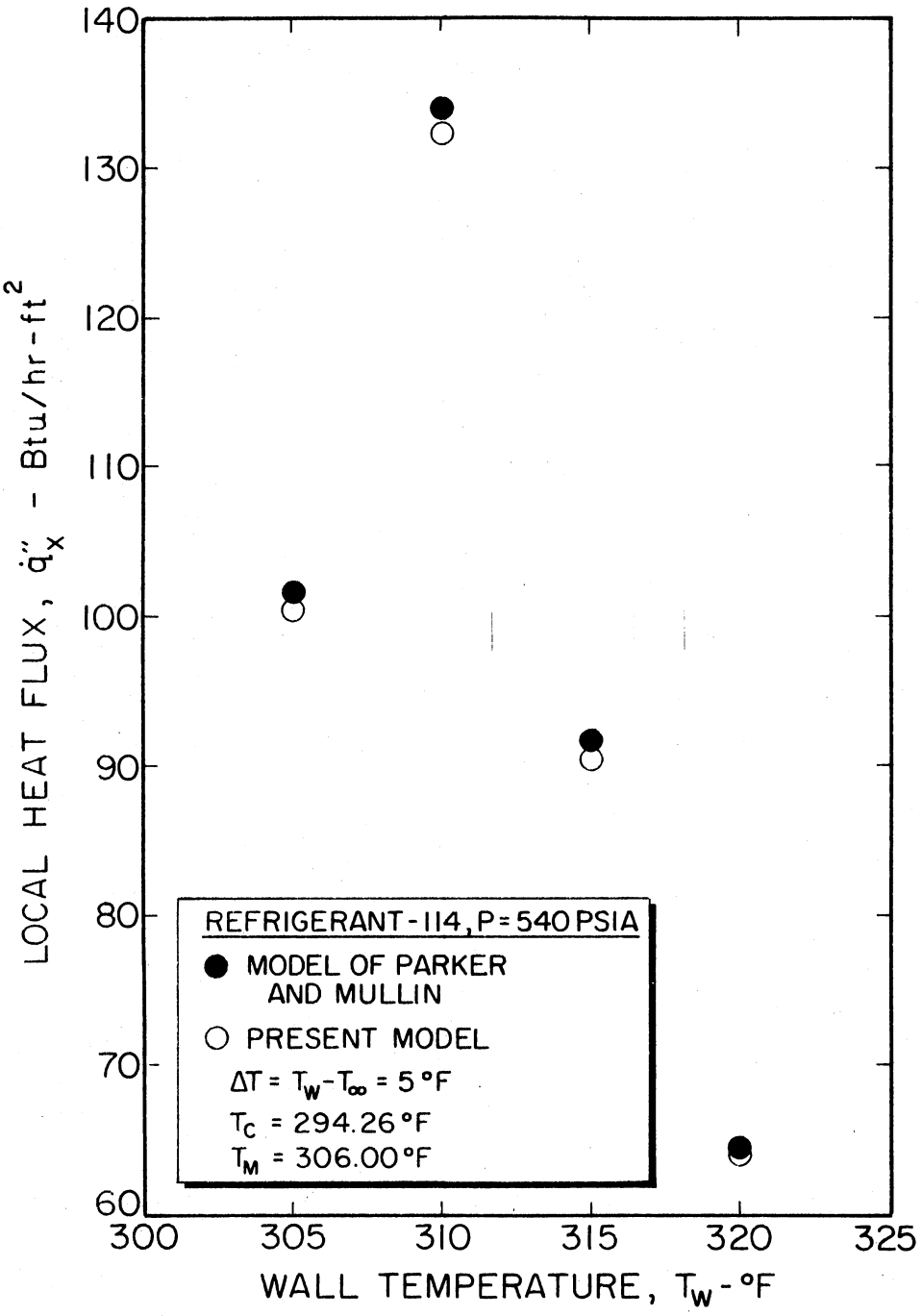

Figure 10. Comparison of Predicted Values of Local Heat Flux With Model of Parker and Mullin (5)

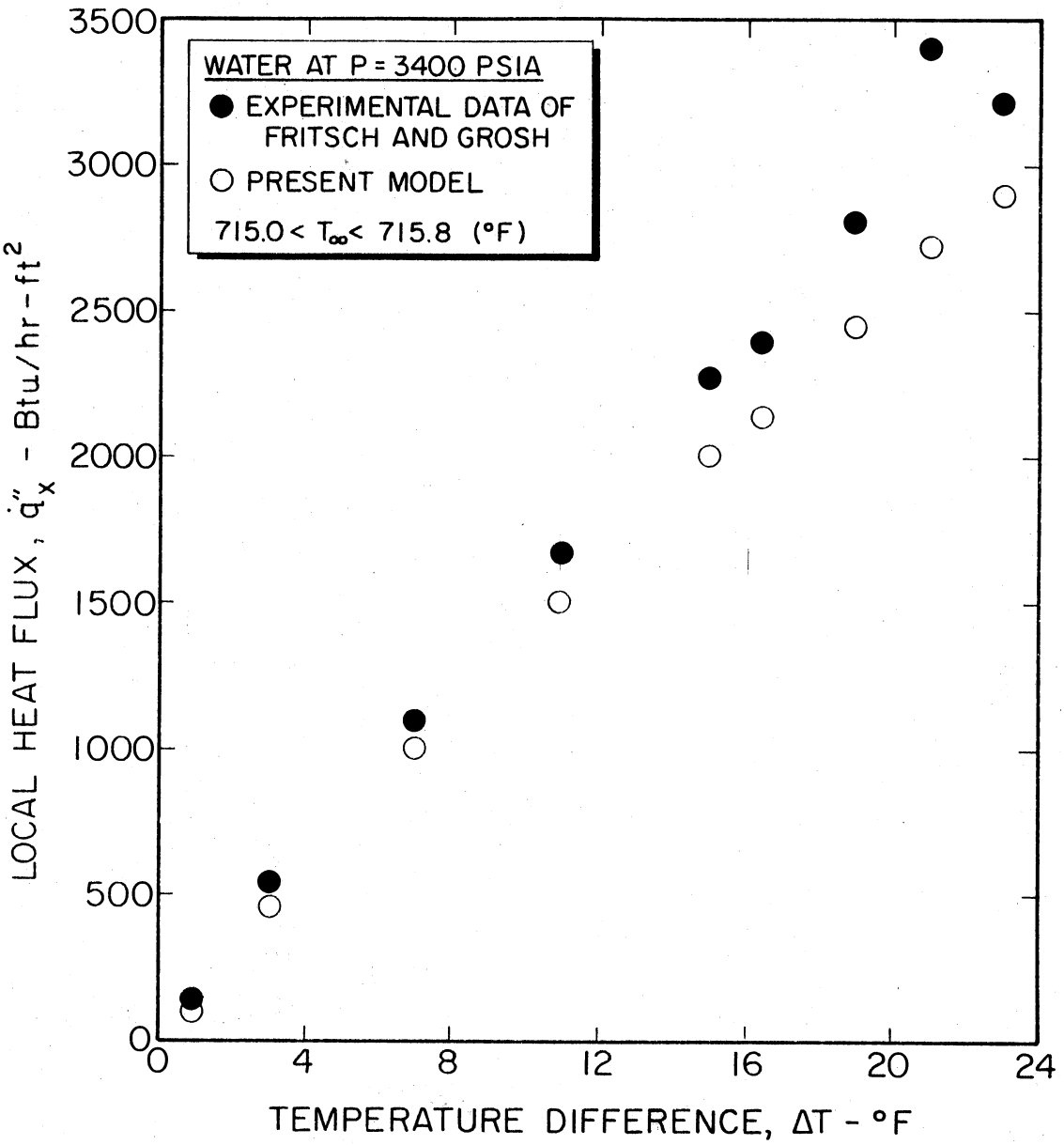

Figure 11. Comparison of Predicted Values of Local Heat Flux With Experimental Data of Fritsch and Grosh (12)

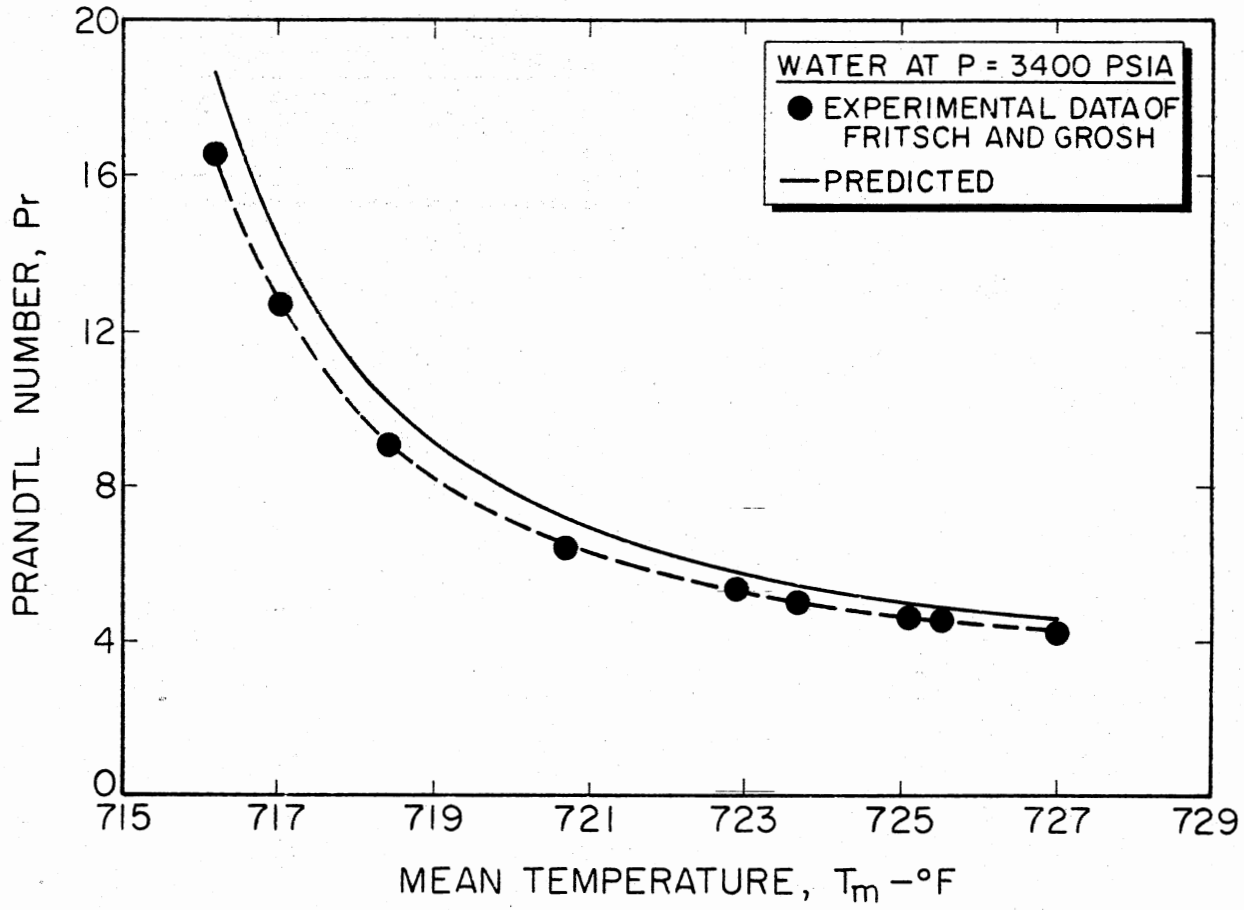

Figure 12. Comparison of Predicted Values of Prandtl Number With Experimental Data of Fritsch and Grosh (12)

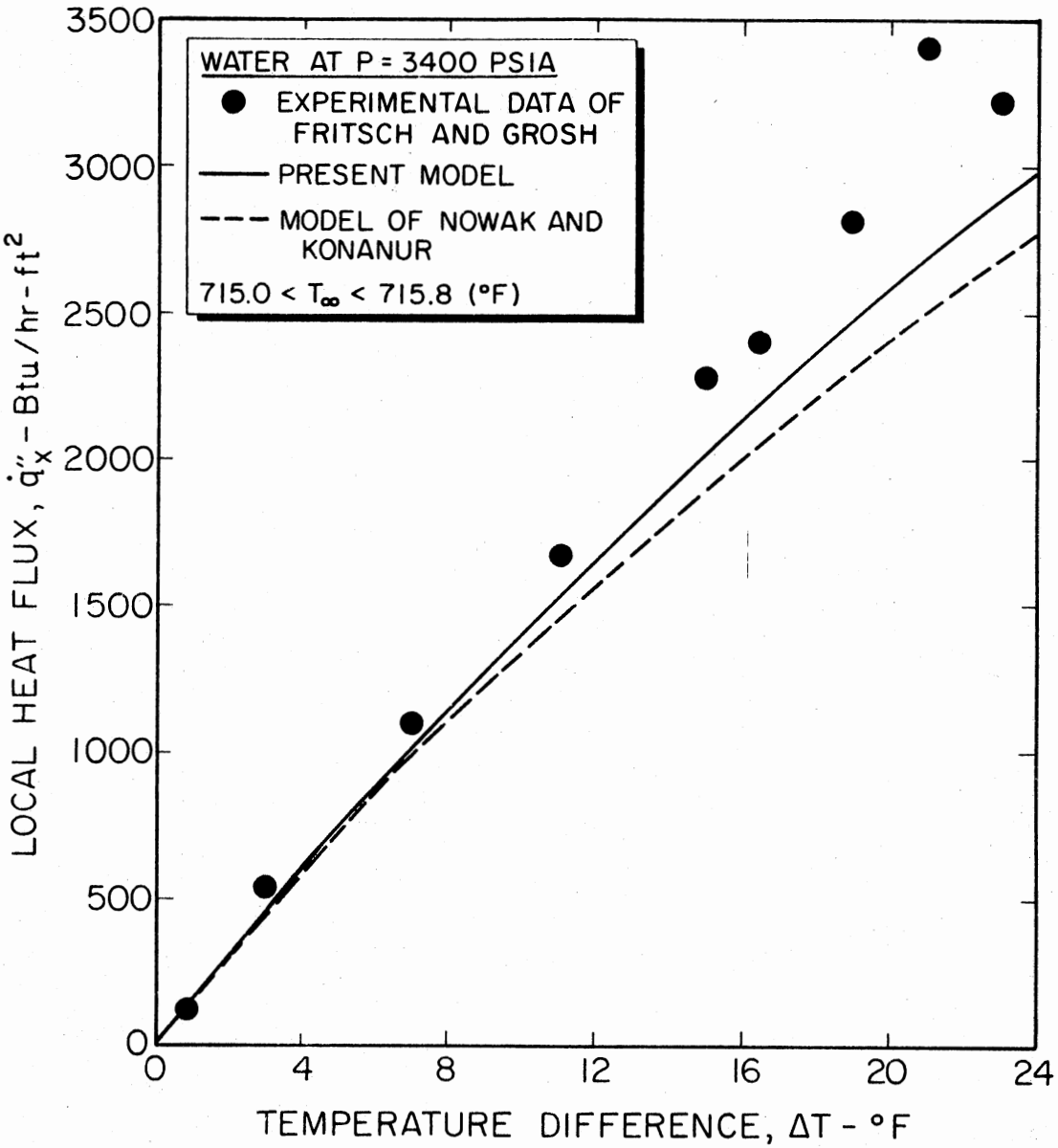

Figure 13. Comparison of Present Model With Model of Nowak and Konanur (7) (Experimental Data From Fritsch and Grosh (12))

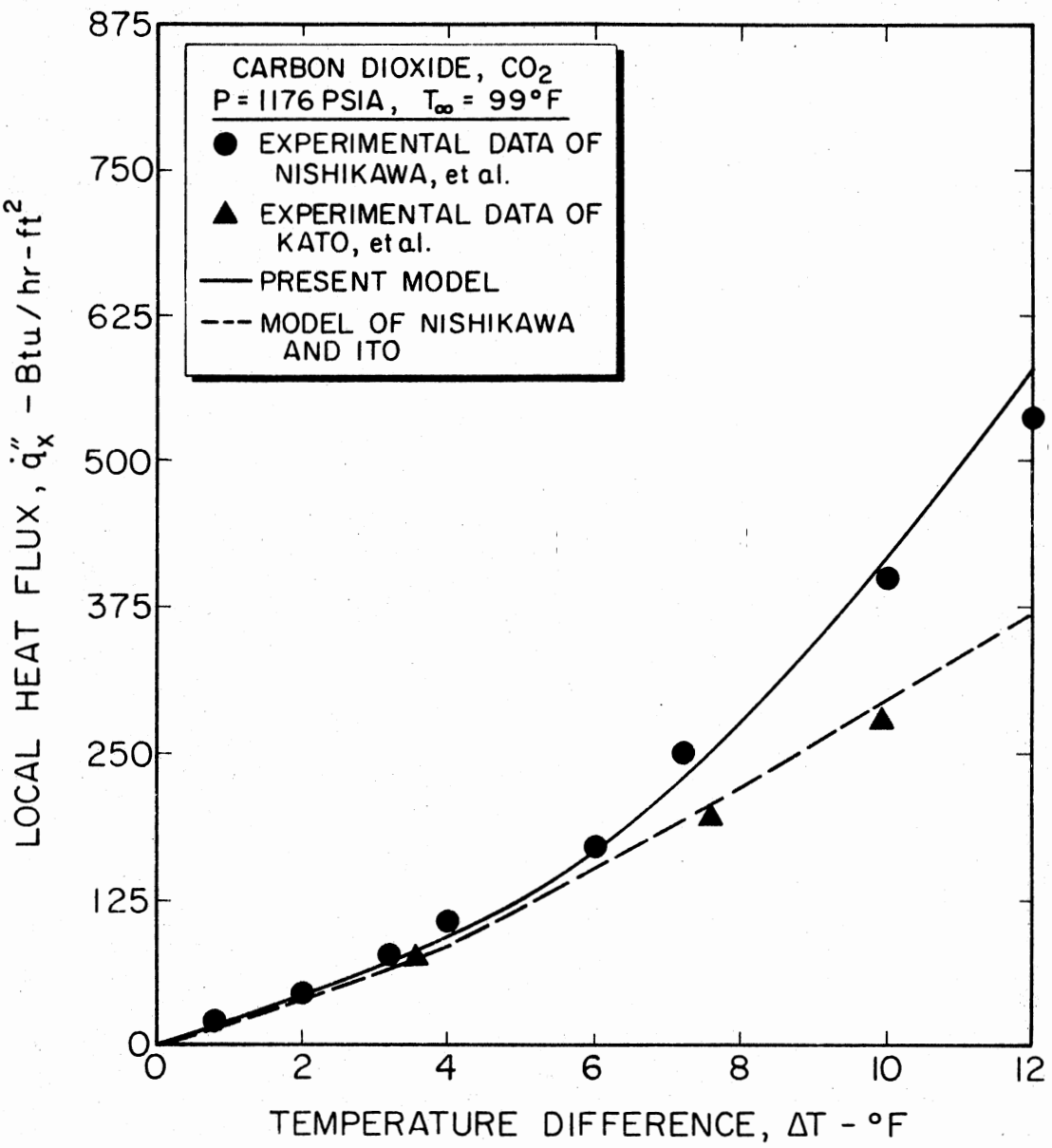

Figure 14. Comparison of Present Model With Experimental Data of Nishikawa et al. (30) and Kato et al. (29), and Model of Nishikawa and Ito (6)

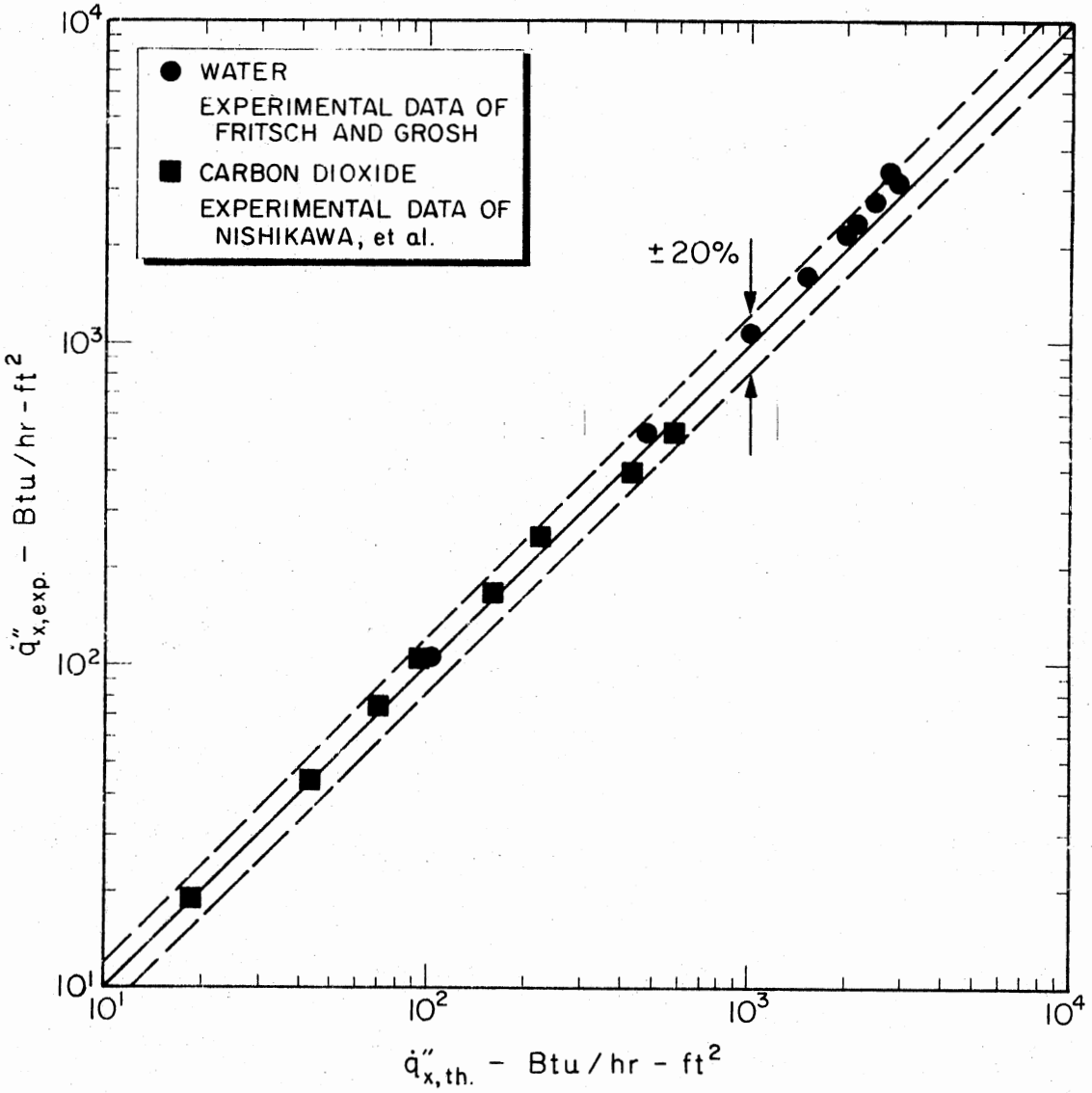

Figure 15. Comparison of Predicted Local Heat Flux Using the Present Model With Measurements

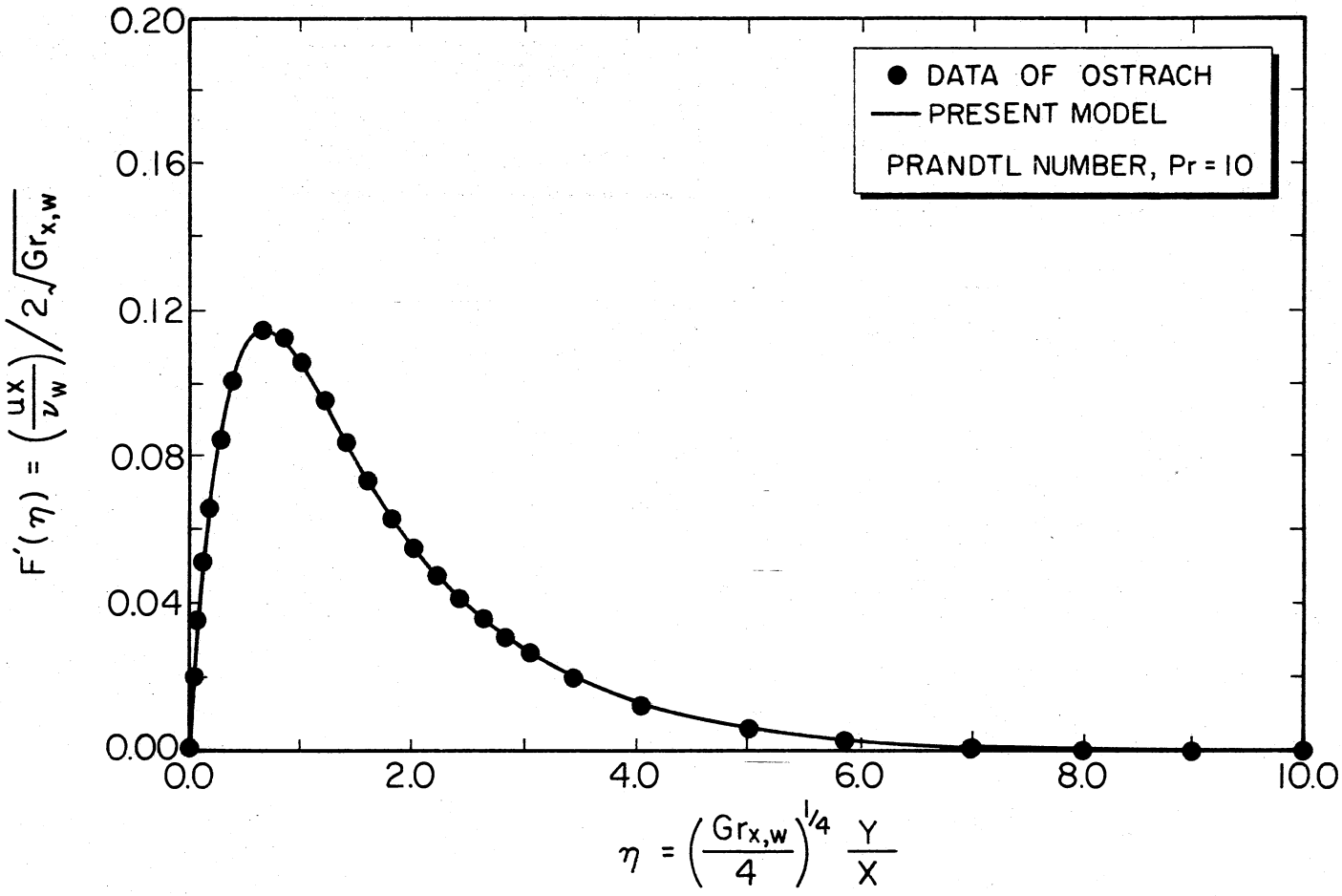

Figure 16. Comparison of Predicted Velocity Profile With Data of Ostrach (10), for the Constant Property Case

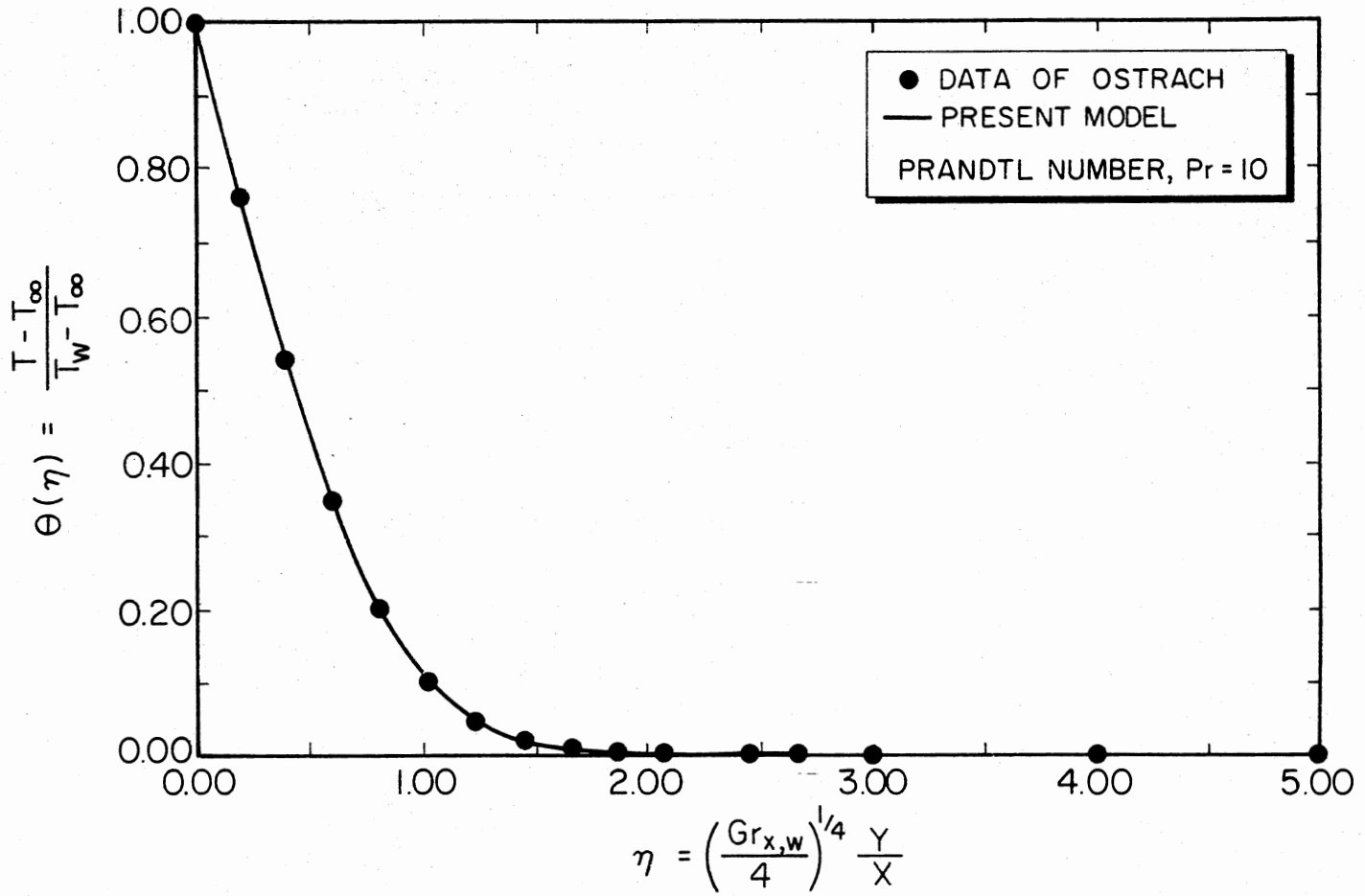

Figure 17. Comparison of Predicted Dimensionless Temperature Profile With Data of Ostrach (10), for the Constant Property Case

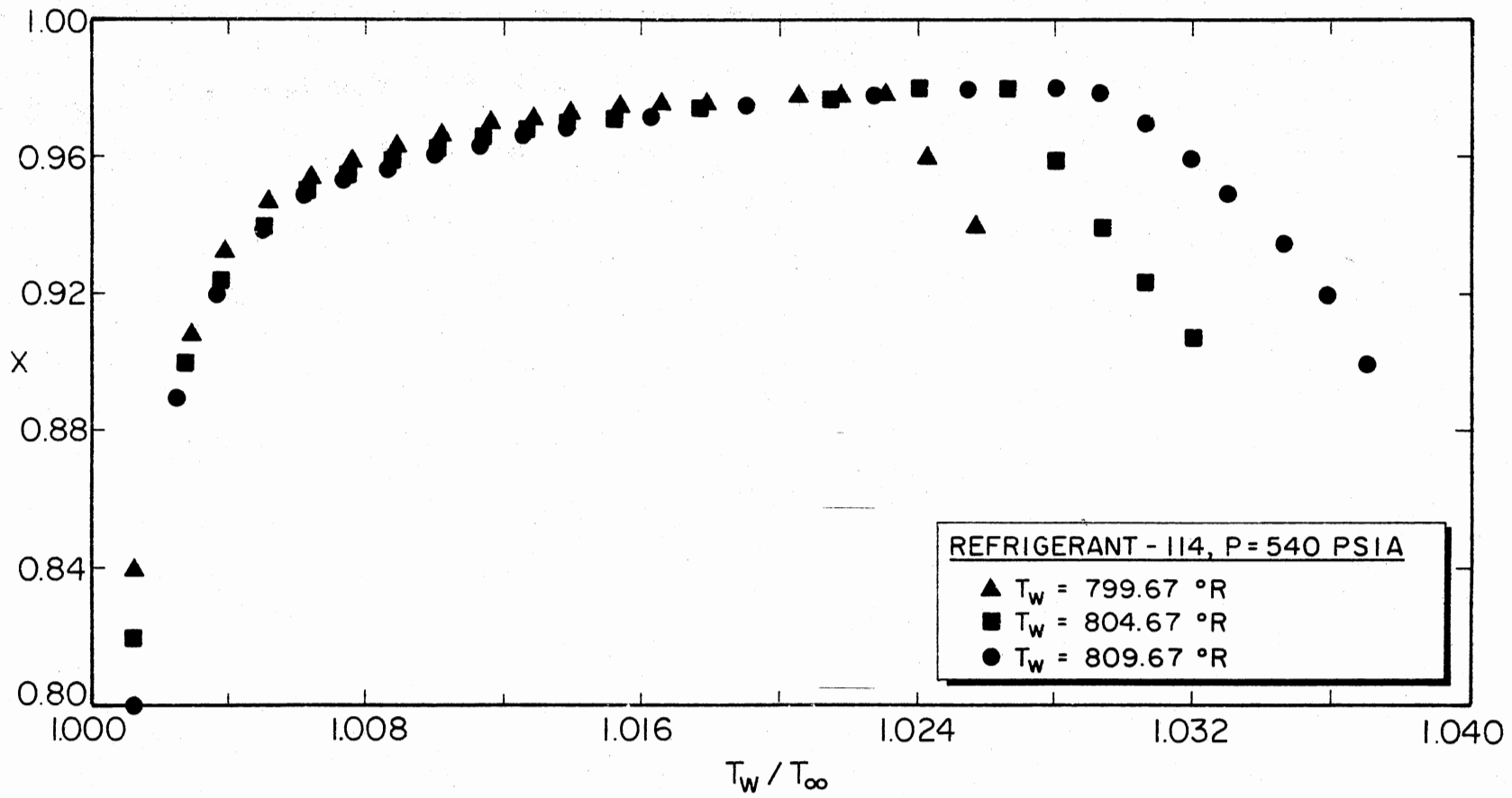

Figure 18. Variation of $x(T_x = T_w - x(T_w - T_\infty))$ With Wall Temperature and T_w/T_∞

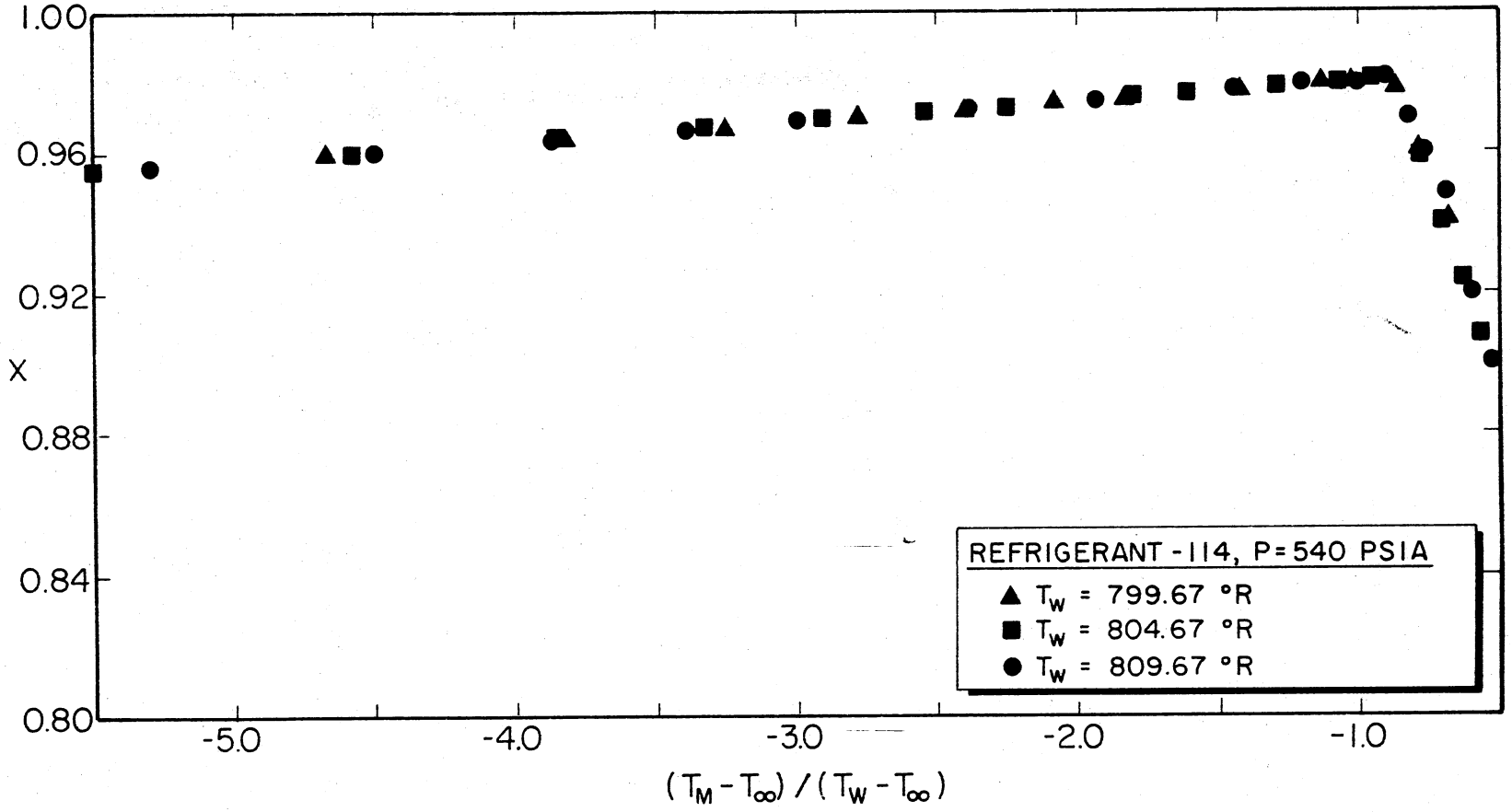

Figure 19. Variation of $x(T_X = T_W - x(T_W - T_\infty))$ With $(T_M - T_\infty)/(T_W - T_\infty)$ at Various Temperatures and a Single Pressure

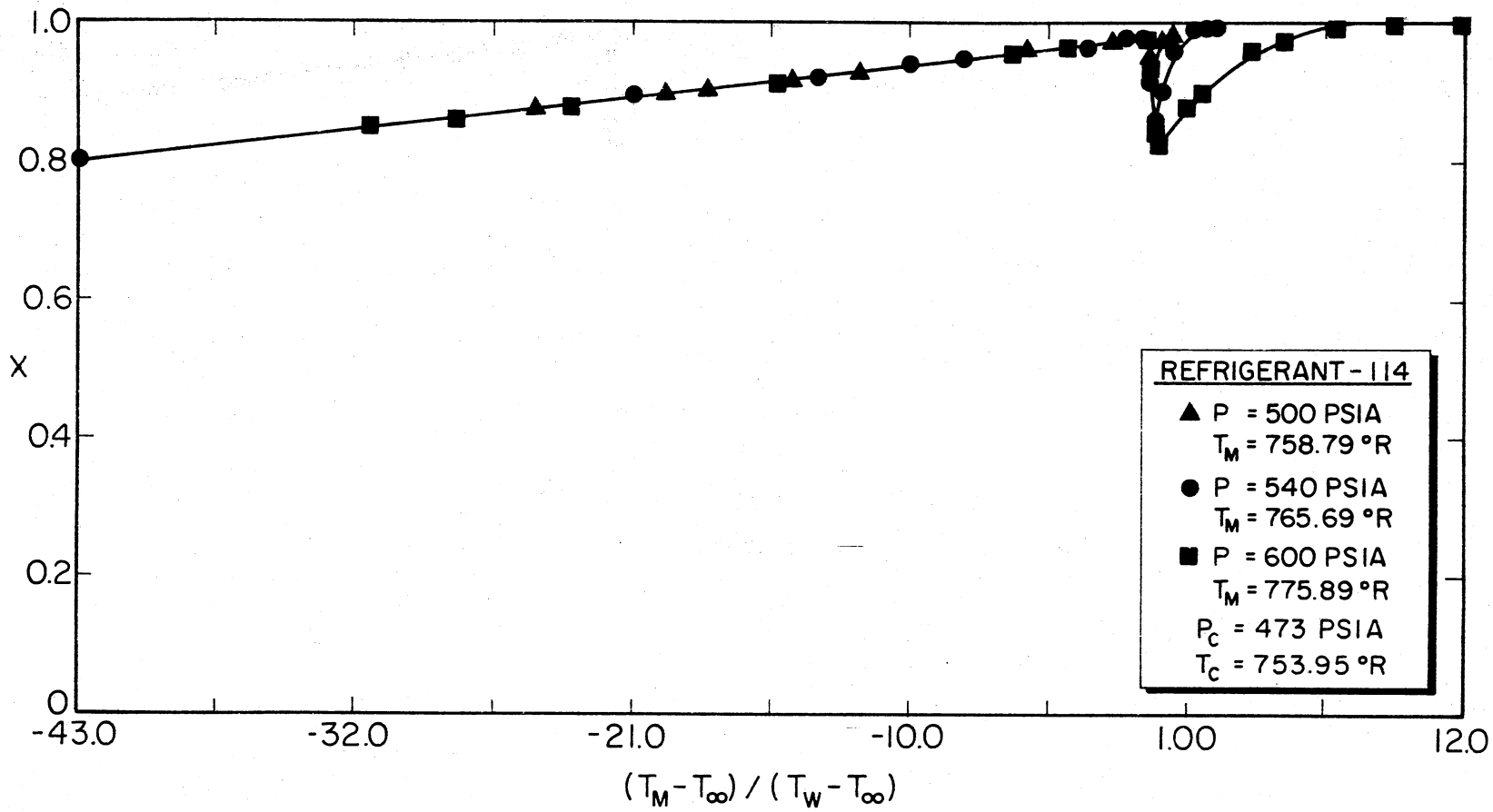

Figure 20. Variation of $x(T_x = T_w - x(T_w - T_\infty))$ With $(T_M - T_\infty)/(T_w - T_\infty)$ at Various Temperatures and Pressures

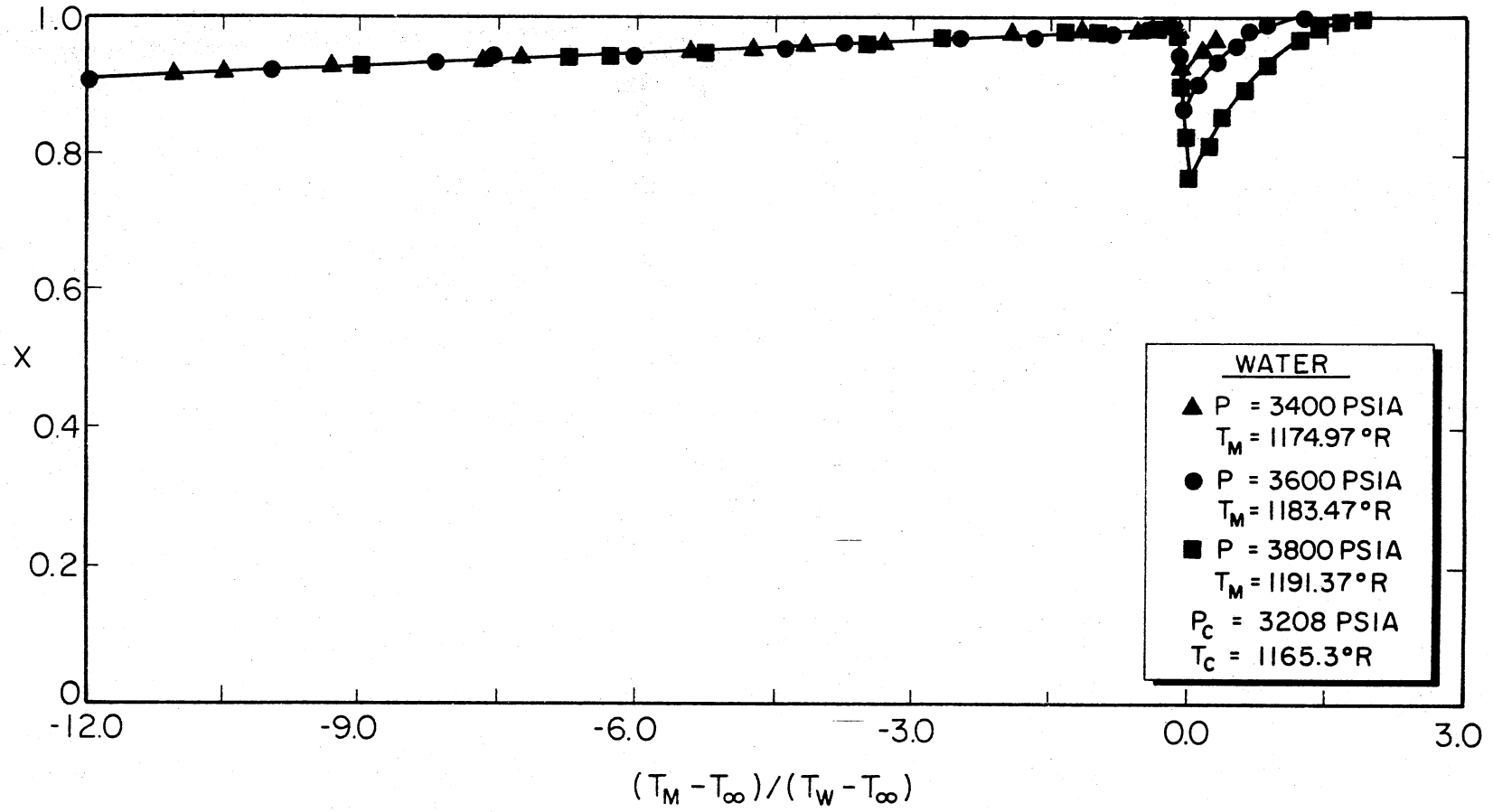

Figure 21. Variation of $x(T_x = T_w - x(T_w - T_\infty))$ With $(T_M - T_\infty)/(T_w - T_\infty)$ at Various Temperatures and Pressures

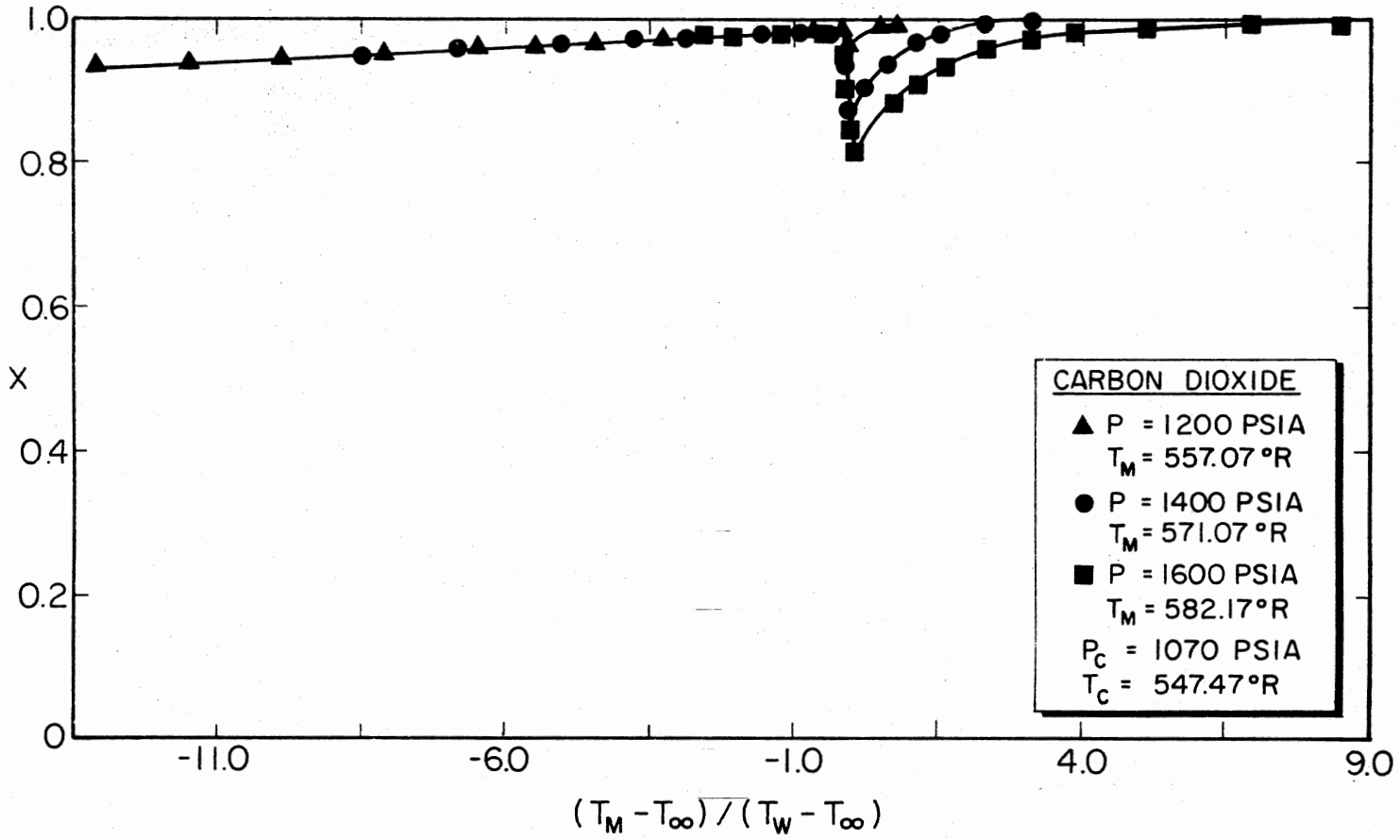

Figure 22. Variation of $x(T_x = T_w - x(T_w - T_\infty))$ With $(T_M - T_\infty)/(T_w - T_\infty)$ at Various Temperatures and Pressures

