Higher Order Approximation

for

Combined Mode Heat Transfer in Building Insulations

by

Sanjeev Gupta

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of
Master of Science
in
Mechanical Engineering

APPROVED:

James R. Thomas

Dr. James R. Thomas, Chairman

Dr. James R. Mahan

Dr. Douglas J. Nelson

January, 1988

Blacksburg, Virginia
Higher Order Approximation
for
Combined Mode Heat Transfer in Building Insulations
by
Sanjeev Gupta
Dr. James R. Thomas, Chairman
Mechanical Engineering
(ABSTRACT)

For heat transfer through building insulations such as fiberglass, radiation and conduction are important modes of heat transfer. Moreover, materials like fiberglass scatter radiation in a highly anisotropic manner. The equations for heat transfer by simultaneous conduction and radiation are a coupled pair, one of which is of the nonlinear integrodifferential type. Exact solution for transient heat transfer in this case is not available, and the approximate solution available is the two-flux model.

The two-flux model does not give good results for transient, combined mode heat transfer, through an absorbing, emitting, and anisotropically scattering medium.

In this thesis a higher order approximate solution has been developed. It is found that this model gives appreciably better results than the two-flux model.
# Table of Contents

Introduction and Literature Review .................................................. 1

Review of previous work ........................................................................... 6

Solution by the $P_n$ Method. ................................................................. 14

3.1 Mathematical formulation. ............................................................... 14

3.2 Solution method. ............................................................................. 19

3.3 Results and Discussion ........................................................................ 24

3.4 Conclusion and Recommendations ..................................................... 29

References .............................................................................................. 34

Particular Solution for Radiation Intensity .................................................. 38

Polynomial Fit for Temperature Profile ..................................................... 44

FORTRAN PROGRAM ............................................................................... 47

Table of Contents
Acknowledgements

The author is extremely grateful to his major professor, Dr. J.R. Thomas. Without his invaluable and active support, patience, and perspicacity, this research could not have been completed. It was indeed a joy to work with him. I am also thankful to Drs. J.R. Mahan and D.J. Nelson, for their insightful comments and suggestions.

I thank the faculty of Virginia Tech for augmenting my understanding of engineering, and my colleagues for being a constant source of encouragement.

I shall forever remain indebted to my parents for inspiring me, and for financing my stay and studies in the United States.
List of Illustrations

Figure 1. Polar Representation of Actual Radiative Heat Flux and its Approximation in the Two-Flux Model ................................ 3

Figure 2. Representation of the Coordinate System Used. .......................... 8

Figure 3. Modelling of the Diurnal Variation of Cold Boundary Temperature; $C_1 = 273$ K, $C_2 = 5$ K, $C_3 = 11$ K. ................................. 10

Figure 4. Comparison of schemes for temperature profile fit, at time = 0.5 hrs. and emissivities = 1.0 ................................................. 46
List of Tables

Table 1. Results from Two-flux Model for Accumulated Heat at the Boundaries, $\sigma_e = 400\text{m}^{-1}$, $\varepsilon_h = \varepsilon_c = 1.0$ ........................................ 11

Table 2. Results from Two-flux Model for Accumulated Heat at the Boundaries, $\sigma_e = 10,000\text{m}^{-1}$, $\varepsilon_h = \varepsilon_c = 0.1$ ............................. 12

Table 3. Properties of the Building Insulation Used in Calculations....... 23

Table 4. Effect of Various Parameters in $P_N$ Model on the Results, $\sigma_e = 400\text{m}^{-1}$, $\varepsilon_h = \varepsilon_c = 1.0$ ..................................................... 25

Table 5. Effect of Various Parameters in $P_N$ Model on the Results, $\sigma_e = 10,000\text{m}^{-1}$, $\varepsilon_h = \varepsilon_c = 0.1$ ..................................................... 26

Table 6. Results from the $P_N$ Model for Accumulated Heat at the Boundaries, $\sigma_e = 400\text{m}^{-1}$, $\varepsilon_h = \varepsilon_c = 1.0$ ........................................ 28

Table 7. Results from the $P_N$ Model for Accumulated Heat at the Boundaries, $\sigma_e = 10,000\text{m}^{-1}$, $\varepsilon_h = \varepsilon_c = 0.1$ ............................. 29

Table 8. Results from the $P_N$ Model for Accumulated Heat at the Boundaries, using Hermite Cubic Interpolation ............................. 30
**Nomenclature**

ACC accumulated heat (per unit area), at cold boundary, J/m².