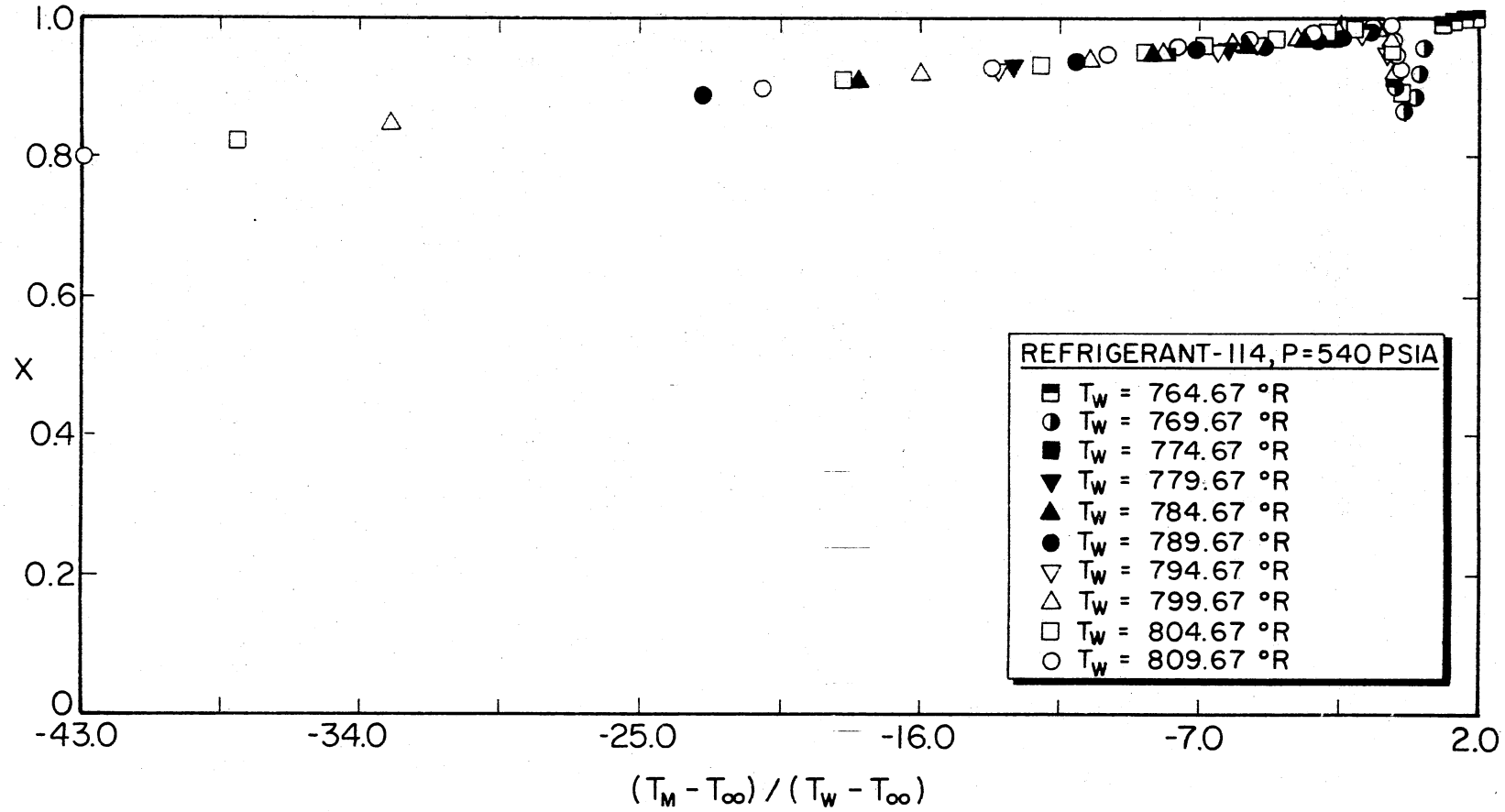

Figure 23. Variation of $x(T_x = T_W - x(T_W - T_\infty))$ With $(T_M - T_\infty)/(T_W - T_\infty)$ at Various Temperatures and a Single Pressure

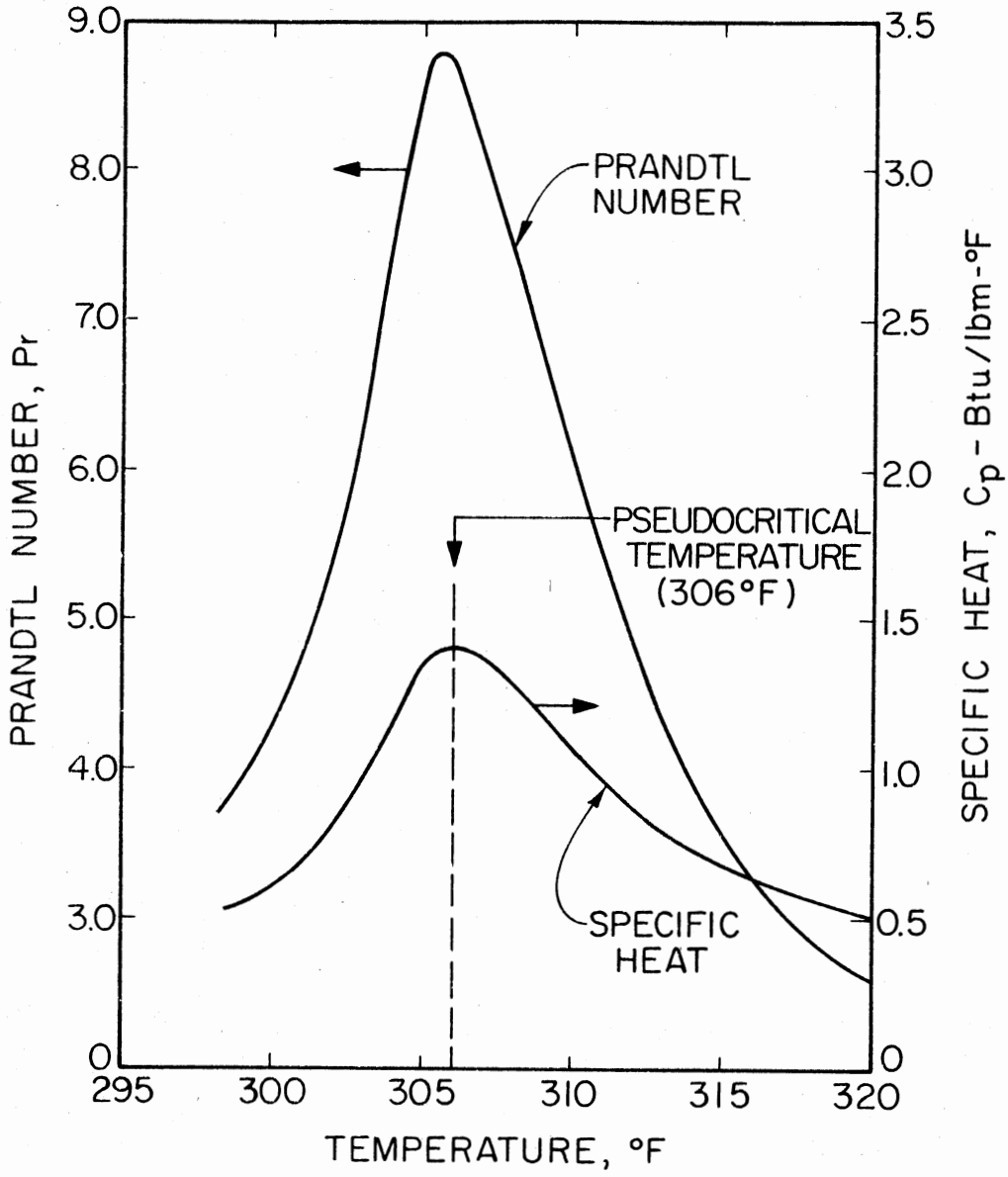

Figure 24. Thermodynamic Properties of Refrigerant-114 in the Supercritical Region at 540 psia

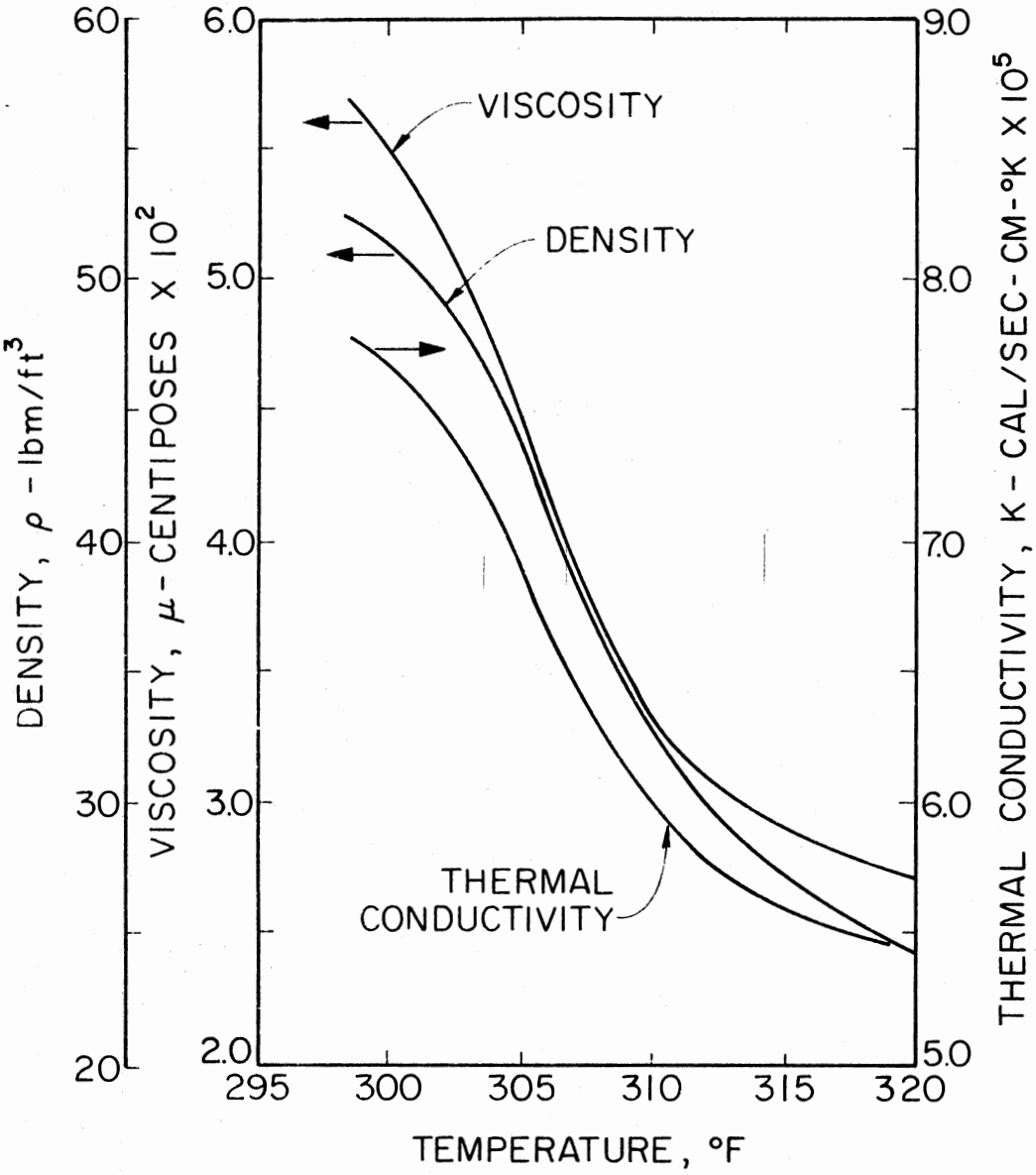

Figure 25. Physical Properties of Refrigerant-114 in the Supercritical Region at 540 psia

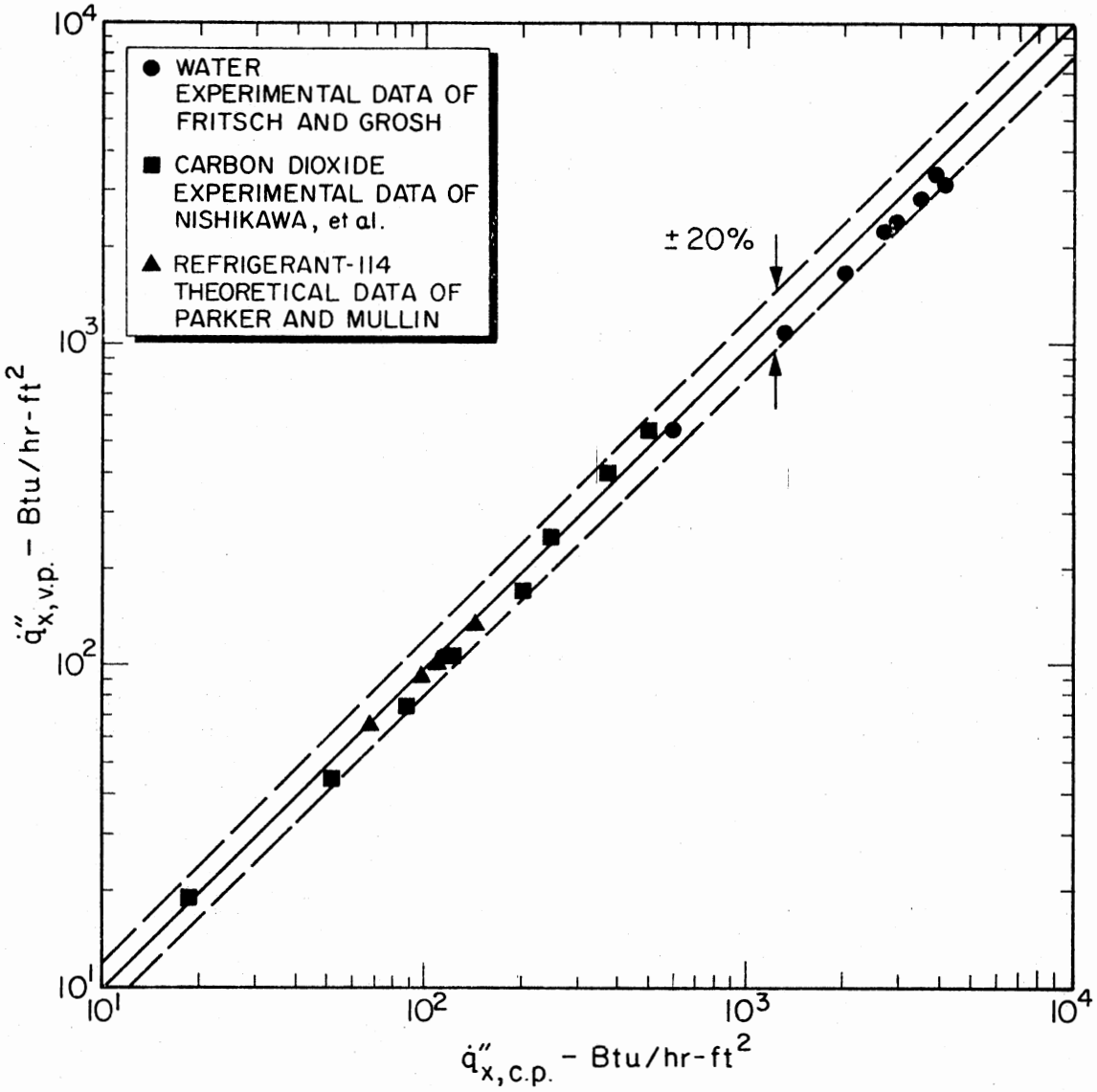

Figure 26. Comparison of Predicted Constant Property Local Heat Flux With Variable Property Local Heat Flux

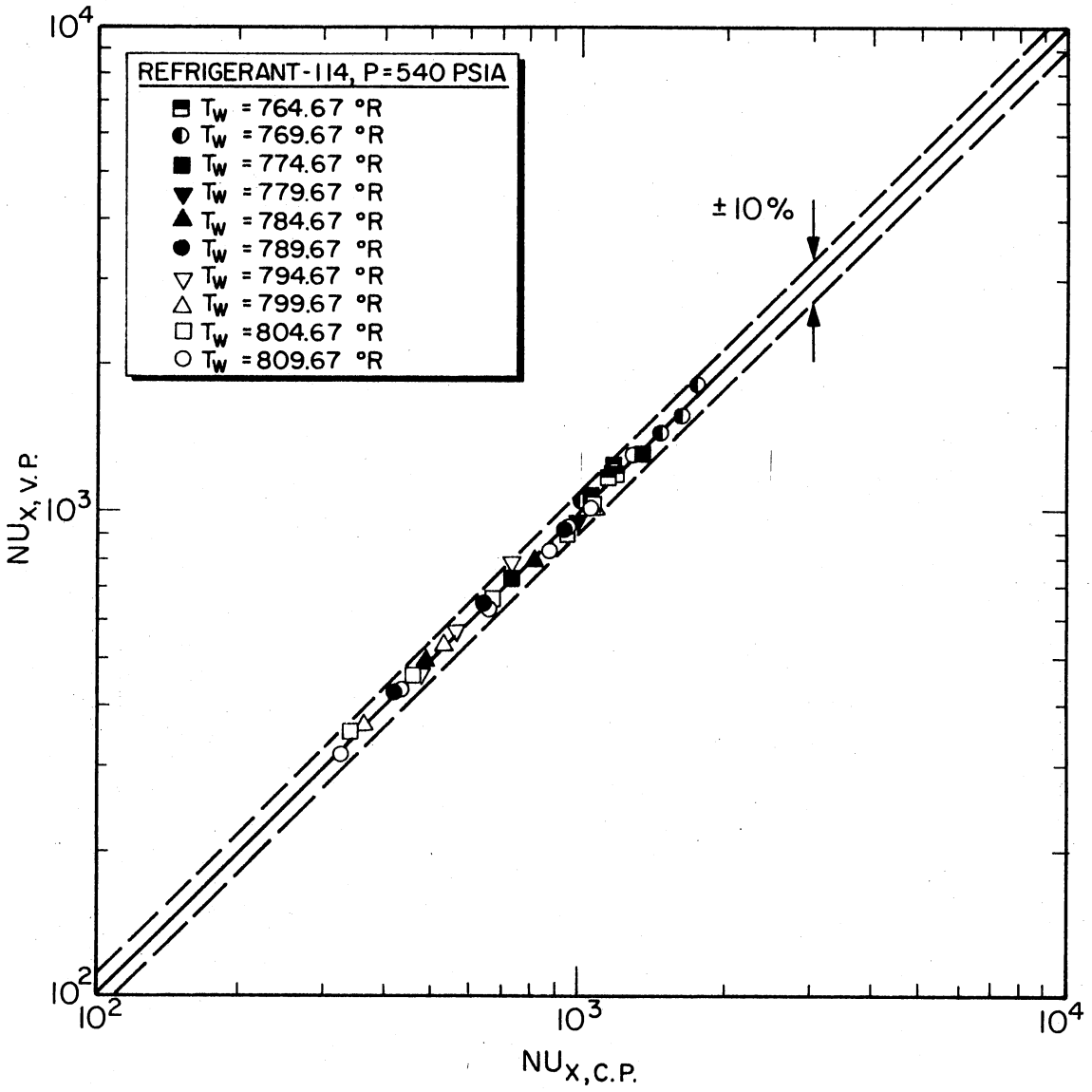

Figure 27. Comparison of Predicted Constant Property Nusselt Numbers With Variable Property Nusselt Numbers

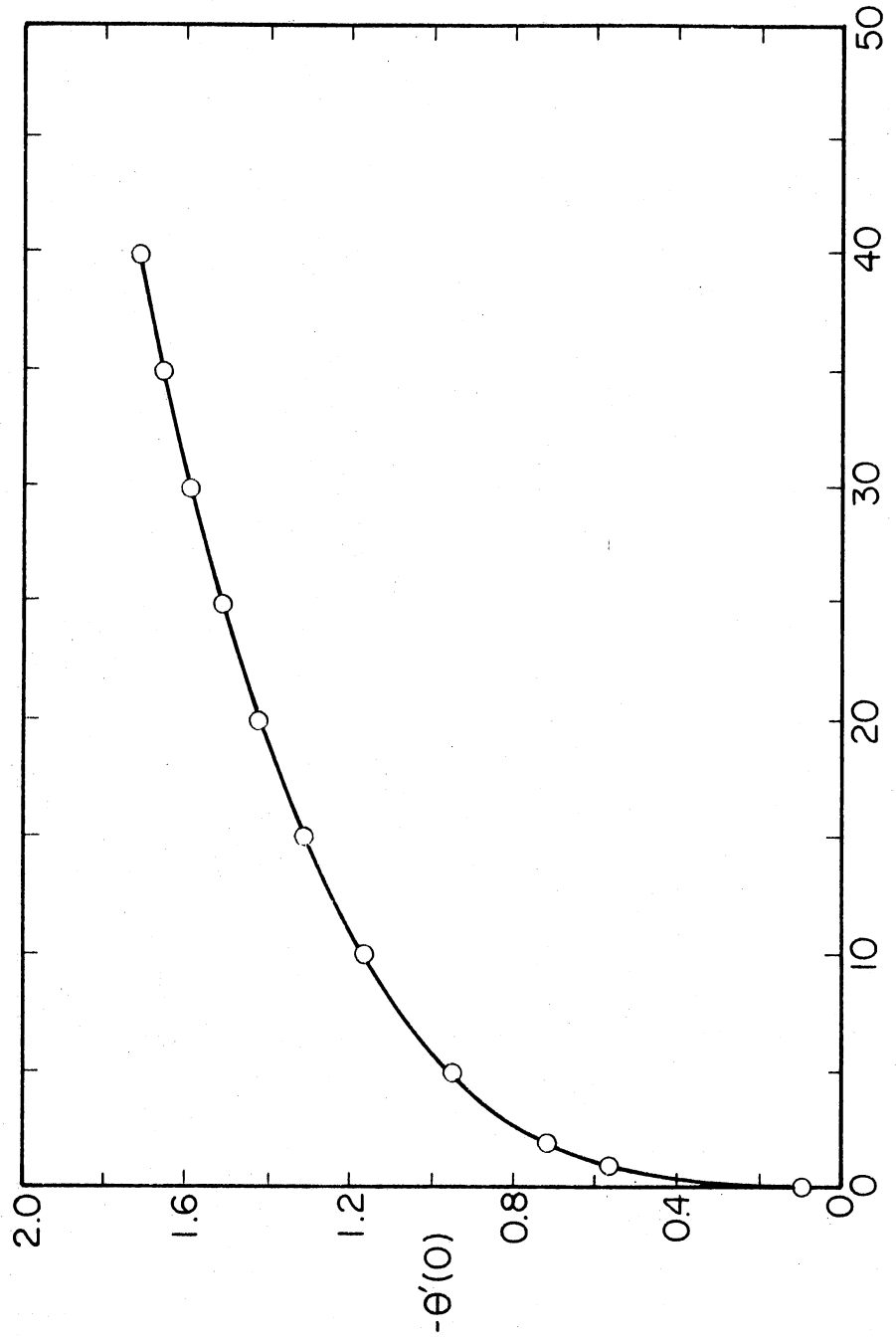

Figure 28. Constant Property Solution

APPENDIX C

EQUATIONS AND DERIVATIONS

Derivation of Equations (3.14) and (3.15)

This section outlines the method by which Equations (3.14) and (3.15) were derived from Equations (3.9) and (3.10). In order to show how this can be accomplished the following is presented:

1. Define the function

$$A = \frac{\rho\mu}{\rho_w \mu_w} = f(\eta) = f(P,T) \tag{C.1}$$

2. The derivative of A with respect to η is

$$\frac{dA}{d\eta} = \frac{\mu}{\rho_w \mu_w}\frac{d\rho}{d\eta} + \frac{\rho}{\rho_w \mu_w}\frac{d\mu}{d\eta} \tag{C.2}$$

3. Note that

$$\rho = \rho(P,T) = f(\eta) \tag{C.3}$$

$$\frac{d\rho}{d\eta} = \left(\frac{\partial\rho}{\partial T}\right)_P \frac{dT}{d\eta} + \left(\frac{\partial\rho}{\partial P}\right)_T \frac{dP}{d\eta} \tag{C.4}$$

4. Since pressure is held constant, this reduces to

$$\frac{d\rho}{d\eta} = \left(\frac{\partial\rho}{\partial T}\right)_P \frac{dT}{d\eta} \tag{C.5}$$

5. The definition of the dimensionless temperature

$$\theta(\eta) = \frac{T - T_\infty}{T_w - T_\infty} \tag{C.6}$$

gives

$$\frac{dT}{d\eta} = (T_w - T_\infty)\,\theta'(\eta) = \Delta T\,\theta'(\eta) \tag{C.7}$$

6. The substitution of Equation (C.7) into (C.5) gives

$$\frac{d\rho}{d\eta} = \left(\frac{\partial\rho}{\partial T}\right)_P \Delta T\ \theta'(\eta) \tag{C.8}$$

7. The same reasoning as above when applied to the viscosity gives

$$\frac{d\mu}{d\eta} = \left(\frac{\partial\mu}{\partial T}\right)_P \Delta T\ \theta'(\eta) \tag{C.9}$$

8. The substitution of Equations (C.8) and (C.9) into Equation (C.2) yields

$$\frac{dA}{d\eta} = \frac{\mu}{\rho_w\ \mu_w}\left(\frac{\partial\rho}{\partial T}\right)_P \Delta T\ \theta'(\eta) + \frac{\rho}{\rho_w\ \mu_w}\left(\frac{\partial\mu}{\partial T}\right) \Delta T\ \theta'(\eta) \tag{C.10}$$

9. If an identical line of reasoning as the above is applied to the term

$$B = \frac{\rho K}{\rho_w\ K_w} = f(\eta) = f(P,T) \tag{C.11}$$

the following expression is obtained:

$$\frac{dB}{d\eta} = \frac{K}{\rho_w\ K_w}\left(\frac{\partial\rho}{\partial T}\right)_P \Delta T\ \theta'(\eta) + \frac{\rho}{\rho_w\ \mu_w}\left(\frac{\partial K}{\partial T}\right)_P \Delta T\ \theta'(\eta) \tag{C.12}$$

10. The substitution of Equations (C.10) and (C.12) into the reduced differential Equations (3.9) and (3.10) yields Equations (3.14) and (3.15).

Derivation of Equation (4.11)

This section outlines the method by which Equation (4.11) was obtained based on the equation of state, thermodynamic potential functions, and Maxwell equations. In order to show how this can be accomplished the following is presented:

1. Start with a functional expression for the specific heat at constant volume in terms of temperature and volume, and then form its derivative:

$$C_V = C_V(T,V) \tag{C.13}$$

$$dC_V = \left(\frac{\partial C_V}{\partial T}\right)_V dT + \left(\frac{\partial C_V}{\partial V}\right)_T dV \tag{C.14}$$

2. Using the definition of C_V and the thermodynamic potential function $(\partial u/\partial s)_V = T$, an expression for the second coefficient of Equation (C.14), $(\partial C_V/\partial V)_T$, was obtained as follows:

$$C_V = \left(\frac{\partial u}{\partial T}\right)_V = \left(\frac{\partial u}{\partial s}\right)_V \left(\frac{\partial s}{\partial T}\right)_V = T\left(\frac{\partial s}{\partial T}\right)_V \tag{C.15}$$

The derivative of Equation (C.15) was taken with respect to the volume while holding the temperature constant; the result was

$$\left(\frac{\partial C_V}{\partial V}\right)_T = T\left(\frac{\partial^2 s}{\partial V_T \, \partial T_V}\right) \tag{C.16}$$

3. Start with the following Maxwell equation

$$\left(\frac{\partial P}{\partial T}\right)_V = \left(\frac{\partial s}{\partial V}\right)_T \tag{C.17}$$

The derivative of the Maxwell Equation (C.17) with respect to volume while holding the temperature constant was obtained as follows:

$$\left(\frac{\partial^2 P}{\partial T^2}\right)_V = \frac{\partial^2 s}{\partial V_T \, \partial T_V} \tag{C.18}$$

4. Equating Equations (C.16) and (C.18) gave

$$\left(\frac{\partial C_V}{\partial V}\right)_T = T\left(\frac{\partial^2 P}{\partial T^2}\right)_V \tag{C.19}$$

5. The substitution of Equation (C.19) into Equation (C.14) and assuming that the temperature was constant gave:

$$dC_V = T\left(\frac{\partial^2 P}{\partial T^2}\right)_V dV \tag{C.20}$$

6. The second derivative of the equation of state, Equation (4.1), with respect to the temperature while holding the volume constant resulted in

$$\left(\frac{\partial^2 P}{\partial T^2}\right)_V = \left(\frac{K}{T_c}\right)^2 EXP(-K\ T_r)\ \left[\frac{C_2}{(V-b)^2} + \frac{C_3}{(V-b)^3} + \frac{C_5}{(V-b)^5}\right] \tag{C.21}$$

7. The substitution of Equation (C.21) into Equation (C.20) gave

$$\int_{C_{V_o}}^{C_V} dC_V = \int_{V_o=\infty}^{V} T\left(\frac{K}{T_c}\right)^2 EXP(-K\ T_r)\ \left[\frac{C_2}{(V-b)^2} + \frac{C_3}{(V-b)^3} + \frac{C_5}{(V-b)^5}\right] dV_T \tag{C.22}$$

8. The integration of Equation (C.22) gave

$$C_V = C_{V_o} - T\left(\frac{K}{T_c}\right)^2 EXP(-K\ T_r)\ \left[\frac{C_2}{(V-b)} + \frac{C_3}{2(V-b)^2} + \frac{C_5}{4(V-b)^4}\right] \tag{C.23}$$

Equation (C.23), which is the same as Equation (4.11), was used to calculate the effects of pressure change on the specific heat at constant volume for a given temperature. This equation along with Equation (4.9)

was used to evaluate specific heat at constant pressure from Equation (4.10).

Coefficients of Equations (5.7) Through (5.12)

The expressions given below represent the coefficients of the dimensionless terms A1, 3(A2), 2(A2), A3, B1, and 3(B2) of Equations (5.7) through (5.12).

Coefficients for the Term A1

$$A1 = CA + CB\,\theta + CC\,\theta^2 + CD\,\theta^3 \tag{5.7}$$

where

$$CA = (A_{11} + B_{11}\,T_\infty + C_{11}\,T_\infty^2 + D_{11}\,T_\infty^3)(\Delta T)$$

$$CB = (B_{11} + 2C_{11}\,T_\infty + 3D_{11}\,T_\infty^2)(\Delta T)^2$$

$$CC = (C_{11} + 3D_{11}\,T_\infty)(\Delta T)^3$$

$$CD = (D_{11})(\Delta T)^4$$

Coefficients for the Term 3(A2)

$$3(A2) = CE + CF\,\theta + CG\,\theta^2 + CH\,\theta^3 \tag{5.8}$$

where

$$CE = 3(A_{22} + B_{22}\,T_\infty + C_{22}\,T_\infty^2 + D_{22}\,T_\infty^3)(\rho_w\,\mu_w)$$

$$CF = 3(B_{22} + 2C_{22}\,T_\infty + 3D_{22}\,T_\infty^2)(\Delta T)(\rho_w\,\mu_w)$$

$$CG = 3(C_{22} + 3D_{22}\,T_\infty)(\Delta T)^2\,(\rho_w\,\mu_w)$$

$$CH = 3(D_{22})(\Delta T)^3\,(\rho_w\,\mu_w)$$

Coefficients for the Term 2(A2)

$$2(A2) = CI + CJ\ \theta + CK\ \theta^2 + CL\ \theta^3 \tag{5.9}$$

where

$$CI = 2(A_{22} + B_{22}\ T_\infty + C_{22}\ T_\infty^2 + D_{22}\ T_\infty^3)(\rho_w\ \mu_w)$$

$$CJ = 2(B_{22} + 2C_{22}\ T_\infty + 3D_{22}\ T_\infty^2)(\Delta T)(\rho_w\ \mu_w)$$

$$CK = 2(C_{22} + 3D_{22}\ T_\infty)(\Delta T)^2\ (\rho_w\ \mu_w)$$

$$CL = 2(D_{22})(\Delta T)^3\ (\rho_w\ \mu_w)$$

Coefficients for the Term A3

$$A3 = (CM + CN\ \theta + CP\ \theta^2 + CR\ \theta^3) - (CS + CT\ \theta + CU\ \theta^2 + CW\ \theta^3) \tag{5.10}$$

where

$$CM = (A_{33} + B_{33}\ T_\infty + C_{33}\ T_\infty^2 + D_{33}\ T_\infty^3)\ \left(\frac{\rho_w^2\ \rho_\infty\ \mu_w}{\rho_\infty - \rho_w}\right)$$

$$CN = (B_{33} + 2C_{33}\ T_\infty + 3D_{33}\ T_\infty^2)\ \left(\frac{\rho_w^2\ \rho_\infty\ \mu_w}{\rho_\infty - \rho_w}\right)\ (\Delta T)$$

$$CP = (C_{33} + 3D_{33}\ T_\infty)\ \left(\frac{\rho_w^2\ \rho_\infty\ \mu_w}{\rho_\infty - \rho_w}\right)\ (\Delta T)^2$$

$$CR = (D_3)\ \left(\frac{\rho_w^2\ \rho_\infty\ \mu_w}{\rho_\infty - \rho_w}\right)\ (\Delta T)^3$$

$$CS = (A_{22} + B_{22}\ T_\infty + C_{22}\ T_\infty^2 + D_{22}\ T_\infty^3)\ \left(\frac{\rho_w^2\ \mu_w}{\rho_\infty - \rho_w}\right)$$

$$CT = (B_{22} + 2C_{22}\, T_\infty + 3D_{22}\, T_\infty^2)\ \left(\frac{\rho_w^2\, \mu_w}{\rho_\infty - \rho_w}\right)\ (\Delta T)$$

$$CU = (C_{22} + 3D_{22}\, T_\infty)\ \left(\frac{\rho_w^2\, \mu_w}{\rho_\infty - \rho_w}\right)\ (\Delta T)^2$$

$$CW = (D_{22})\ \left(\frac{\rho_w^2\, \mu_w}{\rho_\infty - \mu_w}\right)\ (\Delta T)^3$$

Coefficients for the Term B1

$$B1 = CAA + CBB\ \theta + CCC\ \theta^2 + CDD\ \theta^3 \qquad (5.11)$$

where

$$CAA = (A_{44} + B_{44}\, T_\infty + C_{44}\, T_\infty^2 + D_{44}\, T_\infty^3)(\Delta T)$$

$$CBB = (B_{44} + 2C_{44}\, T_\infty + 3D_{44}\, T_\infty^2)(\Delta T)^2$$

$$CCC = (C_{44} + 3D_{44}\, T_\infty)(\Delta T)^3$$

$$CDD = (D_{44})(\Delta T)^4$$

Coefficients for the Term 3(B2)

$$3(B2) = CEE + CFF\ \theta + CGG\ \theta^2 + CHH\ \theta^3 \qquad (5.12)$$

where

$$CEE = 3(A_{55} + B_{55}\, T_\infty + C_{55}\, T_\infty^2 + D_{55}\, T_\infty^3)(\rho_w\, \mu_w)$$

$$CFF = 3(B_{55} + 2C_{55}\, T_\infty + 3D_{55}\, T_\infty^2)(\Delta T)(\rho_w\, \mu_w)$$

$$CGG = 3(C_{55} + 3D_{55}\, T_\infty)(\Delta T)^2\ (\rho_w\, \mu_w)$$

$$CHH = 3(D_{55})(\Delta T)^3\ (\rho_w\, \mu_w)$$

The terms A1, 3(A2), 2(A2), A3, B1, and 3(B2) expressed by Equations (5.7) through (5.12) are the dimensionless coefficients of the reduced ordinary nonlinear differential equations, Equations (3.14) and (3.15).

Jacobian of the First Order Differential Equations

The Jacobians of the first order differential equations, Equation (5.16), are presented here. These Jacobians were presented in the Subroutine JACOB2, which was used in the Lentini-Pereyra Program. The Jacobians of Equation (5.16) were obtained according to

$$JACOB(I,J) = \frac{\partial(x_I)}{\partial(y_J)}$$

Based on the above equation, the following Jacobians were obtained:

$$JACOB(1,2) = 1$$

$$JACOB(2,3) = 1$$

$$JACOB(3,1) = -(CE + CF\ G_4 + CG\ G_4^2 + CH\ G_4^3)\ G_3$$

$$JACOB(3,3) = 2(CI + CJ\ G_4 + CK\ G_4^2 + CL\ G_4^3)\ G_2$$

$$JACOB(3,3) = -(CA + CB\ G_4 + CC\ G_4^2 + CD\ G_4^3)\ G_5$$
$$- (CE + CF\ G_4 + CG\ G_4^2 + CH\ G_4^3)\ G_1$$

$$JACOB(3,4) = -(CB + 2CC\ G_4 + 3CD\ G_4^2)\ G_5 G_3$$
$$- (CF + 2CG\ G_4 + 3CH\ G_4^2)\ G_1 G_3$$
$$+ (CJ + 2CK\ G_4 + 3CL\ G_4^2)\ (G_2)^2$$

$$- (CN + 2CP\ G_4 + 3CR\ G_4^2)$$

$$+ (CT + 2CU\ G_4 + 3CW\ G_4^2)$$

$$JACOB(3,5) = -(CA + CB\ G_4 + CC\ G_4^2 + CD\ G_4^3)\ G_3$$

$$JACOB(4,5) = 1$$

$$JACOB(5,1) = -(CEE + CFF\ G_4 + CGG\ G_4^2 + CHH\ G_4^3)\ G_5$$

$$JACOB(5,4) = -(CBB + 2CCC\ G_4 + 3CDD\ G_4^2)\ (G_5)^2$$

$$- (CFF + 2CGG\ G_4 + 3CHH\ G_4^2)\ G_1 G_5$$

$$JACOB(5,5) = -2(CAA + CBB\ G_4 + CCC\ G_4^2 + CDD\ G_4^3)\ G_5$$

$$- (CEE + CFF\ G_4 + CGG\ G_4^2 + CW\ G_4^3)\ G_1$$

The Jacobians that are not presented by the above equations were zero.

Constant Property Equations for the Lentini-Pereyra Program

The development of the set of first order differential equations and their Jacobians are presented here. These equations were obtained from the constant property reduced differential Equations (3.12) and (3.13).

To obtain the set of first order differential equations, the following procedures were used:

1. Define:

$$G_1 = F \qquad \text{(C.24a)}$$

$$G_2 = F' \qquad \text{(C.24b)}$$

$$G_3 = F'' \qquad \text{(C.24c)}$$

$$G_4 = \theta \tag{C.24d}$$

$$G_5 = \theta' \tag{C.24e}$$

2. The set of differential equations, Equations (3.12) and (3.13), become:

$$Y_1 = \frac{dG_1}{d\eta} = G_2 \tag{C.25a}$$

$$Y_2 = \frac{dG_2}{d\eta} = G_3 \tag{C.25b}$$

$$Y_3 = \frac{dG_3}{d\eta} = -3G_1G_3 + 2(G_2)^2 - G_4 \tag{C.25c}$$

$$Y_4 = \frac{dG_4}{d\eta} = G_5 \tag{C.25d}$$

$$Y_5 = \frac{dG_5}{d\eta} = -3Pr\ G_1G_5 \tag{C.25e}$$

The boundary conditions, Equation (3.11), transform to

$$\left.\begin{aligned} G_1 &= 0 \\ G_2 &= 0 \\ G_3 &= \text{Unknown} \\ G_4 &= 1 \\ G_5 &= \text{Unknown} \end{aligned}\right\} \eta = 0 \qquad \left.\begin{aligned} G_1 &= \text{Unknown} \\ G_2 &= 0 \\ G_3 &= \text{Unknown} \\ G_4 &= 0 \\ G_5 &= \text{Unknown} \end{aligned}\right\} \eta = \infty \tag{C.26}$$

Equation (C.25) along with the boundary conditions, Equation (C.26), are the set of first order nonlinear differential equations of the reduced momentum and energy equations for the constant property case to be solved on the computer.

The Jacobians of the first order differential equations, Equation (C.25), were obtained according to

$$JACOB(I,J) = \frac{\partial(x_I)}{\partial(y_J)}$$

Based on the above equation, the following Jacobians were obtained:

$$JACOB(1,2) = 1 \tag{C.27a}$$

$$JACOB(2,3) = 1 \tag{C.27b}$$

$$JACOB(3,1) = -3G_3 \tag{C.27c}$$

$$JACOB(3,2) = 4G_2 \tag{C.27d}$$

$$JACOB(3,3) = -3G_1 \tag{C.27e}$$

$$JACOB(3,4) = -1 \tag{C.27f}$$

$$JACOB(4,5) = 1 \tag{C.27g}$$

$$JACOB(5,1) = -3PrG_5 \tag{C.27h}$$

$$JACOB(5,5) = -3PrG_1 \tag{C.27i}$$

The Jacobians that are not presented by Equation (C.27) were zero.

APPENDIX D

COMPUTER PROGRAMS

```
                SUBROUTINE PROP                                          PROP0010
C                                                                        PROP0020
C              ********************                                      PROP0030
C              * SUBROUTINE PROP   *                                     PROP0040
C              ********************                                      PROP0050
C                                                                        PROP0060
C     SUBJECT :                                                          PROP0070
C               THERMODYNAMIC AND TRANSPORT PROPERTIES OF                PROP0080
C               WATER IN THE SUPERCRITICAL REGION.                       PROP0090
C                                                                        PROP0100
C     DATA INPUT :                                                       PROP0110
C               P=PRESSURE(PSIA)                                         PROP0120
C               T=TEMPERATURE(DEG.R)                                     PROP0130
C               V=SPECIFIC VOLUME (CUFT/LBM)--INITIAL VALUE              PROP0140
C               DTEMP = TEMP INCREMENT (DEG.R)                           PROP0150
C               NDOLP = MAX NO. OF DOLOOP ITERATION                      PROP0160
C                                                                        PROP0170
C     LIMITATIONS :                                                      PROP0180
C               REDUCED TEMPERATURE RANGE ( TR = T/TC )                  PROP0190
C               FROM 1.008 TO 1.04 ( WHERE TC = 1165.3 DEG R ).          PROP0200
C               REDUCED PRESSURE RANGE ( PR = P/PC )                     PROP0210
C               FROM 1.044 TO 1.247 ( WHERE PC = 3208 PSIA ).            PROP0220
C                                                                        PROP0230
      IMPLICIT REAL*8(A-H,O-Z)                                           PROP0240
      COMMON/CPROP/DENS(50),CPH(50),VISC(50),COND(50),TEMP(50),DTEMP     PROP0250
      COMMON/TERM/AA1(50),AA2(50),AA3(50),BB1(50),BB2(50)                PROP0260
C                                                                        PROP0270
C     CONSTANTS FOR THE EQUATION OF STATE.                               PROP0280
C                                                                        PROP0290
      DATA TC,R/1.1653D+03,0.5956924785D+00/                             PROP0300
      DATA BV,H,A4/-0.24670554D-01,0.86919801D+00,0.71943958D-01/        PROP0310
      DATA A2,B2,C2/-0.22524809D+02,0.11652976D-01,-0.70259944D+02/      PROP0320
      DATA A3,B3,C3/-0.87423115D+01,0.70670168D-02,-0.19906674D+00/      PROP0330
      DATA A5,B5,C5/0.12179538D-01,-0.1558344D-04,0.13679639D-01/        PROP0340
C                                                                        PROP0350
C     CONSTANTS FOR THE SPECIFIC HEAT EQUATION.                          PROP0360
C                                                                        PROP0370
      DATA X1,X2,X3/-0.23690403D-01,-0.22981253D+02,0.11977436D-01/      PROP0380
      DATA X4,X5,X6/-0.72422922D+02,0.84750477D+00,-0.87176941D+01/      PROP0390
      DATA X7,X8,X9/0.71550038D-02,-0.19803821D+00,0.77529884D-01/       PROP0400
      DATA X10,X11,X12/0.11963517D-01,-0.15597259D-04,0.12926977D-01/    PROP0410
      DATA X13,X14,X15/0.11488819D+01,0.449630230-04,0.28090134D-08/     PROP0420
C                                                                        PROP0430
C     CONSTANTS FOR THE ABSOLUTE VISCOSITY EQUATION.                     PROP0440
C                                                                        PROP0450
      DATA AVIS,BVIS,CVIS/0.9717059D+01,0.83352405D-02,0.10242672D+00/   PROP0460
C                                                                        PROP0470
C     CONSTANTS FOR THE THERMAL CONDUCTIVITY EQUATION.                   PROP0480
C                                                                        PROP0490
      DATA ACON,BCON,DCON/3.62D00,0.0380D00,2.667D00/                    PROP0500
C                                                                        PROP0510
      READ (5,40) P,T,V,DTEMP,NDOLP                                      PROP0520
   40 FORMAT(4F10.4,I4)                                                  PROP0530
C                                                                        PROP0540
C     COMPUTING THE SPECIFIC VOLUME (CUFT/LBM).                          PROP0550
C                                                                        PROP0560
      K=1                                                                PROP0570
      ITR = 1                                                            PROP0580
      Z = V-BV                                                           PROP0590
      DO 50 J=1,NDOLP,1                                                  PROP0600
      TEMP(J)=T                                                          PROP0610
      TR = T/TC                                                          PROP0620
      EXPNEG = (1.0D+00)/(DEXP(H*TR))                                    PROP0630
C     COEFF'S FOR 5TH DEGREE POLY.                                       PROP0640
      A00 = P                                                            PROP0650
      A11 = R*T                                                          PROP0660
      A22 = (A2)+(B2*T)+(C2*EXPNEG)                                      PROP0670
      A33 = (A3)+(B3*T)+(C3*EXPNEG)                                      PROP0680
      A44 = (A4)                                                         PROP0690
      A55 = (A5)+(B5*T)+(C5*EXPNEG)                                      PROP0700
C     NEWTON RAPHSON ITERATION.                                          PROP0710
   43 FN = (((((A00*Z+A11)*Z-A22)*Z-A33)*Z-A44)*Z-A55                    PROP0720
      FNDER = (((5.0D+00*A00*Z-4.0D+00*A11)*Z-3.0D+00*A22)*Z             PROP0730
     1-2.0D+00*A33)*Z-A44                                                PROP0740
      ZNEW = Z - (FN/FNDER)                                              PROP0750
      ZDIFF = ZNEW - Z                                                   PROP0760
      VZ=ZNEW+BV                                                         PROP0770
      IF(VZ .LE. 0.0D0) GO TO 31                                         PROP0780
      IF(DABS(ZDIFF).LE. 1.0D-5 )GO TO 45                                PROP0790
   31 Z = ZNEW                                                           PROP0800
      ITR = ITR + 1                                                      PROP0810
      IF(ITR .GT.300) GO TO 44                                           PROP0820
      GO TO 43                                                           PROP0830
   44 WRITE(6,30) P,T                                                    PROP0840
   30 FORMAT(1H0,5X,'MAXIMUM ITERATION REACHED FOR SPECIFIC VOLUME AT'//PROP0850
     *5X,' P = ',F10.5,' AND T = ',F10.5)                                PROP0860
   45 SPVOL = ZNEW + BV                                                  PROP0870
C -                                                                      PROP0880
C     COMPUTING THE DENSITY (LBM/CUFT).                                  PROP0890
C                                                                        PROP0900
      DENSTY = (1.0D+00)/SPVOL                                           PROP0910
      DENS(J)=DENSTY                                                     PROP0920
C-----                                                                   PROP0930
C     COMPUTING THE DERIVATIVES OF SPECIFIC VOLUME AND DENSITY.          PROP0940
C                                                                        PROP0950
      F11 = R/ZNEW                                                       PROP0960
      F22 =((B2)-(H/TC)*(C2)*(EXPNEG))/(ZNEW**2.0)                       PROP0970
      F33 =((B3)-(H/TC)*(C3)*(EXPNEG))/(ZNEW**3.0)                       PROP0980
      F55 =((B5)-(H/TC)*(C5)*(EXPNEG))/(ZNEW**5.0)                       PROP0990
      F66 =(A11)/(ZNEW**2.0)                                             PROP1000
      F77 = ((2.0D+00)*(A22))/(ZNEW**3.0)                                PROP1010
      F88 = ((3.0D+00)*(A33))/(ZNEW**4.0)                                PROP1020
      F99 = ((4.0D+00)*(A44))/(ZNEW**5.0)                                PROP1030
      F10 = ((5.0D+00)*(A55))/(ZNEW**6.0)                                PROP1040
      DERPTV = (F11)+(F22)+(F33)+(F55)                                   PROP1050
      DERPVT =-(F66)-(F77)-(F88)-(F99)-(F10)                             PROP1060
      DERVTP =-(DERPTV/DERPVT)                                           PROP1070
      DEROTP =-(DENSTY**2.0)*(DERVTP)                                    PROP1080
C                                                                        PROP1090
C     COMPUTING THE SPECIFIC HEAT AT CONSTANT PRESSURE (BTU/LBM*DEGR).   PROP1100
C                                                                        PROP1110
      ZCP=SPVOL-X1                                                       PROP1120
      EXPCP=(1.0D0)/(DEXP(X5*TR))                                        PROP1130
      D0=P                                                               PROP1140
      D1=R*T                                                             PROP1150
      D2=X2+(X3*T)+(X4*EXPCP)                                            PROP1160
      D3=X6+(X7*T)+(X8*EXPCP)                                            PROP1170
      D4=X9                                                              PROP1180
      D5=X10+(X11*T)+(X12*EXPCP)                                         PROP1190
```

```
   53 FNCP=(((((D0*ZCP-D1)*ZCP-D2)*ZCP-D3)*ZCP-D4)*ZCP-D5                  PROP1200
      FNDR=(((5.0D0*D0*ZCP-4.0D0*D1)*ZCP-3.0D0*D2)*ZCP-2.0D0*D3)*ZCP-D4   PROP1210
      ZNCP=ZCP-(FNCP/FNDR)                                                PROP1220
      ZDIF=ZNCP-ZCP                                                       PROP1230
      VZCP=ZNCP+X1                                                        PROP1240
      IF(VZCP .LE. 0.0D0) GO TO 35                                        PROP1250
      IF(DABS(ZDIF) .LE. 1.0D-05 ) GO TO 55                               PROP1260
   35 ZCP=ZNCP                                                            PROP1270
      K=K+1                                                               PROP1280
      IF(K .GT. 900 ) GO TO 11                                            PROP1290
      GO TO 53                                                            PROP1300
   11 WRITE(6,21) P,T                                                     PROP1310
   21 FORMAT(1H0,5X,'MAXIMUM ITERATION REACHED FOR SPECIFIC HEAT AT ',//PROP1320
     *5X,' P = ',F10.5,' AND T = ',F10.5)                                 PROP1330
   55 CONTINUE                                                            PROP1340
      F111=R/ZNCP                                                         PROP1350
      F222=((X3)-(X5/TC)*(X4)*(EXPCP))/(ZNCP**2)                          PROP1360
      F333=((X7)-(X5/TC)*(X8)*(EXPCP))/(ZNCP**3)                          PROP1370
      F555=((X11)-(X5/TC)*(X12)*(EXPCP))/(ZNCP**5)                        PROP1380
      F666=(D1)/(ZNCP**2)                                                 PROP1390
      F777=((2.D0)*(D2))/(ZNCP**3)                                        PROP1400
      F888=((3.D0)*(D3))/(ZNCP**4)                                        PROP1410
      F999=((4.D0)*(D4))/(ZNCP**5)                                        PROP1420
      F100=((5.D0)*(D5))/(ZNCP**6)                                        PROP1430
      DRPVT=(F111)+(F222)+(F333)+(F555)                                   PROP1440
      DRPVT=-(F666)-(F777)-(F888)-(F999)-(F100)                           PROP1450
      DRVTP=-(DRPTV/DRPVT)                                                PROP1460
      CVLP=X13+(X14*T)+(X15*T**2)                                         PROP1470
      W22=T*((X5/TC)**2)*(X4*(EXPCP))/(ZNCP)                              PROP1480
      W33=T*((X5/TC)**2)*(X8*(EXPCP))*(2.D0/(ZNCP**2))                    PROP1490
      W55=T*((X5/TC)**2)*(X12*(EXPCP))*(4.D0/(ZNCP**4))                   PROP1500
      CVHP = (CVLP) - ((W22+W33+W55)*(1.44D+02/7.78D+02))                 PROP1510
      CPHP=(CVHP)+T*(DRPTV)*(DRVTP)*(1.44D+02/7.78D+02)                   PROP1520
      CPH(J)=CPHP                                                         PROP1530
C                                                                         PROP1540
C     COMPUTING THE ABSOLUTE VISCOSITY (CENTIPOISE).                      PROP1550
C                                                                         PROP1560
      VISHP=(AVIS+(BVIS*T)+(CVIS*DENSTY**1.57))*(1.0D-06/0.000672D00)     PROP1570
      VISC(J)=VISHP                                                       PROP1580
C                                                                         PROP1590
C     COMPUTING THE DERIVATIVES OF THE ABSOLUTE VISCOSITY EQUATION.       PROP1600
C                                                                         PROP1610
      DERVDT=(CVIS*1.0D-06*1.57D0*DENSTY**0.57)/(0.000672D00)             PROP1620
      DERVTD=(BVIS*1.0D-06)/(0.000672D00)                                 PROP1630
      DERVTP=(DERVDT)*(DERDTP)+(DERVTD)                                   PROP1640
C                                                                         PROP1650
C     COMPUTING THE THERMAL CONDUCTIVITY (BTU/FT-HR-DEG R).               PROP1660
C                                                                         PROP1670
      TF=T-459.67D00                                                      PROP1680
      CONHP=(ACON+(BCON*TF)+(DCON*DENSTY**1.24))*1.0D-03                  PROP1690
      COND(J)=CONHP                                                       PROP1700
C                                                                         PROP1710
C     COMPUTING THE DERIVATIVES OF THE THERMAL CONDUCTIVITY EQUATION.     PROP1720
C                                                                         PROP1730
      DERKDT=1.24D00*DCON*1.0D-03*DENSTY**0.24                            PROP1740
      DERKTD=BCON*1.0D-03                                                 PROP1750
      DERKTP=(DERKDT)*(DERDTP)+(DERKTD)                                   PROP1760
C                                                                         PROP1770
C     COMPUTING THE COEFF'S OF THE DIFF. EQUATIONS.                       PROP1780
C                                                                         PROP1790
      DCOEFF = (DERDTP)/(DENSTY)                                          PROP1800
      ECOEFF = (DERVTP)/(VISHP)                                           PROP1810
      HCOEFF = (DERKTP)/(CONHP)                                           PROP1820
      PRNO = (CPHP*VISHP*2.42D00)/(CONHP)                                 PROP1830
C                                                                         PROP1840
      AA1(J)=DCOEFF + ECOEFF                                              PROP1850
      AA2(J)=1.0D+00/(DENSTY*VISHP)                                       PROP1860
      AA3(J)=1.0D+00/(DENSTY*DENSTY*VISHP)                                PROP1870
      BB1(J)=DCOEFF + HCOEFF                                              PROP1880
      BB2(J)=(CPHP)/(DENSTY*CONHP)                                        PROP1890
C                                                                         PROP1900
      T=T+DTEMP                                                           PROP1910
   50 ITR=1                                                               PROP1920
C                                                                         PROP1930
      RETURN                                                              PROP1940
      END                                                                 PROP1950
```

```
                SUBROUTINE PROP                                          PROP0010
C                                                                        PROP0020
C              ********************                                      PROP0030
C              * SUBROUTINE PROP   *                                     PROP0040
C              ********************                                      PROP0050
C                                                                        PROP0060
C     SUBJECT :                                                          PROP0070
C               THERMODYNAMIC AND TRANSPORT PROPERTIES OF                PROP0080
C               CARBON DIOXIDE IN THE SUPERCRITICAL REGION.              PROP0090
C                                                                        PROP0100
C     DATA INPUT :                                                       PROP0110
C               P=PRESSURE(PSIA)                                         PROP0120
C               T=TEMPERATURE(DEG.R)                                     PROP0130
C               V=SPECIFIC VOLUME (CUFT/LBM)--INITIAL VALUE              PROP0140
C               DTEMP = TEMP INCREMENT (DEG.R)                           PROP0150
C               NDOLP = MAX NO. OF DOLOOP ITERATION                      PROP0160
C                                                                        PROP0170
C     LIMITATIONS :                                                      PROP0180
C               REDUCED TEMPERATURE RANGE ( TR = T/TC )                  PROP0190
C               FROM 1.0028 TO 1.1455 ( WHERE TC = 547.56 DEG R ).       PROP0200
C               REDUCED PRESSURE RANGE ( PR = P/PC )                     PROP0210
C               FROM 1.056 TO 1.955 ( WHERE PC = 1070 PSIA ).            PROP0220
C                                                                        PROP0230
      IMPLICIT REAL*8(A-H,O-Z)                                           PROP0240
      COMMON/CPROP/DENS(50),CPH(50),VISC(50),COND(50),TEMP(50),DTEMP     PROP0250
      COMMON/TERM/AA1(50),AA2(50),AA3(50),BB1(50),BB2(50)                PROP0260
C                                                                        PROP0270
C     CONSTANTS FOR THE EQUATION OF STATE.                               PROP0280
C                                                                        PROP0290
      DATA TC,R/547.56D00,0.243840036 4D00/                              PROP0300
      DATA BV,H,A4/-0.18588948D-01,0.12018118D+01,0.18744415D-01/        PROP0310
      DATA A2,B2,C2/-0.13326704D+02,-0.83298325D-02,0.62417313D+02/      PROP0320
      DATA A3,B3,C3/-0.73090008D+01,0.90982283D-02,0.57368866D+01/       PROP0330
      DATA A5,B5,C5/0.1095804D-01,-0.1265501D-04,-0.13531694D-01/        PROP0340
C                                                                        PROP0350
C     CONSTANTS FOR THE SPECIFIC HEAT EQUATION.                          PROP0360
C                                                                        PROP0370
      DATA X1,X2,X3/-0.30146832D-01,-0.10972492D+02,-0.563071660-02/     PROP0380
      DATA X4,X5,X6/0.47571869D+02,0.12092998D+01,-0.72139233D+01/       PROP0390
      DATA X7,X8,X9/0.90332088D-02,0.63052626D+01,0.13941827D-02/        PROP0400
      DATA X10,X11,X12/0.11295915D-01,-0.12727667D-04,-0.131714620-01/   PROP0410
      DATA X13,X14,X15/0.70600271D+01,-0.17494759D-01,0.10779554D-04/    PROP0420
C                                                                        PROP0430
C     CONSTANTS FOR THE ABSOLUTE VISCOSITY EQUATION.                     PROP0440
C                                                                        PROP0450
      DATA AVIS,BVIS/-0.20033880D+01,0.22677910D-01/                     PROP0460
      DATA CVIS,DVIS/0.10305940D-01,0.20890480D+01/                      PROP0470
C                                                                        PROP0480
C     CONSTANTS FOR THE THERMAL CONDUCTIVITY EQUATION.                   PROP0490
C                                                                        PROP0500
      DATA ACON,BCON/0.84746444D+01,-0.44669005D-01/                     PROP0510
      DATA DCON,ECON/0.11056622D+02,0.46278907D+00/                      PROP0520
C                                                                        PROP0530
      READ (5,40) P,T,V,DTEMP,NDOLP                                      PROP0540
   40 FORMAT(4F10.4,I4)                                                  PROP0550
C                                                                        PROP0560
C     COMPUTING THE SPECIFIC VOLUME (CUFT/LBM).                          PROP0570
C                                                                        PROP0580
      K=1                                                                PROP0590
      ITR = 1                                                            PROP0600
      Z = V-BV                                                           PROP0610
      DO 50 J=1,NDOLP,1                                                  PROP0620
      TEMP(J)=T                                                          PROP0630
      TR = T/TC                                                          PROP0640
      EXPNEG = (1.0D+00)/(DEXP(H*TR))                                    PROP0650
C     COEFF'S FOR 5TH DEGREE POLY.                                       PROP0660
      A00 = P                                                            PROP0670
      A11 = R*T                                                          PROP0680
      A22 = (A2)+(B2*T)+(C2*EXPNEG)                                      PROP0690
      A33 = (A3)+(B3*T)+(C3*EXPNEG)                                      PROP0700
      A44 = (A4)                                                         PROP0710
      A55 = (A5)+(B5*T)+(C5*EXPNEG)                                      PROP0720
C     NEWTON RAPHSON ITERATION.                                          PROP0730
   43 FN = ((((A00*Z-A11)*Z-A22)*Z-A33)*Z-A44)*Z-A55                     PROP0740
      FNDER = (((5.0D+00*A00*Z-4.0D+00*A11)*Z-3.0D+00*A22)*Z             PROP0750
     1-2.0D+00*A33)*Z-A44                                                PROP0760
      ZNEW = Z - (FN/FNDER)                                              PROP0770
      ZDIFF = ZNEW - Z                                                   PROP0780
      VZ=ZNEW+BV                                                         PROP0790
      IF(VZ .LE. 0.0D0) GO TO 31                                         PROP0800
      IF(DABS(ZDIFF).LE. 1.0D-5 )GO TO 45                                PROP0810
   31 Z = ZNEW                                                           PROP0820
      ITR = ITR + 1                                                      PROP0830
      IF(ITR .GT.300) GO TO 44                                           PROP0840
      GO TO 43                                                           PROP0850
   44 WRITE(6,30) P,T                                                    PROP0860
   30 FORMAT(1HO,5X,'MAXIMUM ITERATION REACHED FOR SPECIFIC VOLUME AT'//PROP0870
     *5X,' P = ',F10.5,' AND T = ',F10.5)                                PROP0880
   45 SPVOL = ZNEW + BV                                                  PROP0890
C                                                                        PROP0900
C     COMPUTING THE DENSITY (LBM/CUFT).                                  PROP0910
C                                                                        PROP0920
      DENSTY = (1.0D+00)/SPVOL                                           PROP0930
      DENS(J)=DENSTY                                                     PROP0940
C                                                                        PROP0950
C     COMPUTING THE DERIVATIVES OF SPECIFIC VOLUME AND DENSITY.          PROP0960
C                                                                        PROP0970
      F11 = R/ZNEW                                                       PROP0980
      F22 =((B2)-(H/TC)*(C2)*(EXPNEG))/(ZNEW**2.0)                       PROP0990
      F33 =((B3)-(H/TC)*(C3)*(EXPNEG))/(ZNEW**3.0)                       PROP1000
      F55 =((B5)-(H/TC)*(C5)*(EXPNEG))/(ZNEW**5.0)                       PROP1010
      F66 =(A11)/(ZNEW**2.0)                                             PROP1020
      F77 = ((2.0D+00)*(A22))/(ZNEW**3.0)                                PROP1030
      F88 = ((3.0D+00)*(A33))/(ZNEW**4.0)                                PROP1040
      F99 = ((4.0D+00)*(A44))/(ZNEW**5.0)                                PROP1050
      F10 = ((5.0D+00)*(A55))/(ZNEW**6.0)                                PROP1060
      DERPTV = (F11)+(F22)+(F33)+(F55)                                   PROP1070
      DERPVT =-(F66)-(F77)-(F88)-(F99)-(F10)                             PROP1080
      DERVTP =-(DERPTV/DERPVT)                                           PROP1090
      DERDTP =-(DENSTY**2.0)*(DERVTP)                                    PROP1100
C                                                                        PROP1110
C     COMPUTING THE SPECIFIC HEAT AT CONSTANT PRESSURE (BTU/LBM*DEGR).   PROP1120
C                                                                        PROP1130
      ZCP=SPVOL-X1                                                       PROP1140
      D0=P                                                               PROP1150
      EXPCP=(1.0D0)/(DEXP(X5*TR))                                        PROP1160
      D1=R*T                                                             PROP1170
      D2=X2+(X3*T)+(X4*EXPCP)                                            PROP1180
      D3=X6+(X7*T)+(X8*EXPCP)                                            PROP1190
```

```
      D4=X9                                                                 PROP1200
      D5=X10+(X11*T)+(X12*EXPCP)                                            PROP1210
   53 FNCP=((((D0*ZCP-D1)*ZCP-D2)*ZCP-D3)*ZCP-D4)*ZCP-D5                    PROP1220
      FNDR=(((5.0D0*D0*ZCP-4.0D0*D1)*ZCP-3.0D0*D2)*ZCP-2.0D0*D3)*ZCP-D4     PROP1230
      ZNCP=ZCP-(FNCP/FNDR)                                                  PROP1240
      ZDIF=ZNCP-ZCP                                                         PROP1250
      VZCP=ZNCP+X1                                                          PROP1260
      IF(VZCP .LE. 0.0D0) GO TO 35                                          PROP1270
      IF(DABS(ZDIF) .LE. 1.0D-05 ) GO TO 55                                 PROP1280
   35 ZCP=ZNCP                                                              PROP1290
      K=K+1                                                                 PROP1300
      IF(K .GT. 900 ) GO TO 11                                              PROP1310
      GO TO 53                                                              PROP1320
   11 WRITE(6,21) P,T                                                       PROP1330
   21 FORMAT(1H0,5X,'MAXIMUM ITERATION REACHED FOR SPECIFIC HEAT AT ',//PROP1340
     *5X,' P = ',F10.5,' AND T = ',F10.5)                                   PROP1350
   55 CONTINUE                                                              PROP1360
      F111=R/ZNCP                                                           PROP1370
      F222=((X3)-(X5/TC)*(X4)*(EXPCP))/(ZNCP**2)                            PROP1380
      F333=((X7)-(X5/TC)*(X8)*(EXPCP))/(ZNCP**3)                            PROP1390
      F555=((X11)-(X5/TC)*(X12)*(EXPCP))/(ZNCP**5)                          PROP1400
      F666=(D1)/(ZNCP**2)                                                   PROP1410
      F777=((2.D0)*(D2))/(ZNCP**3)                                          PROP1420
      F888=((3.D0)*(D3))/(ZNCP**4)                                          PROP1430
      F999=((4.D0)*(D4))/(ZNCP**5)                                          PROP1440
      F100=((5.D0)*(D5))/(ZNCP**6)                                          PROP1450
      DRPTV=(F111)+(F222)+(F333)+(F555)                                     PROP1460
      DRPVT=-(F666)-(F777)-(F888)-(F999)-(F100)                             PROP1470
      DRVTP=-(DRPTV/DRPVT)                                                  PROP1480
      CVLP=X13+(X14*T)+(X15*T**2)                                           PROP1490
      W22=T*((X5/TC)**2)*(X4*(EXPCP))/(ZNCP)                                PROP1500
      W33=T*((X5/TC)**2)*(X8*(EXPCP))/(2.D0*(ZNCP**2))                      PROP1510
      W55=T*((X5/TC)**2)*(X12*(EXPCP))/(4.D0*(ZNCP**4))                     PROP1520
      CVHP = (CVLP) - ((W22+W33+W55)*(1.44D+02/7.78D+02))                   PROP1530
      CPHP=(CVHP)+T*(DRPTV)*(DRVTP)*(1.44D+02/7.78D+02)                     PROP1540
      CPH(J)=CPHP                                                           PROP1550
C                                                                           PROP1560
C     COMPUTING THE ABSOLUTE VISCOSITY (CENTIPOISE).                        PROP1570
C                                                                           PROP1580
      VISHP=(AVIS+(BVIS*T)+(CVIS*DENSTY**DVIS))*(1.0D-06/0.000672D00)       PROP1590
      VISC(J)=VISHP                                                         PROP1600
C                                                                           PROP1610
C     COMPUTING THE DERIVATIVES OF THE ABSOLUTE VISCOSITY EQUATION.         PROP1620
C                                                                           PROP1630
      DERUDT=(CVIS*1.0D-06*DVIS*DENSTY**(DVIS-1.0D00))/(0.000672D00)        PROP1640
      DERUTD=BVIS*1.0D-06/(0.000672D00)                                     PROP1650
      DERUTP=(DERUDT)*(DERDTP)+(DERUTD)                                     PROP1660
C                                                                           PROP1670
C     COMPUTING THE THERMAL CONDUCTIVITY (BTU/FT-HR-DEG R).                 PROP1680
C                                                                           PROP1690
      CONHP=(ACON+(BCON*T)+(DCON*DENSTY**ECON))*1.0D-03                     PROP1700
      COND(J)=CONHP                                                         PROP1710
C                                                                           PROP1720
C     COMPUTING THE DERIVATIVES OF THE THERMAL CONDUCTIVITY EQUATION.       PROP1730
C                                                                           PROP1740
      DERKDT=DCON*ECON*1.0D-03*DENSTY**(ECON-1.0D00)                        PROP1750
      DERKTD=BCON*1.0D-03                                                   PROP1760
      DERKTP=(DERKDT)*(DERDTP)+(DERKTD)                                     PROP1770
C                                                                           PROP1780
C     COMPUTING THE COEFF'S OF THE DIFF. EQUATIONS.                         PROP1790
C                                                                           PROP1800
      DCOEFF = (DERDTP)/(DENSTY)                                            PROP1810
      ECCEFF = (DERUTP)/(VISHP)                                             PROP1820
      HCOEFF = (DERKTP)/(CONHP)                                             PROP1830
      PRNO = (CPHP*VISHP*2.420D0)/(CONHP)                                   PROP1840
C                                                                           PROP1850
      AA1(J)=DCOEFF + ECCEFF                                                PROP1860
      AA2(J)=1.0D+00/(DENSTY*VISHP)                                         PROP1870
      AA3(J)=1.0D+00/(DENSTY*DENSTY*VISHP)                                  PROP1880
      BB1(J)=DCOEFF + HCOEFF                                                PROP1890
      BB2(J)=(CPHP)/(DENSTY*CONHP)                                          PROP1900
C                                                                           PROP1910
      T=T+DTEMP                                                             PROP1920
   50 ITR=1                                                                 PROP1930
C                                                                           PROP1940
      RETURN                                                                PROP1950
      END                                                                   PROP1960
```

```
                SUBROUTINE PROP                                          PROP0010
C                                                                        PROP0020
C              ********************                                      PROP0030
C              * SUBROUTINE PROP   *                                     PROP0040
C              ********************                                      PROP0050
C                                                                        PROP0060
C     SUBJECT :                                                          PROP0070
C                THERMODYNAMIC AND TRANSPORT PROPERTIES OF              PROP0080
C                REFRIGERANT 114 IN THE SUPERCRITICAL REGION.           PROP0090
C                                                                        PROP0100
C     DATA INPUT :                                                       PROP0110
C                P=PRESSURE(PSIA)                                       PROP0120
C                T=TEMPERATURE(DEG.R)                                   PROP0130
C                V=SPECIFIC VOLUME (CUFT/LBM)--INITIAL VALUE            PROP0140
C                DTEMP = TEMP INCREMENT (DEG.R)                         PROP0150
C                NDOLP = MAX NO. OF DOLOOP ITERATION                    PROP0160
C                                                                        PROP0170
C     LIMITATIONS :                                                      PROP0180
C                REDUCED TEMPERATURE RANGE ( TR = T/TC )                PROP0190
C                FROM 1.0076 TO 1.074 ( WHERE TC = 753.95 DEG R ).      PROP0200
C                REDUCED PRESSURE RANGE ( PR = P/PC )                   PROP0210
C                FROM 1.0638 TO 1.2766 ( WHERE PC = 473 PSIA ).         PROP0220
C                                                                        PROP0230
      IMPLICIT REAL*8(A-H,O-Z)                                           PROP0240
      COMMON/CPROP/DENS(50),CPH(50),VISC(50),COND(50),TEMP(50),DTEMP     PROP0250
      COMMON/TERM/AA1(50),AA2(50),AA3(50),BB1(50),BB2(50)                PROP0260
C                                                                        PROP0270
C     CONSTANTS FOR THE EQUATION OF STATE.                               PROP0280
C                                                                        PROP0290
      DATA TC,R/7.5395D+02,0.06280187D00/                                PROP0300
      DATA BV,H,A4/5.9149070D-03,3.0D+00,-3.8574810D-04/                 PROP0310
      DATA A2,B2,C2/-2.3856704D+00,1.0801207D-03,-6.5643648D+00/         PROP0320
      DATA A3,B3,C3/3.4055687D-02,-5.3336494D-06,1.6366057D-01/          PROP0330
      DATA A5,B5,C5/1.6017659D-06,6.2632340D-10,-1.0165314D-05/          PROP0340
C                                                                        PROP0350
C     CONSTANTS FOR THE SPECIFIC HEAT EQUATION.                          PROP0360
C                                                                        PROP0370
C     SEE THE APPROPRIATE EQUATIONS IN THE PROGRAM.                      PROP0380
C                                                                        PROP0390
C     CONSTANTS FOR THE ABSOLUTE VISCOSITY EQUATION.                     PROP0400
C                                                                        PROP0410
C     SEE THE APPROPRIATE EQUATIONS IN THE PROGRAM.                      PROP0420
C                                                                        PROP0430
C     CONSTANTS FOR THE THERMAL CONDUCTIVITY EQUATION.                   PROP0440
C                                                                        PROP0450
C     SEE THE APPROPRIATE EQUATIONS IN THE PROGRAM.                      PROP0460
C                                                                        PROP0470
      READ (5,40) P,T,V,DTEMP,NDOLP                                      PROP0480
   40 FORMAT(4F10.4,I4)                                                  PROP0490
C                                                                        PROP0500
C     COMPUTING THE SPECIFIC VOLUME (CUFT/LBM).                          PROP0510
C                                                                        PROP0520
      K=1                                                                PROP0530
      ITR = 1                                                            PROP0540
      Z = V-BV                                                           PROP0550
      DO 50 J=1,NDOLP,1                                                  PROP0560
      TEMP(J)=T                                                          PROP0570
      TR = T/TC                                                          PROP0580
      EXPNEG = (1.0D+00)/(DEXP(H*TR))                                    PROP0590
C     COEFF'S FOR 5TH DEGREE POLY.                                       PROP0600
      A00 = P                                                            PROP0610
      A11 = R*T                                                          PROP0620
      A22 = (A2)+(B2*T)+(C2*EXPNEG)                                      PROP0630
      A33 = (A3)+(B3*T)+(C3*EXPNEG)                                      PROP0640
      A44 = (A4)                                                         PROP0650
      A55 = (A5)+(B5*T)+(C5*EXPNEG)                                      PROP0660
C     NEWTON RAPHSON ITERATION.                                          PROP0670
   43 FN = ((((A00*Z-A11)*Z-A22)*Z-A33)*Z-A44)*Z-A55                     PROP0680
      FNDER = (((5.0D+00*A00*Z-4.0D+00*A11)*Z-3.0D+00*A22)*Z             PROP0690
     1-2.0D+00*A33)*Z-A44                                                PROP0700
      ZNEW = Z - (FN/FNDER)                                              PROP0710
      ZDIFF = ZNEW - Z                                                   PROP0720
      VZ=ZNEW+BV                                                         PROP0730
       IF(VZ .LE. 0.0D0) GO TO 31                                        PROP0740
      IF(DABS(ZDIFF).LE. 1.0D-5 )GO TO 45                                PROP0750
   31 Z = ZNEW                                                           PROP0760
      ITR = ITR + 1                                                      PROP0770
      IF(ITR .GT.300) GO TO 44                                           PROP0780
      GO TO 43                                                           PROP0790
   44 WRITE(6,30) P,T                                                    PROP0800
   30 FORMAT(1H0,5X,'MAXIMUM ITERATION REACHED FOR SPECIFIC VOLUME AT'//PROP0810
     *5X,' P = ',F10.5,' AND T = ',F10.5)                                PROP0820
   45 SPVOL = ZNEW + BV                                                  PROP0830
C                                                                        PROP0840
C     COMPUTING THE DENSITY (LBM/CUFT).                                  PROP0850
C                                                                        PROP0860
      DENSTY = (1.0D+00)/SPVOL                                           PROP0870
      DENS(J)=DENSTY                                                     PROP0880
C                                                                        PROP0890
C     COMPUTING THE DERIVATIVES OF SPECIFIC VOLUME AND DENSITY.          PROP0900
C                                                                        PROP0910
      F11 = R/ZNEW                                                       PROP0920
      F22 =((B2)-(H/TC)*(C2)*(EXPNEG))/(ZNEW**2.0)                       PROP0930
      F33 =((B3)-(H/TC)*(C3)*(EXPNEG))/(ZNEW**3.0)                       PROP0940
      F55 =((B5)-(H/TC)*(C5)*(EXPNEG))/(ZNEW**5.0)                       PROP0950
      F66 =(A11)/(ZNEW**2.0)                                             PROP0960
      F77 = ((2.0D+00)*(A22))/(ZNEW**3.0)                                PROP0970
      F88 = ((3.0D+00)*(A33))/(ZNEW**4.0)                                PROP0980
      F99 = ((4.0D+00)*(A44))/(ZNEW**5.0)                                PROP0990
      F10 = ((5.0D+00)*(A55))/(ZNEW**6.0)                                PROP1000
      DERPTV = (F11)+(F22)+(F33)+(F55)                                   PROP1010
      DERPVT =-(F66)-(F77)-(F88)-(F99)-(F10)                             PROP1020
      DERVTP =-(DERPTV/DERPVT)                                           PROP1030
      DERDTP =-(DENSTY**2.0)*(DERVTP)                                    PROP1040
C                                                                        PROP1050
C     COMPUTING THE SPECIFIC HEAT AT CONSTANT VOLUME (BTU/LBM*DEGR).     PROP1060
C                                                                        PROP1070
      CVLP =(1.75D-02)+(3.49D-04)*(T)-(1.67D-07)*(T**2.0)                PROP1080
      W22 = ((T)*((H/TC)**2.0)*(C2)*(EXPNEG))/(ZNEW)                     PROP1090
      W33 = ((T)*((H/TC)**2.0)*(C3)*(EXPNEG))/(2.0D+00*(ZNEW**2.0))      PROP1100
      W55 = ((T)*((H/TC)**2.0)*(C5)*(EXPNEG))/(4.0D+00*(ZNEW**4.0))      PROP1110
      CVHP = (CVLP) - ((W22+W33+W55)*(1.44D+02/7.78D+02))                PROP1120
C                                                                        PROP1130
C     COMPUTING THE SPECIFIC HEAT AT CONSTANT PRESSURE (BTU/LBM*DEGR).   PROP1140
C                                                                        PROP1150
      CPHP =(CVHP)+(T)*(DERPTV)*(DERVTP)*(1.44D+02/7.78D+02)             PROP1160
      CPH(J)=CPHP                                                        PROP1170
C                                                                        PROP1180
C     COMPUTING THE ABSOLUTE VISCOSITY (CENTIPOISE).                     PROP1190
```

```
C
      VISLP =((1.0167D-03)*(T**1.5))/(T+5.6655D+02)                          PROP1200
      XIN = ((4.18878D+02)**0.16667)                                         PROP1210
      XID =  (1.70936D+02**0.5)*(3.2189D+01**0.66667)                        PROP1220
      XI = (XIN)/(XID)                                                       PROP1230
      DENR = (DENSTY)/(3.632D+01)                                            PROP1240
      XIC = (2.5D-04)/(XI)                                                   PROP1250
      VISHP = (VISLP)+((2.3125-04)*(DEXP(1.079D+00*DENR)))/(XI) -(XIC)       PROP1260
      VISC(J)=VISHP                                                          PROP1270
C                                                                            PROP1280
C     COMPUTING THE DERIVATIVES OF THE ABSOLUTE VISCOSITY EQUATION.          PROP1290
C                                                                            PROP1300
      DERVDT = (1.079D+00/3.632D+01)*(VISHP-VISLP+XIC)                       PROP1310
      DERVTD = (1.5)*(VISLP/T)-(VISLP)/(T+5.6655D+02)                        PROP1320
      DERV1P = (DERVDT)*(DERDTP)+(DERVTD)                                    PROP1330
C                                                                            PROP1340
C     COMPUTING THE THERMAL CONDUCTIVITY (CAL/SEC CM DEGK).                  PROP1350
C                                                                            PROP1360
      CONC = ((1.32D-02)*(2.52D+02)*(1.8D+03))/(4.5360+02)                   PROP1370
      CON1 = (CONC)*(CVLP)*(VISLP)                                           PROP1380
      CON2 = ((3.40D-02)*(VISLP))/(1.70936D+02)                              PROP1390
      CON3 = ((7.0D-03)*(VISLP))/((TR)*(1.70936D+02))                        PROP1400
      CONLP= (CON1)+(CON2)-(CON3)                                            PROP1410
      CON5 = (4.73187D+02)*(1.70936D+02)                                     PROP1420
      CON6 = (3.632D+01)*(1.07315D+01)*(7.5395D+02)                          PROP1430
      CONIC = (CON5)/(CON6)                                                  PROP1440
      CONL1 =  (1.70936D+02**0.5)*(4.18877D+02**0.16667)                     PROP1450
      CONL2 = (3.2189D+01**0.66667)                                          PROP1460
      CONLA =  CONL1/CONL2                                                   PROP1470
      CON7  =  (CONLA)*(CON7C**5.0)                                          PROP1480
      CONHP = (CONLP)+((1.31D-07*DEXP(6.7D-01*DENR))/CON7)                   PROP1490
     1- (1.40039D-07/CON7)                                                   PROP1500
      COND(J)=CONHP                                                          PROP1510
C                                                                            PROP1520
C     COMPUTING THE DERIVATIVES OF THE THERMAL CONDUCTIVITY EQUATION.        PROP1530
C                                                                            PROP1540
      DERKDT =(6.7D-01/3.632D+01)* (CONHP-CONLP+(1.40039D-07/CON7))          PROP1550
      DERCVT = (3.49D-04) -(3.34D-07*T)                                      PROP1560
      DER1 = (CONC)*(CVLP)*(DERUTD)                                          PROP1570
      DER2 = (CONC)*(VISLP)*(DERCVT)                                         PROP1580
      DER3 =  (3.40D-02/1.70936D+02)*(DERUTD)                                PROP1590
      DER4 = ((7.0D-03)*(DERUTD))/((1.70936D+02)*(TR))                       PROP1600
      DER5 = ((7.0D-03)*(VISLP))/((1.70936D+02)*(TR)*(T))                    PROP1610
      DERKTD = (DER1)+(DER2)+(DER3)-(DER4)+(DER5)                            PROP1620
      DERKTP = (DERKDT)*(DERDTP) + (DERKTD)                                  PROP1630
C                                                                            PROP1640
C     COMPUTING THE COEFF'S OF THE DIFF. EQUATIONS.                          PROP1650
C                                                                            PROP1660
      DCOEFF = (DERDTP)/(DENSTY)                                             PROP1670
      ECOEFF = (DERVTP)/(VISHP)                                              PROP1680
      HCOEFF = (DERKTP)/(CONHP)                                              PROP1690
      PRNC = (CPHP*VISHP*1.0D-02)/(CONHP)                                    PROP1700
C                                                                            PROP1710
      AA1(J)=DCOEFF + ECOEFF                                                 PROP1720
      AA2(J)=1.0D+00/(DENSTY*VISHP)                                          PROP1730
      AA3(J)=1.0D+00/(DENSTY*DENSTY*VISHP)                                   PROP1740
      BB1(J)=DCOEFF + HCOEFF                                                 PROP1750
      BB2(J)=(CPHP)/(DENSTY*CONHP)                                           PROP1760
C                                                                            PROP1770
      T=T+DTEMP                                                              PROP1780
   50 ITR=1                                                                  PROP1790
C                                                                            PROP1800
      RETURN                                                                 PROP1810
      END                                                                    PROP1820
                                                                             PROP1830
```

```
      SUBROUTINE STEPT (FUNK)                                           STMARQ  1
C                                                                       STMARQ  2
C  INTERFACE TO MAKE MARQ LOOK LIKE STEPT.                              STMARQ  3
C                                                                       STMARQ  4
C  TO USE THIS ROUTINE, SET THE VALUE OF LPCOL AND THE DIMENSIONS OF    STMARQ  5
C  THE ARRAYS P, FITSV, FIT, Y, AND YSIG.  THE DIMENSIONS ARE....       STMARQ  6
C       P(LPCOL,NVMAX), FITSV(LPCOL), FIT(LPCOL), Y(LPCOL), YSIG(LPCOL) STMARQ  7
C  WHERE LPCOL IS THE MAXIMUM VALUE OF NPTS AND NVMAX IS THE MAXIMUM    STMARQ  8
C  VALUE OF NV.  (IF LEQU.EQ.1, YSIG MAY BE DIMENSIONED YSIG(1).)       STMARQ  9
C  COMMON/CDAT/ DOES NOT APPEAR IN ANY ROUTINE OF THE MARQ PACKAGE      STMARQ10
C  OTHER THAN THIS ONE, SO THAT THIS IS THE ONLY ONE WHICH MUST BE      STMARQ11
C  RECOMPILED WHEN LPCOL AND THE DIMENSIONS OF THE ARRAYS ARE CHANGED.  STMARQ12
C                                                                       STMARQ13
C  THE FOLLOWING EXTERNAL STATEMENT IS REQUIRED BY SOME COMPILERS       STMARQ14
C  (MATFIV, FOR EXAMPLE) AND FORBIDDEN BY OTHERS (MODCOMP II).          STMARQ15
      EXTERNAL FUNK                                                     STMARQ16
C                                                                       STMARQ17
      DOUBLE PRECISION FIT,FITSV                                        STMARQ18
C                                                                       STMARQ19
C     DIMENSION P(275,8),FITSV(275)                                     STMARQ20
      DIMENSION P(25,20),FITSV(25)                                      STMARQ21
C                                                                       STMARQ22
C     COMMON /CDAT/ FIT(275),Y(275),YSIG(275),NPTS                      STMARQ23
      COMMON /CDAT/ FIT(25),Y(25),YSIG(25),NPTS                         STMARQ24
C                                                                       STMARQ25
      LPCOL=1                                                           STMARQ26
      LPCOL=275                                                         STMARQ27
      LPCOL=25                                                          STMARQ28
C                                                                       STMARQ29
C                                                                       STMARQ30
      CALL MARQ (FUNK,Y,YSIG,NPTS,FIT,FITSV,P,LPCOL)                    STMARQ31
C                                                                       STMARQ32
      RETURN                                                            STMARQ33
      END                                                               STMARQ35
C                                                                       STMARQ36
      SUBROUTINE STSET                                                  STSETM  1
C                                                                       STSETM  2
C  STSET SETS SOME INPUT QUANTITIES TO DEFAULT VALUES, FOR MARQ.        STSETM  3
C                                                                       STSETM  4
C  NOTE...........                                                      STSETM  5
C  THIS VERSION OF STSET MAY ALSO BE USED WITH STEPIT, SIMPLEX,         STSETM  6
C  STP, KAUPE, AND MINF.                                               STSETM  7
C                                                                       STSETM  8
C  USAGE.....                                                           STSETM  9
C  CALL STSET.  THEN SET SOME INPUT QUANTITIES (NV AND NPTS, AT LEAST)  1000TM  9
C  AND RESET ANY OF THOSE SET IN STSET (BETTER VALUES OF X, ETC.)       2000TM  9
C  BEFORE CALLING MARQ OR THE STEPT-MARQ INTERFACE ROUTINE.             3000TM  9
C                                                                       STSETM13
      DOUBLE PRECISION X                                                STSETM14
C                                                                       STSETM15
      COMMON /CSTEP/ X(20),XMAX(20),XMIN(20),DELTX(20),DELMN(20),       STSETM16
     *     ERR(20,21),FOBJ,NV,NTRAC,MATRX,MASK(20),                     STSETM17
     *     NFMAX,NFLAT,JVARY,NXTRA,KFLAG,NOREP,KERFL,KW                 STSETM18
      COMMON /ALLS2/FLAMB,RELDF,METHD,KALCP,KORDF,MAXIT,LEQU,MXSUB,MXUPDSTSETM19
C                                                                       STSETM20
C                             KW ...  LOGICAL UNIT NUMBER OF THE PRINTER STSETM21
      KW=6                                                              STSETM22
      NVMAX=20                                                          STSETM23
C                             THE USER MUST SET NV AFTER CALLING STSET. STSETM24
      NV=-1                                                             STSETM25
      NPTS=-1                                                           STSETM26
      HUGE=1.E30                                                        STSETM27
      RZERO=0.                                                          STSETM28
      NTRAC=0                                                           STSETM29
      NFMAX=32767                                                       STSETM30
      MAXIT=20                                                          STSETM31
      MXSUB=50                                                          STSETM32
      METHD=1                                                           STSETM33
      KALCP=0                                                           STSETM34
      KORDF=1                                                           STSETM35
      LEQU=0                                                            STSETM36
      NFLAT=1                                                           STSETM37
      MATRX=106                                                         STSETM38
      NXTRA=0                                                           STSETM39
      FLAMB=1.                                                          STSETM40
      RELDF=1.E-8                                                       STSETM41
C                                                                       STSETM42
      DO 10 JX=1,NVMAX                                                  STSETM43
        A(JX)=RZERO                                                     STSETM44
        XMAX(JX)=HUGE                                                   STSETM45
        XMIN(JX)=-HUGE                                                  STSETM46
        DELTX(JX)=RZERO                                                 STSETM47
        DELMN(JX)=RZERO                                                 STSETM48
        MASK(JX)=0                                                      STSETM49
   10   CONTINUE                                                        STSETM50
C                                                                       STSETM51
      RETURN                                                            STSETM52
      END                                                               STSETM54
C                                                                       STSETM55
      SUBROUTINE FOFX (JPT,NV,X,F)                                      FOFXDUM1
C                                                                       FOFXDUM2
C  THIS IS A DUMMY VERSION OF SUBROUTINE FOFX.                          FOFXDUM3
C  A NON-DUMMY VERSION OF FOFX MAY BE USED (OPTIONALLY) TO SUPPLY       FOFXDUMA
C  TO MARQ VALUES OF THE FUNCTION BEING FITTED, INSTEAD OF USING A      FOFXDUM5
C  -FUNK- SUBROUTINE TO DO THIS.  THE USE OF FOFX REQUIRES              FOFXDUM6
C  SUBSTANTIALLY MORE OVERHEAD TIME DURING EXECUTION, BUT SAVES         FOFXDUM7
C  CONSIDERABLE STORAGE BY NOT REQUIRING THAT THE JACOBIAN MATRIX, P,   FOFXDUM8
C  BE STORED.  THE MINI-MARQ PACKAGE REQUIRES THE USE OF FOFX.          FOFXDUM9
C  TO SEE HOW TO USE A NON-DUMMY FOFX, SEE THE LISTING OF MARQ.         FOFXDU10
C                                                                       FOFXDU11
      DOUBLE PRECISION X,F                                              1000DU11
      DIMENSION X(20)                                                   2000DU11
      RETURN                                                            FOFXDU12
      END                                                               FOFXDU13
C                                                                       FOFXDU14
      SUBROUTINE CALCD (JPT,P,LPCOL)                                    CALCDUM1
C                                                                       CALCDUM2
C  THIS IS A DUMMY VERSION OF SUBROUTINE CALCD.                         CALCDUM3
C  A NON-DUMMY VERSION OF CALCD MAY BE USED (OPTIONALLY) TO SUPPLY      CALCDUMA
C  TO MARQ OR MINI-MARQ ANALYTIC VALUES OF THE ELEMENTS OF THE          CALCDUM5
C  JACOBIAN MATRIX, INSTEAD OF CALCULATING THEM USING FINITE            CALCDUM6
C  DIFFERENCES.  MOST USERS USE FINITE DIFFERENCES MOST OF THE TIME.    CALCDUM7
C  TO SEE HOW TO USE A NON-DUMMY CALCD, SEE THE LISTING OF MARQ.        CALCDUM8
C                                                                       CALCDUM9
      DIMENSION P(LPCOL,1)                                              1000DUM9
      RETURN                                                            CALCDU10
      END                                                               CALCDU11
C                                                                       CALCDU12
      SUBROUTINE MARQ (FUNK,Y,YSIG,NPTS,FIT,FITSV,P,LPCOL)              MARQ   1
```

```
C                                                                       MARQ   2
C  MARQ 2.5        A.N.S.I. STANDARD FORTRAN          AUGUST 1976       MARQ   3
C  COPYRIGHT (C) 1978 J. P. CHANDLER                                    MARQ   4
C                                                                       MARQ   5
C  J. P. CHANDLER AND LEON W. JACKSON,                                  MARQ   6
C               DEPARTMENT OF COMPUTING AND INFORMATION SCIENCES        MARQ   7
C               OKLAHOMA STATE UNIVERSITY, STILLWATER, OKLAHOMA 74074   MARQ   8
C                                                                       MARQ   9
C  MARQ PERFORMS A NONLINEAR LEAST SQUARES FIT OF A USER-SUPPLIED       MARQ  10
C  FUNCTION TO A GIVEN SET OF DATA, USING MARQUARDT-S METHOD, OR THE    MARQ  11
C  GAUSS-NEWTON METHOD, OR A MODIFIED GAUSS-NEWTON METHOD.              MARQ  12
C  D. W. MARQUARDT, J.SOC.IND.APPL.MATH. 11 (1963) 431-441              MARQ  13
C                                                                       MARQ  14
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * MARQ  15
C                                                                       MARQ  16
C  INPUT QUANTITIES.....  FUNK,X(*),XMIN(*),MASK(*),DELMN(*),MATRX,     MARQ  17
C                         Y(*),YSIG(*),NV,NPTS,LPCOL,FLAMB,NFMAX,       MARQ  18
C                         NTRAC,KALCP,KORDF,METHD,NFLAT,RELDF,          MARQ  19
C                         MAXIT,LEQU,KW                                 MARQ  20
C  OUTPUT QUANTITIES....  X(*),FOBJ,ERR(*,*),KFLAG,FIT(*)               MARQ  21
C  UNUSED QUANTITIES (INCLUDED FOR COMPATIBILITY WITH STEPIT COMMON)....MARQ  22
C                         DELTA(*),JVARY,NXTRA,NOREP,KERFL              MARQ  23
C                                                                       MARQ  24
C     FUNK        --  THE NAME OF THE FUNCTION CALLED TO OBTAIN         MARQ  25
C                     THE FITTED VALUES IF KALCP=0                      MARQ  26
C     NV          --  THE NUMBER OF PARAMETERS                          MARQ  27
C     X(JX)       --  THE JX-TH PARAMETER                               MARQ  28
C     FOBJ        --  RETURNS THE FINAL VALUE OF PHI, THE WEIGHTED      MARQ  29
C                     SUM OF SQUARES (PHI IS BEING MINIMIZED AS A       MARQ  30
C                     FUNCTION OF THE X(JX))                            MARQ  31
C     XMAX(JX)    --  THE UPPER LIMIT FOR X(JX)                        MARQ  32
C     XMIN(JX)    --  THE LOWER LIMIT FOR X(JX)                        MARQ  33
C     MASK(JX)    --  NONZERO IF X(JX) IS TO BE HELD FIXED             MARQ  34
C     DELMN(JX)   --  THE CONVERGENCE TOLERANCE FOR X(JX)               MARQ  35
C     NTRAC       --  USER PRINT CONTROL                                MARQ  36
C                         =-3, NO OUTPUT EXCEPT FATAL ERROR MESSAGES    MARQ  37
C                         =-2, NO OUTPUT EXCEPT DIAGNOSTIC MESSAGES     MARQ  38
C                         =-1, STANDARD OUTPUT EXCEPT FINAL FIT VALUES  MARQ  39
C                         = 0, STANDARD OUTPUT                          MARQ  40
C                         = 1, ALSO PRINTS RESULTS OF EACH ITERATION    MARQ  41
C                         = 2, ALSO PRINTS THE COEFFICIENT MATRIX, QSAV MARQ  42
C                         = 3, ALSO PRINTS THE JACOBIAN MATRIX, P       MARQ  43
C     NFMAX       --  THE MAXIMUM NUMBER OF FUNCTION COMPUTATIONS       MARQ  44
C     NFLAT       --  NONZERO IF THE SEARCH IS TO TERMINATE WHEN TWO    MARQ  45
C                     ITERATIONS GIVE IDENTICAL VALUES OF PHI           MARQ  46
C     KW          --  THE LOGICAL UNIT NUMBER OF THE PRINTER            MARQ  47
C     ERR(JX,KX)  --  RETURNS THE ERROR MATRIX                          MARQ  48
C     KFLAG       --  RETURNED .GT. ZERO FOR A NORMAL EXIT,             MARQ  49
C                     RETURNED .LT. ZERO FOR AN ABNORMAL EXIT           MARQ  50
C                                                                       MARQ  51
C     NPTS        --  THE NUMBER OF DATA OBSERVATIONS                   MARQ  52
C     Y(JPT)      --  THE JPT-TH DATA ORDINATE                          MARQ  53
C     YSIG(JPT)   --  THE EXPECTED ERROR IN Y(JPT)                      MARQ  54
C     FIT(JPT)    --  THE JPT-TH FITTED VALUE                          MARQ  55
C     FITSV       --  A SCRATCH VECTOR OF NPTS VALUES                   MARQ  56
C     P(JPT,JX)   --  THE FIRST PARTIAL DERIVATIVE                      MARQ  57
C                     OF FIT(JPT) WITH RESPECT TO X(JX)                 MARQ  58
C     LPCOL       --  THE FIRST DIMENSION OF THE ARRAY CONTAINING P     MARQ  59
C                     (LPCOL MUST BE .GE. NPTS IF KALCP IS .GE. ZERO)   MARQ  60
C     LEQU        --  SAVES STORAGE IF ALL YSIG(JPT) ARE THE SAME       MARQ  61
C                         =1 IF ALL YSIG(JPT) ARE EQUAL (IN THIS CASE   MARQ  62
C                             ONLY YSIG(1) IS REFERENCED)               MARQ  63
C                         =0 OTHERWISE                                  MARQ  64
C     FLAMB       --  MARQUARDT-S LAMBDA, THE RELATIVE AMOUNT BY WHICH  MARQ  65
C                     THE DIAGONAL COEFFICIENTS OF THE NORMAL           MARQ  66
C                     EQUATIONS ARE AUGMENTED                           MARQ  67
C     MXSUB       --  THE MAXIMUM NUMBER OF SUBITERATIONS PERMITTED     MARQ  68
C     KALCP       --  DETERMINES WHICH ROUTINE IS CALLED TO COMPUTE THE MARQ  69
C                     JACOBIAN MATRIX OF FIRST PARTIAL DERIVATIVES      MARQ  70
C                         =-1, ONE ROW OF P RETURNED BY DERIV OR CALCD, MARQ  71
C                              DERIV CALLS FOFX                         MARQ  72
C                         = 0, ALL OF P RETURNED BY DERIV OR CALCD,     MARQ  73
C                              DERIV CALLS FUNK                         MARQ  74
C                         = 1, ALL OF P RETURNED BY DERIV OR CALCD,     MARQ  75
C                              DERIV CALLS FOFX                         MARQ  76
C     KORDF       --  DETERMINES HOW P IS TO BE CALCULATED              MARQ  77
C                         = 1, FIRST ORDER APPROXIMATION USED BY DERIV  MARQ  78
C                         = 2, SECOND ORDER APPROXIMATION USED BY DERIV MARQ  79
C                         = 3, ANALYTICAL DERIVATIVE SUPPLIED BY CALCD  MARQ  80
C     METHD       --  DETERMINES THE METHOD USED                        MARQ  81
C                         =-1, MODIFIED GAUSS-NEWTON METHOD             MARQ  82
C                         = 0, GAUSS-NEWTON METHOD                      MARQ  83
C                         = 1, MODIFIED FORM OF MARQUARDT-S METHOD      MARQ  84
C                         = 2, MARQUARDT-S METHOD                       MARQ  85
C     RELDF       --  DETERMINES THE MAGNITUDE OF THE DIFFERENCING STEP MARQ  86
C     MAXIT       --  THE MAXIMUM NUMBER OF ITERATIONS TO BE PERFORMED  MARQ  87
C                                                                       MARQ  88
C  MODIFICATIONS TO MARQUARDT-S METHOD...                               MARQ  89
C    1.  THE QUANTITY (1+LAMBDA) IS NOT ALLOWED TO INCREASE BY MORE     MARQ  90
C        THAN A FACTOR OF TWO, IN ANY SINGLE INCREASE.                  MARQ  91
C    2.  WHEN CUTSTEPS ARE USED IN A LINE SEARCH (WHEN THE ANGLE        MARQ  92
C        GAMMA .LT. GAMMA SUB ZERO), THE VALUE OF LAMBDA IS INCREASED   MARQ  93
C        PROPORTIONATELY.                                               MARQ  94
C  IN BOTH THE MODIFIED MARQUARDT-S METHOD AND THE MODIFIED             MARQ  95
C  GAUSS-NEWTON METHOD, A HALF STEP IS ATTEMPTED FOLLOWING EACH         MARQ  96
C  SUCCESSFUL STEP.  THIS AVOIDS ONE FORM OF VERY SLOW CONVERGENCE.     MARQ  97
C                                                                       MARQ  98
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * MARQ  99
C                                                                       MARQ 100
C  THE FOLLOWING EXTERNAL STATEMENT IS REQUIRED BY SOME COMPILERS       MARQ 101
C  (WATFIV, FOR EXAMPLE) AND FORBIDDEN BY OTHERS (MODCOMP II).          MARQ 102
      EXTERNAL FUNK                                                     MARQ 103
C                                                                       MARQ 104
C  ON A MACHINE HAVING LESS THAN ABOUT TEN SIGNIFICANT DIGITS IN        MARQ 105
C  SINGLE PRECISION (FOR EXAMPLE THE IBM 360 OR 370), IF P IS           MARQ 106
C  COMPUTED USING FINITE DIFFERENCES (KORDF.LT.3), THIS PART OF THE     MARQ 107
C  COMPUTATION SHOULD BE DONE IN DOUBLE PRECISION.  TO ACCOMPLISH       MARQ 108
C  THIS, ACTIVATE THE DOUBLE PRECISION STATEMENT IN EACH SUBROUTINE.    MARQ 109
C  ORDINARILY THE OTHER COMPUTATIONS MAY BE DONE IN SINGLE PRECISION.   MARQ 110
C                                                                       MARQ 111
      DOUBLE PRECISION X,XSAVE,XTEMP,GRAD,FIT,FITSV,                    MARQ 112
     *                 XSAV,SIG,RTERM,YY,PHI,PHNEW,PHALF,XLIM,STFAC     MARQ 113
C                                                                       MARQ 114
C  THE DIMENSIONS OF THE VECTORS AND MATRICES (AS OPPOSED TO ARRAYS)    MARQ 115
C  ARE....                                                              MARQ 116
C       P(NPTS,NACTV) (OR P(1,NACTV) IF KALCP.EQ.-1),                   MARQ 117
C       FITSV(NPTS) (OR FITSV(1) IF KALCP.EQ.-1),                       MARQ 118
C       X(NV),XMAX(NV),XMIN(NV),DELMN(NV),ERR(NV,NV+1),                 MARQ 119
C       XSAVE(NV),H(NV),MASKT(NV),                                      MARQ 120
C       GRAD(NACTV),SCALE(NACTV),                                       MARQ 121
```

```
C       Y(NPTS),FIT(NPTS),YSIG(NPTS) (OR YSIG(1) IF LEQU.NE.0),          MARQ 122
C  WHERE NACTV IS THE NUMBER OF ACTIVE (UNMASKED) X(J).                  MARQ 123
C                                                                       MARQ 124
      DIMENSION P(LPCOL,1)                                              MARQ 125
      DIMENSION Y(1),YSIG(1),FIT(1),FITSV(1)                            MARQ 126
      DIMENSION XSAVE(20),H(20),GRAD(20),SCALE(20),MASKT(20),XTEMP(20)  MARQ 127
C                                                                       MARQ 128
C  USER COMMON.....                                                     MARQ 129
      COMMON /CSTEP/ X(20),XMAX(20),XMIN(20),DELTX(20),DELMN(20),       MARQ 130
     *     ERR(20,21),FOBJ,NV,NTRAC,MATRX,MASK(20),                     MARQ 131
     *     NF,MAX,NFLAT,JVARY,NXTRA,KFLAG,NOREP,KERFL,KM                MARQ 132
C                                                                       MARQ 133
C  MARQ COMMON.....                                                     MARQ 134
      COMMON /NLLS2/FLAMB,RELDF,METHD,KALCP,KORDF,MAXIT,LEQU,MXSUB,MXUPDMARQ 135
C                                                                       MARQ 136
C  SET THE LIBRARY FUNCTION FOR SINGLE PRECISION (SQRT) OR FOR          MARQ 137
C  DOUBLE PRECISION (DSQRT).  NO OTHER FUNCTIONS ARE USED, EITHER       MARQ 138
C  EXTERNAL OR INTRINSIC.                                               MARQ 139
C  THE ONLY SUBROUTINES CALLED ARE FUNC, DERIV, CALCD, AND MUERR.       MARQ 140
C                                                                       MARQ 141
      DSQRT(ARG)=SQRT(ARG)                                              MARQ 142
C                                                                       MARQ 143
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * MARQ 144
C                                                                       MARQ 145
C  SET FIXED QUANTITIES ....                                            MARQ 146
C                                                                       MARQ 147
C                              HUGE ...  A VERY LARGE REAL NUMBER       MARQ 148
C                                   (DEFAULT VALUE FOR XMAX AND -XMIN)  MARQ 149
      HUGE=1.E37                                                        MARQ 150
C                                                                       MARQ 151
C  NVMAX IS THE MAXIMUM PERMISSIBLE VALUE OF NV.  IT IS ALSO THE        MARQ 152
C  DIMENSION OF THE ARRAYS X, XMAX, XMIN, MASK, DELMN,                  MARQ 153
C  XSAVE, H, MASKT, GRAD, AND SCALE, AND THE FIRST DIMENSION OF ERR.    MARQ 154
C  THE SECOND DIMENSION OF ERR IS NVMAX+1.                              MARQ 155
C                                                                       MARQ 156
      NVMAX=20                                                          MARQ 157
C                              FNU ...  MARQUARDT-S NU, THE FACTOR BY   MARQ 158
C                                            WHICH FLAMB IS CHANGED     MARQ 159
      FNU=10.                                                           MARQ 160
C                              CRIT ...  COSINE OF MARQUARDT-S CRITICAL MARQ 161
C                                            ANGLE, GAMMA SUB ZERO      MARQ 162
      CRIT=.70711                                                       MARQ 163
C                              FLDEF ...  DEFAULT VALUE FOR FLAMB       MARQ 164
      FLDEF=1.                                                          MARQ 165
C                              RELMN ...  USED TO SET DEFAULT VALUE OF  MARQ 166
C                                            DELMN                      MARQ 167
      RELMN=1.E-5                                                       MARQ 168
C                              RLTOL ...  TOLERANCE FOR A WARNING MESSAGE MARQ 169
      RLTOL=1.E-5                                                       MARQ 170
C                                                                       MARQ 171
      RZERO=0.                                                          MARQ 172
      RUNIT=1.                                                          MARQ 173
      RTWO=2.                                                           MARQ 174
C                                                                       MARQ 175
C  NO REAL CONSTANTS ARE USED BEYOND THIS POINT.                        MARQ 176
C                                                                       MARQ 177
      NTRPT=NTRAC-(-2)                                                  MARQ 178
      NTRPU=NTRAC-(-1)                                                  MARQ 179
      NTRMU=NTRAC-1                                                     MARQ 180
      KFLAG=0                                                           MARQ 181
      NOREP=0                                                           MARQ 182
      ITER=0                                                            MARQ 183
      PHI=HUGE                                                          MARQ 184
C                                                                       MARQ 185
      IF(NTRPU)1020,1000,1000                                           MARQ 186
 1000 WRITE(KW,1010)                                                    MARQ 187
 1010 FORMAT(49H1MARQ....  BEGIN NONLINEAR LEAST SQUARES SOLUTION)      MARQ 188
 1020 NACTV=0                                                           MARQ 190
      IF(NV)1070,1070,1030                                              MARQ 191