ACH accumulated heat (per unit area), at hot boundary, J/m².

\( \{A_i\}, \{B_i\} \) coefficients in equation (3-13).

b back scattering coefficient.

\( C_1, C_2, C_3 \) constants in equations (3-3).

\( C_n(\tau) \) coefficients in particular solution of \( \psi(\tau, \mu) \).

\( c_{km} \) see Appendix A.

\( g_0(\xi) \) Chandrashekar polynomials.

\( h_c \) see equation (3-14).

I radiation intensity, W/m².

j defined in equation (2-1).

k thermal conductivity, W/m-K.

N order of \( P_n \) approximation.

\( Q_r \) nondimensionalized radiative flux.

\( q_r \) radiative heat flux, W/m².

\( R_{1,*}, R_{2,*} \) see equation (3-15).

\( S_{*,t} \) see equation (3-15).

\( t \) nondimensional time.

T temperature, K.

x see figure 2.

\( \alpha_k \) see equation (3-15).
\( \varepsilon \)  emissivities.

\( \theta \)  nondimensionalized temperature.

\( \kappa \)  absorption coefficient.

\( \mu \)  \( \cos \gamma \) (see figure 2).

\( \xi \)  see equation (3-13).

\( \rho \)  density, kg/m³.

\( \rho^s \)  specular component of reflectivity

\( \rho^d \)  diffuse component of reflectivity

\( \sigma \)  Stefan-Boltzmann constant, W/(m².K⁴).

\( \sigma_a \)  absorption coefficient

\( \sigma_e \)  extinction coefficient

\( \sigma_s \)  scattering coefficient

\( \tau \)  optical length

\( \psi \)  nondimensionalized radiation intensity

\( \omega \)  single scattering albedo

\( \zeta \)  time, s.
Chapter 1

Introduction and Literature Review

Many engineering applications involve transient heat transfer through porous materials. One such situation is encountered when dealing with building insulations, such as fibreglass, foams, and fine powders in loose-fill form, or in the form of batts and boards.

For the light-weight insulations, radiation, along with conduction, is an important mode of heat transfer [1]. Moreover, typical fibreglass insulations scatter radiation in a highly anisotropic manner [2].

Radiative transfer accounts for a significant amount of heat flow through the building insulations [1]. The radiation and conduction heat flows interact through absorption and reemission, and thus require a coupled treatment. Cess [3] discusses a solution based on a simple addition of the conductive and radiative heat transfers,
in a plane slab. When the boundaries are black, the energy transfer was within 10% of the exact solution, and larger errors were obtained for highly reflective boundaries.

The governing equations of heat transfer through combined radiation and conduction are a coupled pair; one of these is of the nonlinear integrodifferential type [2]. Analytical solutions for the case of transient heat transfer by coupled conduction and radiation are not available, so recourse has to be made to approximate solutions.

The steady-state heat transfer by simultaneous radiation and conduction has been studied by several investigators [4-9]; scattering effect has also been included in the analysis [10,11]. The non-steady case has been studied for an absorbing, emitting, and isotropically scattering medium [12-15]. The only effort towards including anisotropic scattering, in the author’s knowledge, has been Tong’s [16] two-flux [17], finite-difference solution methodology.

One of the discrepancies observed in the above-mentioned two-flux model is that time-averaged energy influx and efflux for the insulation do not match. The actual numerical results from this model and the severity of mismatch are described in Chapter 2. A possible cause of this mismatch is the two-flux approximation. Though it works well for the interior of the region, the two-flux model is very inaccurate at or near the boundaries [17].

Figure 1 illustrates why the two-flux model breaks down at or near the boundaries. The two-flux model treats net radiative heat flux as combination of a forward and a backward heat flux. But as Fig. 1 shows, at any spatial point there is a continuous variation of the magnitude of actual radiative flux with direction. At and near the
a) Actual Radiative Flux

b) Two-Flux Approximation

Length of the arrow is proportional to the magnitude of radiative flux in its direction.