 1030 IF(NV-NVMAX)1040,1040,1070                                        MARQ 192
 1040 IF(NPTS)1070,1070,1050                                            MARQ 193
 1050 IF(KALCP)1080,1060,1060                                           MARQ 194
 1060 IF(NPTS-LPCOL)1080,1080,1070                                      MARQ 195
 1070 KFLAG=-1                                                          MARQ 196
      GO TO 1210                                                        MARQ 197
C                                                                       MARQ 198
C  CHECK SOME INPUT QUANTITIES, AND SET THEM TO DEFAULT VALUES IF       MARQ 199
C  DESIRED.                                                             MARQ 200
C                                  NACTV ...  NUMBER OF ACTIVE X(J)     MARQ 201
 1080 DO 1190 JX=1,NV                                                   MARQ 202
        DELN=DELMN(JX)                                                  MARQ 203
        IF(MASK(JX))1190,1090,1190                                      MARQ 204
 1090   NACTV=NACTV+1                                                  MARQ 205
        IF(DELN)1120,1100,1130                                          MARQ 206
 1100   DELN=RELMN*X(JX)                                               MARQ 207
        IF(DELN)1120,1110,1130                                          MARQ 208
 1110   DELN=RELMN                                                     MARQ 209
        GO TO 1130                                                      MARQ 210
 1120   DELN=-DELN                                                     MARQ 211
 1130   IF(XMAX(JX)-XMIN(JX))1140,1140,1150                            MARQ 212
 1140   XMAX(JX)=HUGE                                                  MARQ 213
        XMIN(JX)=-HUGE                                                  MARQ 214
C                                                                       MARQ 215
C  SET X(JX) TO AMAX1(XMIN(JX),AMIN1(XMAX(JX),X(JX))).                 MARQ 216
C                                                                       MARQ 217
 1150   IF(X(JX)-XMAX(JX))1170,1170,1160                               MARQ 218
 1160   X(JX)=XMAX(JX)                                                 MARQ 219
 1170   IF(X(JX)-XMIN(JX))1180,1190,1190                               MARQ 220
 1180   X(JX)=XMIN(JX)                                                 MARQ 221
 1190   DELMN(JX)=DELN                                                 MARQ 222
C                                                                       MARQ 223
      IF(NACTV)1200,1200,1230                                           MARQ 224
 1200 KFLAG=-2                                                          MARQ 225
 1210 CONTINUE                                                          MARQ 226
      WRITE(KW,1220)NV,NVMAX,NACTV,NPTS,LPCOL,KALCP                     MARQ 227
 1220 FORMAT(//29H ILLEGAL INPUT VALUE IN MARQ..4X,5H NV =,I3,4X,       MARQ 228
     *     8H NVMAX =,I3,4X,8H NACTV =,I3/10X,7H NPTS =,I5,4X,          MARQ 229
     *     8H LPCOL =,I5,4X,8H KALCP =,I3)                              MARQ 230
      GO TO 3690                                                        MARQ 231
 1230 CONTINUE                                                          MARQ 232
      IF(NTRPU)1310,1240,1240                                           MARQ 233
 1240 WRITE(KW,1250)(MASK(J),J=1,NV)                                    MARQ 234
 1250 FORMAT(/10H MASK    = ,I7,7I13/(4X,8I13))                         MARQ 235
      WRITE(KW,1260)(X(J),J=1,NV)                                       MARQ 236
 1260 FORMAT(/10H X       = ,8E13.5/(10X,8E13.5))                       MARQ 237
      WRITE(KW,1270)(XMAX(J),J=1,NV)                                    MARQ 238
 1270 FORMAT(/10H XMAX    = ,8E13.5/(10X,8E13.5))                       MARQ 239
      WRITE(KW,1280)(XMIN(J),J=1,NV)                                    MARQ 240
 1280 FORMAT(/10H XMIN    = ,8E13.5/(10X,8E13.5))                       MARQ 241
      WRITE(KW,1290)(DELMN(J),J=1,NV)                                   MARQ 242
```

```
 1290 FORMAT(/10H DELMN   = ,8E13.5/(10X,8E13.5))                        MARQ 247
      WRITE(KW,1300)NV,NPTS,LPCOL,NTRAC,METHD,KALCP,KORDF,NFLAT,        MARQ 244
     *      NFMAX,MAXIT,MXSUB,CRIT                                      MARQ 245
 1300 FORMAT(//6H NV = ,I4,5X,8H NPTS = ,I6,5X,9H LPCOL = ,I6,          MARQ 246
     *      5X,9H NTRAC = ,I2,5X,9H METHD = ,I2,5X,9H KALCP = ,I2,      MARQ 247
     *      5X,9H KORDF = ,I2//9H NFLAT = ,I2,5X,9H NFMAX = ,I7,        MARQ 248
     *      5X,9H MAXIT = ,I5,5X,9H MXSUB = ,I5,5X,                     MARQ 249
     *      28H COSINE OF CRITICAL ANGLE = ,E12.5)                      MARQ 250
 1310 JWARY=0                                                           MARQ 251
C                                                                       MARQ 252
C  SET FMGN, FLUPB, AND, IF NECESSARY, FLAMB.                           1000 252
C                                                                       MARQ 254
      FMGN=RUNIT                                                        MARQ 255
      FLUPB=RTWO                                                        1000 255
      IF(METHD)1320,1320,1330                                           MARQ 256
 1320 FLAMB=RZERO                                                       MARQ 257
      GO TO 1390                                                        1000 257
 1330 IF(FLAMB)1340,1340,1350                                           MARQ 259
 1340 FLAMB=FLDEF                                                       MARQ 260
 1350 IF(METHD-2)1390,1380,1390                                         1000 260
 1380 FLUPB=RUNIT                                                       MARQ 266
C                                                                       MARQ 267
C  COMPUTE THE INITIAL GOODNESS OF FIT OF THE MODEL TO THE DATA.        MARQ 268
C  CALL FUNC TO CALCULATE THE VECTOR OF FITTED VALUES.                  MARQ 269
C                                                                       MARQ 270
 1390 CALL FUNC (FUNK,Y,YSIG,NPTS,FIT,PHI)                              MARQ 271
C                                                                       MARQ 272
C                              NF ...  EQUIVALENT NUMBER OF CALLS TO FUNC MARQ 273
      NF=1                                                              MARQ 274
      IF(NTRPU)1420,1400,1400                                           MARQ 275
 1400 WRITE(KW,1410)PHI,FLAMB                                           MARQ 276
 1410 FORMAT(//28H PHI (THE SUM OF SQUARES) = ,E15.8,56X,9H LAMBDA =,    MARQ 277
     *      E12.5//1H )                                                 MARQ 278
C                                                                       MARQ 279
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * MARQ 280
C                                                                       MARQ 281
C  BEGIN THE NEXT ITERATION.                                            MARQ 282
C  THIS IS THE ENTRY POINT AFTER A SUCCESSFUL STEP IF THE CONVERGENCE    MARQ 283
C  CRITERION IS NOT MET.                                                MARQ 284
C                                                                       MARQ 285
 1420 JSUB=0                                                            MARQ 286
      ITER=ITER+1                                                       MARQ 287
      IF(NTRMU)1425,1421,1421                                           1000 287
 1421 WRITE(KW,1422)ITER,FMGN,FLAMB                                     2000 287
 1422 FORMAT(//17H BEGIN ITERATION ,I4,44X,7H FMGN =,E12.5,15X,         3000 287
     X      9H LAMBDA =,E12.5)                                          4000 287
 1425 IF(NTRAC-3)1450,1430,1430                                         MARQ 288
 1430 WRITE(KW,1440)                                                    MARQ 289
 1440 FORMAT(/28H P (THE JACOBIAN MATRIX)..../1H )                      MARQ 290
C                                                                       MARQ 291
C                              INITIALIZE FOR THIS ITERATION.           MARQ 292
 1450 STFAC=RUNIT                                                       MARQ 293
      DO 1460 JX=1,NACTV                                                MARQ 294
        GRAD(JX)=RZERO                                                  MARQ 295
        DO 1460 KX=1,NACTV                                              MARQ 296
 1460     ERR(JX,KX)=RZERO                                              MARQ 297
C                                                                       MARQ 298
C  CALL DERIV (OR CALCD) TO COMPUTE THE JACOBIAN MATRIX, P.             MARQ 299
C  DERIV IS CALLED NPTS TIMES IF KALCP=-1.                              MARQ 300
C                                                                       MARQ 301
      SIG=YSIG(1)                                                       MARQ 302
      DO 1590 JPT=1,NPTS                                                MARQ 303
        KPT=JPT                                                         MARQ 304
        IF(KALCP)1470,1490,1490                                         MARQ 305
 1470   KPT=1                                                           MARQ 306
        IF(KORDF-2)1500,1500,1480                                       MARQ 307
 1480   CALL CALCD (JPT,P,LPCOL)                                        MARQ 308
        GO TO 1510                                                      MARQ 309
 1490   IF(JPT-1)1510,1500,1510                                         MARQ 310
 1500   CALL DERIV (JPT,FUNK,NPTS,FIT,FITSV,P,LPCOL)                    MARQ 311
 1510   CONTINUE                                                        MARQ 312
        IF(NTRAC-3)1540,1520,1520                                       MARQ 313
 1520   WRITE(KW,1530)JPT,(P(KPT,JX),JX=1,NACTV)                        MARQ 314
 1530   FORMAT(1X,I3,2X,6E15.7/(6X,6E15.7))                             MARQ 315
C                                                                       MARQ 316
C  COMPUTE QSAV AND GRAD.                                               MARQ 317
C  QSAV, WHICH IS STORED IN ONE HALF OF THE ARRAY ERR(*,*), IS PT*P.     MARQ 318
C  GRAD IS EQUAL TO HALF THE NEGATIVE OF THE GRADIENT VECTOR.           MARQ 319
C                                                                       MARQ 320
 1540   IF(LEQU)1560,1550,1560                                          MARQ 321
 1550   SIG=YSIG(JPT)                                                   MARQ 322
 1560   YY=Y(JPT)                                                       MARQ 323
        RTERM=(FIT(JPT)-YY)/SIG**2                                      MARQ 324
        DO 1580 JX=1,NACTV                                              MARQ 325
          GRAD(JX)=GRAD(JX)-P(KPT,JX)*RTERM                             MARQ 326
          PTERM=P(KPT,JX)/SIG**2                                        MARQ 327
          DO 1570 KX=1,JX                                               MARQ 328
 1570       ERR(JX,KX)=ERR(JX,KX)+P(KPT,KX)*PTERM                       MARQ 329
 1580     CONTINUE                                                      MARQ 330
 1590   CONTINUE                                                        MARQ 331
C                              RESTORE FIT IF IT WAS DESTROYED IN DERIV. MARQ 332
      IF(KORDF-2)1630,1600,1630                                         MARQ 333
 1600 IF(KALCP)1620,1610,1620                                           MARQ 334
 1610 CALL FUNC (FUNK,Y,YSIG,NPTS,FIT,PHI)                              MARQ 335
      NF=NF+1                                                           MARQ 336
 1620 NF=NF+NACTV                                                       MARQ 337
 1630 NF=NF+NACTV                                                       MARQ 338
C                                                                       MARQ 339
C  COMPUTE THE SCALE FACTORS AND SCALE GRAD.                            MARQ 340
C                                                                       MARQ 341
      DO 1660 JX=1,NACTV                                                MARQ 342
        SCALJ=QSQRT(ERR(JX,JX))                                         MARQ 343
        IF(SCALJ)1650,1640,1650                                         MARQ 344
 1640   SCALJ=RUNIT                                                     MARQ 345
 1650   SCALE(JX)=SCALJ                                                 MARQ 346
 1660   GRAD(JX)=GRAD(JX)/SCALJ                                         MARQ 347
      IF(NTRMU)1690,1670,1670                                           MARQ 348
 1670 WRITE(KW,1680)(GRAD(JX),JX=1,NACTV)                               MARQ 349
 1680 FORMAT(/2X,19H SCALED GRADIENT = ,6E15.7/(21X,6E15.7))            MARQ 350
C                                                                       MARQ 351
C  SCALE QSAV.  THE DIAGONAL ELEMENTS OF QSAV ARE SCALED TO UNITY.      MARQ 352
C                                                                       MARQ 353
 1690 DO 1770 JX=1,NACTV                                                MARQ 354
        DO 1770 KX=1,JX                                                 MARQ 355
          SA=ERR(JX,KX)/(SCALE(JX)*SCALE(KX))                           MARQ 356
          IF(KX-JX)1720,1700,1720                                       MARQ 357
 1700     IF(SA)1710,1740,1710                                          MARQ 358
 1710     SA=RUNIT                                                      MARQ 359
          GO TO 1770                                                    MARQ 360
 1720     IF(SA-(RUNIT-RLTOL))1730,1740,1740                            MARQ 361
```

```
 1730      IF(SA+(RUNIT-RLTOL))1740,1740,1770                              MARQ 362
 1740      CONTINUE                                                        MARQ 363
           IF(NTRPT)1770,1750,1750                                         MARQ 364
 1750      WRITE(KW,1760)JX,KX,SA                                          MARQ 365
 1760      FORMAT(38H ******** POSSIBLY DANGEROUS VALUE OF ,               MARQ 366
     *        16H COEFFICIENT.....,5X,6H QSAV(,I3,1H,,I3,4H) = ,E14.7)    MARQ 367
 1770      ERR(JX,KX)=SA                                                   MARQ 368
       IF(NTRAC-2)1810,1780,1780                                           MARQ 369
 1780  WRITE(KW,1790)                                                      MARQ 370
 1790  FORMAT(/49H QSAV (PT=P, SCALED, WHERE P IS THE JACOBIAN)..../1H )   MARQ 371
       DO 1800 JX=1,NACTV                                                  MARQ 372
 1800    WRITE(KW,1530)JX,(ERR(JX,KX),KX=1,JX)                             MARQ 373
 1810  DO 1820 JX=1,NV                                                     MARQ 374
 1820    XSAVE(JX)=X(JX)                                                   MARQ 375
C                              INITIALIZE MASKT AND NACT.                 MARQ 376
 1830  NACT=NACTV                                                          MARQ 377
       DO 1840 JX=1,NV                                                     MARQ 378
 1840    MASKT(JX)=MASK(JX)                                                MARQ 379
C                                                                          MARQ 380
C  COPY QSAV INTO Q AND GRAD INTO H, AND SET THE DIAGONAL ELEMENTS OF Q.   MARQ 381
C  THIS IS THE ENTRY POINT FOR SUBITERATIONS IN WHICH FLAMB IS             MARQ 382
C  INCREASED OR CONSTRAINTS ARE IMPOSED.                                   MARQ 383
C                                                                          MARQ 384
 1850  KRANK=0                                                             MARQ 385
       JQ=0                                                                MARQ 386
       JT=0                                                                MARQ 387
       DO 1940 JX=1,NV                                                     MARQ 388
         IF(MASK(JX))1940,1860,1940                                        MARQ 389
 1860    JQ=JQ+1                                                           MARQ 390
         IF(MASKT(JX))1940,1870,1940                                       MARQ 391
 1870    JT=JT+1                                                           MARQ 392
         H(JT)=GRAD(JQ)                                                    MARQ 393
         KQ=0                                                              MARQ 394
         KT=0                                                              MARQ 395
         DO 1930 KX=1,JX                                                   MARQ 396
           IF(MASK(KX))1930,1880,1930                                      MARQ 397
 1880      KQ=KQ+1                                                         MARQ 398
           IF(MASKT(KX))1930,1890,1930                                     MARQ 399
 1890      KT=KT+1                                                         MARQ 400
           SA=ERR(JQ,KQ)                                                   MARQ 401
           IF(KX-JX)1920,1900,1920                                         MARQ 402
 1900      IF(SA)1910,1920,1910                                            MARQ 403
 1910      SA=RUNIT+FLAMB                                                  MARQ 404
           KRANK=KRANK+1                                                   MARQ 405
 1920      ERR(KT,JT+1)=SA                                                 MARQ 406
 1930      CONTINUE                                                        MARQ 407
 1940    CONTINUE                                                          MARQ 408
C                                                                          MARQ 409
C  SOLVE THE NORMAL EQUATIONS FOR H, THE CORRECTION VECTOR.                MARQ 410
C  THE METHOD USED IS GAUSSIAN ELIMINATION WITHOUT PIVOTING.               MARQ 411
C  (PIVOTING IS NOT NECESSARY FOR A POSITIVE DEFINITE MATRIX.)             MARQ 412
C  ONLY ABOUT N**3/6 MULTIPLICATIONS ARE DONE.                             MARQ 413
C  THE CHOICE OF GAUSSIAN ELIMINATION RATHER THAN CHOLESKY                 MARQ 414
C  DECOMPOSITION IS INTENTIONAL.                                           MARQ 415
C                                                                          MARQ 416
       NSMAL=0                                                             MARQ 417
       NMU=NACT-1                                                          MARQ 418
       IF(NMU)2100,2010,1950                                               MARQ 419
C                              REDUCE THE SYSTEM TO TRIANGULAR FORM,       MARQ 420
C                                   UTILIZING THE SYMMETRY OF THE MATRIX.  MARQ 421
 1950  DO 2000 J=1,NMU                                                     MARQ 422
         PIVOT=ERR(J,J+1)                                                  MARQ 423
         IF(PIVOT)1960,2000,1960                                           MARQ 424
 1960    JPU=J+1                                                           MARQ 425
         DO 1990 K=JPU,NACT                                                MARQ 426
           EM=ERR(J,K+1)/PIVOT                                             MARQ 427
           IF(EM)1970,1990,1970                                            MARQ 428
 1970      DO 1980 L=K,NACT                                                MARQ 429
 1980        ERR(K,L+1)=ERR(K,L+1)-ERR(J,L+1)*EM                           MARQ 430
 1990      H(K)=H(K)-H(J)*EM                                               MARQ 431
 2000    CONTINUE                                                          MARQ 432
C                                DO THE BACK SOLUTION.                     MARQ 433
 2010  DO 2090 JINV=1,NACT                                                 MARQ 434
         J=(NACT+1)-JINV                                                   MARQ 435
         PIVOT=ERR(J,J+1)                                                  MARQ 436
         IF(PIVOT)2020,2020,2030                                           MARQ 437
 2020    NSMAL=NSMAL+1                                                     MARQ 438
 2030    IF(PIVOT)2050,2040,2050                                           MARQ 439
 2040    H(J)=RZERO                                                        MARQ 440
         GO TO 2090                                                        MARQ 441
 2050    SUM=RZERO                                                         MARQ 442
         IF(J-NACT)2060,2080,2080                                          MARQ 443
 2060    JPU=J+1                                                           MARQ 444
         DO 2070 K=JPU,NACT                                                MARQ 445
 2070      SUM=SUM+ERR(J,K+1)*H(K)                                         MARQ 446
 2080    H(J)=(H(J)-SUM)/PIVOT                                             MARQ 447
 2090    CONTINUE                                                          MARQ 448
C                                                                          MARQ 449
 2100  MRANK=NACT-NSMAL                                                    MARQ 450
C                                                                          MARQ 451
C  IF THE COEFFICIENT MATRIX WAS RANK DEFICIENT, PRINT A MESSAGE.          MARQ 452
C                                                                          MARQ 453
       IF(MRANK-NACT)2110,2180,2110                                        MARQ 454
 2110  COSIN=HUGE                                                          MARQ 455
       IF(NTRPT)2140,2120,2120                                             MARQ 456
 2120  WRITE(KW,2130)MRANK,NACT                                            MARQ 457
 2130  FORMAT(/41H RANK-DEFICIENT NORMAL EQUATIONS IN MARQ.,9X,             1000 457
     X      7H RANK =,I3,7X,18H ORDER OF MATRIX =,I3)                       2000 457
 2140  IF(MRANK)2150,2150,2160                                             MARQ 460
 2150  KFLAG=-4                                                            MARQ 461
       GO TO 3470                                                          MARQ 462
 2160  IF(METHD)2180,2180,2170                                             MARQ 463
 2170  IF(MRANK-KRANK)2770,2180,2180                                       MARQ 464
C                                                                          MARQ 465
C  UNPACK AND DE-SCALE THE CORRECTION VECTOR H.                            MARQ 466
C  COMPUTE THE INNER PRODUCTS SA, SB, AND SC.                              MARQ 467
C                                                                          MARQ 468
 2180  SA=RZERO                                                            MARQ 469
       SB=RZERO                                                            MARQ 470
       SC=RZERO                                                            MARQ 471
       KX=NV                                                               MARQ 472
       KQ=NACTV                                                            MARQ 473
       KT=NACT                                                             MARQ 474
       DO 2230 JX=1,NV                                                     MARQ 475
         HH=RZERO                                                          MARQ 476
         IF(MASKT(KX))2190,2200,2190                                       MARQ 477
 2190    IF(MASK(KX))2220,2210,2220                                        MARQ 478
 2200    HH=H(KT)                                                          MARQ 479
         SA=SA+HH*GRAD(KQ)                                                 MARQ 480
         SB=SB+HH*HH                                                       MARQ 481
```

```
      SC=SC+GRAD(KQ)**2                                                 MARQ 482
      HH=HH*FMGN/SCALE(KQ)                                              MARQ 483
      KT=KT-1                                                           MARQ 484
 2210 KQ=KQ-1                                                           MARQ 485
 2220 H(KX)=HH                                                          MARQ 486
 2230 KX=KX-1                                                           MARQ 487
C                                                                       MARQ 488
C  ADD THE CORRECTION VECTOR TO THE PARAMETER VECTOR AND                MARQ 489
C  INSURE THAT NO CONSTRAINTS ARE VIOLATED.                             MARQ 490
C  THIS IS THE ENTRY POINT FOLLOWING A CUTSTEP.                         MARQ 491
C                                                                       MARQ 492
 2240 CONTINUE                                                          MARQ 493
      IF(NTRMU)2270,2250,2250                                           MARQ 494
 2250 WRITE(KW,2260)(H(JX),JX=1,NV)                                     MARQ 495
 2260 FORMAT(/7X,14H CORRECTION = ,6E15.7/(21X,6E15.7))                 MARQ 496
 2270 NACSV=NACT                                                        MARQ 497
      FRMIN=RUNIT                                                       MARQ 498
      NLOOP=0                                                           MARQ 499
      JXLIM=0                                                           MARQ 500
 2280 DO 2500 JX=1,NV                                                   MARQ 501
      IF(MASKT(JX))2500,2290,2500                                       MARQ 502
 2290 XSAV=XSAVE(JX)                                                    MARQ 503
      XMX=XMAX(JX)                                                      MARQ 504
      XMN=XMIN(JX)                                                      MARQ 505
      HH=H(JX)                                                          MARQ 506
      IF(XSAV-XMX)2310,2300,2300                                        MARQ 507
 2300 IF(HH)2310,2310,2330                                              MARQ 508
 2310 IF(XSAV-XMN)2320,2320,2360                                        MARQ 509
 2320 IF(HH)2330,2360,2360                                              MARQ 510
 2330 MASKT(JX)=1                                                       MARQ 511
      NACT=NACT-1                                                       MARQ 512
      X(JX)=XSAV                                                        MARQ 513
      IF(NTRPU)2500,2340,2340                                           MARQ 514
 2340 WRITE(KW,2350)JX,XSAV                                             MARQ 515
 2350 FORMAT(/8H FIX  X(,I3,4H) = ,E12.5,                               MARQ 516
     *       46H TEMPORARILY, TO AVOID VIOLATING A CONSTRAINT. )        MARQ 517
      GO TO 2500                                                        MARQ 518
 2360 XLIM=XSAV+HH*FRMIN                                                MARQ 519
      IF(JX-JXLIM)2380,2370,2380                                        MARQ 520
 2370 IF(KBND)2410,2390,2390                                            MARQ 521
 2380 X(JX)=XLIM                                                        MARQ 522
      IF(XLIM-XMX)2400,2400,2390                                        MARQ 523
 2390 X(JX)=XMX                                                         MARQ 524
      JBND=1                                                            MARQ 525
      GO TO 2420                                                        MARQ 526
 2400 IF(XLIM-XMN)2410,2500,2500                                        MARQ 527
 2410 X(JX)=XMN                                                         MARQ 528
      JBND=-1                                                           MARQ 529
 2420 IF(JX-JXLIM)2460,2430,2460                                        MARQ 530
 2430 CONTINUE                                                          MARQ 531
      IF(NTRPU)2500,2440,2440                                           MARQ 532
 2440 WRITE(KW,2450)JX,X(JX),FRMIN                                      MARQ 533
 2450 FORMAT(/26H CONSTRAINT VIOLATED BY X(,I3,                         MARQ 534
     *       20H).   VALUE RESET TO ,E15.8,24H USING CUTSTEP FACTOR = , MARQ 535
     *       E12.5 )                                                    MARQ 536
      GO TO 2500                                                        MARQ 537
 2460 IF(NLOOP)2500,2470,2500                                           MARQ 538
 2470 DENOM=XLIM-XSAV                                                   MARQ 539
      IF(DENOM)2480,2500,2480                                           MARQ 540
 2480 FRAC=(X(JX)-XSAV)/DENOM                                           MARQ 541
      IF(FRAC-FRMIN)2490,2500,2500                                      MARQ 542
 2490 FRMIN=FRAC                                                        MARQ 543
      JXLIM=JX                                                          MARQ 544
      KBND=JBND                                                         MARQ 545
 2500 CONTINUE                                                          MARQ 546
C                                                                       MARQ 547
C  IF THE PROPOSED STEP WOULD VIOLATE ANY ALREADY ACTIVE CONSTRAINTS,   MARQ 548
C  FIX THOSE COMPONENTS OF H EQUAL TO ZERO AND RECOMPUTE THE            MARQ 549
C  OTHER COMPONENTS.                                                    MARQ 550
C                                                                       MARQ 551
      IF(NACT)2510,2510,2540                                            MARQ 552
 2510 KFLAG=3                                                           MARQ 553
      IF(NTRPT)3470,2520,2520                                           MARQ 554
 2520 WRITE(KW,2530)                                                    MARQ 555
 2530 FORMAT(////47H APPARENT CONSTRAINED OPTIMUM LIES IN A CORNER. )    MARQ 556
      GO TO 3470                                                        MARQ 557
 2540 IF(NACT-NACSV)1850,2550,2550                                      MARQ 558
 2550 IF(NLOOP)2570,2560,2570                                           MARQ 559
 2560 NLOOP=1                                                           MARQ 560
      IF(JXLIM)2280,2570,2280                                           MARQ 561
 2570 CONTINUE                                                          MARQ 562
      IF(NTRMU)2600,2580,2580                                           MARQ 563
 2580 WRITE(KW,2590)(X(JX),JX=1,NV)                                     MARQ 564
 2590 FORMAT(/16X,5H X = ,6E15.7/(21X,6E15.7))                          MARQ 565
C                                                                       MARQ 566
C  CALCULATE THE NEW FITTED VALUES.                                     MARQ 567
C                                                                       MARQ 568
 2600 CALL FUNC (FUNK,Y,YSIG,NPTS,FIT,PHNEW)                            MARQ 569
      NF=NF+1                                                           MARQ 570
      IF(PHNEW-PHI)2910,2610,2650                                       MARQ 571
C                                                                       MARQ 572
C  THE NEW VALUE OF PHI IS EXACTLY EQUAL TO THE OLD VALUE.              MARQ 573
C  CHECK FOR CONVERGENCE UNDER THE NFLAT OPTION.                        MARQ 574
C                                                                       MARQ 575
 2610 IF(NFLAT)2620,2910,2620                                           MARQ 576
 2620 KFLAG=2                                                           MARQ 577
      IF(NTRPU)2730,2630,2630                                           MARQ 578
 2630 WRITE(KW,2640)                                                    MARQ 579
 2640 FORMAT(//45H CONVERGENCE ACHIEVED UNDER THE NFLAT OPTION.)        MARQ 580
      GO TO 2730                                                        MARQ 581
C                                                                       MARQ 582
C  THE NEW VALUE OF PHI IS GREATER THAN THE OLD VALUE.                  MARQ 583
C                                                                       MARQ 584
 2650 CONTINUE                                                          MARQ 585
      IF(NTRMU)2660,2660,2660                                           MARQ 586
 2660 WRITE(KW,2670)PHI,PHNEW                                           MARQ 587
 2670 FORMAT(/33X,11H OLD PHI = ,E15.8,4X,11H NEW PHI = ,E15.8)         MARQ 588
C                                                                       MARQ 590
C  INSURE THAT JSUB HAS NOT EXCEEDED MXSUB.                             MARQ 591
C                                                                       MARQ 592
 2680 JSUB=JSUB+1                                                       MARQ 593
      IF(JSUB-MXSUB)2690,2690,2700                                      MARQ 594
 2690 IF(METHD)2840,3240,2750                                           MARQ 595
 2700 KFLAG=-1                                                          MARQ 596
      IF(NTRPU)2730,2710,2710                                           MARQ 597
 2710 WRITE(KW,2720)MXSUB                                               MARQ 598
 2720 FORMAT(//44H EXCEEDED MAXIMUM NUMBER OF SUBITERATIONS = ,I5,      MARQ 599
     *       9H IN MARQ.)                                               MARQ 600
C                              RESTORE X TO THE BASE POINT.             MARQ 601
 2730 DO 2740 JX=1,NV                                                   MARQ 602
```

```
 2740    X(JX)=XSAVE(JX)                                                  MARQ 603
         CALL FUNC (FUNK,Y,YSIG,NPTS,FIT,PHI)                             MARQ 604
         GO TO 3470                                                       MARQ 605
C                                                                         MARQ 606
C  THE NEW FIT IS WORSE THAN THE OLD FIT.  COMPUTE COSIN, THE COSINE      MARQ 607
C  OF THE ANGLE BETWEEN THE SCALED GRADIENT AND THE SCALED CORRECTION     MARQ 608
C  VECTOR.                                                                MARQ 609
C                                                                         MARQ 610
 2750 DENOM=SB*SC                                                         MARQ 611
         IF(DENOM)2770,2770,2760                                          MARQ 612
 2760 COSIN=SA/DSQRT(DENOM)                                               MARQ 613
         IF(COSIN-CRIT)2770,2770,2840                                     MARQ 614
C                                                                         MARQ 615
C  COSIN IS NOT GREATER THAN CRIT.  INCREASE THE VALUE OF LAMBDA.         MARQ 616
C                                                                         MARQ 617
 2770 UPFAC=FNU                                                           MARQ 618
         IF(METHD-1)2800,2780,2800                                        MARQ 619
 2780 IF(FLAMB*(FNU-RTMO)-RUNIT)2800,2800,2790                            MARQ 620
 2790 UPFAC=RTMO+RUNIT/FLAMB                                              MARQ 621
 2800 FLAMB=FLAMB*UPFAC                                                   MARQ 622
         IF(NTRMU)2830,2810,2810                                          MARQ 623
 2810 WRITE(KW,2820)JSUB,COSIN,FLAMB                                      MARQ 624
 2820 FORMAT(/18H **** SUBITERATION,I3,4X,17H INCREASE LAMBDA.,           MARQ 625
     *      4X,14H COS(GAMMA) = ,E12.5,27X,9H LAMBDA =,E12.5)            MARQ 626
 2830 CONTINUE                                                            MARQ 627
C                                GO BACK AND FORM THE NORMAL EQUATIONS    MARQ 628
C                                     USING A LARGER VALUE OF LAMBDA.     MARQ 629
         GO TO 1830                                                       MARQ 630
C                                                                         MARQ 631
C  COSIN IS GREATER THAN CRIT.  CUT THE MAGNITUDE OF THE STEP, H.         MARQ 632
C                                                                         MARQ 633
 2840 STFAC=STFAC/RTMO                                                    MARQ 634
         FLAMB=FLAMB*FLUPB                                                MARQ 635
         IF(METHD)2850,2860,2860                                          MARQ 636
 2850 FMGN=FMGN/RTMO                                                      MARQ 637
 2860 DO 2870 JX=1,NV                                                     MARQ 638
 2870    H(JX)=(X(JX)-XSAVE(JX))/RTMO                                     MARQ 639
         IF(NTRMU)2900,2880,2880                                          MARQ 640
 2880 WRITE(KW,2890)JSUB,COSIN,STFAC                                      MARQ 641
 2890 FORMAT(/18H **** SUBITERATION,I3,4X,16H TAKE CUT STEPS.,4X,         MARQ 642
     *       14H COS(GAMMA) = ,E12.5,4X,17H CUTSTEP FACTOR =,E12.5 )      MARQ 643
 2900 CONTINUE                                                            MARQ 644
C                                GO BACK AND TRY A SMALLER CUTSTEP.       MARQ 645
         GO TO 2240                                                       MARQ 646
C                                                                         MARQ 647
C  THE VALUE OF PHI HAS DECREASED.  TRY A HALF STEP.                      MARQ 648
C                                                                         MARQ 649
 2910 IF(METHD)2930,3240,2920                                             MARQ 650
 2920 IF(METHD-2)2930,3240,2930                                           MARQ 651
 2930 DO 2980 JX=1,NV                                                     MARQ 652
        XTEMP(JX)=X(JX)                                                   MARQ 653
         IF(MASK(JX))2980,2940,2980                                       MARQ 654
 2940    X(JX)=XSAVE(JX)+(X(JX)-XSAVE(JX))/RTMO                           MARQ 655
         IF(X(JX)-XMAX(JX))2960,2960,2950                                 MARQ 656
 2950    X(JX)=XMAX(JX)                                                   MARQ 657
 2960    IF(X(JX)-XMIN(JX))2970,2980,2980                                 MARQ 658
 2970    X(JX)=XMIN(JX)                                                   MARQ 659
 2980    CONTINUE                                                         MARQ 660
       DO 2990 JPT=1,NPTS                                                 MARQ 661
 2990    FITSV(JPT)=FIT(JPT)                                              MARQ 662
         CALL FUNC (FUNK,Y,YSIG,NPTS,FIT,PHALF)                           MARQ 663
         NF=NF+1                                                          MARQ 664
C                                                                         MARQ 665
C  USE QUADRATIC INTERPOLATION, IN ORDER TO TRY TO REFINE THE             MARQ 666
C  POSITION OF THE MINIMUM OF PHI.                                        MARQ 667
C                                                                         MARQ 668
         RLFAC=RUNIT                                                      MARQ 669
         DENOM=RTMO*((PHNEW-PHALF)-(PHALF-PHI))                           MARQ 670
         IF(DENOM)3020,3020,3000                                          MARQ 671
 3000 STFAC=(PHI-PHNEW)/DENOM                                             MARQ 672
         RSFAC=(RUNIT+STFAC)/RTMO                                         MARQ 673
C                                                                         MARQ 674
C                                DO NOT EXTRAPOLATE.                      MARQ 675
         IF(STFAC-RUNIT)3030,3020,3020                                    MARQ 677
 3020 STFAC=RZERO                                                         MARQ 678
 3030 DO 3040 JX=1,NV                                                     MARQ 679
        XSAVE(JX)=X(JX)                                                   MARQ 680
 3040    X(JX)=X(JX)+(XTEMP(JX)-X(JX))*STFAC                              MARQ 681
         IF(PHALF-PHNEW)3050,3100,3100                                    MARQ 682
 3050 RLFAC=RUNIT/RTMO                                                    MARQ 684
         JSUB=JSUB+1                                                      MARQ 685
         DO 3060 JX=1,NV                                                  MARQ 686
 3060    XTEMP(JX)=XSAVE(JX)                                              MARQ 687
         DO 3070 JPT=1,NPTS                                               MARQ 688
 3070    FITSV(JPT)=FIT(JPT)                                              MARQ 689
         IF(NTRMU)3095,3080,3080                                          MARQ 690
 3080 WRITE(KW,3090)PHNEW,PHALF                                           MARQ 691
 3090 FORMAT(/21H HALF STEP SUCCEEDED.,15X,8H PHNEW =,E15.8,18X,          1000 691
     X       8H PHALF =,E15.8)                                            2000 691
 3095 PHNEW=PHALF                                                         3000 691
 3100 IF(STFAC)3110,3120,3110                                             MARQ 693
 3110 CALL FUNC (FUNK,Y,YSIG,NPTS,FIT,PHI)                                MARQ 694
         NF=NF+1                                                          MARQ 695
         IF(PHI-PHNEW)3180,3120,3120                                      MARQ 696
 3120 DO 3130 JX=1,NV                                                     MARQ 697
 3130    X(JX)=XTEMP(JX)                                                  MARQ 698
         DO 3140 JPT=1,NPTS                                               MARQ 699
 3140    FIT(JPT)=FITSV(JPT)                                              MARQ 700
         IF(STFAC)3150,3210,3150                                          MARQ 701
 3150 CONTINUE                                                            MARQ 702
         IF(NTRMU)3210,3160,3160                                          MARQ 703
 3160 WRITE(KW,3170)RSFAC,PHI                                             MARQ 704
 3170 FORMAT(/32H QUADRATIC INTERPOLATION FAILED.,15X,8H RSFAC =,E12.5,    1000 704
     X       12X,6H PHI =,E15.8)                                          2000 704
         GO TO 3210                                                       MARQ 706
 3180 RLFAC=RSFAC                                                         MARQ 707
         PHNEW=PHI                                                        MARQ 708
         IF(NTRMU)3210,3190,3190                                          MARQ 709
 3190 WRITE(KW,3200)RLFAC,PHI                                             MARQ 710
 3200 FORMAT(/35H QUADRATIC INTERPOLATION SUCCEEDED.,12X,8H RLFAC =,      MARQ 711
     *       E12.5,12X,6H PHI =,E15.8)                                    MARQ 712
 3210 IF(RLFAC)3240,3240,3220                                             MARQ 713
 3220 FLAMB=FLAMB/RLFAC                                                   MARQ 714
         IF(METHD)3230,3240,3240                                          MARQ 715
 3230 FMGN=FMGN*RLFAC                                                     MARQ 716
C                                                                         MARQ 717
C  THE STEP IS ACCEPTED.  TEST FOR CONVERGENCE IF NO CONSTRAINT           MARQ 718
C  BECAME ACTIVE DURING THIS ITERATION.                                   MARQ 719
C                                                                         MARQ 720
 3240 CONTINUE                                                            MARQ 721
```

```
      IF(NTRMU)3270,3250,3250                                          MARQ 722
 3250 WRITE(KW,3260)ITER,PHNEW                                         1000 722
 3260 FORMAT(//17H END    ITERATION ,I4,58X,6H PHI =,E15.8)            2000 722
 3270 PHI=PHNEW                                                        MARQ 726
      IF(JXLIM)3280,3280,3350                                          MARQ 727
 3280 DO 3320 JX=1,NV                                                  MARQ 728
        IF(MASK(JX))3320,3290,3320                                     MARQ 729
 3290   DIF=X(JX)-XSAVE(JX)                                           MARQ 730
        IF(DIF)3300,3310,3310                                          MARQ 731
 3300   DIF=-DIF                                                      MARQ 732
 3310   IF(DIF-DELMN(JX))3320,3320,3350                               MARQ 733
 3320   CONTINUE                                                      MARQ 734
      KFLAG=1                                                          MARQ 735
      IF(NTRPU)3470,3330,3330                                          MARQ 736
 3330 WRITE(KW,3340)                                                   MARQ 737
 3340 FORMAT(//36H CONVERGED WHEN THE STEP BECAME SMALL.)              MARQ 738
      GO TO 3470                                                       MARQ 739
C                                                                      MARQ 740
C  THE ITERATION HAS NOT YET CONVERGED.                                MARQ 741
C                                                                      MARQ 742
 3350 IF(ITER-MAXIT)3380,3380,3360                                     MARQ 743
 3360 KFLAG=-6                                                         MARQ 744
      WRITE(KW,3370)MAXIT                                              MARQ 745
 3370 FORMAT(//47H MAXIMUM NUMBER OF ITERATIONS EXCEEDED IN MARQ.,5X,  MARQ 746
     *     9H MAXIT = ,I4 )                                           MARQ 747
      GO TO 3470                                                       MARQ 748
C                                                                      MARQ 749
C  IF SUBITERATIONS WERE NOT PERFORMED THIS ITERATION, DECREASE LAMBDA. MARQ 750
C                                                                      MARQ 751
 3380 IF(NF-NFMAX)3390,3390,3450                                       MARQ 752
 3390 IF(JSUB)3400,3400,3440                                           MARQ 753
 3400 FMGN=FMGN*RTWO                                                   MARQ 754
      IF(FMGN-RUNIT)3420,3420,3410                                     MARQ 755
 3410 FMGN=RUNIT                                                       MARQ 756
 3420 SCALJ=RUNIT+FLAMB                                                MARQ 757
      IF(SCALJ-RUNIT)3440,3440,3430                                    MARQ 758
 3430 FLAMB=FLAMB/FMU                                                  MARQ 759
 3440 CONTINUE                                                         MARQ 760
C                              GO BACK AND DO ANOTHER ITERATION.       MARQ 761
      GO TO 1420                                                       MARQ 762
 3450 KFLAG=-7                                                         MARQ 763
      WRITE(KW,3460)NFMAX                                              MARQ 764
 3460 FORMAT(//25H NF HAS EXCEEDED NFMAX = ,I7,9H IN MARQ. )           MARQ 765
C                                                                      MARQ 766
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * MARQ 767
C                                                                      MARQ 768
C  THE ITERATION HAS TERMINATED.                                       MARQ 769
C  PRINT OUT THE DATA, FITTED VALUES, AND RESIDUALS.                   MARQ 770
C  COMPUTE AND PRINT THE STANDARD DEVIATION OF THE DATA FROM THE FIT.   MARQ 771
C                                                                      MARQ 772
 3470 CONTINUE                                                         MARQ 773
      IF(NTRPU)3670,3480,3480                                          MARQ 774
 3480 WRITE(KW,3490)ITER,NF,PHI,FMGN,FLAMB                             MARQ 775
 3490 FORMAT(/1X,I4,11H ITERATIONS,7X,5H NF =,I5,9X,6H PHI =,E15.8,    1000 775
     X     10X,7H FMGN =,E12.5,7X,9H LAMBDA =,E12.5)                  2000 775
      WRITE(KW,2590)(X(JX),JX=1,NV)                                    MARQ 778
      IF(NTRAC)3520,3500,3500                                          MARQ 779
 3500 WRITE(KW,3510)                                                   MARQ 780
 3510 FORMAT(///14X,1HJ,9X,5H Y(J),14X,7H FIT(J),10X,                  MARQ 781
     *     12H Y(J)-FIT(J),7X,8H YSIG(J),11X,13H (Y-FIT)/YSIG/1H )    MARQ 782
 3520 SIG=YSIG(1)                                                      MARQ 783
      RMSDV=RZERO                                                      MARQ 784
      SDVMX=RZERO                                                      MARQ 785
      DO 3610 JPT=1,NPTS                                               MARQ 786
        IF(LEQU)3540,3530,3540                                         MARQ 787
 3530   SIG=YSIG(JPT)                                                 MARQ 788
 3540   YY=Y(JPT)                                                     MARQ 789
        PTERM=YY-FIT(JPT)                                              MARQ 790
        RTERM=PTERM/SIG                                                MARQ 791
        IF(NTRAC)3570,3550,3550                                        MARQ 792
 3550   WRITE(KW,3560)JPT,YY,FIT(JPT),PTERM,SIG,RTERM                 MARQ 793
 3560   FORMAT(10X,I5,5X,E15.8,5X,E15.8,5X,E12.5,5X,E12.5,10X,E12.5)  MARQ 794
 3570   RMSDV=RMSDV+RTERM**2                                          MARQ 795
        IF(RTERM)3580,3590,3590                                        MARQ 796
 3580   RTERM=-RTERM                                                  MARQ 797
 3590   IF(RTERM-SDVMX)3610,3610,3600                                 MARQ 798
 3600   SDVMX=RTERM                                                   MARQ 799
 3610   CONTINUE                                                      MARQ 900
      DENOM=NPTS-NACTV                                                 MARQ 801
      WRITE(KW,3620)DENOM                                              MARQ 802
 3620 FORMAT(//32H NUMBER OF DEGREES OF FREEDOM = ,E12.5 )             MARQ 803
      IF(DENOM)3650,3650,3630                                          MARQ 804
 3630 RMSDV=DSQRT(RMSDV/DENOM)                                         MARQ 805
      WRITE(KW,3640)RMSDV                                              MARQ 806
 3640 FORMAT(/43H R.M.S. SCALED DEVIATION OF DATA FROM FIT =,E12.5)     MARQ 807
 3650 CONTINUE                                                         MARQ 808
      WRITE(KW,3660)SDVMX                                              MARQ 809
 3660 FORMAT(/27H MAXIMUM SCALED DEVIATION =,E12.5)                    MARQ 810
C                                                                      MARQ 811
C  CALL FUNC TO SET THE FINAL VALUES.                                  MARQ 812
C                                                                      MARQ 813
 3670 CALL FUNC (FUNK,Y,YSIG,NPTS,FIT,PHI)                             MARQ 814
      FOBJ=PHI                                                         MARQ 815
C                                                                      MARQ 816
C  CALL MQERR TO PRINT THE PARAMETER ERRORS AND CORRELATIONS.          MARQ 817
C  A DUMMY ROUTINE MAY BE SUBSTITUTED FOR MQERR IF THESE ARE NOT NEEDED.MARQ 818
C                                                                      MARQ 819
      IF(MATRX)3690,3690,3680                                          MARQ 820
 3680 CALL MQERR (NACTV,H,SCALE,NPTS)                                  MARQ 821
C                                                                      MARQ 822
 3690 RETURN                                                           MARQ 823
C  END MARQ.                                                           MARQ 824
      END                                                              MARQ 825
C                                                                      MARQ 826
      SUBROUTINE FUNC (FUNK,Y,YSIG,NPTS,FIT,PHI)                       FUNC   1
C                                                                      FUNC   2
C  FUNC CALLS FUNK OR FOFX TO COMPUTE THE ARRAY OF FITTED VALUES FITM. FUNC   3
C                                                                      FUNC   4
      DOUBLE PRECISION X,FIT,F,PHI,SIG,YY                              FUNC   5
C                                                                      FUNC   6
      DIMENSION Y(1),YSIG(1),FIT(1)                                    FUNC   7
C                                                                      FUNC   8
      COMMON /CSTEP/ X(20),XMAX(20),XMIN(20),DELTX(20),DELMN(20),      FUNC   9
     *     ERR(20,21),FOBJ,NV,NTRAC,MATRX,MASK(20),                    FUNC  10
     *     NFMAX,NFLAT,JVARY,NXTRA,KFLAG,NOREP,KERFL,KW                FUNC  11
      COMMON /NLLS2/FLAMB,RELDF,METHD,KALCP,KORDF,MAXIT,LEQU,MXSUB,MXUPDFUNC  12
C                                                                      FUNC  13
      RZERO=0.                                                         FUNC  14
      IF(KALCP)5050,5040,5050                                          FUNC  21
C                                                                      FUNC  22
```

```
 5040 CALL FUNK (FIT)                                                    FUNC  23
      GO TO 5070                                                         FUNC  24
C                                                                        FUNC  25
 5050 DO 5060 JPT=1,NPTS                                                 FUNC  26
        CALL FOFX(JPT,NV,X,F)                                            FUNC  27
 5060   FIT(JPT)=F                                                       FUNC  28
C                                                                        FUNC  29
 5070 PHI=RZERO                                                          FUNC  30
      SIG=YSIG(1)                                                        FUNC  31
      DO 5130 JPT=1,NPTS                                                 FUNC  32
        IF(LEQU)5090,5080,5090                                           FUNC  33
 5080   SIG=YSIG(JPT)                                                    FUNC  34
C                                CHECK FOR AN ILLEGAL VALUE OF SIG.      FUNC  35
 5090   IF(SIG)5100,5100,5120                                            FUNC  36
 5100   WRITE(KW,5110)LEQU,JPT,SIG                                       FUNC  37
 5110   FORMAT(/28H ERROR IN MARQ....   LEQU = ,I1,5X,6HJPT = ,I5,5X,     FUNC  38
     *     7HYSIG = ,E12.5,14H IS .LE. ZERO.)                           FUNC  39
        STOP                                                             FUNC  40
C                                                                        FUNC  41
 5120   YY=Y(JPT)                                                        FUNC  42
 5130   PHI=PHI+((FIT(JPT)-YY)/SIG)**2                                   FUNC  43
C                                                                        FUNC  44
      RETURN                                                             FUNC  45
C  END FUNC.                                                             FUNC  46
      END                                                                FUNC  47
C                                                                        FUNC  48
      SUBROUTINE DERIV (JPT,FUNK,NPTS,FIT,FITSV,P,LPCOL)                 DERIV  1
C                                                                        DERIV  2
C  DERIV 4.0          A.N.S.I. STANDARD FORTRAN          JUNE 1975      DERIV  3
C                                                                        DERIV  4
C  DERIV COMPUTES THE JACOBIAN MATRIX P USING FINITE DIFFERENCES.        DERIV  5
C                                                                        DERIV  6
C  P(J,K) IS THE PARTIAL DERIVATIVE OF FIT(J) WITH RESPECT TO X(K).      DERIV  7
C  IF KORDF.EQ.1, DERIV USES A NONCENTRAL DIFFERENCE FORMULA.            DERIV  8
C  IF KORDF.EQ.2, DERIV USES A CENTRAL DIFFERENCE FORMULA.               DERIV  9
C  KORDF.EQ.1 IS ABOUT TWICE AS FAST AS KORDF.EQ.2, BUT LESS ACCURATE.   DERIV 10
C                                                                        DERIV 11
C  ON A MACHINE HAVING LESS THAN ABOUT TEN SIGNIFICANT DIGITS IN         DERIV 12
C  SINGLE PRECISION (FOR EXAMPLE THE IBM 360 OR 370), THE                DERIV 13
C  DIFFERENCING SHOULD BE DONE IN DOUBLE PRECISION.  TO ACCOMPLISH       DERIV 14
C  THIS, ACTIVATE THE DOUBLE PRECISION STATEMENT BELOW.                  DERIV 15
C                                                                        DERIV 16
      DOUBLE PRECISION X,FIT,FITSV,DEL,TWODL,XSAVE,FX0,FX1               DERIV 17
C                                                                        DERIV 18
      DIMENSION FIT(1),FITSV(1),P(LPCOL,1)                               DERIV 19
C                                                                        DERIV 20
      COMMON /CSTEP/ X(20),XMAX(20),XMIN(20),DELTX(20),DELMN(20),        DERIV 21
     *     ERR(20,21),FOBJ,NV,NTRAC,MATRX,MASK(20),                      DERIV 22
     *     NFMAX,NFLAT,JVARY,NXTRA,KFLAG,NOREP,KERFL,KW                  DERIV 23
      COMMON /NLLS2/FLAMB,RELDF,METHD,KALCP,KORDF,MAXIT,LEQU,MXSUB,MXUPD1000V 23
C                                                                        DERIV 26
      JVARY=0                                                            DERIV 27
C                                                                        DERIV 28
C  SAVE FIT IF KALCP.GE.0 .                                              DERIV 29
      IF(KALCP)6020,6000,6000                                            DERIV 30
 6000 DO 6010 J=1,NPTS                                                   DERIV 31
 6010   FITSV(J)=FIT(J)                                                  DERIV 32
C                                                                        DERIV 33
C  LOOP OVER THE ACTIVE PARAMETERS X(JX).                                DERIV 34
C                                                                        DERIV 35
 6020 KX=0                                                               DERIV 36
      DO 6210 JX=1,NV                                                    DERIV 37
        IF(MASK(JX))6210,6030,6210                                       DERIV 38
 6030   KX=KX+1                                                          DERIV 39
        DEL=RELDF*X(JX)                                                  DERIV 40
        IF(DEL)6050,6040,6050                                            DERIV 41
 6040   DEL=RELDF                                                        DERIV 42
 6050   XSAVE=X(JX)                                                      DERIV 43
        X(JX)=XSAVE+DEL                                                  DERIV 44
        TWODL=DEL+DEL                                                    DERIV 45
        IF(KALCP)6160,6060,6120                                          DERIV 46
C                                                                        DERIV 47
C  KALCP.EQ.0 .  COMPUTE P, ONE COLUMN AT A TIME.                        DERIV 48
C                                                                        DERIV 49
 6060   CALL FUNK (FIT)                                                  DERIV 50
        IF(KORDF-2)6070,6090,6240                                        DERIV 51
 6070   DO 6080 J=1,NPTS                                                 DERIV 52
 6080     P(J,KX)=(FIT(J)-FITSV(J))/DEL                                  DERIV 53
        GO TO 6200                                                       DERIV 54
C                                                                        DERIV 55
C  KALCP.EQ.0 AND KORDF.EQ.2 .  IN THIS CASE, THE INPUT VALUES OF        DERIV 56
C  FIT(J) WILL BE DESTROYED.                                             DERIV 57
C                                                                        DERIV 58
 6090   X(JX)=XSAVE-DEL                                                  DERIV 59
        DO 6100 J=1,NPTS                                                 DERIV 60
 6100     FITSV(J)=FIT(J)                                                DERIV 61
        JVARY=JX                                                         DERIV 62
        CALL FUNK (FIT)                                                  DERIV 63
        JVARY=0                                                          DERIV 64
        DO 6110 J=1,NPTS                                                 DERIV 65
 6110     P(J,KX)=(FITSV(J)-FIT(J))/TWODL                                DERIV 66
        GO TO 6200                                                       DERIV 67
C                                                                        DERIV 68
C  KALCP.GT.0 .  COMPUTE P, ONE ELEMENT AT A TIME.                       DERIV 69
C                                                                        DERIV 70
 6120   DO 6150 J=1,NPTS                                                 DERIV 71
          CALL FOFX (J,NV,X,FX1)                                         DERIV 72
          IF(KORDF-2)6130,6140,6240                                      DERIV 73
 6130     P(J,KX)=(FX1-FITSV(J))/DEL                                     DERIV 74
          GO TO 6150                                                     DERIV 75
 6140     X(JX)=XSAVE-DEL                                                DERIV 76
          CALL FOFX (J,NV,X,FX0)                                         DERIV 77
          P(J,KX)=(FX1-FX0)/TWODL                                        DERIV 78
          X(JX)=XSAVE+DEL                                                DERIV 79
 6150     CONTINUE                                                       DERIV 80
        GO TO 6200                                                       DERIV 81
C                                                                        DERIV 82
C  KALCP.LT.0 .  COMPUTE ONE ROW OF P, ONE ELEMENT AT A TIME.            DERIV 83
C                                                                        DERIV 84
 6160   FITSV(1)=FIT(JPT)                                                DERIV 85
        CALL FOFX (JPT,NV,X,FX1)                                         DERIV 86
        IF(KORDF-2)6170,6180,6170                                        DERIV 87
 6170   P(1,KX)=(FX1-FITSV(1))/DEL                                       DERIV 88
        GO TO 6190                                                       DERIV 89
 6180   X(JX)=XSAVE-DEL                                                  DERIV 90
        CALL FOFX (JPT,NV,X,FX0)                                         DERIV 91
        P(1,KX)=(FX1-FX0)/TWODL                                          DERIV 92
 6190   FIT(JPT)=FITSV(1)                                                DERIV 93
 6200   X(JX)=XSAVE                                                      DERIV 94
 6210   CONTINUE                                                         DERIV 95
```

```
C                                                                       DERIV 96
      IF(KALCP)6240,6220,6220                                           DERIV 97
 6220 DO 6230 J=1,NPTS                                                  DERIV 98
 6230   FIT(J)=FITSV(J)                                                 DERIV 99
C                                                                       DERIV100
 6240 RETURN                                                            DERIV101
C  END DERIV.                                                           DERIV102
      END                                                               DERIV103
C                                                                       DERIV104
      SUBROUTINE MQERR (NACTV,H,SCALE,NPTS)                             MQERR  1
C                                                                       MQERR  2
C  MQERR 1.3        A.N.S.I. STANDARD FORTRAN        FEBRUARY 1978      MQERR  3
C                                                                       MQERR  4
C  MQERR IS CALLED BY MARQ TO COMPUTE AND PRINT APPROXIMATE VALUES OF   MQERR  5
C  THE PARAMETER ERRORS AND CORRELATIONS.                               MQERR  6
C                                                                       MQERR  7
C  FOR THE MEANING OF THE -MAXIMUM VARIANCE INFLATION FACTOR- BELOW,    MQERR  8
C  SEE...      D. W. MARQUARDT AND R. D. SNEE,                          MQERR  9
C              RIDGE REGRESSION IN PRACTICE,                            MQERR 10
C              THE AMERICAN STATISTICIAN 29 (1975) 3-20                 MQERR 11
C                                                                       MQERR 12
C  INPUT QUANTITIES.....  KW,ERR,NACTV,SCALE,NPTS,NV,NTRAC,MASK,FOBJ    MQERR 13
C  OUTPUT QUANTITIES....  ERR                                           MQERR 14
C  SCRATCH STORAGE......  H                                             MQERR 15
C                                                                       MQERR 16
      DOUBLE PRECISION X                                                MQERR 17
C                                                                       MQERR 18
      DIMENSION H(1),SCALE(1)                                           MQERR 19
C                                                                       MQERR 20
      COMMON /CSTEP/ X(20),XMAX(20),XMIN(20),DELTX(20),DELMN(20),       MQERR 21
     *     ERR(20,21),FOBJ,NV,NTRAC,MATRX,MASK(20),                     MQERR 22
     *     NFMAX,NFLAT,JVARY,NXTRA,KFLAG,NOREP,KERFL,KW                 MQERR 23
C                                                                       MQERR 24
      QSQRT(ARG)=SQRT(ARG)                                              MQERR 25
C                                                                       MQERR 26
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * MQERR 27
C                                                                       MQERR 28
      RZERO=0.                                                          MQERR 29
      RUNIT=1.                                                          MQERR 30
      HUGE=1.E30                                                        MQERR 31
C                              PRINT QSAV.                              MQERR 32
      IF(NTRAC-(-1))4040,4000,4000                                      MQERR 33
 4000 WRITE(KW,4010)                                                    MQERR 34
 4010 FORMAT(///18H SUBROUTINE MQERR.//26H QSAV (PT*P, SCALED, WHERE,   MQERR 35
     *     23H P IS THE JACOBIAN)....)                                  MQERR 36
      DO 4030 JX=1,NACTV                                                MQERR 37
        WRITE(KW,4020)JX,(ERR(JX,KX),KX=1,JX)                           MQERR 38
 4020   FORMAT(/1X,I3,2X,6E15.7/(6X,6E15.7))                           MQERR 39
 4030   CONTINUE                                                       MQERR 40
C                                                                       MQERR 41
C  COMPUTE THE SCALED ERROR MATRIX, WHICH IS THE INVERSE OF QSAV.       MQERR 42
C  INVERT QSAV USING THE GAUSS-JORDAN METHOD WITHOUT PIVOTING.          MQERR 43
C  (PIVOTING IS NOT NECESSARY FOR A POSITIVE DEFINITE MATRIX.)          MQERR 44
C  ONLY ABOUT N**3/2 MULTIPLICATIONS ARE DONE.                          MQERR 45
C  F. L. BAUER AND C. REINSCH, P. 45 IN -LINEAR ALGEBRA-                MQERR 46
C  BY J. H. WILKINSON AND C. REINSCH (SPRINGER-VERLAG, 1971)            MQERR 47
C  THE INVERSE IS NOT GUARANTEED TO BE POSITIVE DEFINITE, DUE TO        MQERR 48
C  ROUNDOFF ERROR.  NEVERTHELESS, THE CHOICE OF THE GAUSS-JORDAN METHOD MQERR 49
C  RATHER THAN CHOLESKY DECOMPOSITION IS INTENTIONAL.                   MQERR 50
C  H(*) IS USED AS A SCRATCH VECTOR.                                    MQERR 51
C                                                                       MQERR 52
 4040 NSMAL=0                                                           MQERR 53
      DO 4200 LINV=1,NACTV                                              MQERR 54
        L=(NACTV+1)-LINV                                                MQERR 55
        PIVOT=ERR(1,1)                                                  MQERR 56
        IF(PIVOT)4050,4050,4060                                         MQERR 57
 4050   NSMAL=NSMAL+1                                                  MQERR 58
 4060   IF(NACTV-2)4140,4070,4070                                      MQERR 59
 4070   DO 4130 K=2,NACTV                                              MQERR 60
          Q=ERR(K,1)                                                    MQERR 61
          IF(PIVOT)4090,4080,4090                                       MQERR 62
 4080     H(K)=RZERO                                                    MQERR 63
          GO TO 4120                                                    MQERR 64
 4090     IF(K-L)4100,4100,4110                                         MQERR 65
 4100     H(K)=-Q/PIVOT                                                 MQERR 66
          GO TO 4120                                                    MQERR 67
 4110     H(K)=Q/PIVOT                                                  MQERR 68
 4120     DO 4130 M=2,K                                                 MQERR 69
 4130       ERR(K-1,M-1)=ERR(K,M)+Q*H(M)                                MQERR 70
 4140   IF(PIVOT)4160,4150,4160                                        MQERR 71
 4150   ERR(NACTV,NACTV)=RZERO                                         MQERR 72
        GO TO 4170                                                      MQERR 73
 4160   ERR(NACTV,NACTV)=RUNIT/PIVOT                                   MQERR 74
 4170   IF(NACTV-2)4200,4180,4180                                      MQERR 75
 4180   DO 4190 K=2,NACTV                                              MQERR 76
 4190     ERR(NACTV,K-1)=H(K)                                          MQERR 77
 4200   CONTINUE                                                       MQERR 78
C                                                                       MQERR 79
      NRANK=NACTV-NSMAL                                                 MQERR 80
      IF(NRANK-NACTV)4210,4230,4230                                     MQERR 81
 4210 CONTINUE                                                          MQERR 82
      WRITE(KW,4220)NRANK,NACTV                                         MQERR 83
 4220 FORMAT(//44H THE SECOND DERIVATIVE MATRIX IS SINGULAR IN,         MQERR 84
     X     7H MQERR,,9X,7H RANK =,I3,7X,8H ORDER =,I3//                 1000R 84
     *     45H THEREFORE ALL PARAMETER ERRORS ARE INFINITE. )           MQERR 86
C                                                                       MQERR 87
C                              UNPACK THE ERROR MATRIX INTO THE UPPER    MQERR 88
C                                   TRIANGLE OF ERR(*,*), DE-SCALING IT. MQERR 89
 4230 JV=0                                                              MQERR 90
      WIFMX=RZERO                                                       MQERR 91
      DO 4380 JX=1,NV                                                   MQERR 92
        IF(MASK(JX))4250,4240,4250                                      MQERR 93
 4240   JV=JV+1                                                        MQERR 94
 4250   KV=0                                                           MQERR 95
        DO 4370 KX=1,JX                                                 MQERR 96
          ER=RZERO                                                      MQERR 97
          IF(MASK(JX))4370,4260,4370                                    MQERR 98
 4260     IF(MASK(KX))4370,4270,4370                                    MQERR 99
 4270     KV=KV+1                                                       MQERR100
          TEMP=RZERO                                                    MQERR101
          DENOM=SCALE(JV)*SCALE(KV)                                     MQERR102
          IF(DENOM)4280,4290,4280                                       MQERR103
 4280     TEMP=ERR(JV,KV)                                               MQERR104
          ER=TEMP/DENOM                                                 MQERR105
 4290     IF(JX-KX)4370,4300,4370                                       MQERR106
 4300     IF(TEMP)4310,4310,4350                                        MQERR107
 4310     CONTINUE                                                      MQERR108
          IF(NTRAC-(-2))4340,4320,4320                                  MQERR109
 4320     WRITE(KW,4330)JX,KX,TEMP                                      MQERR110
 4330     FORMAT(/6H THE (,I3,1H,,I3,23H) ELEMENT OF QSAV**-1 =,E12.5,  MQERR111
```

```
     *      5X,45H THEREFORE ALL PARAMETER ERRORS ARE INFINITE. )          MQERR112
 4340      TEMP=-TEMP                                                     MQERR113
 4350      IF(TEMP-VIFMX)4370,4370,4360                                   MQERR114
 4360      VIFMX=TEMP                                                     MQERR115
 4370      ERR(KX,JX+1)=ER                                                MQERR116
 4380    CONTINUE                                                         MQERR117
C                                  SYMMETRIZE ERR.                       MQERR118
       DO 4390 JX=1,NV                                                   MQERR119
         DO 4390 KX=1,JX                                                 MQERR120
           ERR(JX,KX)=ERR(KX,JX+1)                                       MQERR121
 4390      ERR(KX,JX)=ERR(JX,KX)                                          MQERR122
C                                                                        MQERR123
C                                  COMPUTE AND PRINT THE STANDARD ERRORS. MQERR124
       NDF=NPTS-NACTV                                                    MQERR125
       SCFAC=HUGE                                                        MQERR126
       IF(NDF)4410,4410,4400                                             MQERR127
 4400  SCFAC=NDF                                                         MQERR128
       SCFAC=QSQRT(FOBJ/SCFAC)                                           MQERR129
 4410  RESCL=NDF+NDF                                                     MQERR130
       IF(RESCL)4430,4430,4420                                           MQERR131
 4420  RESCL=QSQRT(RESCL)                                                MQERR132
 4430  CONTINUE                                                          MQERR133
       IF(NTRAC-(-1))4520,4440,4440                                      MQERR134
 4440  WRITE(KW,4450)NDF,NDF,RESCL,FOBJ,SCFAC                            MQERR135
 4450  FORMAT(///41H NUMBER OF DEGREES OF FREEDOM (N.D.F.) = ,           MQERR136
      *      15H(NPTS-NACTV) = ,I5//24H EXPECTED VALUE OF PHI = ,        MQERR137
      *      39H N.D.F. PLUS OR MINUS SQRT(2*N.D.F.) = ,I5,             MQERR138
      X      16H  PLUS OR MINUS ,E12.5//23H ACTUAL VALUE OF PHI = ,E12.5 1000R138
      *      //39H RESCALING FACTOR = SQRT(PHI/N.D.F.) = ,E12.5 )       MQERR140
       WRITE(KW,4460)VIFMX                                               MQERR141
 4460  FORMAT(///36H MAXIMUM VARIANCE INFLATION FACTOR =,E12.5///        MQERR142
      *       32H APPROXIMATE STANDARD ERRORS..../68X,9H RESCALED/      MQERR143
      *      12X,1HJ,6X,7HMASK(J),9X,4HX(J),14X,5HERROR,12X,5HERROR )   MQERR144
       RESCL=HUGE                                                        MQERR145
       DO 4540 JX=1,NV                                                   MQERR146
         SCALJ=RUNIT                                                     MQERR147
         ER=ERR(JX,JX)                                                   MQERR148
         IF(ER)4470,4490,4480                                            MQERR149
 4470    ER=-QSQRT(-ER)                                                  MQERR150
         SCALJ=-ER                                                       MQERR151
         GO TO 4490                                                      MQERR152
 4480    ER=QSQRT(ER)                                                    MQERR153
         SCALJ=ER                                                        MQERR154
 4490    IF(NDF)4520,4520,4500                                           MQERR155
 4500    IF(NRANK-NACTV)4520,4510,4510                                   MQERR156
 4510    RESCL=SCFAC*ER                                                  MQERR157
 4520    SCALE(JX)=SCALJ                                                 MQERR158
         WRITE(KW,4530)JX,MASK(JX),X(JX),ER,RESCL                        MQERR159
 4530    FORMAT(/10X,I3,I10,6X,E16.8,4X,E13.5,4X,E13.5 )                 MQERR160
 4540    CONTINUE                                                        MQERR161
C                                  COMPUTE AND PRINT THE CORRELATIONS.   MQERR162
       IF(NV-2)4620,4550,4550                                            MQERR163
 4550  CONTINUE                                                          MQERR164
       WRITE(KW,4560)(K,K=1,NV)                                          MQERR165
 4560  FORMAT(///45H LOWER TRIANGLE OF THE CORRELATION MATRIX..../       MQERR166
      *    12X,7H K ....,I5,7I13/(11X,8I13))                            MQERR167
       WRITE(KW,4570)(MASK(K),K=1,NV)                                    MQERR168
 4570  FORMAT(/7X,12H MASK(K)....,I5,7I13/(11X,8I13))                    MQERR169
       WRITE(KW,4580)                                                    MQERR170
 4580  FORMAT(/3X,1HJ,4X,7HMASK(J) )                                     MQERR171
       DO 4610 JX=1,NV                                                   MQERR172
         DO 4590 KX=1,JX                                                 MQERR173
 4590      H(KX)=ERR(JX,KX)/(SCALE(JX)*SCALE(KX))                       MQERR174
         WRITE(KW,4600)JX,MASK(JX),(H(KX),KX=1,JX)                       MQERR175
 4600    FORMAT(/1X,I3,I8,5X,8E13.5/(17X,8E13.5))                        MQERR176
 4610    CONTINUE                                                        MQERR177
C                                                                        MQERR178
C  END MQERR.                                                            MQERR180
 4620  RETURN                                                            MQERR179
       END                                                               MQERR181
C                                                                        MQERR182
```

```
C                                                                       POLRG001
C     ****************************************************************POLRG002
C                                                                       POLRG003
C         SAMPLE MAIN PROGRAM FOR POLYNOMIAL REGRESSION - POLRG       POLRG004
C                                                                       POLRG005
C     PURPOSE                                                           POLRG006
C        (1) READ THE PROBLEM PARAMETER CARD FOR A POLYNOMIAL REGRES-   POLRG007
C        SION, (2) CALL SUBROUTINES TO PERFORM THE ANALYSIS, (3)        POLRG008
C        PRINT THE REGRESSION COEFFICIENTS AND ANALYSIS OF VARIANCE     POLRG009
C        TABLE FOR POLYNOMIALS OF SUCCESSIVELY INCREASING DEGREES,      POLRG010
C        AND (4) PRINT THE TABLE OF RESIDUALS OF Y VALUES AND Y         POLRG011
C        ESTIMATES.                                                     POLRG012
C                                                                       POLRG013
C     REMARKS                                                           POLRG014
C        THE NUMBER OF OBSERVATIONS, N, MUST BE GREATER THAN M+1,       POLRG015
C        WHERE M IS THE HIGHEST DEGREE POLYNOMIAL SPECIFIED.            POLRG016
C        IF THERE IS NO REDUCTION IN THE RESIDUAL SUM OF SQUARES        POLRG017
C        BETWEEN TWO SUCCESSIVE DEGREES OF THE POLYNOMIALS, THE         POLRG018
C        PROGRAM TERMINATES THE PROBLEM BEFORE COMPLETING THE ANALY-    POLRG019
C        SIS FOR THE HIGHEST DEGREE POLYNOMIAL SPECIFIED.               POLRG020
C                                                                       POLRG021
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                     POLRG022
C        GDATA                                                          POLRG023
C        ORDER                                                          POLRG024
C        MINV                                                           POLRG025
C        MULTR                                                          POLRG026
C        PROP                                                           POLRG027
C                                                                       POLRG028
C     METHOD                                                            POLRG029
C        REFER TO B. OSTLE, *STATISTICS IN RESEARCH*, THE IOWA STATE    POLRG030
C        COLLEGE PRESS, 1954, CHAPTER 6, AND T. E. MULLIN,              POLRG031
C        PH.D. THESIS, OKLA. STATE UNIV., 1968.                         POLRG032
C                                                                       POLRG033
C     ****************************************************************POLRG034
C                                                                       POLRG035
C     THE FOLLOWING COMMON STATEMENTS TRANSFER THE APPROPRIATE TERMS    POLRG036
C     FROM SUBROUTINE PROP.                                             POLRG037
C                                                                       POLRG038
      COMMON/CPROP/DENS(50),CPH(50),VISC(50),COND(50),TEMP(50),DTEMP    POLRG039
      COMMON/TERM/AA1(50),AA2(50),AA3(50),BB1(50),BB2(50)               POLRG040
C                                                                       POLRG041
C     THE FOLLOWING DIMENSION MUST BE GREATER THAN OR EQUAL TO THE      POLRG042
C     PRODUCT OF N*(M+1), WHERE N IS THE NUMBER OF OBSERVATIONS AND M   POLRG043
C     IS THE HIGHEST DEGREE POLYNOMIAL SPECIFIED..                      POLRG044
C                                                                       POLRG045
      DIMENSION X(1100)                                                 POLRG046
C                                                                       POLRG047
C     THE FOLLOWING DIMENSION MUST BE GREATER THAN OR EQUAL TO THE      POLRG048
C     PRODUCT OF M*M..                                                  POLRG049
C                                                                       POLRG050
      DIMENSION DI(100)                                                 POLRG051
C                                                                       POLRG052
C     THE FOLLOWING DIMENSION MUST BE GREATER THAN OR EQUAL TO          POLRG053
C     (M+2)*(M+1)/2..                                                   POLRG054
C                                                                       POLRG055
      DIMENSION D(66)                                                   POLRG056
C                                                                       POLRG057
C     THE FOLLOWING DIMENSIONS MUST BE GREATER THAN OR EQUAL TO M..     POLRG058
C                                                                       POLRG059
      DIMENSION B(10),E(10),SS(10),T(10)                                POLRG060
C                                                                       POLRG061
C     THE FOLLOWING DIMENSIONS MUST BE GREATER THAN OR EQUAL TO (M+1).. POLRG062
C                                                                       POLRG063
      DIMENSION XBAR(11),STD(11),COE(11),SUMSQ(11),ISAVE(11)            POLRG064
C                                                                       POLRG065
C     THE FOLLOWING DIMENSION MUST BE GREATER THAN OR EQUAL TO 10..     POLRG066
C                                                                       POLRG067
      DIMENSION ANS(10)                                                 POLRG068
C                                                                       POLRG069
C     THE FOLLOWING DIMENSION MUST BE GREATER THAN OR EQUAL TO N*3..    POLRG070
C                                                                       POLRG071
      DIMENSION P(300)                                                  POLRG072
C                                                                       POLRG073
C     ****************************************************************POLRG074
C                                                                       POLRG075
C        IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE  POLRG076
C        C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION      POLRG077
C        STATEMENT WHICH FOLLOWS.                                       POLRG078
C                                                                       POLRG079
      DOUBLE PRECISION X,XBAR,STD,D,SUMSQ,DI,E,B,SS,T,ANS,DET,COE       POLRG080
      DOUBLE PRECISION AA1,AA2,AA3,BB1,BB2,TEMP                         POLRG081
      DOUBLE PRECISION DENS,CPH,VISC,COND,TM,DTEMP                      POLRG082
C                                                                       POLRG083
C        THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS    POLRG084
C        APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS      POLRG085
C        ROUTINE.                                                       POLRG086
C                                                                       POLRG087
C     ****************************************************************POLRG088
C                                                                       POLRG089
    2 FORMAT(A4,3I2,F10.5)                                              POLRG090
    3 FORMAT(27H1POLYNOMIAL REGRESSION.....,A4/)                        POLRG091
    4 FORMAT(23HONUMBER OF OBSERVATIONS,I6/)                            POLRG092
    5 FORMAT(32HOPOLYNOMIAL REGRESSION OF DEGREE,I3)                    POLRG093
    6 FORMAT(12HO  INTERCEPT,D25.16)                                    POLRG094
    7 FORMAT(26HO  REGRESSION COEFFICIENTS/(3D25.16))                   POLRG095
    8 FORMAT(1HO/24X,24HANALYSIS OF VARIANCE FOR,I4,19H  DEGREE POLYNOMPOLRG096
     1IAL/)                                                             POLRG097
    9 FORMAT(1HO,5X,19HSOURCE OF VARIATION,7X,9HDEGREE OF,7X,6HSUM OF,9XPOLRG098
     1,4HMEAN,10X,1HF,9X,20HIMPROVEMENT IN TERMS/33X,7HFREEDOM,8X,7HSQUAPOLRG099
     2RES,7X,6HSQUARE,7X,5HVALUE,8X,17HOF SUM OF SQUARES)               POLRG100
   10 FORMAT(20HO  DUE TO REGRESSION,12X,I6,F17.5,F14.5,F13.5,F20.5)    POLRG101
   11 FORMAT(32H   DEVIATION ABOUT REGRESSION    ,I6,F17.5,F14.5)      POLRG102
   12 FORMAT(8X,5HTOTAL,19X,I6,F17.5//)                                 POLRG103
   13 FORMAT(17HO  NO IMPROVEMENT)                                      POLRG104
   14 FORMAT(1HO//27X,18HTABLE OF RESIDUALS//16H OBSERVATION NO.,5X,7HX  POLRG105
     1VALUE,7X,7HY VALUE,7X,10HY ESTIMATE,7X,8HRESIDUAL,7X,5HERROR/)    POLRG106
   15 FORMAT(1HO,3X,I6,F18.5,F14.5,F17.5,F15.5,F15.5)                   POLRG107
   16 FORMAT(1HO,*STARTING TEMPERATURE = *,F6.2/)                       POLRG108
   17 FORMAT(1HO,*TEMPERATURE INCREMENT = *,F4.2/)                      POLRG109
   18 FORMAT(1HO,*TEMPERATURE DIFFERENCE = *,I4/)                       POLRG110
C                                                                       POLRG111
C     ****************************************************************POLRG112
C                                                                       POLRG113
C     READ PROBLEM PARAMETER CARD                                       POLRG114
C                                                                       POLRG115
      CALL PROP                                                         POLRG116
      DO 300 KK=1,5                                                     POLRG117
      READ (5,2) PR,N,M,NDT,TM                                          POLRG118
C                                                                       POLRG119
```

```
C         PR....PROBLEM NUMBER (MAY BE ALPHAMERIC)                          POLRG120
C         N.....NUMBER OF OBSERVATIONS                                      POLRG121
C         M.....HIGHEST DEGREE POLYNOMIAL SPECIFIED                         POLRG122
C         NDT...TEMPERATURE DIFFERENCE                                      POLRG123
C         TM....STARTING TEMPERATURE                                        POLRG124
C                                                                           POLRG125
C     PRINT PROBLEM PARAMETERS.                                             POLRG126
C                                                                           POLRG127
      WRITE (6,3) PR                                                        POLRG128
      WRITE (6,4) N                                                         POLRG129
      WRITE(6,16) TM                                                        POLRG130
      WRITE(6,18) NDT                                                       POLRG131
      WRITE(6,17) DTEMP                                                     POLRG132
C                                                                           POLRG133
C     READ INPUT DATA                                                       POLRG134
C                                                                           POLRG135
      L=N*M                                                                 POLRG136
      DO 110 I=1,N                                                          POLRG137
      J=L+I                                                                 POLRG138
C                                                                           POLRG139
C        X(I) IS THE INDEPENDENT VARIABLE, AND X(J) IS THE DEPENDENT        POLRG140
C        VARIABLE.                                                          POLRG141
C                                                                           POLRG142
      X(I)=TEMP(I)                                                          POLRG143
      IF(KK .EQ. 1) X(J)=AA1(I)                                             POLRG144
      IF(KK .EQ. 2) X(J)=AA2(I)                                             POLRG145
      IF(KK .EQ. 3) X(J)=AA3(I)                                             POLRG146
      IF(KK .EQ. 4) X(J)=BB1(I)                                             POLRG147
      IF(KK .EQ. 5) X(J)=BB2(I)                                             POLRG148
  110 CONTINUE                                                              POLRG149
C                                                                           POLRG150
      CALL GDATA (N,M,X,XBAR,STD,D,SUMSQ)                                   POLRG151
C                                                                           POLRG152
      MM=M+1                                                                POLRG153
      SUM=0.0                                                               POLRG154
      NT=N-1                                                                POLRG155
C                                                                           POLRG156
      DO 200 I=1,M                                                          POLRG157
      ISAVE(I)=I                                                            POLRG158
C                                                                           POLRG159
C     FORM SUBSET OF CORRELATION COEFFICIENT MATRIX                         POLRG160
C                                                                           POLRG161
      CALL ORDER (MM,D,MM,I,ISAVE,DI,E)                                     POLRG162
C                                                                           POLRG163
C     INVERT THE SUBMATRIX OF CORRELATION COEFFICIENTS                      POLRG164
C                                                                           POLRG165
      CALL MINV (DI,I,DET,B,I)                                              POLRG166
C                                                                           POLRG167
      CALL MULTR (N,I,XBAR,STD,SUMSQ,DI,E,ISAVE,B,SB,T,ANS)                 POLRG168
C                                                                           POLRG169
C     PRINT THE RESULT OF CALCULATION                                       POLRG170
C                                                                           POLRG171
      WRITE (6,5) I                                                         POLRG172
      SUMIP=ANS(4)-SUM                                                      POLRG173
      IF(SUMIP) 140, 140, 150                                               POLRG174
  140 WRITE (6,13)                                                          POLRG175
      GO TO 210                                                             POLRG176
  150 WRITE (6,6) ANS(1)                                                    POLRG177
      WRITE (6,7) (B(J),J=1,I)                                              POLRG178
      WRITE (6,8) I                                                         POLRG179
      WRITE (6,9)                                                           POLRG180
      SUM=ANS(4)                                                            POLRG181
      WRITE (6,10) I,ANS(4),ANS(6),ANS(10),SUMIP                            POLRG182
      NI=ANS(8)                                                             POLRG183
      WRITE (6,11) NI,ANS(7),ANS(9)                                         POLRG184
      WRITE (6,12) NT,SUMSQ(MM)                                             POLRG185
C                                                                           POLRG186
C     SAVE COEFFICIENTS FOR CALCULATION OF Y ESTIMATES                      POLRG187
C                                                                           POLRG188
      COE(1)=ANS(1)                                                         POLRG189
      DO 160 J=1,I                                                          POLRG190
  160 COE(J+1)=B(J)                                                         POLRG191
      LA=I                                                                  POLRG192
  200 CONTINUE                                                              POLRG193
C                                                                           POLRG194
C                                                                           POLRG195
C         CALCULATE ESTIMATES                                               POLRG196
C                                                                           POLRG197
  210 NP3=N+N                                                               POLRG198
      DO 230 I=1,N                                                          POLRG199
      NP3=NP3+1                                                             POLRG200
      P(NP3)=COE(1)                                                         POLRG201
      L=I                                                                   POLRG202
      DO 230 J=1,LA                                                         POLRG203
      P(NP3)=P(NP3)+X(L)*COE(J+1)                                           POLRG204
  230 L=L+N                                                                 POLRG205
C                                                                           POLRG206
C         COPY OBSERVED DATA                                                POLRG207
C                                                                           POLRG208
      N2=N                                                                  POLRG209
      L=N*M                                                                 POLRG210
      DO 240 I=1,N                                                          POLRG211
      P(I)=X(I)                                                             POLRG212
      N2=N2+1                                                               POLRG213
      L=L+1                                                                 POLRG214
  240 P(N2)=X(L)                                                            POLRG215
C                                                                           POLRG216
C     PRINT TABLE OF RESIDUALS                                              POLRG217
C                                                                           POLRG218
      WRITE (6,3) PR                                                        POLRG219
      WRITE (6,5) LA                                                        POLRG220
      WRITE (6,14)                                                          POLRG221
      NP2=N                                                                 POLRG222
      NP3=N+N                                                               POLRG223
      DO 250 I=1,N                                                          POLRG224
      NP2=NP2+1                                                             POLRG225
      NP3=NP3+1                                                             POLRG226
      RESID=P(NP2)-P(NP3)                                                   POLRG227
      PRERR=RESID*100./(P(NP2))                                             POLRG228
  250 WRITE (6,15) I,P(I),P(NP2),P(NP3),RESID,PRERR                         POLRG229
C                                                                           POLRG230
  300 CONTINUE                                                              POLRG231
      STOP                                                                  POLRG232
      END                                                                   POLRG233
C                                                                           MINV 001
C     ..................................................................... MINV 002
C                                                                           MINV 003
C         SUBROUTINE MINV                                                   MINV 004
C                                                                           MINV 005
C         PURPOSE                                                           MINV 006
```

```
C          INVERT A MATRIX                                                  MINV 007
C                                                                           MINV 008
C       USAGE                                                               MINV 009
C          CALL MINV(A,N,D,L,M)                                             MINV 010
C                                                                           MINV 011
C       DESCRIPTION OF PARAMETERS                                           MINV 012
C          A - INPUT MATRIX, DESTROYED IN COMPUTATION AND REPLACED BY       MINV 013
C              RESULTANT INVERSE.                                           MINV 014
C          N - ORDER OF MATRIX A                                            MINV 015
C          D - RESULTANT DETERMINANT                                        MINV 016
C          L - WORK VECTOR OF LENGTH N                                      MINV 017
C          M - WORK VECTOR OF LENGTH N                                      MINV 018
C                                                                           MINV 019
C       REMARKS                                                             MINV 020
C          MATRIX A MUST BE A GENERAL MATRIX                                MINV 021
C                                                                           MINV 022
C       SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                       MINV 023
C          NONE                                                             MINV 024
C                                                                           MINV 025
C       METHOD                                                              MINV 026
C          THE STANDARD GAUSS-JORDAN METHOD IS USED. THE DETERMINANT        MINV 027
C          IS ALSO CALCULATED. A DETERMINANT OF ZERO INDICATES THAT         MINV 028
C          THE MATRIX IS SINGULAR.                                          MINV 029
C                                                                           MINV 030
C  .........................................................................MINV 031
C                                                                           MINV 032
      SUBROUTINE MINV(A,N,D,L,M)                                            MINV 033
      DIMENSION A(1),L(1),M(1)                                              MINV 034
C                                                                           MINV 035
C                                                                           MINV 037
C  .........................................................................MINV 036
C     IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE         MINV 038
C     C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION             MINV 039
C     STATEMENT WHICH FOLLOWS.                                              MINV 040
C                                                                           MINV 041
C     DOUBLE PRECISION A,D,BIGA,HOLD,DABS                                   MINV 042
C                                                                           MINV 043
C     THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS           MINV 044
C     APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS             MINV 045
C     ROUTINE.                                                              MINV 046
C                                                                           MINV 047
C     THE DOUBLE PRECISION VERSION OF THIS SUBROUTINE MUST ALSO             MINV 048
C     CONTAIN DOUBLE PRECISION FORTRAN FUNCTIONS.  ABS IN STATEMENT         MINV 049
C     10 MUST BE CHANGED TO DABS.                                           MINV 050
C                                                                           MINV 051
C  .........................................................................MINV 052
C                                                                           MINV 053
C     SEARCH FOR LARGEST ELEMENT                                            MINV 054
C                                                                           MINV 055
      D=1.0                                                                 MINV 056
      NK=-N                                                                 MINV 057
      DO 80 K=1,N                                                           MINV 058
      NK=NK+N                                                               MINV 059
      L(K)=K                                                                MINV 060
      M(K)=K                                                                MINV 061
      KK=NK+K                                                               MINV 062
      BIGA=A(KK)                                                            MINV 063
      DO 20 J=K,N                                                           MINV 064
      IZ=N*(J-1)                                                            MINV 065
      DO 20 I=K,N                                                           MINV 066
      IJ=IZ+I                                                               MINV 067
   10 IF(DABS(BIGA)-DABS(A(IJ))) 15,20,20                                   MINV 068
   15 BIGA=A(IJ)                                                            MINV 069
      L(K)=I                                                                MINV 070
      M(K)=J                                                                MINV 071
   20 CONTINUE                                                              MINV 072
C                                                                           MINV 073
C        INTERCHANGE ROWS                                                   MINV 074
C                                                                           MINV 075
      J=L(K)                                                                MINV 076
      IF(J-K) 35,35,25                                                      MINV 077
   25 KI=K-N                                                                MINV 078
      DO 30 I=1,N                                                           MINV 079
      KI=KI+N                                                               MINV 080
      HOLD=-A(KI)                                                           MINV 081
      JI=KI-K+J                                                             MINV 082
      A(KI)=A(JI)                                                           MINV 083
   30 A(JI) =HOLD                                                           MINV 084
C                                                                           MINV 085
C        INTERCHANGE COLUMNS                                                MINV 086
C                                                                           MINV 087
   35 I=M(K)                                                                MINV 088
      IF(I-K) 45,45,38                                                      MINV 089
   38 JP=N*(I-1)                                                            MINV 090
      DO 40 J=1,N                                                           MINV 091
      JK=NK+J                                                               MINV 092
      JI=JP+J                                                               MINV 093
      HOLD=-A(JK)                                                           MINV 094
      A(JK)=A(JI)                                                           MINV 095
   40 A(JI) =HOLD                                                           MINV 096
C                                                                           MINV 097
C        DIVIDE COLUMN BY MINUS PIVOT (VALUE OF PIVOT ELEMENT IS            MINV 098
C        CONTAINED IN BIGA)                                                 MINV 099
C                                                                           MINV 100
   45 IF(BIGA) 48,46,48                                                     MINV 101
   46 D=0.0                                                                 MINV 102
      RETURN                                                                MINV 103
   48 DO 55 I=1,N                                                           MINV 104
      IF(I-K) 50,55,50                                                      MINV 105
   50 IK=NK+I                                                               MINV 106
      A(IK)=A(IK)/(-BIGA)                                                   MINV 107
   55 CONTINUE                                                              MINV 108
C                                                                           MINV 109
C        REDUCE MATRIX                                                      MINV 110
C                                                                           MINV 111
      DO 65 I=1,N                                                           MINV 112
      IK=NK+I                                                               MINV 113
      IJ=I-N                                                                MINV 114
      DO 65 J=1,N                                                           MINV 115
      IJ=IJ+N                                                               MINV 116
      IF(I-K) 60,65,60                                                      MINV 117
   60 IF(J-K) 62,65,62                                                      MINV 118
   62 KJ=IJ-I+K                                                             MINV 119
      A(IJ)=A(IK)*A(KJ)+A(IJ)                                               MINV 120
   65 CONTINUE                                                              MINV 121
C                                                                           MINV 122
C        DIVIDE ROW BY PIVOT                                                MINV 123
C                                                                           MINV 124
      KJ=K-N                                                                MINV 125
      DO 75 J=1,N                                                           MINV 126
```

```
      KJ=KJ+N                                                           MINV 127
      IF(J-K) 70,75,70                                                  MINV 128
   70 A(KJ)=A(KJ)/BIGA                                                  MINV 129
   75 CONTINUE                                                          MINV 130
C                                                                       MINV 131
C        PRODUCT OF PIVOTS                                              MINV 132
C                                                                       MINV 133
      D=D*BIGA                                                          MINV 134
C                                                                       MINV 135
C        REPLACE PIVOT BY RECIPROCAL                                    MINV 136
C                                                                       MINV 137
      A(KK)=1.0/BIGA                                                    MINV 138
   80 CONTINUE                                                          MINV 139
C                                                                       MINV 140
C        FINAL ROW AND COLUMN INTERCHANGE                               MINV 141
C                                                                       MINV 142
      K=N                                                               MINV 143
  100 K=(K-1)                                                           MINV 144
      IF(K) 150,150,105                                                 MINV 145
  105 I=L(K)                                                            MINV 146
      IF(I-K) 120,120,108                                               MINV 147
  108 JQ=N*(K-1)                                                        MINV 148
      JR=N*(I-1)                                                        MINV 149
      DO 110 J=1,N                                                      MINV 150
      JK=JQ+J                                                           MINV 151
      HOLD=A(JK)                                                        MINV 152
      JI=JR+J                                                           MINV 153
      A(JK)=-A(JI)                                                      MINV 154
  110 A(JI) =HOLD                                                       MINV 155
  120 J=M(K)                                                            MINV 156
      IF(J-K) 100,100,125                                               MINV 157
  125 KI=K-N                                                            MINV 158
      DO 130 I=1,N                                                      MINV 159
      KI=KI+N                                                           MINV 160
      HOLD=A(KI)                                                        MINV 161
      JI=KI-K+J                                                         MINV 162
      A(KI)=-A(JI)                                                      MINV 163
  130 A(JI) =HOLD                                                       MINV 164
      GO TO 100                                                         MINV 165
  150 RETURN                                                            MINV 166
      END                                                               MINV 167
C                                                                       ORDER001
C     ..................................................................ORDER002
C                                                                       ORDER003
C        SUBROUTINE ORDER                                               ORDER004
C                                                                       ORDER005
C        PURPOSE                                                        ORDER006
C           CONSTRUCT FROM A LARGER MATRIX OF CORRELATION COEFFICIENTS  ORDER007
C           A SUBSET MATRIX OF INTERCORRELATIONS AMONG INDEPENDENT      ORDER008
C           VARIABLES AND A VECTOR OF INTERCORRELATIONS OF INDEPENDENT  ORDER009
C           VARIABLES WITH DEPENDENT VARIABLE.  THIS SUBROUTINE IS      ORDER010
C           NORMALLY USED IN THE PERFORMANCE OF MULTIPLE AND POLYNOMIAL ORDER011
C           REGRESSION ANALYSES.                                        ORDER012
C                                                                       ORDER013
C        USAGE                                                          ORDER014
C           CALL ORDER (M,R,NDEP,K,ISAVE,RX,RY)                         ORDER015
C                                                                       ORDER016
C        DESCRIPTION OF PARAMETERS                                      ORDER017
C           M      - NUMBER OF VARIABLES AND ORDER OF MATRIX R.         ORDER018
C           R      - INPUT MATRIX CONTAINING CORRELATION COEFFICIENTS.  ORDER019
C                    THIS SUBROUTINE EXPECTS ONLY UPPER TRIANGULAR      ORDER020
C                    PORTION OF THE SYMMETRIC MATRIX TO BE STORED (BY   ORDER021
C                    COLUMNS) IN R.  (STORAGE MODE OF 1)                ORDER022
C           NDEP   - THE SUBSCRIPT NUMBER OF THE DEPENDENT VARIABLE.    ORDER023
C           K      - NUMBER OF INDEPENDENT VARIABLES TO BE INCLUDED     ORDER024
C                    IN THE FORTHCOMING REGRESSION.                     ORDER025
C           ISAVE  - INPUT VECTOR OF LENGTH K+1 CONTAINING, IN ASCENDING ORDER026
C                    ORDER, THE SUBSCRIPT NUMBERS OF K INDEPENDENT      ORDER027
C                    VARIABLES TO BE INCLUDED IN THE FORTHCOMING REGRES- ORDER028
C                    SION.                                              ORDER029
C                    UPON RETURNING TO THE CALLING ROUTINE, THIS VECTOR ORDER030
C                    CONTAINS, IN ADDITION, THE SUBSCRIPT NUMBER OF     ORDER031
C                    THE DEPENDENT VARIABLE IN K+1 POSITION.            ORDER032
C           RX     - OUTPUT MATRIX (K X K) CONTAINING INTERCORRELATIONS ORDER033
C                    AMONG INDEPENDENT VARIABLES TO BE USED IN FORTH-    ORDER034
C                    COMING REGRESSION.                                 ORDER035
C           RY     - OUTPUT VECTOR OF LENGTH K CONTAINING INTERCORRELA- ORDER036
C                    TIONS OF INDEPENDENT VARIABLES WITH DEPENDENT       ORDER037
C                    VARIABLES.                                         ORDER038
C                                                                       ORDER039
C        REMARKS                                                        ORDER040
C           NONE                                                        ORDER041
C                                                                       ORDER042
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                  ORDER043
C           NONE                                                        ORDER044
C                                                                       ORDER045
C        METHOD                                                         ORDER046
C           FROM THE SUBSCRIPT NUMBERS OF THE VARIABLES TO BE INCLUDED  ORDER047
C           IN THE FORTHCOMING REGRESSION, THE SUBROUTINE CONSTRUCTS THEORDER048
C           MATRIX RX AND THE VECTOR RY.                                ORDER049
C                                                                       ORDER050
C     ..................................................................ORDER051
C                                                                       ORDER052
      SUBROUTINE ORDER (M,R,NDEP,K,ISAVE,RX,RY)                         ORDER053
      DIMENSION R(1),ISAVE(1),RX(1),RY(1)                               ORDER054
C                                                                       ORDER055
C     ..................................................................ORDER056
C                                                                       ORDER057
C     IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE     ORDER058
C     C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION         ORDER059
C     STATEMENT WHICH FOLLOWS.                                          ORDER060
C                                                                       ORDER061
      DOUBLE PRECISION R,RX,RY                                          ORDER062
C                                                                       ORDER063
C     THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS       ORDER064
C     APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS         ORDER065
C     ROUTINE.                                                          ORDER066
C                                                                       ORDER067
C     ..................................................................ORDER068
C                                                                       ORDER069
C     COPY INTERCORRELATIONS OF INDEPENDENT VARIABLES                   ORDER070
C     WITH DEPENDENT VARIABLE                                           ORDER071
C                                                                       ORDER072
      MM=0                                                              ORDER073
      DO 130 J=1,K                                                      ORDER074
      L2=ISAVE(J)                                                       ORDER075
      IF(NDEP-L2) 122, 123, 123                                         ORDER076
  122 L=NDEP+(L2*L2-L2)/2                                               ORDER077
      GO TO 125                                                         ORDER078
  123 L=L2+(NDEP*NDEP-NDEP)/2                                           ORDER079
```

```
  125 RY(J)=R(L)                                                        ORDER080
C                                                                       ORDER081
C     COPY A SUBSET MATRIX OF INTERCORRELATIONS AMONG                   ORDER082
C     INDEPENDENT VARIABLES                                             ORDER083
C                                                                       ORDER084
      DO 130 I=1,K                                                      ORDER085
      L1=ISAVE(I)                                                       ORDER086
      IF(L1-L2) 127, 128, 128                                           ORDER087
  127 L=L1+(L2*L2-L2)/2                                                 ORDER088
      GO TO 129                                                         ORDER089
  128 L=L2+(L1*L1-L1)/2                                                 ORDER090
  129 MM=MM+1                                                           ORDER091
  130 RX(MM)=R(L)                                                       ORDER092
C                                                                       ORDER093
C     PLACE THE SUBSCRIPT NUMBER OF THE DEPENDENT                       ORDER094
C     VARIABLE IN ISAVE(K+1)                                            ORDER095
C                                                                       ORDER096
      ISAVE(K+1)=NDEP                                                   ORDER097
      RETURN                                                            ORDER098
      END                                                               ORDER099
C                                                                       GDATA001
C     ..................................................................GDATA002
C                                                                       GDATA003
C        SUBROUTINE GDATA                                               GDATA004
C                                                                       GDATA005
C        PURPOSE                                                        GDATA006
C           GENERATE INDEPENDENT VARIABLES UP TO THE M-TH POWER (THE    GDATA007
C           HIGHEST DEGREE POLYNOMIAL SPECIFIED) AND COMPUTE MEANS,     GDATA008
C           STANDARD DEVIATIONS, AND CORRELATION COEFFICIENTS.  THIS    GDATA009
C           SUBROUTINE IS NORMALLY CALLED BEFORE SUBROUTINES ORDER,     GDATA010
C           MINV AND MULTR IN THE PERFORMANCE OF A POLYNOMIAL           GDATA011
C           REGRESSION.                                                 GDATA012
C                                                                       GDATA013
C        USAGE                                                          GDATA014
C           CALL GDATA (N,M,X,XBAR,STD,D,SUMSQ)                         GDATA015
C                                                                       GDATA016
C        DESCRIPTION OF PARAMETERS                                      GDATA017
C           N      - NUMBER OF OBSERVATIONS.                            GDATA018
C           M      - THE HIGHEST DEGREE POLYNOMIAL TO BE FITTED.        GDATA019
C           X      - INPUT MATRIX (N BY M+1) .  WHEN THE SUBROUTINE IS  GDATA020
C                    CALLED, DATA FOR THE INDEPENDENT VARIABLE ARE      GDATA021
C                    STORED IN THE FIRST COLUMN OF MATRIX X, AND DATA FORGDATA022
C                    THE DEPENDENT VARIABLE ARE STORED IN THE LAST      GDATA023
C                    COLUMN OF THE MATRIX.  UPON RETURNING TO THE       GDATA024
C                    CALLING ROUTINE, GENERATED POWERS OF THE INDEPENDENTGDATA025
C                    VARIABLE ARE STORED IN COLUMNS 2 THROUGH M.        GDATA026
C           XBAR   - OUTPUT VECTOR OF LENGTH M+1 CONTAINING MEANS OF    GDATA027
C                    INDEPENDENT AND DEPENDENT VARIABLES.               GDATA028
C           STD    - OUTPUT VECTOR OF LENGTH M+1 CONTAINING STANDARD    GDATA029
C                    DEVIATIONS OF INDEPENDENT AND DEPENDENT VARIABLES. GDATA030
C           D      - OUTPUT MATRIX (ONLY UPPER TRIANGULAR PORTION OF THE GDATA031
C                    SYMMETRIC MATRIX OF M+1 BY M+1) CONTAINING CORRELA-GDATA032
C                    TION COEFFICIENTS.  (STORAGE MODE OF 1)            GDATA033
C           SUMSQ  - OUTPUT VECTOR OF LENGTH M+1 CONTAINING SUMS OF     GDATA034
C                    PRODUCTS OF DEVIATIONS FROM MEANS  OF INDEPENDENT  GDATA035
C                    AND DEPENDENT VARIABLES.                           GDATA036
C                                                                       GDATA037
C        REMARKS                                                        GDATA038
C           N MUST BE GREATER THAN M+1.                                 GDATA039
C           IF M IS EQUAL TO 5 OR GREATER, SINGLE PRECISION MAY NOT BE  GDATA040
C              SUFFICIENT TO GIVE SATISFACTORY COMPUTATIONAL RESULTS.    GDATA041
C                                                                       GDATA042
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                  GDATA043
C           NONE                                                        GDATA044
C                                                                       GDATA045
C        METHOD                                                         GDATA046
C           REFER TO 8. OSTLE, 'STATISTICS IN RESEARCH', THE IOWA STATE GDATA047
C           COLLEGE PRESS, 1954, CHAPTER 6.                             GDATA048
C                                                                       GDATA049
C     ..................................................................GDATA050
C                                                                       GDATA051
      SUBROUTINE GDATA (N,M,X,XBAR,STD,D,SUMSQ)                         GDATA052
      DIMENSION X(1),XBAR(1),STD(1),D(1),SUMSQ(1)                       GDATA053
C                                                                       GDATA054
C        ..............................................................GDATA055
C                                                                       GDATA056
C        IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE  GDATA057
C        C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION      GDATA058
C        STATEMENT WHICH FOLLOWS.                                       GDATA059
C                                                                       GDATA060
      DOUBLE PRECISION X,XBAR,STD,D,SUMSQ,T1,T2,DABS,DSQRT              GDATA061
C                                                                       GDATA062
C        THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS    GDATA063
C        APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS      GDATA064
C        ROUTINE.                                                       GDATA065
C                                                                       GDATA066
C        THE DOUBLE PRECISION VERSION OF THIS SUBROUTINE MUST ALSO      GDATA067
C        CONTAIN DOUBLE PRECISION FORTRAN FUNCTIONS.  SQRT AND ABS IN   GDATA068
C        STATEMENT 180 MUST BE CHANGED TO DSQRT AND DABS.               GDATA069
C                                                                       GDATA070
C        ..............................................................GDATA071
C     GENERATE INDEPENDENT VARIABLES                                    GDATA072
C                                                                       GDATA073
      IF(M-1) 105, 105, 90                                              GDATA074
   90 L1=0                                                              GDATA075
      DO 100 I=2,M                                                      GDATA076
      L1=L1+N                                                           GDATA077
      DO 100 J=1,N                                                      GDATA078
      L=L1+J                                                            GDATA079
      K=L-N                                                             GDATA080
  100 X(L)=X(K)*X(J)                                                    GDATA081
C                                                                       GDATA082
C     CALCULATE MEANS                                                   GDATA083
C                                                                       GDATA084
  105 MM=M+1                                                            GDATA085
      DF=N                                                              GDATA086
      L=0                                                               GDATA087
      DO 115 I=1,MM                                                     GDATA088
      XBAR(I)=0.0                                                       GDATA089
      DO 110 J=1,N                                                      GDATA090
      L=L+1                                                             GDATA091
  110 XBAR(I)=XBAR(I)+X(L)                                              GDATA092
  115 XBAR(I)=XBAR(I)/DF                                                GDATA093
C                                                                       GDATA094
      DO 130 I=1,MM                                                     GDATA095
  130 STD(I)=0.0                                                        GDATA096
C                                                                       GDATA097
C     CALCULATE SUMS OF CROSS-PRODUCTS OF DEVIATIONS                    GDATA098
C                                                                       GDATA099
      L=((MM+1)*MM)/2                                                   GDATA100
```

```
      DO 150 I=1,L                                                      GDATA101
  150 D(I)=0.0                                                          GDATA102
      DO 170 K=1,N                                                      GDATA103
      L=0                                                               GDATA104
      DO 170 J=1,MM                                                     GDATA105
      L2=N*(J-1)+K                                                      GDATA106
      T2=X(L2)-XBAR(J)                                                  GDATA107
      STD(J)=STD(J)+T2                                                  GDATA108
      DO 170 I=1,J                                                      GDATA109
      L1=N*(I-1)+K                                                      GDATA110
      T1=X(L1)-XBAR(I)                                                  GDATA111
      L=L+1                                                             GDATA112
  170 D(L)=D(L)+T1*T2                                                   GDATA113
      L=0                                                               GDATA114
      DO 175 J=1,MM                                                     GDATA115
      DO 175 I=1,J                                                      GDATA116
      L=L+1                                                             GDATA117
  175 D(L)=D(L)-STD(I)*STD(J)/DF                                        GDATA118
      L=0                                                               GDATA119
      DO 180 I=1,MM                                                     GDATA120
      L=L+I                                                             GDATA121
      SUMSQ(I)=D(L)                                                     GDATA122
  180 STD(I)=DSQRT(DABS(D(L)))                                          GDATA123
C                                                                       GDATA124
C     CALCULATE CORRELATION COEFFICIENTS                                GDATA125
C                                                                       GDATA126
      L=0                                                               GDATA127
      DO 190 J=1,MM                                                     GDATA128
      DO 190 I=1,J                                                      GDATA129
      L=L+1                                                             GDATA130
  190 D(L)=D(L)/(STD(I)*STD(J))                                         GDATA131
C                                                                       GDATA132
C     CALCULATE STANDARD DEVIATIONS                                     GDATA133
C                                                                       GDATA134
      DF=SQRT(DF-1.0)                                                   GDATA135
      DO 200 I=1,MM                                                     GDATA136
  200 STD(I)=STD(I)/DF                                                  GDATA137
      RETURN                                                            GDATA138
      END                                                               GDATA139
C                                                                       MULTR001
C     ..................................................................MULTR002
C                                                                       MULTR003
C        SUBROUTINE MULTR                                               MULTR004
C                                                                       MULTR005
C        PURPOSE                                                        MULTR006
C           PERFORM A MULTIPLE LINEAR REGRESSION ANALYSIS FOR A DEPEN-  MULTR007
C           DENT VARIABLE AND A SET OF INDEPENDENT VARIABLES.  THIS     MULTR008
C           SUBROUTINE IS NORMALLY USED IN THE PERFORMANCE OF MULTIPLE  MULTR009
C           AND POLYNOMIAL REGRESSION ANALYSES.                         MULTR010
C                                                                       MULTR011
C        USAGE                                                          MULTR012
C           CALL MULTR (N,K,XBAR,STD,D,RX,RY,ISAVE,B,SB,T,ANS)          MULTR013
C                                                                       MULTR014
C        DESCRIPTION OF PARAMETERS                                      MULTR015
C           N      - NUMBER OF OBSERVATIONS.                           MULTR016
C           K      - NUMBER OF INDEPENDENT VARIABLES IN THIS REGRESSION.MULTR017
C           XBAR   - INPUT VECTOR OF LENGTH M CONTAINING MEANS OF ALL  MULTR018
C                    VARIABLES. M IS NUMBER OF VARIABLES IN OBSERVATIONS.MULTR019
C           STD    - INPUT VECTOR OF LENGTH M CONTAINING STANDARD DEVI-MULTR020
C                    ATIONS OF ALL VARIABLES.                           MULTR021
C           D      - INPUT VECTOR OF LENGTH M CONTAINING THE DIAGONAL OF MULTR022
C                    THE MATRIX OF SUMS OF CROSS-PRODUCTS OF DEVIATIONS MULTR023
C                    FROM MEANS FOR ALL VARIABLES.                      MULTR024
C           RX     - INPUT MATRIX (K X K) CONTAINING THE INVERSE OF    MULTR025
C                    INTERCORRELATIONS AMONG INDEPENDENT VARIABLES.     MULTR026
C           RY     - INPUT VECTOR OF LENGTH K CONTAINING INTERCORRELA- MULTR027
C                    TIONS OF INDEPENDENT VARIABLES WITH DEPENDENT      MULTR028
C                    VARIABLE.                                          MULTR029
C           ISAVE  - INPUT VECTOR OF LENGTH K+1 CONTAINING SUBSCRIPTS OF MULTR030
C                    INDEPENDENT VARIABLES IN ASCENDING ORDER.  THE     MULTR031
C                    SUBSCRIPT OF THE DEPENDENT VARIABLE IS STORED IN   MULTR032
C                    THE LAST, K+1, POSITION.                           MULTR033
C           B      - OUTPUT VECTOR OF LENGTH K CONTAINING REGRESSION    MULTR034
C                    COEFFICIENTS.                                      MULTR035
C           SB     - OUTPUT VECTOR OF LENGTH K CONTAINING STANDARD      MULTR036
C                    DEVIATIONS OF REGRESSION COEFFICIENTS.             MULTR037
C           T      - OUTPUT VECTOR OF LENGTH K CONTAINING T-VALUES.     MULTR038
C           ANS    - OUTPUT VECTOR OF LENGTH 10 CONTAINING THE FOLLOWING MULTR039
C                    INFORMATION..                                      MULTR040
C                    ANS(1)  INTERCEPT                                  MULTR041
C                    ANS(2)  MULTIPLE CORRELATION COEFFICIENT           MULTR042
C                    ANS(3)  STANDARD ERROR OF ESTIMATE                 MULTR043
C                    ANS(4)  SUM OF SQUARES ATTRIBUTABLE TO REGRES-     MULTR044
C                            SION (SSAR)                                MULTR045
C                    ANS(5)  DEGREES OF FREEDOM ASSOCIATED WITH SSAR    MULTR046
C                    ANS(6)  MEAN SQUARE OF SSAR                        MULTR047
C                    ANS(7)  SUM OF SQUARES OF DEVIATIONS FROM REGRES-  MULTR048
C                            SION (SSDR)                                MULTR049
C                    ANS(8)  DEGREES OF FREEDOM ASSOCIATED WITH SSDR    MULTR050
C                    ANS(9)  MEAN SQUARE OF SSDR                        MULTR051
C                    ANS(10) F-VALUE                                    MULTR052
C                                                                       MULTR053
C        REMARKS                                                        MULTR054
C           N MUST BE GREATER THAN K+1.                                 MULTR055
C                                                                       MULTR056
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                  MULTR057
C           NONE                                                        MULTR058
C                                                                       MULTR059
C        METHOD                                                         MULTR060
C           THE GAUSS-JORDAN METHOD IS USED IN THE SOLUTION OF THE      MULTR061
C           NORMAL EQUATIONS.  REFER TO W. W. COOLEY AND P. R. LOHNES,  MULTR062
C           'MULTIVARIATE PROCEDURES FOR THE BEHAVIORAL SCIENCES',      MULTR063
C           JOHN WILEY AND SONS, 1962, CHAPTER 3; AND B. OSTLE,         MULTR064
C           'STATISTICS IN RESEARCH', THE IOWA STATE COLLEGE PRESS,     MULTR065
C           1964, CHAPTER 8.                                            MULTR066
C                                                                       MULTR067
C     ..................................................................MULTR068
C                                                                       MULTR069
      SUBROUTINE MULTR (N,K,XBAR,STD,D,RX,RY,ISAVE,B,SB,T,ANS)          MULTR070
      DIMENSION XBAR(1),STD(1),D(1),RX(1),RY(1),ISAVE(1),B(1),SB(1),    MULTR071
     1          T(1),ANS(1)                                             MULTR072
C                                                                       MULTR073
C     ..................................................................MULTR074
C                                                                       MULTR075
C     IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE     MULTR076
C     C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION         MULTR077
C     STATEMENT WHICH FOLLOWS.                                          MULTR078
C                                                                       MULTR079
C     DOUBLE PRECISION XBAR,STD,D,RX,RY,B,SB,T,ANS,RM,BO,SSAR,SSDR,SY,  MULTR080
     1                 FN,FK,SSARM,SSDRM,F,DABS,DSQRT                   MULTR081
```

```
C                                                                       MULTR082
C         THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS   MULTR083
C         APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS     MULTR084
C         ROUTINE.                                                      MULTR085
C                                                                       MULTR086
C         THE DOUBLE PRECISION VERSION OF THIS SUBROUTINE MUST ALSO     MULTR087
C         CONTAIN DOUBLE PRECISION FORTRAN FUNCTIONS.  SQRT AND ABS IN  MULTR088
C         STATEMENTS 122, 125, AND 135 MUST BE CHANGED TO DSQRT AND DABS.MULTR089
C                                                                       MULTR090
C         ******************************************************************MULTR091
C                                                                       MULTR092
      MM=K+1                                                            MULTR093
C                                                                       MULTR094
C         BETA WEIGHTS                                                  MULTR095
C                                                                       MULTR096
      DO 100 J=1,K                                                      MULTR097
  100 B(J)=0.0                                                          MULTR098
      DO 110 J=1,K                                                      MULTR099
      L1=K*(J-1)                                                        MULTR100
      DO 110 I=1,K                                                      MULTR101
      L=L1+I                                                            MULTR102
  110 B(J)=B(J)+RY(I)*RX(L)                                             MULTR103
      RM=0.0                                                            MULTR104
      B0=0.0                                                            MULTR105
      L1=ISAVE(MM)                                                      MULTR106
C                                                                       MULTR107
C         COEFFICIENT OF DETERMINATION                                  MULTR108
C                                                                       MULTR109
      DO 120 I=1,K                                                      MULTR110
      RM=RM+B(I)*RY(I)                                                  MULTR111
C                                                                       MULTR112
C         REGRESSION COEFFICIENTS                                       MULTR113
C                                                                       MULTR114
      L=ISAVE(I)                                                        MULTR115
      B(I)=B(I)*(STD(L1)/STD(L))                                        MULTR116
C                                                                       MULTR117
C         INTERCEPT                                                     MULTR118
C                                                                       MULTR119
  120 B0=B0+B(I)*XBAR(L)                                                MULTR120
      B0=XBAR(L1)-B0                                                    MULTR121
C                                                                       MULTR122
C         SUM OF SQUARES ATTRIBUTABLE TO REGRESSION                     MULTR123
C                                                                       MULTR124
      SSAR=RM*D(L1)                                                     MULTR125
C                                                                       MULTR126
C         MULTIPLE CORRELATION COEFFICIENT                              MULTR127
C                                                                       MULTR128
  122 RM=DSQRT(DABS(RM))                                                MULTR129
C                                                                       MULTR130
C         SUM OF SQUARES OF DEVIATIONS FROM REGRESSION                  MULTR131
C                                                                       MULTR132
      SSDR=D(L1)-SSAR                                                   MULTR133
C                                                                       MULTR134
C         VARIANCE OF ESTIMATE                                          MULTR135
C                                                                       MULTR136
      FN=N-K-1                                                          MULTR137
      SY=SSDR/FN                                                        MULTR138
C                                                                       MULTR139
C         STANDARD DEVIATIONS OF REGRESSION COEFFICIENTS                MULTR140
C                                                                       MULTR141
      DO 130 J=1,K                                                      MULTR142
      L1=K*(J-1)+J                                                      MULTR143
      L=ISAVE(J)                                                        MULTR144
  125 SB(J)=DSQRT(DABS((RY(L1)/D(L))*SY))                               MULTR145
C                                                                       MULTR146
C         COMPUTED T-VALUES                                             MULTR147
C                                                                       MULTR148
  130 T(J)=B(J)/SB(J)                                                   MULTR149
C                                                                       MULTR150
C         STANDARD ERROR OF ESTIMATE                                    MULTR151
C                                                                       MULTR152
  135 SY=DSQRT(DABS(SY))                                                MULTR153
C                                                                       MULTR154
C         F VALUE                                                       MULTR155
C                                                                       MULTR156
      FK=K                                                              MULTR157
      SSARM=SSAR/FK                                                     MULTR158
      SSDRM=SSDR/FN                                                     MULTR159
      F=SSARM/SSDRM                                                     MULTR160
C                                                                       MULTR161
      ANS(1)=B0                                                         MULTR162
      ANS(2)=RM                                                         MULTR163
      ANS(3)=SY                                                         MULTR164
      ANS(4)=SSAR                                                       MULTR165
      ANS(5)=FK                                                         MULTR166
      ANS(6)=SSARM                                                      MULTR167
      ANS(7)=SSDR                                                       MULTR168
      ANS(8)=FN                                                         MULTR169
      ANS(9)=SSDRM                                                      MULTR170
      ANS(10)=F                                                         MULTR171
      RETURN                                                            MULTR172
      END                                                               MULTR173
```