Figure 1. Polar Representation of Actual Radiative Heat Flux and its Approximation in the Two-Flux Model
boundaries, this variation is more pronounced; at the boundaries, the magnitude of
the radiative flux varies from a zero to a maximum, and then back to zero.

A better approximation of the directional dependence of radiative flux might yield
more accurate results; the \( P_n \) [17] method makes such an approximation, by an ex-
pansion in Legendre Polynomials. As the number of terms in the expansion ap-
proaches infinity, the ideal representation of the directional dependence of intensity
is obtained. Thus the \( P_n \) method provides a means to obtain a higher order approxi-
mate solution to the equation of radiative transfer, at the expense of additional labor
and calculations. This method was first suggested by Jeans [18], in connection with
the problem of radiative transfer in stars. In 1945 Mark [19] described the \( P_n \) method
and its potential for application to nuclear energy transport. A good general de-
scription of the \( P_n \) method can be found in the book by Kourganoff [20] for radiative
transfer applications, and the books by Davison [21] and Murray [22] for neutron
transport applications. Initial work in the application of the \( P_n \) method to radiative
transfer concerned the vacuum boundary conditions. For example, one can look at
the work of Federighi [23] and Pellaud [24].

The objective of the work presented here is to analyze the effect of application of
the \( P_n \) approximation to the angular distribution of intensity in the problem of coupled
conduction and radiation heat transfer, with anisotropic scattering. Once this model
is verified and established, it can be used to determine the sensitivity of the thermal
resistance of the insulation to the radiative properties of its constituent materials.
This, in turn, can help in the development of insulations with significantly improved
thermal resistance. Development of such insulations can result in very large energy
savings.
Chapter 2 of this report is a review of Tong's work [16]. The mathematical formulation of the problem, application of $P_n$ and finite difference approximations to the equations, the solution procedure, and results are discussed in Chapter 3. The FORTRAN computer program is given in the Appendix.
Chapter 2

Review of previous work

This chapter is review of Tong’s [16] work on heat transfer in building insulations, using the two-flux model.

In the two-flux model, radiative heat flux is treated as a combination of a forward and a backward heat flux \( (q^+ \text{ and } q^-) \), respectively. Similarly, radiation intensity within the insulation is divided into a forward and backward radiation stream, \( j^+ \) for \( 0 < \mu \leq +1 \) and \( j^- \) for \( -1 < \mu \leq 0 \) (see Fig.2), respectively, defined as,

\[
\begin{align*}
    j^+ &\equiv \int_0^1 I^+(\tau, \mu) d\mu \\
    j^- &\equiv \int_{-1}^0 I^-(\tau, \mu) d\mu
\end{align*}
\]
The two-flux approximation refers to treating $I^+ (\tau, \mu)$ and $I^- (\tau, \mu)$ as constant over the respective ranges of $\mu$. Tong assumed a one dimensional heat transfer across the test specimen, and treated the test specimen as isotropic and homogenous.

The governing equations, therefore, are [16]

$$\frac{k}{\partial x^2} T - \frac{\partial}{\partial x} q_r = \rho_p C_p \frac{\partial}{\partial \zeta} T ; \quad (2-2)$$

$$\frac{\partial}{\partial x} q^+ = -2\sigma_a q^+ - 2\sigma_a q^+ + 2\sigma_a \sigma T^4 + 2\sigma_s b q^- ; \quad (2-3)$$

$$\frac{\partial}{\partial x} q^- = -2\sigma_a q^- - 2\sigma_a q^- + 2\sigma_a \sigma T^4 + 2\sigma_s b q^+ . \quad (2-4)$$

where the definition of symbols is given in the Nomenclature ($\sigma_a$ is the absorption coefficient, for which many authors use $\kappa$). Equation (2-2) is the energy equation. The first term on the left-hand side pertains to conduction, and the second term on the left-hand side is for radiation; the term on right-hand side is the heat storage term. Equations (2-3) and (2-4) constitute the two-flux equations for radiative heat transfer and are coupled to Equation (2-2) by