POLYNOMIAL REGRESSION.....AA1

NUMBER OF OBSERVATIONS 10

STARTING TEMPERATURE = 715.00

TEMPERATURE DIFFERENCE = 9

TEMPERATURE INCREMENT = 1.00

POLYNOMIAL REGRESSION OF DEGREE 1

INTERCEPT 0.22428377851647370 01

REGRESSION COEFFICIENTS
-0.19275652754302330-02

ANALYSIS OF VARIANCE FOR 1 DEGREE POLYNOMIAL

SOURCE OF VARIATION	DEGREE OF FREEDOM	SUM OF SQUARES	MEAN SQUARE	F VALUE	IMPROVEMENT IN TERMS OF SUM OF SQUARES
DUE TO REGRESSION	1	0.00031	0.00031	421.25456	0.00031
DEVIATION ABOUT REGRESSION	8	0.00001	0.00000		
TOTAL	9	0.00031			

POLYNOMIAL REGRESSION OF DEGREE 2

INTERCEPT -0.14270623065806340 03

REGRESSION COEFFICIENTS
0.24392320854704500 00 -0.10424738325366490-03

ANALYSIS OF VARIANCE FOR 2 DEGREE POLYNOMIAL

SOURCE OF VARIATION	DEGREE OF FREEDOM	SUM OF SQUARES	MEAN SQUARE	F VALUE	IMPROVEMENT IN TERMS OF SUM OF SQUARES
DUE TO REGRESSION	2	0.00031	0.00016	13133.43067	0.00001
DEVIATION ABOUT REGRESSION	7	0.00000	0.00000		
TOTAL	9	0.00031			

POLYNOMIAL REGRESSION OF DEGREE 3

NO IMPROVEMENT

POLYNOMIAL REGRESSION.....AA1

POLYNOMIAL REGRESSION OF DEGREE 2

TABLE OF RESIDUALS

OBSERVATION NO.	X VALUE	Y VALUE	Y ESTIMATE	RESIDUAL	ERROR
1	1174.66900	-0.02255	-0.02267	0.00012	-0.54195
2	1175.66900	-0.02379	-0.02376	-0.00004	0.15022
3	1176.66900	-0.02516	-0.02506	-0.00011	0.42278
4	1177.66900	-0.02667	-0.02658	-0.00009	0.34602
5	1178.66900	-0.02834	-0.02829	-0.00005	0.15947
6	1179.66900	-0.03019	-0.03022	0.00002	-0.08216
7	1180.66900	-0.03226	-0.03235	0.00009	-0.27111
8	1181.66900	-0.03458	-0.03470	0.00012	-0.35321
9	1182.66900	-0.03720	-0.03725	0.00006	-0.15144
10	1183.66900	-0.04015	-0.04001	-0.00014	0.35469

```
C                                                                       SYSS  10
C     ******************************************************************SYSS  20
C                                                                       SYSS  30
      SUBROUTINE SYSSOL(M,N,A,B,ALPHA,A1,B1,TOL,DELEPS,Y,Y,            SYSS  40
     *                  ABT,FF,JACOB,JERROR)                           SYSS  50
      IMPLICIT REAL*8(A-H,O-Z)                                         SYSS  60
      EXTERNAL JACOB                                                   SYSS  70
      DIMENSION X(1)                                                   SYSS  80
      DIMENSION ALPHA(10),A1(10,10),B1(10,10),Y(2600),ABT(10)          SYSS  90
      DIMENSION F(2600),LU(2600),RES(2600),SK(2600),                   SYSS 100
     *ABSEX(10),ABR(10),TEMP(10)                                       SYSS 110
      DIMENSION AA(50),BB(50),CC(20)                                   SYSS 120
      COMMON /CI/ EPSNU                                                SYSS 130
      LOGICAL DIVNEW                                                   SYSS 140
C*********************************************************************SYSS 150
C*                                                                     *SYSS 160
C*    PURPOSE :  THIS IS A VARIABLE ORDER, VARIABLE                    *SYSS 170
C* (UNIFORM) STEP SOLVER FOR TWO-POINT BOUNDARY VALUE FIRST            *SYSS 180
C* ORDER SMOOTH NONLINEAR SYSTEMS OF THE FORM                          *SYSS 190
C*                                                                     *SYSS 200
C*  (1)      Y' = F(X,Y)  ;  A1 . Y(A) + B1 . Y(B) = ALPHA             *SYSS 210
C*                                                                     *SYSS 220
C* IT ATTEMPTS TO PRODUCE A DISCRETE SOLUTION WITH MAXIMUM             *SYSS 230
C* ABSOLUTE ERROR LESS THAN  TOL  ON A UNIFORM GRID CONTAI-            *SYSS 240
C* NING THE ONE DEFINED BY THE USER WITH THE PARAMETER  N.             *SYSS 250
C* THE METHOD, MORE DETAILS AND TESTS ARE DESCRIBED IN                 *SYSS 260
C* 'A VARIABLE ORDER, VARIABLE STEP, FINITE DIFFERENCE                 *SYSS 270
C* METHOD FOR NONLINEAR MULTIPOINT BOUNDARY VALUE PROBLEMS'            *SYSS 280
C* BY M. LENTINI AND V. PEREYRA, PUB. 73-06, DEPTO. DE                 *SYSS 290
C* COMPUTACION, FAC. DE CIENCIAS, UNIVERSIDAD CENTRAL DE               *SYSS 300
C* VENEZUELA (1973).                                                   *SYSS 310
C*                                                                     *SYSS 320
C*********************************************************************SYSS 330
      CONTINUE                                                         SYSS 340
C*********************************************************************SYSS 350
C*                                                                     *SYSS 360
C*  USE :  PRESENT DIMENSIONING OF ARRAYS ALLOWS PROCESSING            *SYSS 370
C*         OF SYSTEMS OF UP TO                                         *SYSS 380
C*                            MMAX = 10 EQUATIONS                      *SYSS 390
C*         ON GRIDS WITH UP TO                                         *SYSS 400
C*                           NMAX = 2600/M POINTS.                     *SYSS 410
C*         THESE SIZES CAN BE VARIED BY CHANGING THE                   *SYSS 420
C*         DIMENSIONS APPROPRIATELY.                                   *SYSS 430
C*                                                                     *SYSS 440
C*       THE USER MUST SPECIFY EACH OF THE FOLLOWINGS:                 *SYSS 450
C*                                                                     *SYSS 460
C*   M  -  NUMBER OF EQUATIONS                                         *SYSS 470
C*   N  -  NUMBER OF POINTS IN THE BASIC GRID (COUNTING THE            *SYSS 480
C*         END POINTS). N  MUST BE  > 3.                               *SYSS 490
C*         UPON EXIT, N  WILL CONTAIN THE FINAL GRID SIZE              *SYSS 500
C*         IN WHICH  Y  HAS BEEN COMPUTED.                             *SYSS 510
C*   A  -  LEFT BOUNDARY POINT                                         *SYSS 520
C*   B  -  RIGHT BOUNDARY POINT                                        *SYSS 530
C*   ALPHA A1 ,B1  -  ARRAYS OF DIMENSIONS  MMAX                       *SYSS 540
C*                    CONTAINING BOUNDARY CONDITIONS.SEE(1)            *SYSS 550
C*   TOL  -  DESIRED FINAL ABSOLUTE ERROR.                             *SYSS 560
C*                                                                     *SYSS 570
C* DELEPS - (A) IF DELEPS <= 0 THEN THE PROGRAM SETS Y=0 AS            *SYSS 580
C*              INITIAL VALUES.                                        *SYSS 590
C*              IF DELEPS IS SET TO ANY VALUE >= 1 THEN THE            *SYSS 600
C*              PROGRAM ASSUMES THAT THE USER WILL GIVE                *SYSS 610
C*              INITIAL VALUES FOR Y. THESE INITIAL VALUES             *SYSS 620
C*              (ON THE INITIAL MESH) MUST BE STORED AS                *SYSS 630
C*              INDICATED BELOW (SEE OUTPUT PARAMETERS).               *SYSS 640
C*          (B) A CONTINUATION METHOD CAN BE EMPLOYED AS AN            *SYSS 650
C*              OPTION IN SEVERELY NON LINEAR CASES IN                 *SYSS 660
C*              ORDER TO GENERATE GOOD INITIAL VALUES. THIS            *SYSS 670
C*              PRESUPOSES THAT,THE USER HAS EMBEDDED HIS              *SYSS 680
C*              PROBLEM IN A ONE PARAMETER FAMILY:                     *SYSS 690
C*              Y' = F(X, Y, EPSNU),                                   *SYSS 700
C*              SUCH THAT FOR EPSNU =0 THE RESULTING                   *SYSS 710
C*              PROBLEM IS ''SIMPLER'', AND FOR EPSNU= 1               *SYSS 720
C*              THE ORGINAL PROBLEM IS RECOVERED. THE                  *SYSS 730
C*              PROGRAM WILL AUTOMATICALLY ATTEMPT TO GO               *SYSS 740
C*              FROM EPSNU = 0 TO EPSNU = 1 IN STEPS OF                *SYSS 750
C*              DELEPS. THUS IN THIS CASE                              *SYSS 760
C*              0 < DELEPS < 1                                         *SYSS 770
C*              ONCE EPSNU > = 1 THE REGULAR PROCEDURE                 *SYSS 780
C*              CONTINUES. THE USER MUST GIVE INITIAL                  *SYSS 790
C*              VALUES FOR Y AS IN (A).                                *SYSS 800
C*                                                                     *SYSS 810
C*              WHEN USING THIS OPTION THE USER MUST                   *SYSS 820
C*              INCLUDE IN HIS SUBROUTINES FF, JACOB                   *SYSS 830
C*              THE COMMON CARDS                                       *SYSS 840
C*              COMMON /C1/ EPSNU                                      *SYSS 850
C*              IN ORDER TO DESCRIBE THE PARAMETRIZED                  *SYSS 860
C*              PROBLEM                                                *SYSS 870
C*                                                                     *SYSS 880
C*********************************************************************SYSS 890
      CONTINUE                                                         SYSS 900
C*********************************************************************SYSS 910
C*                                                                     *SYSS 920
C*     *** USER PROVIDED SUBROUTINES ***                               *SYSS 930
C*                                                                     *SYSS 940
C*                                                                     *SYSS 950
C*   FF  -  NAME OF SUBROUTINE FOR EVALUATING  F(X,Y)  ON              *SYSS 960
C*          THE WHOLE MESH.  THE SUBROUTINE ITSELF MUST BE             *SYSS 970
C*          PROVIDED BY THE USER AND IT SHOULD HAVE THE                *SYSS 980
C*          FOLLOWING HEADING                                          *SYSS 990
C*                                                                     *SYSS1000
C*                SUBROUTINE FF(X,Y,N,F)                               *SYSS1010
C*                                                                     *SYSS1020
C*          WHERE  F  IS THE VECTOR CONTAINING THE VALUES              *SYSS1030
C*          OF  F(X,Y). STORAGE OF  F  IS THE SAME AS THAT             *SYSS1040
C*          OF  Y  (SEE BELOW) .                                       *SYSS1050
C*                                                                     *SYSS1060
C*   JACOB  -  NAME OF A SUBROUTINE FOR EVALUATING THE                 *SYSS1070
C*             JACOBIAN OF  F(X,Y)  AT A GIVEN GRID POINT.             *SYSS1080
C*             THE SUBROUTINE ITSELF MUST BE PROVIDED BY               *SYSS1090
C*             THE USER, AND IT SHOULD HAVE THE FOLLOWING              *SYSS1100
C*             HEADING                                                 *SYSS1110
C*                                                                     *SYSS1120
C*                                                                     *SYSS1130
C*                SUBROUTINE JACOB(XX,YY,XJAC)                         *SYSS1140
C*                                                                     *SYSS1150
C*             WHERE  XX  IS THE GIVEN GRID POINT,  YY  IS             *SYSS1160
C*             THE CURRENT VALUE OF  Y  AT  XX, AND THE                *SYSS1170
C*             MMAX*NMAX  ARRAY  XJAX  SHOULD BE FILLED BY             *SYSS1180
C*             THE USER WITH THE CORRESPONDING PARTIAL                 *SYSS1190
```

```
C*              DERIVATIVES ACCORDING TO                          *
C*                   XJAC(I,J) = DFI/DYJ                          *
C*                                                                *
C******************************************************************
      CONTINUE
C******************************************************************
C*                                                                *
C*              *** OUTPUT  PARAMETERS ***                        *
C*                                                                *
C*   SEE  N  ABOVE                                                *
C*                                                                *
C*   Y  -  NMAX*MMAX-VECTOR CONTAINING THE COMPUTED               *
C*         SOLUTION ON THE FINAL GRID                             *
C*         THE GRID VECTOR FUNCTION  Y(X)  IS STORED              *
C*         SEQUENTIALLY IN THE FORM:                              *
C*            Y1(X1),Y2(X1),....,YM(X1),Y1(X2),....YM(XN),        *
C*      V  THE PROGRAM SETS  Y  TO ZERO INITIALLY WHEN            *
C*         DELEPS <= 0.                                           *
C*   X  -  NMAX-VECTOR CONTAINING THE FINAL GRID POINTS.          *
C*   ABT - M-VECTOR CONTAINING MAX ABSOLUTE ERROR ON THE          *
C*         GRID FOR EACH COMPONENT OF THE APPROXIMATE             *
C*         SOLUTION                                               *
C*                                                                *
C*   JERROR  -  ERROR CODED INTEGER VARIABLE.                     *
C*     JERROR = 0  REQUIRED TOLERANCE WAS REACHED                 *
C*     JERROR = 1  M  OR INITIAL  N  ARE OUT OF RANGE.            *
C*            CORRECTING ACTION:  CHECK THAT INPUT DATA           *
C*            SATISFIES  0< M <= MMAX  AND  3 < N <= NMAX         *
C*            AND RERUN.                                          *
C*     JERROR = 2  THE PROGRAM HAS ATTEMPTED TO USE A GRID        *
C*          WITH MORE THAN  NMAX  POINTS                          *
C*          CORRECTING ACTION:  IF PARTIAL RESULTS SEEM           *
C*          REASONABLE, INCREASE  MMAX*NMAX = 2600 AND            *
C*          CHANGE DIMENSIONS IN ALL SUBROUTINES ACCORDING-       *
C*          LY IN ORDER TO ALLOW FINER GRIDS                      *
C******************************************************************
      CONTINUE
C******************************************************************
C*                                                                *
C* *** OTHER SUBROUTINES NEEDED BY SYSSOL ***                     *
C*                                                                *
C* SYSLIN - SOLVES A BLOCK LINEAR SYSTEM OF SPECIAL TYPE.         *
C* DGELG  - SOLVES A LINEAR SYSTEM OF EQUATIONS                   *
C* DARRAY - TRANSFORMS BETWEEN TWO AND ONE DIMENSIONAL            *
C*          STORAGE.                                              *
C* U2DCGS - DEFERRED CORRECTOIN GENERATOR AND INTERPOLATION       *
C* CDEGEN - WEIGHT GENERATOR                                      *
C*                                                                *
C******************************************************************
C
C******************************************************************
C*                                                                *
C* IN CALLING PROGRAM THE PARAMETER ARRAY X MUST BE               *
C* DIMENSIONED AS X( 2600/M )                                     *
C*                                                                *
C* DIMENSIONS INVOLVED IN THE FOLLOWING PARAMETER ARRAYS          *
C* ARE MMAX = 10 AND NMAX*MMAX = 2600, AND THEY MUST BE           *
C*  DIMENSIONED ACCORDINGLY IN THE CALLING PROGRAM                *
C*                                                                *
C*  DIMENSION ALPHA(10),A1(10,10),B1(10,10),Y(2600),ABT(10)*
C*                                                                *
C*  FOLLOWING ARRAYS ARE WORKING AREAS.                           *
C*  DEMENSION F(2600),UU(2600),SK(2600),                          *
C*  ARSEX(10),ARR(10),TEMP(10)                                    *
C*                                                                *
C* WORKING AREAS WITH SIZES RELATED TO MAX. NUMBER OF             *
C* DEFERRED CORRECTIONS = 20 , WHICH SHOULD BE ADEQUATED          *
C*    FOR ALL PURPOSES.                                           *
C*                                                                *
C******************************************************************
C
C
      MMAX=10
      NMAX=2600/M
      IF(M.LE.0).OR.(M.GT.MMAX).OR.(N.LE.3).OR.(N.GT.NMAX)) GO TO 1
      GO TO 10
    1 JERROR=1
      RETURN
C
C<<<...            ERROR  EXIT  1        .........
C
C     INITIALIZATION
C
   10 MPN=M*N
C
C******************************************************************
C*                                                                *
C*    THIS CONSTANT IS MACHINE DEPENDENT:                         *
C*    EPSMAC  IS APPROXIMATELY 10* RELATIVE PRECISION OF          *
C*    FLOATING POINT ARITHMETIC                                   *
C*                                                                *
C******************************************************************
C
      EPSMAC=5.D-15
      EPSNU=0.D0
      JERROR=0
      M1=M+1
      M2=M*M
      IF (DELEPS.GT.0.D0) GO TO 60
C
C     FIRST APPROXIMATION FOR  Y
C
   40 DO 50 I=1,MPN
   50 Y(I)=0.D0
C
   60 BB(1)=1.D0
      DO 105 I=2,50
  105 BB(I)=0.D0
C******************************************************************
C*                                                                *
C* IN CORRECTION  NU=0 THE RESIDUAL IN THE NEWTON ITERATION       *
C* MUST BE REDUCED IN FORM BELOW  EPS=C*H**4 , WHERE  C  IS       *
C* A SMALL CONSTANT. FOR  NU > 0, WE USE THE ERROR ESTIMATE       *
C* CORRESPONDING TO CORRECTION (NU-1) : ERROLD, IN ORDER TO       *
C* OBTAIN THE NEW  EPS= C*H**2*ERROLD.  IN ALL CASES  EPS         *
C* IS NOT ALLOWED TO BE SMALLER THAN  EPSMAC.                     *
C*                                                                *
C******************************************************************
      EPS=DMAX1(EPSMAC,.01D0*((B-A)/(N-1))**4)
      NU=0
```

```
C                                                                       SYSS2400
C....>>>            ENTER  UPON  STEP  HALVING        ......           SYSS2410
C                                                                       SYSS2420
  120 ERROLD=1.0D20                                                     SYSS2430
      KMAX=(N-2)/2                                                      SYSS2440
      MPNM=MPN-M                                                        SYSS2450
      C1=0.8D0                                                          SYSS2460
      N1=N-1                                                            SYSS2470
      H=(B-A)/N1                                                        SYSS2480
      HCUB=H**2                                                         SYSS2490
      DO 150 I=1,MPNM                                                   SYSS2500
  150 SK(I)=0.D0                                                        SYSS2510
      X(1)=A                                                            SYSS2520
      X(N)=B                                                            SYSS2530
      DO 200 I=2,N1                                                     SYSS2540
          I1=I-1                                                        SYSS2550
  200 X(I)=A+I1*H                                                       SYSS2560
      IF(NU .EQ. 0) GO TO 405                                           SYSS2570
C                                                                       SYSS2580
C     WHEN STEP-HALVING WE HAVE TO INITIALIZE  SK  IF  NU .GT. 0        SYSS2590
C                                                                       SYSS2600
      CALL FF(X,Y,N,F)                                                  SYSS2610
      CALL U2DCGS(NU,2,2,N1,M,AA,F,RES,IERROR)                          SYSS2620
      DO 300 I=1,MPNM                                                   SYSS2630
  300 SK(I)=H*RES(I)                                                    SYSS2640
C                                                                       SYSS2650
C....>>>            NEWTON ITERATION STARTS     .........              SYSS2660
C                                                                       SYSS2670
  405 INWT=0                                                            SYSS2680
      REOLD=1.0D20                                                      SYSS2690
      DIVNEW=.FALSE.                                                    SYSS2700
      IF(NU+1.LE.KMAX) GO TO 410                                        SYSS2710
C                                                                       SYSS2720
C     MAXIMUM NUMBER OF CORRECTIONS ON THIS MESH HAS BEEN               SYSS2730
C     REACHED. GO TO *STEP HALVING*.                                    SYSS2740
C                                                                       SYSS2750
      NU=NU+1                                                           SYSS2760
      GO TO 2600                                                        SYSS2770
C...>>>    LABEL 410 IS INPUT FOR NEWTON ITERATION     ..........       SYSS2780
C                                                                       SYSS2790
C                                                                       SYSS2800
C     RESIDUAL  COMPUTATION                                             SYSS2810
C                                                                       SYSS2820
  410 RABS=0.D0                                                         SYSS2830
      DO 700 I=1,M                                                      SYSS2840
         SUM=ALPHA(I)                                                   SYSS2850
         DO 600 J=1,M                                                   SYSS2860
  600    SUM=SUM-A1(I,J)*Y(J)-B1(I,J)*Y(MPNM+J)                         SYSS2870
         RES(I)=SUM                                                     SYSS2880
         TEM=DABS(RES(I))                                               SYSS2890
         IF(TEM .GT. RABS) RABS=TEM                                     SYSS2900
  700 CONTINUE                                                          SYSS2910
  800 CALL FF (X,Y,N,F)                                                 SYSS2920
      DO 900 I=2,N                                                      SYSS2930
         K1=(I-1)*M                                                     SYSS2940
         DO 900 J=1,M                                                   SYSS2950
         K1J=K1+J                                                       SYSS2960
         K1JM=K1J-M                                                     SYSS2970
         RES(K1J)=-Y(K1J)+Y(K1JM)+H/2*(F(K1J)+F(K1JM))+SK(K1JM)         SYSS2980
         TEM=DABS(RES(K1J))                                             SYSS2990
         IF(TEM .GT. RABS) RABS=TEM                                     SYSS3000
  900 CONTINUE                                                          SYSS3010
C                                                                       SYSS3020
C     THE FIRST TIME THROUGH WE DON'T CHECK ANYTHING                    SYSS3030
C                                                                       SYSS3040
      IF(INWT .EQ. 0) GO TO 950                                         SYSS3050
  910 IF(RABS .LT. REOLD .OR. INWT .EQ. 1) GO TO 920                    SYSS3060
C                                                                       SYSS3070
C     IF THE RESIDUAL INCREASES AFTER THE FIRST ITERATION               SYSS3080
C     WE ASSUME DIVERGENCE AND GO TO HALVE THE STEP SIZE                SYSS3090
C                                                                       SYSS3100
      DIVNEW=.TRUE.                                                     SYSS3110
      NU=NU+1                                                           SYSS3120
      GO TO 2600                                                        SYSS3130
  920 IF(RABS .LE. EPS .OR. INWT .GE. 5) GO TO 1500                     SYSS3140
C                                                                       SYSS3150
C....          NEWTON EXIT (CONVERGENCE OR TOO MANY ITERATIONS)         SYSS3160
C                                                                       SYSS3170
  950 CALL SYSLIN(M,N,X,Y,H,JACOB,RES,A1,B1,UU)                         SYSS3180
      REOLD=RABS                                                        SYSS3190
C                                                                       SYSS3200
C     APPROXIMATE SOLUTION IS CORRECTED                                 SYSS3210
C                                                                       SYSS3220
 1100 DO 1300 I=1,MPN                                                   SYSS3230
 1300 Y(I)=Y(I)+UU(I)                                                   SYSS3240
C                                                                       SYSS3250
C     NEXT TWO INSTRUCTION ARE FOR CONTROL OF PARAMETER                 SYSS3260
C     IN CONTINUATION METHOD                                            SYSS3270
C                                                                       SYSS3280
      EPSNU=DMIN1(EPSNU+DELEPS,1.D0)                                    SYSS3290
      IF(EPSNU .GE. 1.D0 .OR. DELEPS .LE. 0.D0) INWT=INWT+1             SYSS3300
      GO TO 410                                                         SYSS3310
C                                                                       SYSS3320
C     CORRECTION AND ERROR CONTROL STARTS                               SYSS3330
C                                                                       SYSS3340
 1500 NU=NU+1                                                           SYSS3350
      NU2=2*NU+1                                                        SYSS3360
      AA(NU2)=-DFLOAT(NU)/DFLOAT((2**(2*NU-1)*NU2))                     SYSS3370
      AA(NU2+1)=0.D0                                                    SYSS3380
      CALL U2DCGS(NU,2,2,N1,M,AA,F,RES,IERROR)                          SYSS3390
      IF(IERROR .EQ. 1) GO TO 2600                                      SYSS3400
      DO 1700 I=1,MPNM                                                  SYSS3410
         AUXI=RES(I)*H                                                  SYSS3420
         RES(I)=SK(I)-AUXI                                              SYSS3430
         SK(I)=AUXI                                                     SYSS3440
 1700 CONTINUE                                                          SYSS3450
      CALL SYSLIN(M,N,X,Y,H,JACOB,RES,A1,B1,UU)                         SYSS3460
C                                                                       SYSS3470
C     ESTIMATE FOR MAX. ABSOLUTE ERROR (BY COMPONENTS)                  SYSS3480
C                                                                       SYSS3490
      ERRNEW=0.D0                                                       SYSS3500
      DO 1900 J=1,M                                                     SYSS3510
 1900 ABT(J)=0.D0                                                       SYSS3520
      DO 2100 I=1,N                                                     SYSS3530
      DO 2100 J=1,M                                                     SYSS3540
         KI=(I-1)*M+J                                                   SYSS3550
         UI=DABS(UU(KI))                                                SYSS3560
         IF(UI .GT. ABT(J)) ABT(J)=UI                                   SYSS3570
 2100 CONTINUE                                                          SYSS3580
      DO 2300 J=1,M                                                     SYSS3590
```

```
      IF(ABT(J) .GT. ERRNEW) ERRNEW=ABT(J)
 2300 CONTINUE
      K=NU-1
 2500 IF(ERRNEW .LE. TOL) RETURN
C
C<<<...              PRECISION  ACHIEVED     ..............
C
C     CC(K+1) CONTAINS ESTIMATED ERROR FOR CORRECTION  K
C     ON MESH SIZE H/2 (UNDER THE HYPOTHESIS : ERROR IN
C          CORRECTION  K(H) = C*H**(2*K+2)).
C
      CC(NU)=ERRNEW*4.D0**(-NU)
      IF(ERRNEW .LE. .1*ERROLD) GO TO 2550
      IF(ERRNEW .GT. C1*ERROLD) GO TO 2600
      C1=.5D0*C1
C
C     EITHER KEEP CORRECTING  ...
C
C*************************************************************
C*                                                           *
C* THE ERROR REDUCTION THRESHOLD C1 IS SET ORIGINALLY (AND   *
C* ARBITRARARILY) TO 0.8. IF  C1*ERROLD < ERRNEW WE HALVE    *
C* THE STEP. EACH TIME THAT 0.1*ERROLD < ERRNEW <C1*EPROLD   *
C* WE SET C1 TO 0.5*C1, THUS ACTUALLY ALLOWING THIS TO HA-   *
C* PPEN A MAXIMUM OF THREE TIMES, BEFORE THE MORE STRICT     *
C* TEST WITH 0.1*ERROLD TAKES OVER COMPLETELY.               *
C* ERROLD IS THE ERROR ESTIMATE FOR THE LAST CORRECTION BUT  *
C* ONE, WHILE ERRNEW IS THE ONE CORRESPONDING TO THE LAST    *
C* CORRECTION.                                               *
C*                                                           *
C*************************************************************
 2550 ERROLD=ERRNEW
      EPS=DMAX1(EPSMAC,.01D0*H**2*ERROLD)
      GO TO 405
C
C....>>>             OR  HALVE  THE  STEP  SIZE      ......
C
 2600 IF(2*N-1 .LE. NMAX) GO TO 2625
      JERROR=2
      RETURN
C
C<<<....             TOO MANY GRID POINTS      .....
C
 2625 N=2*N-1
      MPN=M*N
C
C     IF NEWTON DIVERGED WE START AGAIN WITH NU=0
C
      IF(DIVNEW) GO TO 40
C*************************************************************
C*                                                           *
C* NOW WE DECIDE THE LEVEL OF CORRECTION ON THE NEW GRID     *
C* WE ASSUME THAT THE LAST ESTIMATED ERROR (PRESENTLY  IN    *
C* ERROLD) WILL BE PRESERVED AFTER INTERPOLATING, AND        *
C* THEREFORE WE LOCATE THE FIRST INDEX I FOR WHICH           *
C* CC(I)<=ERROLD, WHERE  CC(I) IS THE PREDICTED ERROR FOR    *
C* THE (I-1) CORRECTION ON THE NEW GRID.                     *
C*                                                           *
C*************************************************************
      NUINT=NU
      IF(ERRNEW .GE. ERROLD) GO TO 2650
      ERROLD=ERRNEW
 2650 DO 2700 I=1,NU
      IF(ERROLD .LT. CC(I)) GO TO 2700
      GO TO 2750
 2700 CONTINUE
 2750 NU=I-1
      EPS=DMAX1(EPSMAC,.005D0*H**2*ERROLD)
C*************************************************************
C*                                                           *
C* COMPUTATION OF FIRST APPROXIMATION FOR  Y  ON NEW GRID    *
C* BY MEANS OF U2DCGS WILL GIVE ORDER OF INTERPOLATION       *
C* (2*NU+2), WHERE  NU  IS THE LAST SUCCESFULL CORRECTION    *
C* PERFORMED ON THE COARSER GRID.                            *
C*                                                           *
C*************************************************************
 2800 NO2=(N+1)/2
      MPO2=M*NO2
      DO 2900 I=1,MPO2
 2900 RES(I)=Y(I)
      NO21=NO2-1
      CALL U2DCGS(NUINT,2,0,NO21,M,BB,Y,SK,IERROR)
      DO 3100 I=1,NO21
      KI=(I-1)*M
      KI2=2*KI
      DO 3100 L=1,M
      Y(KI2+L)=RES(KI+L)
 3100 Y(KI2+M+L)=SK(KI+L)
      DO 3200 L=1,M
 3200 Y(MPN-M+L)=RES(MPO2-M+L)
      GO TO 120
C.....               START  ON  NEW  GRID      .......
      END
C
C     ..................................................
C
      SUBROUTINE SYSLIN(M,N,X,Y,H,JACOB,RES,A1,B1,UU)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION X(1),Y(1),UU(1),RES(1)
C*************************************************************
C*                                                           *
C*            *** SOLUTION OF LINEAR SYSTEM ***              *
C*                                                           *
C*************************************************************
C*                                                           *
C* FOLLOWING ARRAYS ARE WORKING AREAS. DIMENSIONS INVOLVED   *
C* ARE :  MMAX=10 , MMAX+1 = 11 , MMAX**2=100 ,              *
C*        NMAX*MMAX=2600, MMAX*(MMAX+1)=110                  *
C*                                                           *
      DIMENSION T(110),U(10),S(10,10),VM(10,11),R(100),
     *          V(2600,11),AUX(10,11),A1(10,10),B1(10,10)
      DOUBLE PRECISION JAB(10,10)
C*                                                           *
C*************************************************************
C
C*************************************************************
C*                                                           *
C* IT SOLVES THE 2*2 BLOCK SYSTEM                            *
C*                  ! A ! B !   ! X0 !   ! B0 !              *
C*                  !---!---!   !----! = !----!              *
```

```
C*                  | C | D |  | X |  | B1 |                          *
C* WHERE A IS M*M AND D IS (M*N)*(M*N) AND ALL THE OTHER          *
C* BLOCKS HAVE THE APPROPIATE DIMENSIONS. D IS BLOCK LOWER        *
C* BIDIAGONAL, WITH MAIN DIAGONAL BLOCKS R(I) AND                 *
C* SUB-DIAGONAL S(I), ALL OF SIZES M*M.                           *
C* C HAS ONLY THE FIRST BLOCK DIFFERENT FROM ZERO, AND B          *
C* ONLY THE LAST BLOCK DIFFERENT FROM ZERO.                       *
C*                                                                *
C****************************************************************
      CONTINUE
C****************************************************************
C*                                                                *
C*              *** OUTLINE OF THE METHOD ***                     *
C*                                                                *
C* FIRST WE FORM C*=(C|B1) AND THEN WE SOLVE THE MATRIX           *
C* SYSTEM  DV*=C*  (V*=(V|W))                                     *
C* BY THE RECURSION: V*(0)=0,                                     *
C* R(J)V*(J)=(C*(J)-S(J)V*(J-1)), J=1,...,N                       *
C* THESE LINEAR SYSTEMS ARE SOLVED BY A STANDARD GAUSSIAN         *
C* ELIMINATION CODE (SUBROUTINE DGELG).                           *
C* FINALLY X0 IS THE SOLUTION OF THE LINEAR SYSTEM                *
C* (A - B V) X0 = B0 - B W  AND                                   *
C* X = W - V X0                                                   *
C*                                                                *
C****************************************************************
      CONTINUE
C****************************************************************
C*                                                                *
C*                        *** CAUTION ***                         *
C*                                                                *
C* THIS SUBROUTINE MANIPULATES SOME MATRICES AS ONE               *
C* DIMENSIONAL ARRAYS.                                            *
C* SUBROUTINES DGELG AND DARRAY ARE FROM THE IBM/360              *
C* SCIENTIFIC SUBROUTINES PACKAGE.                                *
C* THE SUBROUTINE DARRAY TRANSFORMS BETWEEN TYPES OF              *
C* STORAGE.                                                       *
C*                                                                *
C****************************************************************
C
  950 CALL JACOB(X(1),Y,JAB)
      M2=M*M
      MPN=M*N
      M1=M+1
C
C              *** SOLUTION OF D.V* = C* ***
C
      DO 1000 I=1,M
 1000 T(M2+I)=RES(M+I)
      DO 1800 L=2,N
      K1=(L-1)*M
      DO 1100 J=1,M
 1100 U(J)=Y(K1+J)
C
C         *** GENERATION OF JACOBIAN ***
C
      H2=.5D0*H
      DO 50 J=1,M
      K1=(J-1)*M
      DO 50 I=1,M
      S(I,J)=H2*JAB(I,J)
      IF(I .EQ. J) S(I,J)=S(I,J)+1.D0
   50 CONTINUE
      CALL JACOB(X(L),U,JAB)
      DO 60 I=1,M
      DO 60 J=1,M
         KI=(J-1)*M+I
      R(KI)=-H2*JAB(I,J)
      IF(I .EQ. J) R(KI)=R(KI)+1.D0
   60 CONTINUE
      IF(L .NE. 2) GO TO 1300
      DO 1200 J=1,M
      K1=(J-1)*M
      DO 1200 I=1,M
 1200 T(K1+I)=-S(I,J)
      GO TO 1700
C
C                    COMPUTATION OF (C* - S.V*)
C
 1300 DO 1500 K1=1,M1
      DO 1500 I=1,M
      SUM=0.D0
      DO 1400 J=1,M
 1400 SUM=SUM+S(I,J)*VM(J,K1)
 1500 T((K1-1)*M+I)=SUM
      DO 1600 I=1,M
 1600 T(M2+I)=T(M2+I)+RES((L-1)*M+I)
 1700 CALL DGELG (T,R,M,M1,1.D-7,IER)
 1750 CALL DARRAY (1,M,M1,10,11,T,VM)
      DO 1800 J=1,M1
      DO 1800 I=1,M
 1800 V((L-1)*M+I,J)=VM(I,J)
C
C                *** END OF RECURSION ***
C
      DO 2000 J=1,M1
      DO 2000 I=1,M
      SUM=0.D0
C
C     PRODUCTS B.V AND B.W
C
      DO 1900 K=1,M
 1900 SUM=SUM+B1(I,K)*VM(K,J)
 2000 AUX(I,J)=SUM
C
C     (A - B.V)
C
      DO 2100 J=1,M
      K1=(J-1)*M
      DO 2100 I=1,M
 2100 R(K1+I)=A1(I,J)-AUX(I,J)
      DO 2200 I=1,M
 2200 UU(I)=RES(I)-AUX(I,M1)
C
C     SOLUTION OF LINEAR SYSTEM
C     (A - B.V) X0 = (B0 - B.W)
C     AND COMPUTATION OF X
C
 2210 CALL DGELG(UU,R,M,1,1.D-7,IER)
 2250 DO 2400 I=M1,MPN
      SUM=V(I,M1)
```

```
      DO 2300 J=1,M                                                     SYSL1480
 2300 SUM=SUM-V(I,J)*UU(J)                                              SYSL1490
 2400 UU(I)=SUM                                                         SYSL1500
      RETURN                                                            SYSL1510
      END                                                               SYSL1520
C                                                                       U2DC  10
C     ..................................................................U2DC  20
C                                                                       U2DC  30
      SUBROUTINE U2DCGS(K,P,Q,N,M,A,Y,S,IERROR)                         U2DC  40
      IMPLICIT REAL*8(A-H,O-Z)                                          U2DC  50
      INTEGER P,Q                                                       U2DC  60
      DIMENSION A(50),Y(2600),S(2600),C(50)                             U2DC  70
C     ******************************************************************U2DC  80
C     *                                                                *U2DC  90
C     *   THIS IS A TWO POINT BOUNDARY VALUE DEFERRED CORRECTION GENERA-*U2DC 100
C     *   TOR FOR SYSTEMS OF M EQUATIONS. GIVEN THE ASYMPTOTIC EXPANSION*U2DC 110
C     *      Y(X) = SUM(A(J)*(D**(J-1))Y/(J-1))                        *U2DC 120
C     *               J = Q+1,...,Q+P*K                                *U2DC 130
C     *   AND VECTOR FUNCTION VALUES  Y(1),...,Y(N+1), CORRESPONDING TO *U2DC 140
C     *   AN UNIFORMLY H-SPACED MESH : X(I) = X(1) + (I-1)*H ,I=1,...,N+1*U2DC 150
C     *   U2DCGS WILL PRODUCE  S(1),...,S(N-1) AN  H**(Q+P*K)  ORDER    *U2DC 160
C     *   APPROXIMATION TO  Y(X)  AT MIDWAY BETWEEN EACH PAIR OF CONSEC-*U2DC 170
C     *   UTIVE GRID POINTS                                            *U2DC 180
C     *   FOR FIXED INTEGERS N,P,Q A RESTRICTION ON  K  IS             *U2DC 190
C     ***********        K .LE. (N+1-Q)/P                 ************U2DC 200
C     *   ALSO  P .GE. 1  ,  K .GE.  1                                 *U2DC 210
C     *   IERROR = 1  MEANS THAT ONE OF THESE CONDITIONS HAVE BEEN VIOL-*U2DC 220
C     *   ATED AND NO CORRECTION HAS BEEN COMPUTED. A(1),...,A(Q) ARE SET*U2DC 230
C     *   TO ZERO BY U2DCGS.                                           *U2DC 240
C     *   BOTH  Y  AND  S ARE STORED AS VECTORS: Y(1,X(1)),Y(2,X(1)),... *U2DC 250
C     *                                                                *U2DC 260
C     *   FOR MORE DETAILES SEE CHAPTER III OF 'HIGH ORDER FINITE DIFFE-*U2DC 270
C     *   RENCE SOLUTION OF DIFFERENTIAL EQUATIONS' BY V. PEREYRA. TECHN*U2DC 280
C     *   REP. STAN CS-73-348 , STANFORD UNIVERSITY (1973).            *U2DC 290
C     *                                                                *U2DC 300
C     *   APRIL 1973  ********* M. LENTINI P V.PEREYRA  ***************U2DC 310
C     *                                                                *U2DC 320
C     ******************************************************************U2DC 330
      IF(K .GT. (N+1-Q)/P .OR. P .LT. 1 .OR. K .LT. 1) GO TO 100        U2DC 340
      IF(Q .EQ. 0) GO TO 10                                             U2DC 350
      DO 20 I=1,Q                                                       U2DC 360
   20 A(I)=0.                                                           U2DC 370
   10 KK1=Q+P*K                                                         U2DC 380
      KK=KK1-1                                                          U2DC 390
      KMID=KK1/2                                                        U2DC 400
      IERROR=0                                                          U2DC 410
      KMID1=KMID-1                                                      U2DC 420
C                                                                       U2DC 430
C     UNSYMMETRIC APPROXIMATION LEFT BOUNDARY                           U2DC 440
C                                                                       U2DC 450
    1 IF(KMID1 .LT. 1) GO TO 25                                         U2DC 460
      DO 5 I=1,KMID1                                                    U2DC 470
      CALL COEGEN(KK1,I,C,A)                                            U2DC 480
      DO 7 L=1,M                                                        U2DC 490
      ACUM=0.                                                           U2DC 500
      DO 4 J=1,KK1                                                      U2DC 510
    4 ACUM=ACUM+C(J)*Y((J-1)*M+L)                                       U2DC 520
      IT=(I-1)*M+L                                                      U2DC 530
    7 S(IT)=ACUM                                                        U2DC 540
    5 CONTINUE                                                          U2DC 550
C                                                                       U2DC 560
C     CENTER RANGE                                                      U2DC 570
C                                                                       U2DC 580
   25 CALL COEGEN(KK1,KMID,C,A)                                         U2DC 590
      NF=N+1-KK1+KMID                                                   U2DC 600
      DO 40 I=KMID,NF                                                   U2DC 610
      II=I-KMID                                                         U2DC 620
      DO 39 L=1,M                                                       U2DC 630
      ACUM=0.                                                           U2DC 640
      DO 38 J=1,KK1                                                     U2DC 650
   38 ACUM=ACUM+C(J)*Y((II+J-1)*M+L)                                    U2DC 660
      IT=(I-1)*P+L                                                      U2DC 670
   39 S(IT)=ACUM                                                        U2DC 680
   40 CONTINUE                                                          U2DC 690
C                                                                       U2DC 700
C     RIGHT  BOUNDARY                                                   U2DC 710
C                                                                       U2DC 720
      KMIDP1=KMID+1                                                     U2DC 730
      DO 50 I=KMIDP1,KK                                                 U2DC 740
      CALL COEGEN(KK1,I,C,A)                                            U2DC 750
      II=N-KK                                                           U2DC 760
      DO 49 L=1,M                                                       U2DC 770
      ACUM=0.                                                           U2DC 780
      DO 48 J=1,KK1                                                     U2DC 790
   48 ACUM=ACUM+C(J)*Y((II+J-1)*M+L)                                    U2DC 800
      IT=(I+II-1)*M+L                                                   U2DC 810
   49 S(IT)=ACUM                                                        U2DC 820
   50 CONTINUE                                                          U2DC 830
      RETURN                                                            U2DC 840
  100 IERROR=1                                                          U2DC 850
      RETURN                                                            U2DC 860
      END                                                               U2DC 870
C                                                                       COEG  10
C     ..................................................................COEG  20
C                                                                       COEG  30
      SUBROUTINE COEGEN(N,NP,C,BB)                                      COEG  40
      IMPLICIT REAL*8(A-H,O-Z)                                          COEG  50
      DIMENSION C(50),BB(50),ALF(50)                                    COEG  60
C     ******************************************************************COEG  70
C     *  THIS IS A SLIGHTLY MODIFIED VERSION IN FORTRAN IV OF THE ALGOL*COEG  80
C     *  PROCEDURE  PVAND , P. 901 OF **SOLUTION OF VANDERMONDE SYSTEMS*COEG  90
C     *  OF EQUATIONS** BY A. BJORCK AND V.PEREYRA MATH. COMP. VOL. 24 *COEG 100
C     *  PP. 893-903 (1970), WHERE A COMPLETE DESCRIPTION OF THE METHOD*COEG 110
C     *  USED CAN BE FOUND.                                           *COEG 120
C     ******************************************************************COEG 130
      DO 1 I=1,N                                                        COEG 140
    1 C(I)=BB(I)                                                        COEG 150
      DO 11 I=1,N                                                       COEG 160
   11 ALF(I)=I-NP-0.5D0                                                 COEG 170
    2 NN=N-1                                                            COEG 180
      DO 6 I=1,NN                                                       COEG 190
      LL=N-I                                                            COEG 200
      DO 6 J=1,LL                                                       COEG 210
      K=N-J+1                                                           COEG 220
    6 C(K)=C(K)-ALF(I)*C(K-1)                                           COEG 230
      DO 8 I=1,NN                                                       COEG 240
      K=N-I                                                             COEG 250
      XKIN=1.D0/K                                                       COEG 260
      KM1=K+1                                                           COEG 270
      DO 8 J=KM1,N                                                      COEG 280
```

```
      C(J)=C(J)*XKIN                                                    COEG 290
      JM1=J-1                                                           COEG 300
    8 C(JM1)=C(JM1)-C(J)                                                COEG 310
      RETURN                                                            COEG 320
      END                                                               COEG 330
C                                                                       DELG  10
C     ..................................................................DELG  20
C                                                                       DELG  30
C        SUBROUTINE DGELG                                               DELG  40
C                                                                       DELG  50
C        PURPOSE                                                        DELG  60
C           TO SOLVE A GENERAL SYSTEM OF SIMULTANEOUS LINEAR EQUATIONS. DELG  70
C                                                                       DELG  80
C        USAGE                                                          DELG  90
C           CALL DGELG(R,A,M,N,EPS,IER)                                 DELG 100
C                                                                       DELG 110
C        DESCRIPTION OF PARAMETERS                                      DELG 120
C           R       - DOUBLE PRECISION M BY N RIGHT HAND SIDE MATRIX    DELG 130
C                     (DESTROYED). ON RETURN R CONTAINS THE SOLUTIONS   DELG 140
C                     OF THE EQUATIONS.                                 DELG 150
C           A       - DOUBLE PRECISION M BY M COEFFICIENT MATRIX        DELG 160
C                     (DESTROYED).                                      DELG 170
C           M       - THE NUMBER OF EQUATIONS IN THE SYSTEM.            DELG 180
C           N       - THE NUMBER OF RIGHT HAND SIDE VECTORS.            DELG 190
C           EPS     - SINGLE PRECISION INPUT CONSTANT WHICH IS USED AS  DELG 200
C                     RELATIVE TOLERANCE FOR TEST ON LOSS OF            DELG 210
C                     SIGNIFICANCE.                                     DELG 220
C           IER     - RESULTING ERROR PARAMETER CODED AS FOLLOWS        DELG 230
C                     IER=0  - NO ERROR.                                DELG 240
C                     IER=-1 - NO RESULT BECAUSE OF M LESS THAN 1 OR    DELG 250
C                              PIVOT ELEMENT AT ANY ELIMINATION STEP    DELG 260
C                              EQUAL TO 0.                              DELG 270
C                     IER=K  - WARNING DUE TO POSSIBLE LOSS OF SIGNIFI- DELG 280
C                              CANCE INDICATED AT ELIMINATION STEP K+1, DELG 290
C                              WHERE PIVOT ELEMENT WAS LESS THAN OR     DELG 300
C                              EQUAL TO THE INTERNAL TOLERANCE EPS TIMES DELG 310
C                              ABSOLUTELY GREATEST ELEMENT OF MATRIX A. DELG 320
C                                                                       DELG 330
C        REMARKS                                                        DELG 340
C           INPUT MATRICES R AND A ARE ASSUMED TO BE STORED COLUMNWISE  DELG 350
C           IN M*N RESP. M*M SUCCESSIVE STORAGE LOCATIONS. ON RETURN    DELG 360
C           SOLUTION MATRIX R IS STORED COLUMNWISE TOO.                 DELG 370
C           THE PROCEDURE GIVES RESULTS IF THE NUMBER OF EQUATIONS M IS DELG 380
C           GREATER THAN 0 AND PIVOT ELEMENTS AT ALL ELIMINATION STEPS  DELG 390
C           ARE DIFFERENT FROM 0. HOWEVER WARNING IER=K - IF GIVEN -    DELG 400
C           INDICATES POSSIBLE LOSS OF SIGNIFICANCE. IN CASE OF A WELL  DELG 410
C           SCALED MATRIX A AND APPROPRIATE TOLERANCE EPS, IER=K MAY BE DELG 420
C           INTERPRETED THAT MATRIX A HAS THE RANK K. NO WARNING IS     DELG 430
C           GIVEN IN CASE M=1.                                          DELG 440
C                                                                       DELG 450
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                  DELG 460
C           NONE                                                        DELG 470
C                                                                       DELG 480
C        METHOD                                                         DELG 490
C           SOLUTION IS DONE BY MEANS OF GAUSS-ELIMINATION WITH         DELG 500
C           COMPLETE PIVOTING.                                          DELG 510
C                                                                       DELG 520
C     ..................................................................DELG 530
C                                                                       DELG 540
      SUBROUTINE DGELG(R,A,M,N,EPS,IER)                                 DELG 550
C                                                                       DELG 560
C                                                                       DELG 570
      IMPLICIT REAL*8(A-H,O-Z)                                          DELG 580
      DIMENSION A(1),R(1)                                               DELG 590
      IF(M) 23,23,1                                                     DELG 600
C                                                                       DELG 610
C                                                                       DELG 620
C     SEARCH FOR GREATEST ELEMENT IN MATRIX A                           DELG 630
    1 IER=0                                                             DELG 640
      PIV=0.D0                                                          DELG 650
      MM=M*M                                                            DELG 660
      NM=N*M                                                            DELG 670
      DO 3 L=1,MM                                                       DELG 680
      TB=DABS(A(L))                                                     DELG 690
      IF(TB-PIV) 3,3,2                                                  DELG 700
    2 PIV=TB                                                            DELG 710
      I=L                                                               DELG 720
    3 CONTINUE                                                          DELG 730
      TOL=EPS*PIV                                                       DELG 740
C     A(I) IS PIVOT ELEMENT. PIV CONTAINS THE ABSOLUTE VALUE OF A(I)    DELG 750
C                                                                       DELG 760
C                                                                       DELG 770
C     START ELIMINATION LOOP                                            DELG 780
      LST=1                                                             DELG 790
      DO 17 K=1,M                                                       DELG 800
C                                                                       DELG 810
C     TEST ON SINGULARITY                                               DELG 820
      IF(PIV) 23,23,4                                                   DELG 830
    4 IF(IER) 7,5,7                                                     DELG 840
    5 IF(PIV-TOL) 6,6,7                                                 DELG 850
    6 IER=K-1                                                           DELG 860
    7 PIVI=1.D0/A(I)                                                    DELG 870
      J=(I-1)/M                                                         DELG 880
      I=I-J*M-K                                                         DELG 890
      J=J+1-K                                                           DELG 900
C     I+K IS ROW-INDEX, J+K COLUMN-INDEX OF PIVOT ELEMENT               DELG 910
C                                                                       DELG 920
C     PIVOT ROW REDUCTION AND ROW INTERCHANGE IN RIGHT HAND SIDE R      DELG 930
      DO 8 L=K,NM,M                                                     DELG 940
      LL=L+I                                                            DELG 950
      TB=PIVI*R(LL)                                                     DELG 960
      R(LL)=R(L)                                                        DELG 970
    8 R(L)=TB                                                           DELG 980
C                                                                       DELG 990
C     IS ELIMINATION TERMINATED                                         DELG1000
      IF(K-M) 9,18,18                                                   DELG1010
C                                                                       DELG1020
C     COLUMN INTERCHANGE IN MATRIX A                                    DELG1030
    9 LEND=LST+M-K                                                      DELG1040
      IF(J) 12,12,10                                                    DELG1050
   10 II=J*M                                                            DELG1060
      DO 11 L=LST,LEND                                                  DELG1070
      TB=A(L)                                                           DELG1080
      LL=L+II                                                           DELG1090
      A(L)=A(LL)                                                        DELG1100
   11 A(LL)=TB                                                          DELG1110
C                                                                       DELG1120
C     ROW INTERCHANGE AND PIVOT ROW REDUCTION IN MATRIX A               DELG1130
   12 DO 13 L=LST,MM,M                                                  DELG1140
      LL=L+I                                                            DELG1150
```

```
      TB=PIVI*A(LL)                                                     DELG1160
      A(LL)=A(L)                                                        DELG1170
   13 A(L)=TB                                                           DELG1180
C                                                                       DELG1190
C     SAVE COLUMN INTERCHANGE INFORMATION                               DELG1200
      A(LST)=J                                                          DELG1210
C                                                                       DELG1220
C     ELEMENT PRODUCTION AND NEXT PIVOT SEARCH                          DELG1230
      PIV=0.D0                                                          DELG1240
      LST=LST+1                                                         DELG1250
      J=0                                                               DELG1260
      DO 16 II=LST,LEND                                                 DELG1270
      PIVI=-A(II)                                                       DELG1280
      IST=II+M                                                          DELG1290
      J=J+1                                                             DELG1300
      DO 15 L=IST,MM,M                                                  DELG1310
      LL=L-J                                                            DELG1320
      A(L)=A(L)+PIVI*A(LL)                                              DELG1330
      TB=DABS(A(L))                                                     DELG1340
      IF(TB-PIV) 15,15,14                                               DELG1350
   14 PIV=TB                                                            DELG1360
      I=L                                                               DELG1370
   15 CONTINUE                                                          DELG1380
      LL=L+J                                                            DELG1400
   16 R(LL)=R(LL)+PIVI*R(L)                                             DELG1410
   17 LST=LST+M                                                         DELG1420
C     END OF ELIMINATION LOOP                                           DELG1430
C                                                                       DELG1440
C     BACK SUBSTITUTION AND BACK INTERCHANGE                            DELG1450
   18 IF(M-1) 23,22,19                                                  DELG1460
   19 IST=MM+M                                                          DELG1470
      LST=M+1                                                           DELG1480
      DO 21 I=2,M                                                       DELG1490
      II=LST-I                                                          DELG1500
      IST=IST-LST                                                       DELG1510
      L=IST-M                                                           DELG1520
      L=A(L)+.500                                                       DELG1530
      DO 21 J=II,MM,M                                                   DELG1540
      TB=R(J)                                                           DELG1550
      LL=J                                                              DELG1560
      DO 20 K=IST,MM,M                                                  DELG1570
      LL=LL+1                                                           DELG1580
   20 TB=TB-A(K)*R(LL)                                                  DELG1590
      K=J+L                                                             DELG1600
      R(J)=R(K)                                                         DELG1610
   21 R(K)=TB                                                           DELG1620
   22 RETURN                                                            DELG1630
C                                                                       DELG1640
C     ERROR RETURN                                                      DELG1650
   23 IER=-1                                                            DELG1660
      RETURN                                                            DELG1670
      END                                                               DELG1680
C                                                                       DARR  10
C     ..................................................................DARR  20
C                                                                       DARR  30
C        SUBROUTINE DARRAY                                              DARR  40
C                                                                       DARR  50
C        PURPOSE                                                        DARR  60
C           CONVERT DATA ARRAY FROM SINGLE TO DOUBLE DIMENSION OR VICE  DARR  70
C           VERSA.  THIS SUBROUTINE IS USED TO LINK THE USER PROGRAM    DARR  80
C           WHICH HAS DOUBLE DIMENSION ARRAYS AND THE SSP SUBROUTINES   DARR  90
C           WHICH OPERATE ON ARRAYS OF DATA IN A VECTOR FASHION.        DARR 100
C                                                                       DARR 110
C        USAGE                                                          DARR 120
C           CALL DARRAY (MODE,I,J,N,M,S,D)                              DARR 130
C                                                                       DARR 140
C        DESCRIPTION OF PARAMETERS                                      DARR 150
C           MODE - CODE INDICATING TYPE OF CONVERSION                   DARR 160
C                  1 - FROM SINGLE TO DOUBLE DIMENSION                  DARR 170
C                  2 - FROM DOUBLE TO SINGLE DIMENSION                  DARR 180
C           I    - NUMBER OF ROWS IN ACTUAL DATA MATRIX                 DARR 190
C           J    - NUMBER OF COLUMNS IN ACTUAL DATA MATRIX              DARR 200
C           N    - NUMBER OF ROWS SPECIFIED FOR THE MATRIX D IN         DARR 210
C                  DIMENSION STATEMENT                                  DARR 220
C           M    - NUMBER OF COLUMNS SPECIFIED FOR THE MATRIX D IN      DARR 230
C                  DIMENSION STATEMENT                                  DARR 240
C           S    - IF MODE=1, THIS VECTOR IS INPUT WHICH CONTIANS THE   DARR 250
C                  ELEMENTS OF A DATA MATRIX OF SIZE I BY J. COLUMN I+1 DARR 260
C                  OF DATA MATRIX FOLLOWS COLUMN I., ETC. IF MODE=2,    DARR 270
C                  THIS VECTOR IS OUTPUT REPRESENTING A DATA MATRIX OF  DARR 280
C                  SIZE I BY J CONTAINIG ITS COLUMNS CONSECUTIVELY.     DARR 290
C                  THE LENGTH OF S IS IJ, WHERE IJ=I*J.                 DARR 300
C           D    - IF MODE=1, THIS MATRIX OF SIZE N BY M IS OUTPUT,     DARR 310
C                  CONTAINING A DATA MATRIX OF SIZE I BY J IN THE FIRST DARR 320
C                  I ROWS AND J COLUMNS. IF MODE=2, THIS N BY M MATRIX  DARR 330
C                  IS INPUT CONTAINING A DATA MATRIX OF SIZE I BY J IN  DARR 340
C                  THE FIRST I ROWS AND J COLUMNS.                      DARR 350
C                                                                       DARR 360
C        REMARKS                                                        DARR 370
C           VECTOR S CAN BE IN THE SAME LOCATION AS MATRIX D. VECTOR S  DARR 380
C           IS REFERRED AS A MATRIX IN OTHER SSP ROUTINES, SINCE IT     DARR 390
C           CONTAINS A DATA MATRIX.                                     DARR 400
C           THIS SUBROUTINE CONVERTS ONLY GENERAL DATA MATRICES (STORAGEDARR 410
C           MODE OF 0).                                                 DARR 420
C                                                                       DARR 430
C        SUBROUTINES AND FUNCTION SUBROUTINES REQUIRED                  DARR 440
C           NONE                                                        DARR 450
C        METHOD                                                         DARR 470
C           REFER TO THE DISCUSSION ON VARIABLE DATA SIZE IN THE SECTIONDARR 480
C           DESCRIBING OVERALL RULES FOR USAGE IN THIS MANUAL.          DARR 490
C                                                                       DARR 500
C     ..................................................................DARR 510
C                                                                       DARR 520
      SUBROUTINE DARRAY (MODE,I,J,N,M,S,D)                              DARR 530
      IMPLICIT REAL*8(A-H,O-Z)                                          DARR 540
      DIMENSION S(1),D(1)                                               DARR 550
      NI=N-I                                                            DARR 560
C                                                                       DARR 570
C        TEST TYPE OF CONVERSION                                        DARR 580
C                                                                       DARR 590
      IF(MODE-1) 100, 100, 120                                          DARR 600
C                                                                       DARR 610
C        CONVERT FROM SINGLE TO DOUBLE DIMENSION                        DARR 620
C                                                                       DARR 630
  100 IJ=I*J+1                                                          DARR 640
      NM=N*J+1                                                          DARR 650
      DO 110 K=1,J                                                      DARR 660
      NM=NM-NI                                                          DARR 670
      DO 110 L=1,I                                                      DARR 680
      IJ=IJ-1                                                           DARR 690
```

```
      NM=NM-1                                                          DARR 700
  110 D(NM)=S(IJ)                                                      DARR 710
      GO TO 140                                                        DARR 720
C                                                                      DARR 730
C         CONVERT FROM DOUBLE TO SINGLE DIMENSION                      DARR 740
C                                                                      DARR 750
  120 IJ=0                                                             DARR 760
      NM=0                                                             DARR 770
      DO 130 K=1,J                                                     DARR 780
      DO 125 L=1,I                                                     DARR 790
      IJ=IJ+1                                                          DARR 800
      NM=NM+1                                                          DARR 810
  125 S(IJ)=D(NM)                                                      DARR 820
  130 NM=NM+NI                                                         DARR 830
C                                                                      DARR 840
  140 RETURN                                                           DARR 850
      END                                                              DARR 860
```

```
C                                                                       MAIN0010
C     ****************************************************************MAIN0020
C                                                                       MAIN0030
C        SAMPLE MAIN PROGRAM FOR VARIABLE FLUID PROPERTY PROBLEM.       MAIN0040
C                                                                       MAIN0050
C        PURPOSE                                                        MAIN0060
C           1) SOLVE THE DIFFERENTIAL EQUATIONS FOR VARIABLE FLUID      MAIN0070
C           PROPERTY PROBLEM IN LAMINAR FREE CONVECTION ON A            MAIN0080
C           VERTICAL FLAT PLATE, 2) CALCULATE HEAT TRANSFER COEFFICIENT.MAIN0090
C                                                                       MAIN0100
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                  MAIN0110
C           PROP                                                        MAIN0120
C           FF2                                                         MAIN0130
C           JACOB2                                                      MAIN0140
C           SYSSOL                                                      MAIN0150
C           BLOCK DATA                                                  MAIN0160
C           COEFF                                                       MAIN0170
C           HEAT                                                        MAIN0180
C                                                                       MAIN0190
C     ****************************************************************MAIN0200
C                                                                       MAIN0210
      IMPLICIT REAL*8(A-H,O-Z)                                          MAIN0220
      EXTERNAL FF2,JACOB2                                               MAIN0230
      COMMON /CCOEFF/ CA,CB,CC,CD,CE,CF,CG,CH,CI,CJ,CK,CL,CM,CN,CP,CR,  MAIN0240
     *   CS,CT,CU,CW,CAA,CBB,CCC,CDD,CEE,CFF,CGG,CHH                    MAIN0250
      COMMON/CPROP/DENS(50),CPH(50),VISC(50),COND(50),TEMP(50),DTEMP    MAIN0260
      COMMON/TERM/AA1(50),AA2(50),AA3(50),BB1(50),BB2(50)               MAIN0270
      COMMON /CHEAT/ QWX,HWX,RNUGR,GRX,CNUX,RATIO,Y,PRW,PRIN            MAIN0280
      COMMON /CTEMP/ TIN,TW,DT                                          MAIN0290
      COMMON /EQU/ M                                                    MAIN0300
      DIMENSION A1(10,10),B1(10,10),ALPHA(10),Y(2600),X(521),ABT(10)    MAIN0310
C                                                                       MAIN0320
C     READ PROBLEM PARAMETERS.                                          MAIN0330
C                                                                       MAIN0340
C                                                                       MAIN0350
C     THE FOLLOWING SUBROUTINE REQUIRES DATA INPUT,FOR SPECIFICATION    MAIN0360
C     OF INPUT DATA REFER TO THE LISTING OF SUBROUTINE PROP.            MAIN0370
C                                                                       MAIN0380
      CALL PROP                                                         MAIN0390
C                                                                       MAIN0400
C     TW IS THE WALL TEMPERATURE IN DEG. R.                             MAIN0410
C     TIN IS THE FREE STREAM TEMPERATURE IN DEG. R.                     MAIN0420
C     DT IS THE TEMPERATURE DIFFERENCE IN DEG. R.                       MAIN0430
      READ(5,1) TW,TIN                                                  MAIN0440
      DT=TW-TIN                                                         MAIN0450
C                                                                       MAIN0460
C     THE FOLLOWING INFORMATION IS REQUIRED FOR SUBROUTINE SYSSOL,      MAIN0470
C     FOR THE DEFINITION OF THE PARAMETERS REFER TO THE LISTING OF      MAIN0480
C     SUBROUTINE SYSSOL.                                                MAIN0490
C                                                                       MAIN0500
      N0=4                                                              MAIN0510
      TOL=1.D-6                                                         MAIN0520
      N=N0                                                              MAIN0530
      DELEPS=0.D0                                                       MAIN0540
      M=5                                                               MAIN0550
      A=0.D0                                                            MAIN0560
      B=10.0D00                                                         MAIN0570
      DO 2000 I=1,M                                                     MAIN0580
      ALPHA(I)=0.D0                                                     MAIN0590
      DO 2000 J=1,M                                                     MAIN0500
      A1(I,J)=0.D0                                                      MAIN0610
 2000 B1(I,J)=0.D0                                                      MAIN0620
      A1(1,1)=1.D0                                                      MAIN0630
      A1(2,2)=1.D0                                                      MAIN0640
      A1(4,4)=1.D0                                                      MAIN0650
      B1(3,2)=1.D0                                                      MAIN0660
      B1(5,4)=1.D0                                                      MAIN0670
      ALPHA(4)=1.D0                                                     MAIN0680
C                                                                       MAIN0690
      CALL COEFF                                                        MAIN0700
      CALL SYSSOL(M,N,A,B,ALPHA,A1,B1,TOL,DELEPS,X,Y,ABT,FF2,JACOB2,    MAIN0710
     *JERROR)                                                           MAIN0720
C                                                                       MAIN0730
      WRITE(6,200)                                                      MAIN0740
      WRITE(6,16) M,N,A,B,(ALPHA(I),I=1,M)                              MAIN0750
      WRITE(6,18)((A1(I,J),J=1,M),I=1,M)                                MAIN0760
      WRITE(6,24) ((B1(I,J),J=1,M),I=1,M)                               MAIN0770
      WRITE(6,19) TOL                                                   MAIN0780
      WRITE(6,13) (ABT(J),J=1,M)                                        MAIN0790
      WRITE(6,23) JERROR                                                MAIN0800
      WRITE(6,200)                                                      MAIN0810
      WRITE(6,21)                                                       MAIN0820
      DO 15 J=1,N,16                                                    MAIN0830
      KA=M*(J-1)+1                                                      MAIN0840
      KB=M*J                                                            MAIN0850
      WRITE(6,22) J,X(J),(Y(K),K=KA,KB)                                 MAIN0860
   15 CONTINUE                                                          MAIN0870
      WRITE(6,200)                                                      MAIN0880
      WRITE(6,41) TW,TIN,DT                                             MAIN0890
      CALL  HEAT                                                        MAIN0900
      WRITE(6,42) PRW,PRIN                                              MAIN0910
      WRITE(6,43) CNUX,GRX                                              MAIN0920
      WRITE(6,30)                                                       MAIN0930
      WRITE(6,31)                                                       MAIN0940
      WRITE(6,32)                                                       MAIN0950
      WRITE(6,33)                                                       MAIN0960
      WRITE(6,34) QWX,HWX,RNUGR                                         MAIN0970
C                                                                       MAIN0980
    1 FORMAT(2D12.5)                                                    MAIN0990
   13 FORMAT(1HO,' ERROR ESTIMATE FOR COMPONENTS '//5D12.3)             MAIN1000
   16 FORMAT(1HO,' NUMBER OF EQUATIONS ',I3//'  FINAL NUMBER OF MESH POIMAIN1010
     $NTS ',I5//'  LEFT BOUNDARY POINT = ',F7.2,5X,'RIGHT BOUNDARY POINTMAIN1020
     $ = ',F7.2//'  BOUNDARY CONDITIONS'//5(F7.2,2X))                  MAIN1030
   18 FORMAT(1HO,' MATRIX OF LEFT BOUNDARY CONDITONS '//5(F7.2,2X))     MAIN1040
   24 FORMAT(1HO,' MATRIX OF RIGHT BOUNDARY CONDITIONS '//5(F7.2,2X))   MAIN1050
   19 FORMAT(1HO,' TOLERANCE ',D12.2)                                   MAIN1060
   21 FORMAT(1HO//7X,'N',7X,'ETA',15X,'F',15X,'FP',15X,'FPP',15X,'T',15XMAIN1070
     $,'TP')                                                            MAIN1080
   22 FORMAT(1HO,3X,I4,3X,6(D12.5,5X))                                  MAIN1090
   23 FORMAT(1HO,' JERROR = ',I1)                                       MAIN1100
   30 FORMAT(1HO///14X,'Q*X**(1/4) = HEAT TRANSFER IN BTU/HR-FT**(7/4)')MAIN1110
   31 FORMAT(1HO,13X,'H*X**(1/4) = COEFFICIENT OF HEAT TRANSFER IN BTU/HMAIN1120
     $R-FT**(7/4)-R')                                                   MAIN1130
   32 FORMAT(1HO,13X,'NU/GR = RATIO OF NUSSELT NO. AND GRASHOF NO.')    MAIN1140
   33 FORMAT(1HO///16X,'Q*X**(1/4)',10X,'H*X**(1/4)',12X,'NU/GR')       MAIN1150
   34 FORMAT(1HO,13X,E15.8,5X,E15.8,5X,E15.8)                           MAIN1160
   41 FORMAT(1HO,' WALL TEMPERATURE = ',F7.2,'  R'//'  FREE STREAM TEMPRMAIN1170
     $ATURE = ',F7.2,'  R'//'  TEMPERATURE DIFFERENCE = ',F6.2,'  R'//) MAIN1180
   42 FORMAT(1HO,' WALL PRANDTL NUMBER = ',F6.2//'  FREE STREAM PRANDTL MAIN1190
```

```
     $NUMBER = ',F6.2//)                                                MAIN1200
    43 FORMAT(1HO,' NUSSELT NUMBER = ',D12.5//,'  GRASHOF NUMBER = ',D12.MAIN1210
     $5//)                                                             MAIN1220
   150 FORMAT(F15.5,F15.5,F15.5,D15.5,D15.5)                           MAIN1230
   200 FORMAT(1H1,' VARIABLE PROPERTY PROBLEM--REFRIGERANT 114')       MAIN1240
C                                                                      MAIN1250
      STOP                                                             MAIN1260
      END                                                              MAIN1270
C                                                                      FF200010
C     ..................................................................FF200020
C                                                                      FF200030
C        SUBROUTINE FF2(X,Y,N,FF)                                      FF200040
C                                                                      FF200050
C        PURPOSE                                                       FF200060
C           THIS SUBROUTINE EVALUATES THE FIRST ORDER DIFFERENTIAL     FF200070
C           EQUATIONS ON THE WHOLE GRID. FOR FURTHER INFORMATION SEE   FF200080
C           THE LISTING OF SUBROUTINE SYSSOL AND EQUATIONS (5.13) AND  FF200090
C           (5.14) OF THE THESIS.                                      FF200100
C                                                                      FF200110
C     ..................................................................FF200120
C                                                                      FF200130
      SUBROUTINE FF2(X,Y,N,FF)                                         FF200140
      IMPLICIT REAL*8(A-H,O-Z)                                         FF200150
      COMMON /CCOEFF/ CA,CB,CC,CD,CE,CF,CG,CH,CI,CJ,CK,CL,CM,CN,CP,CR, FF200160
     *   CS,CT,CU,CW,CAA,CBB,CCC,CDD,CEE,CFF,CGG,CHH                   FF200170
      DIMENSION X(1),Y(1),FF(1)                                        FF200180
C                                                                      FF200190
      DO 10 I=1,N                                                      FF200200
      K=(I-1)*5+1                                                      FF200210
      FF(K)=Y(K+1)                                                     FF200220
      FF(K+1)=Y(K+2)                                                   FF200230
      FF(K+2)=-(CA+CB*Y(K+3)+CC*Y(K+3)**2+CD*Y(K+3)**3)*Y(K+4)*Y(K+2)  FF200240
     $        -(CE+CF*Y(K+3)+CG*Y(K+3)**2+CH*Y(K+3)**3)*Y(K)*Y(K+2)     FF200250
     $        +(CI+CJ*Y(K+3)+CK*Y(K+3)**2+CL*Y(K+3)**3)*Y(K+1)**2       FF200260
     $        -(CM+CN*Y(K+3)+CP*Y(K+3)**2+CR*Y(K+3)**3)                 FF200270
     $        +(CS+CT*Y(K+3)+CU*Y(K+3)**2+CW*Y(K+3)**3)                 FF200280
      FF(K+3)=Y(K+4)                                                   FF200290
   10 FF(K+4)=-(CAA+CBB*Y(K+3)+CCC*Y(K+3)**2+CDD*Y(K+3)**3)*Y(K+4)**2  FF200300
     $        -(CEE+CFF*Y(K+3)+CGG*Y(K+3)**2+CHH*Y(K+3)**3)*Y(K)*Y(K+4) FF200310
C                                                                      FF200320
      RETURN                                                           FF200330
      END                                                              FF200340
C                                                                      JAC20010
C     ..................................................................JAC20020
C                                                                      JAC20030
C        SUBROUTINE JACOB2(X,YX,JAB)                                   JAC20040
C                                                                      JAC20050
C        PURPOSE                                                       JAC20060
C           THIS SUBROUTINE EVALUATES THE JACOBIAN OF THE DIFFERENTIAL JAC20070
C           EQUATIONS AT A GIVEN GRID POINT. FOR FURTHER INFORMATION   JAC20080
C           SEE THE LISTING OF SUBROUTINE SYSSOL AND APPENDIX C        JAC20090
C           OF THE THESIS.                                             JAC20100
C                                                                      JAC20110
C     ..................................................................JAC20120
C                                                                      JAC20130
      SUBROUTINE JACOB2(X,YX,JAB)                                      JAC20140
      IMPLICIT REAL*8(A-H,O-Z)                                         JAC20150
      COMMON /CCOEFF/ CA,CB,CC,CD,CE,CF,CG,CH,CI,CJ,CK,CL,CM,CN,CP,CR, JAC20160
     *   CS,CT,CU,CW,CAA,CBB,CCC,CDD,CEE,CFF,CGG,CHH                   JAC20170
      COMMON /EQU/ M                                                   JAC20180
      DIMENSION YX(1)                                                  JAC20190
      DOUBLE PRECISION JAB(10,10)                                      JAC20200
C                                                                      JAC20210
      DO 10 I=1,M                                                      JAC20220
      DO 10 J=1,M                                                      JAC20230
   10 JAB(I,J)=0.D0                                                    JAC20240
      JAB(1,2)=1.D0                                                    JAC20250
      JAB(2,3)=1.D0                                                    JAC20260
      JAB(3,1)=-CE*YX(3)-CF*YX(3)*YX(4)-CG*YX(3)*YX(4)**2              JAC20270
     $         -CH*YX(3)*YX(4)**3                                      JAC20280
      JAB(3,2)=2.D0*CI*YX(2)+2.D0*CJ*YX(2)*YX(4)+2.D0*CK*YX(2)*YX(4)**2 JAC20290
     $         +2.D0*CL*YX(2)*YX(4)**3                                 JAC20300
      JAB(3,3)=-CA*YX(5)-CB*YX(4)*YX(5)-CC*YX(5)*YX(4)**2              JAC20310
     $         -CD*YX(5)*YX(4)**3-CE*YX(1)-CF*YX(1)*YX(4)              JAC20320
     $         -CG*YX(1)*YX(4)**2-CH*YX(1)*YX(4)**3                    JAC20330
      JAB(3,4)=-CB*YX(3)*YX(5)-2.D0*CC*YX(3)*YX(4)*YX(5)               JAC20340
     $         -3.D0*CD*YX(3)*YX(5)*YX(4)**2-CF*YX(1)*YX(3)            JAC20350
     $         -2.D0*CG*YX(1)*YX(3)*YX(4)-3.D0*CH*YX(1)*YX(3)*YX(4)**2 JAC20360
     $         +CJ*YX(2)**2+2.D0*CK*YX(4)*YX(2)**2                     JAC20370
     $         +3.D0*CL*YX(2)**2*YX(4)**2-CN-2.D0*CP*YX(4)             JAC20380
     $         -3.D0*CR*YX(4)**2+CT+2.D0*CU*YX(4)+3.D0*CW*YX(4)**2     JAC20390
      JAB(3,5)=-CA*YX(3)-CB*YX(3)*YX(4)-CC*YX(3)*YX(4)**2              JAC20400
     $         -CD*YX(3)*YX(4)**3                                      JAC20410
      JAB(4,5)=1.D0                                                    JAC20420
      JAB(5,1)=-CEE*YX(5)-CFF*YX(5)*YX(4)-CGG*YX(5)*YX(4)**2           JAC20430
     $         -CHH*YX(5)*YX(4)**3                                     JAC20440
      JAB(5,4)=-CBB*YX(5)**2-2.D0*CCC*YX(4)*YX(5)**2                   JAC20450
     $         -3.D0*CDD*YX(4)**2*YX(5)**2-CFF*YX(1)*YX(5)             JAC20460
     $         -2.D0*CGG*YX(1)*YX(4)*YX(5)-3.D0*CHH*YX(1)*YX(4)**2*YX(5)JAC20470
      JAB(5,5)=-2.D0*CAA*YX(5)-2.D0*CBB*YX(4)*YX(5)                    JAC20480
     $         -2.D0*CCC*YX(4)**2*YX(5)-2.D0*CDD*YX(5)*YX(4)**3        JAC20490
     $         -CEE*YX(1)-CFF*YX(1)*YX(4)-CGG*YX(1)*YX(4)**2           JAC20500
     $         -CHH*YX(1)*YX(4)**3                                     JAC20510
C                                                                      JAC20520
      RETURN                                                           JAC20530
      END                                                              JAC20540
C                                                                      BLK00010
C     ..................................................................BLK00020
C                                                                      BLK00030
C        BLOCK DATA                                                    BLK00040
C                                                                      BLK00050
C        PURPOSE                                                       BLK00060
C           THE COEFFICIENTS FOR THE CURVE FITTED TERMS AA1 , AA2 ,    BLK00070
C           AA3 , BB1 , BB2 ARE STORED IN THIS BLOCK DATA.FOR MORE     BLK00080
C           INFORMATION SEE EQUATIONS (5.1) THROUGH (5.5) OF THE THESIS.BLK00090
C           THESE COEFFICIENTS SHOULD BE SUPPLIED BY THE USER.         BLK00100
C                                                                      BLK00110
C     ..................................................................BLK00120
C                                                                      BLK00130
      BLOCK DATA                                                       BLK00140
      IMPLICIT REAL*8(A-H,O-Z)                                         BLK00150
      COMMON /CONST/ A(25),B(25),C(25),D(25)                           BLK00160
C                                                                      BLK00170
C     FLUID : REFRIGERANT-114                                          BLK00180
C     PRESSURE : 540 PSIA                                              BLK00190
C     TEMP. RANGE : FROM 774.69 TO 779.69 DEG. R                       BLK00200
C                                                                      BLK00210
      DATA A                                                           BLK00220
     1/-.10139867066526D05,-.28249157766885AD05,-.65434837330367D03,   BLK00230
     2-.74599396750486670D04,.22110136418748850D06/                    BLK00240
```

```
C                                                                       BLK00250
      DATA B                                                            BLK00260
     1/.3891121782049853D02,-.1078387778419250D03,.2482935111410940D01, BLK00270
     2.2862293720560248D02,-.8461731580827346D03/                       BLK00280
C                                                                       BLK00290
      DATA C                                                            BLK00300
     1/-.4977596213409AD-01,-.1372623554228000,-.3143400138466530-02,   BLK00310
     2-.3660972533462388D-01,.1079552444084574D01/                      BLK00320
C                                                                       BLK00330
      DATA D                                                            BLK00340
     1/.2122588086603510-04,.5825886723323334D-04,.1327920211839620-05, BLK00350
     2.1560921327554040-04,-.4591322997601475D-03/                      BLK00360
C                                                                       BLK00370
      END                                                               BLK00380
C                                                                       COEF0010
C     ..................................................................COEF0020
C                                                                       COEF0030
C        SUBROUTINE COEFF                                               COEF0040
C                                                                       COEF0050
C        PURPOSE                                                        COEF0060
C           THIS SUBROUTINE CALCULATES THE COEFFICIENTS OF THE REDUCED  COEF0070
C           DIFFERENTIAL EQUATIONS. COEFFICIENT FOR TERMS A1 , 3A2 ,    COEF0080
C           2A2 , A3 , B1 , 3B2 .SEE EQUATIONS (5.7) THROUGH (5.12)     COEF0090
C           OF THE THESIS FOR MORE INFORMATION.                         COEF0100
C                                                                       COEF0110
C     ..................................................................COEF0120
C                                                                       COEF0130
      SUBROUTINE COEFF                                                  COEF0140
      IMPLICIT REAL*8(A-H,O-Z)                                          COEF0150
      COMMON /CCOEFF/ CA,CB,CC,CD,CE,CF,CG,CH,CI,CJ,CK,CL,CM,CN,CP,CR,  COEF0160
     *    CS,CT,CU,CW,CAA,CBB,CCC,CDD,CEE,CFF,CGG,CHH                   COEF0170
      COMMON/TERM/AA1(50),AA2(50),AA3(50),BB1(50),BB2(50)               COEF0180
      COMMON/CPROP/DENS(50),CPH(50),VISC(50),COND(50),TEMP(50),DTEMP    COEF0190
      COMMON /CONST/ A(25),B(25),C(25),D(25)                            COEF0200
      COMMON /CTEMP/ TIN,TW,DT                                          COEF0210
C                                                                       COEF0220
C     THE FOLLOWING IF STATEMENTS ARE FOR THE CURVE FITTED              COEF0230
C     TERMS AA1,AA2,AA3,BB1,BB2, AT DIFFERENT TEMP. INTERVALS           COEF0240
C     AND ONE SPECIFIC PRESSURE AND SHOULD BE SUPPLIED BY THE USER.     COEF0250
C     THE IF STATEMENTS SHOULD BE ARRANGED IN ORDER OF INCREASING TEMP. COEF0260
C     FLUID : REFRIGERANT-114                                           COEF0270
C     PRESSURE : 540 PSIA                                               COEF0280
C     TEMP. RANGE : FROM 774.69 TO 779.69 DEG. R                        COEF0290
C                                                                       COEF0300
      IF(TIN .GE. 774.69 .AND. TIN .LE. 779.69) GO TO 10                COEF0310
C     IF(TIN .GT. 779.69 .AND. TIN .LE. 809.69) GO TO 20                COEF0320
C     "       "         "          "          "         "              COEF0330
C     "       "         "          "          "         "              COEF0340
C                                                                       COEF0350
C     IF THE NEXT STATEMENT IS READ THE TEMP. IS OUT OF RANGE.          COEF0360
      GO TO 70                                                          COEF0370
C                                                                       COEF0380
   10 I=1                                                               COEF0390
      GO TO 60                                                          COEF0400
   20 I=2                                                               COEF0410
      GO TO 60                                                          COEF0420
   30 I=3                                                               COEF0430
      GO TO 60                                                          COEF0440
   40 I=4                                                               COEF0450
      GO TO 60                                                          COEF0460
   50 I=5                                                               COEF0470
   60 CONTINUE                                                          COEF0480
C                                                                       COEF0490
      J=(I-1)*5                                                         COEF0500
      A1=A(J+1)                                                         COEF0510
      A2=A(J+2)                                                         COEF0520
      A3=A(J+3)                                                         COEF0530
      A4=A(J+4)                                                         COEF0540
      A5=A(J+5)                                                         COEF0550
C                                                                       COEF0560
      B1=B(J+1)                                                         COEF0570
      B2=B(J+2)                                                         COEF0580
      B3=B(J+3)                                                         COEF0590
      B4=B(J+4)                                                         COEF0600
      B5=B(J+5)                                                         COEF0610
C                                                                       COEF0620
      C1=C(J+1)                                                         COEF0630
      C2=C(J+2)                                                         COEF0640
      C3=C(J+3)                                                         COEF0650
      C4=C(J+4)                                                         COEF0660
      C5=C(J+5)                                                         COEF0670
C                                                                       COEF0680
      D1=D(J+1)                                                         COEF0690
      D2=D(J+2)                                                         COEF0700
      D3=D(J+3)                                                         COEF0710
      D4=D(J+4)                                                         COEF0720
      D5=D(J+5)                                                         COEF0730
C                                                                       COEF0740
      TIN1=TIN                                                          COEF0750
      TIN2=TIN1*TIN                                                     COEF0760
      TIN3=TIN2*TIN                                                     COEF0770
C                                                                       COEF0780
      DT1=DT                                                            COEF0790
      DT2=DT1*DT                                                        COEF0800
      DT3=DT2*DT                                                        COEF0810
      DT4=DT3*DT                                                        COEF0820
C                                                                       COEF0830
      K=IDINT(((TW-TEMP(1))/DTEMP)+0.5)+1                               COEF0840
      JJ=IDINT(((TIN-TEMP(1))/DTEMP)+0.5)+1                             COEF0850
C                                                                       COEF0860
C     CALCULATE COEFFICIENTS FOR TERM A1                                COEF0870
C                                                                       COEF0880
      CA=(A1+B1*TIN1+C1*TIN2+D1*TIN3)*DT1                               COEF0890
      CB=(B1+2.D0*C1*TIN1+3.D0*D1*TIN2)*DT2                             COEF0900
      CC=(C1+3.D0*D1*TIN1)*DT3                                          COEF0910
      CD=D1*DT4                                                         COEF0920
C                                                                       COEF0930
C     CALCULATE COEFFICIENTS FOR TERM 3A2                               COEF0940
C                                                                       COEF0950
      DENVIS=DENS(K)*VISC(K)                                            COEF0960
      CE=3.D0*(A2+B2*TIN1+C2*TIN2+D2*TIN3)*DENVIS                       COEF0970
      CF=3.D0*(B2+2.D0*C2*TIN1+3.D0*D2*TIN2)*DENVIS*DT1                 COEF0980
      CG=3.D0*(C2+3.D0*D2*TIN1)*DENVIS*DT2                              COEF0990
      CH=3.D0*D2*DENVIS*DT3                                             COEF1000
C                                                                       COEF1010
C     CALCULATE COEFFICIENTS FOR TERM 2A2                               COEF1020
C                                                                       COEF1030
      CI=2.D0*CE/3.D0                                                   COEF1040
      CJ=2.D0*CF/3.D0                                                   COEF1050
      CK=2.D0*CG/3.D0                                                   COEF1060
```

```
      CL=2.D0*CH/3.D0                                                   COEF1070
C                                                                       COEF1080
C     CALCULATE COEFFICIENTS FOR TERM A3                                COEF1090
C                                                                       COEF1100
      RATIO1=DENS(JJ)*VISC(K)*DENS(K)**2/(DENS(JJ)-DENS(K))            COEF1110
      CM=(A3+B3*TIN1+C3*TIN2+D3*TIN3)*RATIO1                            COEF1120
      CN=(B3+2.D0*C3*TIN1+3.D0*D3*TIN2)*RATIO1*DT1                      COEF1130
      CP=(C3+3.D0*D3*TIN1)*RATIO1*DT2                                   COEF1140
      CR=D3*RATIO1*DT3                                                  COEF1150
C                                                                       COEF1160
      RATIO2=RATIO1/DENS(JJ)                                            COEF1170
      CS=(A2+B2*TIN1+C2*TIN2+D2*TIN3)*RATIO2                            COEF1180
      CT=(B2+2.D0*C2*TIN1+3.D0*D2*TIN2)*RATIO2*DT1                      COEF1190
      CU=(C2+3.D0*D2*TIN1)*RATIO2*DT2                                   COEF1200
      CW=D2*RATIO2*DT3                                                  COEF1210
C                                                                       COEF1220
C     CALCULATE COEFFICIENTS FOR TERM B1                                COEF1230
C                                                                       COEF1240
      CAA=(A4+B4*TIN1+C4*TIN2+D4*TIN3)*DT                               COEF1250
      CBB=(B4+2.D0*C4*TIN1+3.DC*D4*TIN2)*DT2                            COEF1260
      CCC=(C4+3.D0*D4*TIN1)*DT3                                         COEF1270
      CDD=D4*DT4                                                        COEF1280
C                                                                       COEF1290
C     CALCULATE COEFFICIENTS FOR TERM 3B2                               COEF1300
C                                                                       COEF1310
      CEE=3.D0*(A5+B5*TIN1+C5*TIN2+D5*TIN3)*DENVIS                      COEF1320
      CFF=3.D0*(B5+2.D0*C5*TIN1+3.D0*D5*TIN2)*DENVIS*DT1                COEF1330
      CGG=3.D0*(C5+3.D0*D5*TIN1)*DENVIS*DT2                             COEF1340
      CHH=3.D0*D5*DENVIS*DT3                                            COEF1350
C                                                                       COEF1360
      GO TO 80                                                          COEF1370
   70 WRITE (6,1) TIN                                                   COEF1380
    1 FORMAT(1H0,5X,'TIN = ',F10.5,5X,'TEMPERATURE OUT OF RANGE')       COEF1390
   80 CONTINUE                                                          COEF1400
C                                                                       COEF1410
      RETURN                                                            COEF1420
      END                                                               COEF1430
C                                                                       HEAT0010
C     ******************************************************************HEAT0020
C                                                                       HEAT0030
C        SUBROUTINE HEAT                                                HEAT0040
C                                                                       HEAT0050
C        PURPOSE                                                        HEAT0060
C           CALCULATE HEAT TRANSFER COEFFICIENT.                        HEAT0070
C        REMARK                                                         HEAT0080
C           FOR REFRIGERANT-114 STATEMENTS 10, 20, AND 30 SHOULD BE     HEAT0090
C           REPLACED BY THE FOLLOWING STATEMENTS.                       HEAT0100
C           CONDW=COND(J)*241.9D00                                      HEAT0110
C           PRIN=(VISC(JJ)*CPH(JJ)*(1.0D-02))/COND(JJ)                  HEAT0120
C           PRW=(VISC(J)*CPH(J)*(1.0D-02))/COND(J)                      HEAT0130
C                                                                       HEAT0140
C     ******************************************************************HEAT0150
C                                                                       HEAT0160
      SUBROUTINE HEAT                                                   HEAT0170
      IMPLICIT REAL*8(A-H,O-Z)                                          HEAT0180
      COMMON /CHEAT/ QWX,HWX,RNUGR,GRX,CNUX,RATIO,Y,PRW,PRIN            HEAT0190
      COMMON /CTEMP/ TIN,TW,DT                                          HEAT0200
      COMMON/TERM/AA1(50),AA2(50),AA3(50),BB1(50),BB2(50)               HEAT0210
      COMMON/CPROP/DENS(50),CPH(50),VISC(50),COND(50),TEMP(50),DTEMP    HEAT0220
      COMMON /EQU/ M                                                    HEAT0230
      DIMENSION Y(2500)                                                 HEAT0240
C                                                                       HEAT0250
C     CALCULATE CONSTANT C IN SIMILARITY VARIABLE ETA                   HEAT0260
C                                                                       HEAT0270
      J=IDINT(((TW-TEMP(1))/DTEMP)+0.5)+1                               HEAT0280
      JJ=IDINT(((TIN-TEMP(1))/DTEMP)+0.5)+1                             HEAT0290
      DENR=(DENS(JJ)-DENS(J))/DENS(J)                                   HEAT0300
      VISKM=VISC(J)/(1490.000)                                          HEAT0310
      VISKM=VISKM/DENS(J)                                               HEAT0320
      VISKM2=VISKM*VISKM                                                HEAT0330
      C1=(32.2D0*DENR)/(4.D0*VISKM2)                                    HEAT0340
      C=C1**0.25                                                        HEAT0350
C                                                                       HEAT0360
C     CALCULATE RATE OF HEAT TRANSFER                                   HEAT0370
C                                                                       HEAT0380
      TPW=Y(M)                                                          HEAT0390
   10 CONDW=COND(J)                                                     HEAT0400
      QWX=-CONDW*(TW-TIN)*C*TPW                                         HEAT0410
C                                                                       HEAT0420
C     CALCULATE HEAT TRANSFER COEFFICIENT                               HEAT0430
C                                                                       HEAT0440
      HWX=QWX/(TW-TIN)                                                  HEAT0450
C                                                                       HEAT0460
C     CALCULATE GRASHOF NUMBER                                          HEAT0470
C                                                                       HEAT0480
      GRX=(32.D0*DENR)/VISKM2                                           HEAT0490
C                                                                       HEAT0500
C     CALCULATE NUSSELT NUMBER                                          HEAT0510
C                                                                       HEAT0520
      CNUX=HWX/CONDW                                                    HEAT0530
C                                                                       HEAT0540
C     CALCULATE RATIO OF NUSSELT NO. AND GRASHOF NO.                    HEAT0550
C                                                                       HEAT0560
      RNUGR=-TPW                                                        HEAT0570
      RATIO=CNUX/(GRX/4.D0)**0.25                                       HEAT0580
C                                                                       HEAT0590
C     CALCULATE PRANDTL NUMBER                                          HEAT0600
C                                                                       HEAT0610
   20 PRIN=(VISC(JJ)*CPH(JJ)*(2.42000))/COND(JJ)                        HEAT0620
   30 PRW=(VISC(J)*CPH(J)*(2.42D00))/COND(J)                            HEAT0630
C                                                                       HEAT0640
      RETURN                                                            HEAT0650
      END                                                               HEAT0660
```