$$q_r = q^+ - q^- . \quad (2-5)$$

The boundary conditions are,

$$T = T_h, \quad x = 0 \quad (2-6a)$$

and

$$T = T_c, \quad x = L. \quad (2-6b)$$

Review of previous work
Figure 2. Representation of the Coordinate System Used.
The temperature at $x = 0$ is taken to be constant in the analysis, and to model the diurnal variation of temperature at the cold boundary we use (see Fig. 3),

$$T_c = C_1 + C_2 \sin\left(\frac{\pi \zeta}{12}\right), \quad 0 < \zeta \leq 12 \text{ hours} ; \quad (2-7a)$$

and

$$T_c = C_1 + C_3 \sin\left(\frac{\pi \zeta}{12}\right), \quad 12 < \zeta \leq 24 \text{ hours} . \quad (2-7b)$$

The partial heat fluxes $q^+$ and $q^-$ are related to $j^+$ and $j^-$ by

$$q^+ = 2\pi \int_0^1 l^+(\tau, \mu) \mu d\mu = \pi j^+ ; \quad (2-8a)$$

and

$$q^- = 2\pi \int_{-1}^0 l^-(\tau, \mu) \mu d\mu = \pi j^- . \quad (2-8b)$$

In the method of solution used by Tong, equations (2-2)-(2-5) are written in finite difference form. Manipulation yields equations which are solved as follows:

1. The initial temperature distribution across the insulation is specified and is used to calculate $q^+$ and $q^-$ for all the nodes, using equations (2-3 and 2-4).

2. $q^+$ and $q^-$ are used to find the temperatures at one time increment later, using equations (2-2) and (2-5).
Figure 3. Modelling of the Diurnal Variation of Cold Boundary Temperature; $C_1 = 273\, \text{K}, \quad C_2 = 5\, \text{K}, \quad C_3 = 11\, \text{K}$.

Review of previous work
In Ref. [16] Tong gives a FORTRAN program for this methodology; also given are the values of heat accumulated, per unit area, at the hot and cold boundaries up to 168 hrs. His calculations were repeated and verified by the present author. The results using his solution methodology are given in Tables 1 and 2. The percent differences, defined as \( \frac{ACH - ACC}{ACC} \times 100 \), remain in the neighborhood of 7.5 percent for \( \varepsilon_c = \varepsilon_h = 1.0 \) and \( \sigma_* = 400 \text{ m}^{-1} \); after the initial 12 hrs the differences remain at less than 1 percent for \( \varepsilon_h = \varepsilon_c = 0.1 \) and \( \sigma_* = 10,000 \text{ m}^{-1} \). These differences in accumulated heats can not be accounted for by heat storage in the insulation. As explained in the next chapter, the quantity of heat stored is negligible compared to the accumulated heats, particularly after large times of simulation.
Table 1. Results from Two-flux Model for Accumulated Heat at the Boundaries, $\sigma = 400 \text{m}^{-1}$, $\xi = \xi_c = 1.0$.