VARIABLE PROPERTY PROBLEM--REFRIGERANT 114

NUMBER OF EQUATIONS 5

FINAL NUMBER OF MESH POINTS 97

LEFT BOUNDARY POINT = 0.0 RIGHT BOUNDARY POINT = 10.00

BOUNDARY CONDITIONS

0.0	0.0	0.0	1.00	0.0

MATRIX OF LEFT BOUNDARY CONDITIONS

1.00	0.0	0.0	0.0	0.0
0.0	1.00	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	1.00	0.0
0.0	0.0	0.0	0.0	0.0

MATRIX OF RIGHT BOUNDARY CONDITIONS

0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	1.00	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	1.00	0.0

TOLERANCE 0.100-05

ERROR ESTIMATE FOR COMPONENTS

0.7560-06	0.1270-06	0.4330-07	0.3010-07	0.2750-07

JERROR = 0

VARIABLE PROPERTY PROBLEM--REFRIGERANT 114

N	ETA	F	FP	FPP	T	TP
1	0.0	0.0	0.13235D-22	0.546940+00	0.100000+01	-0.854410+00
5	0.41667D+00	0.35850D-01	0.14718D+00	0.188390+00	0.69502D+00	-0.785360+00
9	0.83333D+00	0.13682D+00	0.18002D+00	-0.594680-02	0.36368D+00	-0.593530+00
13	0.12500D+01	0.17351D+00	0.15910D+00	-0.792610-01	0.16823D+00	-0.346830+00
17	0.16667D+01	0.23736D+00	0.12277D+00	-0.684990-01	0.65867D-01	-0.159780+00
21	0.20833D+01	0.28117D+00	0.88553D-01	-0.739440-01	0.22547D-01	-0.618110-01
25	0.25000D+01	0.31222D+00	0.61680D-01	-0.551720-01	0.69828D-02	-0.203120-01
29	0.29167D+01	0.33360D+00	0.42180D-01	-0.390840-01	0.20137D-02	-0.614410-02
33	0.33333D+01	0.34816D+00	0.28544D-01	-0.270080-01	0.55263D-03	-0.174030-02
37	0.37500D+01	0.35798D+00	0.19185D-01	-0.184350-01	0.14662D-03	-0.471450-03
41	0.41667D+01	0.36456D+00	0.12818D-01	-0.125010-01	0.38023D-04	-0.123960-03
45	0.45833D+01	0.36895D+00	0.85086D-02	-0.844110-02	0.97105D-05	-0.319490-04
49	0.50000D+01	0.37185D+00	0.56037D-02	-0.568350-02	0.24549D-05	-0.812430-05
53	0.54167D+01	0.37375D+00	0.36520D-02	-0.380900-02	0.61649D-06	-0.204810-05
57	0.58333D+01	0.37496D+00	0.23459D-02	-0.254250-02	0.15415D-06	-0.513400-06
61	0.62500D+01	0.37576D+00	0.14765D-02	-0.168630-02	0.38440D-07	-0.128220-06
65	0.66667D+01	0.37626D+00	0.90228D-03	-0.115770-02	0.95690D-08	-0.319490-07
69	0.70833D+01	0.37654D+00	0.52748D-03	-0.716860-03	0.23795D-08	-0.794960-08
73	0.75000D+01	0.37671D+00	0.28737D-03	-0.452920-03	0.59131D-09	-0.197640-08
77	0.79167D+01	0.37680D+00	0.13821D-03	-0.274680-03	0.14682D-09	-0.491130-09
81	0.83333D+01	0.37684D+00	0.50467D-04	-0.154330-03	0.36369D-10	-0.122020-09
85	0.87500D+01	0.37684D+00	0.41598D-05	-0.730600-04	0.89305D-11	-0.303130-10

VARIABLE PROPERTY PROBLEM--REFRIGERANT 114

WALL TEMPERATURE = 779.69 R

FREE STREAM TEMPRATURE = 774.69 R

TEMPERATURE DIFFERENCE = 5.00 R

WALL PRANDTL NUMBER = 2.55

FREE STREAM PRANDTL NUMBER = 3.55

NUSSELT NUMBER = 0.97578D+03

GRASHOF NUMBER = 0.676230+13

Q*X**(1/4) = HEAT TRANSFER IN BTU/HR-FT**(7/4)

H*X**(1/4) = COEFFICIENT OF HEAT TRANSFER IN BTU/HR-FT**(7/4)-R

NU/GR = RATIO OF NUSSELT NO. AND GRASHOF NO.

Q*X**(1/4)	H*X**(1/4)	NU/GR
0.64244913D+02	0.12848983D+02	0.85440824D+00

```
C                                                                       MAIN0010
C     ****************************************************************MAIN0020
C                                                                       MAIN0030
C        SAMPLE MAIN PROGRAM FOR CONSTANT PROPERTY PROBLEM.             MAIN0040
C                                                                       MAIN0050
C        PURPOSE                                                        MAIN0060
C           1) SOLVE THE DIFFERENTIAL EQUATIONS FOR COSTANT FLUID       MAIN0070
C           PROPERTY PROBLEM IN LAMINAR FREE CONVECTION ON A            MAIN0080
C           VERTICAL FLAT PLATE, 2) CALCULATE HEAT TRANSFER COEFFICIENT.MAIN0090
C                                                                       MAIN0100
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                  MAIN0110
C           PROP                                                        MAIN0120
C           FF1                                                         MAIN0130
C           JACOB1                                                      MAIN0140
C           SYSSOL                                                      MAIN0150
C           HEAT                                                        MAIN0160
C                                                                       MAIN0170
C     ****************************************************************MAIN0180
C                                                                       MAIN0190
      IMPLICIT REAL*8(A-H,O-Z)                                          MAIN0200
      EXTERNAL FF1,JACOB1                                               MAIN0210
      COMMON/CPROP/DENS(50),CPH(50),VISC(50),COND(50),TEMP(50),DTEMP    MAIN0220
      COMMON/TERM/AA1(50),AA2(50),AA3(50),BB1(50),BB2(50)               MAIN0230
      COMMON /CHEAT/ QWX,HWX,RNUGR,GRX,CNUX,RATIO,Y,PRW,PRIN            MAIN0240
      COMMON /CTEMP/ TIN,TW,DT,TX                                       MAIN0250
      COMMON /EQU/ M                                                    MAIN0260
      COMMON /CONS/ PR                                                  MAIN0270
      DIMENSION A1(10,10),B1(10,10),ALPHA(10),Y(2600),X(521),ABT(10)    MAIN0280
C                                                                       MAIN0290
C     READ PROBLEM PARAMETERS.                                          MAIN0300
C                                                                       MAIN0310
C                                                                       MAIN0320
C     THE FOLLOWING SUBROUTINE REQUIRES DATA INPUT,FOR SPECIFICATION    MAIN0330
C     OF INPUT DATA REFER TO THE LISTING OF SUBROUTINE PROP.            MAIN0340
C                                                                       MAIN0350
      CALL PROP                                                         MAIN0360
C                                                                       MAIN0370
C     TW IS THE WALL TEMPERATURE IN DEG. R.                             MAIN0380
C     TIN IS THE FREE STREAM TEMPERATURE IN DEG. R.                     MAIN0390
C     TX IS THE REFERENCE TEMPERATURE IN DEG. R.                        MAIN0400
C     DT IS THE TEMPERATURE DIFFERENCE IN DEG. R.                       MAIN0410
      READ(5,1) TW,TIN,TX                                               MAIN0420
      DT=TW-TIN                                                         MAIN0430
C                                                                       MAIN0440
C     CALCULATE PRANDTL NUMBER AT THE SPECIFIED REFERENCE TEMPERATURE.  MAIN0450
      L=IDINT(((TX-TEMP(1))/DTEMP)+0.5)+1                               MAIN0460
      PR=(VISC(L)*CPH(L)*(1.0D-02))/COND(L)                             MAIN0470
C                                                                       MAIN0480
C     THE FOLLOWING INFORMATION IS REQUIRED FOR SUBROUTINE SYSSOL,      MAIN0490
C     FOR THE DEFINITION OF THE PARAMETERS PEFER TO THE LISTING OF      MAIN0500
C     SUBROUTINE SYSSOL.                                                MAIN0510
C                                                                       MAIN0520
      N0=4                                                              MAIN0530
      TOL=1.D-6                                                         MAIN0540
      N=N0                                                              MAIN0550
      DELEPS=0.D0                                                       MAIN0560
      M=5                                                               MAIN0570
      A=0.D0                                                            MAIN0580
      B=10.D0                                                           MAIN0590
      DO 1000 I=1,M                                                     MAIN0600
      ALPHA(I)=0.D0                                                     MAIN0610
      DO 1000 J=1,M                                                     MAIN0620
      A1(I,J)=0.D0                                                      MAIN0630
 1000 B1(I,J)=0.D0                                                      MAIN0640
      A1(1,1)=1.D0                                                      MAIN0650
      A1(2,2)=1.D0                                                      MAIN0660
      A1(4,4)=1.D0                                                      MAIN0670
      B1(3,2)=1.D0                                                      MAIN0680
      B1(5,4)=1.D0                                                      MAIN0690
      ALPHA(4)=1.D0                                                     MAIN0700
C                                                                       MAIN0710
      CALL SYSSOL(M,N,A,B,ALPHA,A1,B1,TOL,DELEPS,X,Y,ABT,FF1,JACOB1,    MAIN0720
     *JERROR)                                                           MAIN0730
C                                                                       MAIN0740
      WRITE(6,100) PR                                                   MAIN0750
      WRITE(6,16) M,N,A,B,(ALPHA(I),I=1,M)                              MAIN0760
      WRITE(6,18)((A1(I,J),J=1,M),I=1,M)                                MAIN0770
      WRITE(6,24) ((B1(I,J),J=1,M),I=1,M)                               MAIN0780
      WRITE(6,19) TOL                                                   MAIN0790
      WRITE(6,13) (ABT(J),J=1,M)                                        MAIN0800
      WRITE(6,23) JERROR                                                MAIN0810
      WRITE(6,100) PR                                                   MAIN0820
      WRITE(6,21)                                                       MAIN0830
      DO 15 J=1,M,24                                                    MAIN0840
      KA=M*(J-1)+1                                                      MAIN0850
      KB=M*J                                                            MAIN0860
      WRITE(6,22) J,X(J),(Y(K),K=KA,KB)                                 MAIN0870
   15 CONTINUE                                                          MAIN0880
      WRITE(6,100) PR                                                   MAIN0890
      WRITE(6,41) TW,TIN,DT,TX                                          MAIN0900
      CALL  HEAT                                                        MAIN0910
      WRITE(6,42) PRW,PRIN                                              MAIN0920
      WRITE(6,43) CNUX,GRX                                              MAIN0930
      WRITE(6,30)                                                       MAIN0940
      WRITE(6,31)                                                       MAIN0950
      WRITE(6,32)                                                       MAIN0960
      WRITE(6,33)                                                       MAIN0970
      WRITE(6,34) QWX,HWX,RNUGR                                         MAIN0980
C                                                                       MAIN0990
    1 FORMAT(3D12.5)                                                    MAIN1000
   13 FORMAT(1HO,' ERROR ESTIMATE FOR COMPONENTS '//5D12.3)             MAIN1010
   16 FORMAT(1HO,' NUMBER OF EQUATIONS ',I3//'  FINAL NUMBER OF MESH POIMAIN1020
     $NTS ',I5//'  LEFT BOUNDARY POINT = ',F7.2,5X,'RIGHT BOUNDARY POINTMAIN1030
     $ = ',F7.2//'  BOUNDARY CONDITIONS'//5(F7.2,2X))                  MAIN1040
   18 FORMAT(1HO,' MATRIX OF LEFT BOUNDARY CONDITIONS '//5(F7.2,2X))    MAIN1050
   24 FORMAT(1HO,' MATRIX OF RIGHT BOUNDARY CONDITIONS '//5(F7.2,2X))   MAIN1060
   19 FORMAT(1HO,' TOLERANCE ',D12.2)                                   MAIN1070
   21 FORMAT(1HO//7X,'N',7X,'ETA',15X,'F',15X,'FP',15X,'FPP',15X,'T',15XMAIN1080
     $,'TP')                                                            MAIN1090
   22 FORMAT(1HO,3X,I4,3X,6(D12.5,5X))                                  MAIN1100
   23 FORMAT(1HO,' JERROR = ',I1)                                       MAIN1110
   30 FORMAT(1HO///14X,'Q*X**(1/4) = HEAT TRANSFER IN BTU/HR-FT**(7/4)')MAIN1120
   31 FORMAT(1HO,13X,'H*X**(1/4) = COEFFICIENT OF HEAT TRANSFER IN BTU/HMAIN1130
     $R-FT**(7/4)-R')                                                   MAIN1140
   32 FORMAT(1HO,13X,'NU/GR = RATIO OF NUSSELT NO. AND GRASHOF NO.')    MAIN1150
   33 FORMAT(1HO///16X,'G*X**(1/4)',10X,'H*X**(1/4)',12X,'NU/GR')       MAIN1160
   34 FORMAT(1HO,13X,E15.8,5X,E15.8,5X,E15.8)                           MAIN1170
   41 FORMAT(1HO,' WALL TEMPERATURE = ',F7.2,'  R'///'  FREE STREAM TEMPRMAIN1180
     $ATURE = ',F7.2,'  R'///'  TEMPERATURE DIFFERENCE = ',F6.2,'  R'//' MAIN1190
```

```
     $ REFERENCE TEMPERATURE = ',F6.2,'  R'//)                          MAIN1200
   42 FORMAT(1H0,' WALL PRANDTL NUMBER = ',F6.2//' FREE STREAM PRANDTL  MAIN1210
     $NUMBER = ',F6.2//)                                                MAIN1220
   43 FORMAT(1H0,' NUSSELT NUMBER = ',D12.5//,' GRASHOF NUMBER = ',D12. MAIN1230
     $5//)                                                              MAIN1240
  100 FORMAT(1H1,' REFRIGERANT 114 -- CONSTANT PROPERTY PROBLEM WITH PR  MAIN1250
     $= ',F6.2)                                                         MAIN1260
C                                                                       MAIN1270
      STOP                                                              MAIN1280
      END                                                               MAIN1290
C                                                                       FF100010
C     ******************************************************************FF100020
C                                                                       FF100030
C        SUBROUTINE FF1(X,Y,N,FF)                                       FF100040
C                                                                       FF100050
C        PURPOSE                                                        FF100060
C           THIS SUBROUTINE EVALUATES THE FIRST ORDER DIFFERENTIAL      FF100070
C           EQUATIONS ON THE WHOLE GRID. FOR FURTHER INFORMATION SEE    FF100080
C           THE LISTING OF SUBROUTINE SYSSOL AND APPENDIX C OF          FF100090
C           THE THESIS.                                                 FF100100
C                                                                       FF100110
C     ******************************************************************FF100120
C                                                                       FF100130
      SUBROUTINE FF1(X,Y,N,FF)                                          FF100140
      IMPLICIT REAL*8(A-H,O-Z)                                          FF100150
      COMMON /CONS/ PR                                                  FF100160
      DIMENSION X(1),Y(1),FF(1)                                         FF100170
C                                                                       FF100180
      DO 10 I=1,N                                                       FF100190
      KI=(I-1)*5+1                                                      FF100200
      FF(KI)=Y(KI+1)                                                    FF100210
      FF(KI+1)=Y(KI+2)                                                  FF100220
      FF(KI+2)=-3.D0*Y(KI)*Y(KI+2)+2.D0*Y(KI+1)**2-Y(KI+3)              FF100230
      FF(KI+3)=Y(KI+4)                                                  FF100240
   10 FF(KI+4)=-3.D0*PR*Y(KI)*Y(KI+4)                                   FF100250
C                                                                       FF100260
      RETURN                                                            FF100270
      END                                                               FF100280
C                                                                       JAC10010
C     ******************************************************************JAC10020
C                                                                       JAC10030
C        SUBROUTINE JACOB1(X,YX,JAB)                                    JAC10040
C                                                                       JAC10050
C        PURPOSE                                                        JAC10060
C           THIS SUBROUTINE EVALUATES THE JACOBIAN OF THE DIFFERENTIAL  JAC10070
C           EQUATIONS AT A GIVEN GRID POINT. FOR FURTHER INFORMATION    JAC10080
C           SEE THE LISTING OF SUBROUTINE SYSSOL AND APPENDIX C         JAC10090
C           OF THE THESIS.                                              JAC10100
C                                                                       JAC10110
C     ******************************************************************JAC10120
C                                                                       JAC10130
      SUBROUTINE JACOB1(X,YX,JAB)                                       JAC10140
      IMPLICIT REAL*8(A-H,O-Z)                                          JAC10150
      COMMON /CONS/ PR                                                  JAC10160
      COMMON /EQU/ M                                                    JAC10170
      DIMENSION YX(1)                                                   JAC10180
      DOUBLE PRECISION JAB(10,10)                                       JAC10190
C                                                                       JAC10200
      DO 10 I=1,M                                                       JAC10210
      DO 10 J=1,M                                                       JAC10220
   10 JAB(I,J)=0.D0                                                     JAC10230
      JAB(1,2)=1.D0                                                     JAC10240
      JAB(2,3)=1.D0                                                     JAC10250
      JAB(3,1)=-2.D0*YX(3)                                              JAC10260
      JAB(3,2)=4.D0*YX(2)                                               JAC10270
      JAB(3,3)=-3.D0*YX(1)                                              JAC10280
      JAB(3,4)=-1.D0                                                    JAC10290
      JAB(4,5)=1.D0                                                     JAC10300
      JAB(5,1)=-3.D0*PR*YX(5)                                           JAC10310
      JAB(5,5)=-3.D0*PR*YX(1)                                           JAC10320
C                                                                       JAC10330
      RETURN                                                            JAC10340
      END                                                               JAC10350
C                                                                       HEAT0010
C     ******************************************************************HEAT0020
C                                                                       HEAT0030
C        SUBROUTINE HEAT                                                HEAT0040
C                                                                       HEAT0050
C        PURPOSE                                                        HEAT0060
C           CALCULATE HEAT TRANSFER COEFFICIENT.                        HEAT0070
C        REMARK                                                         HEAT0080
C           FOR REFRIGERANT-114 STATEMENTS 10, 20, AND 30 SHOULD BE     HEAT0090
C           REPLACED BY THE FOLLOWING STATEMENTS.                       HEAT0100
C           CONDX=COND(J)*241.9D00                                      HEAT0110
C           PRIN=(VISC(JJ)*CPH(JJ)*(1.0D-02))/COND(JJ)                  HEAT0120
C           PRW=(VISC(J)*CPH(J)*(1.0D-02))/COND(J)                      HEAT0130
C                                                                       HEAT0140
C     ******************************************************************HEAT0150
C                                                                       HEAT0160
      SUBROUTINE HEAT                                                   HEAT0170
      IMPLICIT REAL*8(A-H,O-Z)                                          HEAT0180
      COMMON /CHEAT/ QWX,HWX,RNUGR,GRX,CNUX,RATIO,Y,PRW,PRIN            HEAT0190
      COMMON /CTEMP/ TIN,TW,DT,TA                                       HEAT0200
      COMMON/TERM/AA1(50),AA2(50),AA3(50),BB1(50),BB2(50)               HEAT0210
      COMMON/CPROP/DENS(50),CPH(50),VISC(50),COND(50),TEMP(50),DTEMP    HEAT0220
      COMMON /EQU/ M                                                    HEAT0230
      DIMENSION Y(2500)                                                 HEAT0240
C                                                                       HEAT0250
C     CALCULATE CONSTANT C IN SIMILARITY VARIABLE ETA                   HEAT0260
C                                                                       HEAT0270
      J=IDINT(((TW-TEMP(1))/DTEMP)+0.5)+1                               HEAT0280
      K=IDINT(((TX-TEMP(1))/DTEMP)+0.5)+1                               HEAT0290
      JJ=IDINT(((TIN-TEMP(1))/DTEMP)+0.5)+1                             HEAT0300
      DENR=(DENS(JJ)-DENS(J))/DENS(J)                                   HEAT0310
      VISWX=VISC(K)/(1490.D0)                                           HEAT0320
      VISNX=VISWX/DENS(K)                                               HEAT0330
      VISNX2=VISNX*VISNX                                                HEAT0340
      C1X=(32.2D0*DENR)/(4.D0*VISNX2)                                   HEAT0350
      CX=C1X**0.25                                                      HEAT0360
C                                                                       HEAT0370
C     CALCULATE RATE OF HEAT TRANSFER                                   HEAT0380
C                                                                       HEAT0390
      TPWX=Y(M)                                                         HEAT0400
   10 CONDX=COND(J)                                                     HEAT0410
      QWX=-CONDX*(TW-TIN)*CX*TPWX                                       HEAT0420
C                                                                       HEAT0430
C     CALCULATE HEAT TRANSFER COEFFICIENT                               HEAT0440
C                                                                       HEAT0450
      HWX=QWX/(TW-TIN)                                                  HEAT0460
C                                                                       HEAT0470
```

```
C     CALCULATE GRASHOF NUMBER                                          HEAT0480
C                                                                       HEAT0490
      GRX=(32.D0*DENR)/VISNX2                                           HEAT0500
C                                                                       HEAT0510
C     CALCULATE NUSSELT NUMBER                                          HEAT0520
C                                                                       HEAT0530
      CNUX=HWX/CONDX                                                    HEAT0540
C                                                                       HEAT0550
C     CALCULATE RATIO OF NUSSELT NO. AND GRASHOF NO.                    HEAT0560
C                                                                       HEAT0570
      RNUGR=-TPWX                                                       HEAT0580
      RATIO=CNUX/(GRX/4.D0)**0.25                                       HEAT0590
C                                                                       HEAT0600
C     CALCULATE PRANDTL NUMBER                                          HEAT0610
C                                                                       HEAT0620
   20 PRIN=(VISC(JJ)*CPH(JJ)*(2.*2D00))/COND(JJ)                        HEAT0630
   30 PR#=(VISC(J)*CPH(J)*(2.*2D00))/COND(J)                            HEAT0640
C                                                                       HEAT0650
      RETURN                                                            HEAT0660
      END                                                               HEAT0670
```

REFRIGERANT 114 -- CONSTANT PROPERTY PROBLEM WITH PR = 3.55

NUMBER OF EQUATIONS 5

FINAL NUMBER OF MESH POINTS 97

LEFT BOUNDARY POINT = 0.0 RIGHT BOUNDARY POINT = 10.00

BOUNDARY CONDITIONS

0.0	0.0	0.0	1.00	0.0

MATRIX OF LEFT BOUNDARY CONDITIONS

1.00	0.0	0.0	0.0	0.0
0.0	1.00	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	1.00	0.0
0.0	0.0	0.0	0.0	0.0

MATRIX OF RIGHT BOUNDARY CONDITIONS

0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	1.00	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	1.00	0.0

TOLERANCE 0.10D-05

ERROR ESTIMATE FOR COMPONENTS

0.7050-06 0.120D-06 0.4410-07 0.285D-07 0.298D-07

JERROR = 0

REFRIGERANT 114 -- CONSTANT PROPERTY PROBLEM WITH PR = 3.55

N	ETA	F	FP	FPP	T	TP
1	0.0	0.0	0.41359D-27	0.51455D+00	0.10000D+01	-0.85x14D+00
6	0.52083D+00	0.48949D-01	0.15524D+00	0.11462D+00	0.56392D+00	-0.777950+00
11	0.10417D+01	0.13426D+00	0.15954D+00	-0.57683D-01	0.23313D+00	-0.468450+00
16	0.15625D+01	0.20714D+00	0.11784D+00	-0.87784D-01	0.70562D-01	-0.18021D+00
21	0.20833D+01	0.25724D+00	0.76266D-01	-0.69767D-01	0.16761D-01	-0.49274D-01
26	0.26042D+01	0.28876D+00	0.46817D-01	-0.45124D-01	0.33795D-02	-0.10796D-01
31	0.31250D+01	0.30783D+00	0.26129D-01	-0.27812D-01	0.61444D-03	-0.20547D-02
36	0.36458D+01	0.31933D+00	0.16755D-01	-0.16747D-01	0.10487D-03	-0.36946D-03
41	0.41667D+01	0.32513D+00	0.99390D-02	-0.99920D-02	0.17231D-04	-0.60188D-04
46	0.46875D+01	0.33016D+00	0.58409D-02	-0.59375D-02	0.27676D-05	-0.97532D-05
51	0.52083D+01	0.33254D+00	0.34720D-02	-0.35213D-02	0.43858D-06	-0.15552D-05
56	0.57292D+01	0.33394D+00	0.20441D-02	-0.20861D-02	0.68952D-07	-0.24530D-06
61	0.62500D+01	0.33477D+00	0.11984D-02	-0.12352D-02	0.10790D-07	-0.38458D-07
66	0.67708D+01	0.33525D+00	0.69772D-03	-0.73112D-03	0.16837D-08	-0.60082D-08
71	0.72917D+01	0.33553D+00	0.40142D-03	-0.43267D-03	0.26232D-09	-0.93767D-09
76	0.78125D+01	0.33569D+00	0.22607D-03	-0.25603D-03	0.40920D-10	-0.14579D-09
81	0.83333D+01	0.33578D+00	0.12232D-03	-0.15149D-03	0.63371D-11	-0.22599D-10
86	0.88542D+01	0.33582D+00	0.60922D-04	-0.89638D-04	0.97178D-12	-0.35310D-11
91	0.93750D+01	0.33585D+00	0.24597D-04	-0.53038D-04	0.13722D-12	-0.54917D-12
96	0.98958D+01	0.33585D+00	0.31033D-05	-0.31342D-04	0.74281D-14	-0.85412D-13

REFRIGERANT 114 -- CONSTANT PROPERTY PROBLEM WITH PR = 3.55

WALL TEMPERATURE = 779.69 R

FREE STREAM TEMPRATURE = 774.69 R

TEMPERATURE DIFFERENCE = 5.00 R

REFERENCE TEMPERATURE = 774.69 R

WALL PRANDTL NUMBER = 2.55

FREE STREAM PRANDTL NUMBER = 3.55

NUSSELT NUMBER = 0.10022D+04

GRASHOF NUMBER = 0.73613D+1?

Q*X**(1/4) = HEAT TRANSFER IN BTU/HR-FT**(7/4)

H*X**(1/4) = COEFFICIENT OF HEAT TRANSFER IN BTU/HR-FT**(7/4)-R

NU/GR = RATIO OF NUSSELT NO. AND GRASHOF NO.

Q*X**(1/4)	H*X**(1/4)	NU/GR
0.65986312D+02	0.13197262D+02	0.85914446D+00

VITA[2]

Afshin Jahanshahi Ghajar

Candidate for the Degree of

Doctor of Philosophy

Thesis: LAMINAR FREE CONVECTIVE HEAT TRANSFER IN THE SUPERCRITICAL REGION WITH VARIABLE FLUID PROPERTIES

Major Field: Mechanical Engineering

Biographical:

Personal Data: Born in Tehran, Iran, June 10, 1951, the son of Iraj and Parvin Jahanshahi Ghajar.

Education: Graduated from Hadaf High School, Tehran, Iran, May, 1969; received the Bachelor of Science degree in Mechanical Engineering from Oklahoma State University, Stillwater, Oklahoma, May, 1974; received the Master of Science degree in Mechanical Engineering from Oklahoma State University, December, 1975; completed requirements for the Doctor of Philosophy degree at Oklahoma State University, July, 1979.

Professional Experience: Graduate Teaching and Research Assistant, School of Mechanical and Aerospace Engineering, Oklahoma State University, August, 1974, to December, 1975; Graduate Teaching and Research Associate, School of Mechanical and Aerospace Engineering, Oklahoma State University, January, 1976, to July, 1979.

Professional Organizations: Associate Member, American Society of Mechanical Engineers (ASME); American Institute of Aeronautics and Astronautics (AIAA); American Society of Heating, Refrigeration and Air-Conditioning Engineers (ASHRAE).

Honors and Awards: Phi Eta Sigma; Pi Tau Sigma; Dean's Honor Roll of Distinguished Students, Oklahoma State University, September, 1970, to May, 1974; recipient of National Iranian Oil Company Scholarship (NIOC), February, 1972, to May, 1974; two-time recipient of International Student Scholarship, Oklahoma State University, September, 1971, to May, 1972, and September, 1973, to May, 1974.