<table>
<thead>
<tr>
<th>time (hrs.)</th>
<th>ACH (J/m²)</th>
<th>ACC (J/m²)</th>
<th>Percent Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4.563 E04</td>
<td>4.171 E04</td>
<td>+9.40</td>
</tr>
<tr>
<td>2</td>
<td>8.900 E04</td>
<td>8.159 E04</td>
<td>+9.08</td>
</tr>
<tr>
<td>3</td>
<td>1.305 E05</td>
<td>1.198 E05</td>
<td>+8.93</td>
</tr>
<tr>
<td>4</td>
<td>1.704 E05</td>
<td>1.566 E05</td>
<td>+8.81</td>
</tr>
<tr>
<td>5</td>
<td>2.092 E05</td>
<td>1.924 E05</td>
<td>+8.73</td>
</tr>
<tr>
<td>6</td>
<td>2.474 E05</td>
<td>2.278 E05</td>
<td>+8.60</td>
</tr>
<tr>
<td>12</td>
<td>4.933 E05</td>
<td>4.577 E05</td>
<td>+7.78</td>
</tr>
<tr>
<td>18</td>
<td>8.381 E05</td>
<td>7.819 E05</td>
<td>+7.19</td>
</tr>
<tr>
<td>24</td>
<td>1.186 E06</td>
<td>1.102 E06</td>
<td>+7.62</td>
</tr>
<tr>
<td>30</td>
<td>1.433 E06</td>
<td>1.330 E06</td>
<td>+7.74</td>
</tr>
<tr>
<td>36</td>
<td>1.679 E06</td>
<td>1.559 E06</td>
<td>+7.70</td>
</tr>
<tr>
<td>42</td>
<td>2.024 E06</td>
<td>1.884 E06</td>
<td>+7.43</td>
</tr>
<tr>
<td>48</td>
<td>2.371 E06</td>
<td>2.204 E06</td>
<td>+7.58</td>
</tr>
<tr>
<td>72</td>
<td>3.557 E06</td>
<td>3.305 E06</td>
<td>+7.62</td>
</tr>
<tr>
<td>96</td>
<td>4.742 E06</td>
<td>4.407 E06</td>
<td>+7.60</td>
</tr>
<tr>
<td>120</td>
<td>5.926 E06</td>
<td>5.509 E06</td>
<td>+7.61</td>
</tr>
<tr>
<td>144</td>
<td>7.113 E06</td>
<td>6.611 E06</td>
<td>+7.60</td>
</tr>
<tr>
<td>168</td>
<td>8.299 E06</td>
<td>7.712 E06</td>
<td>+7.61</td>
</tr>
</tbody>
</table>
Table 2. Results from Two-flux Model for Accumulated Heat at the Boundaries, $\sigma = 10,000\text{m}^{-1}$, $\varepsilon = 0.1$

<table>
<thead>
<tr>
<th>t (hrs.)</th>
<th>ACH (J/m²)</th>
<th>ACC (J/m²)</th>
<th>Percent Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.931 E04</td>
<td>2.896 E04</td>
<td>+ 1.21</td>
</tr>
<tr>
<td>2</td>
<td>5.721 E04</td>
<td>5.652 E04</td>
<td>+ 1.22</td>
</tr>
<tr>
<td>3</td>
<td>8.358 E04</td>
<td>8.286 E04</td>
<td>+ 1.19</td>
</tr>
<tr>
<td>4</td>
<td>1.094 E05</td>
<td>1.082 E05</td>
<td>+ 1.11</td>
</tr>
<tr>
<td>5</td>
<td>1.343 E05</td>
<td>1.330 E05</td>
<td>+ 0.98</td>
</tr>
<tr>
<td>6</td>
<td>1.587 E05</td>
<td>1.574 E05</td>
<td>+ 0.83</td>
</tr>
<tr>
<td>12</td>
<td>3.161 E05</td>
<td>3.169 E05</td>
<td>-0.25</td>
</tr>
<tr>
<td>18</td>
<td>5.402 E05</td>
<td>5.454 E05</td>
<td>-0.95</td>
</tr>
<tr>
<td>24</td>
<td>7.670 E05</td>
<td>7.691 E05</td>
<td>-0.27</td>
</tr>
<tr>
<td>30</td>
<td>9.257 E05</td>
<td>9.264 E05</td>
<td>-0.08</td>
</tr>
<tr>
<td>36</td>
<td>1.083 E06</td>
<td>1.086 E06</td>
<td>-0.28</td>
</tr>
<tr>
<td>42</td>
<td>1.307 E06</td>
<td>1.314 E06</td>
<td>-0.53</td>
</tr>
<tr>
<td>48</td>
<td>1.534 E06</td>
<td>1.538 E06</td>
<td>-0.26</td>
</tr>
</tbody>
</table>
Chapter 3

Solution by the Pₙ Method.

The Spherical Harmonics (Pₙ) method refers to expressing the directional dependence of \( I(\tau, \mu) \) by an expansion in Legendre polynomials. The system to be modelled is the same as sketched in Fig. 2.

3.1 Mathematical formulation.

Heat transfer through the wall is assumed to be one-dimensional, and the wall is treated as isotropic and homogenous. Thus \( T(x, \xi) \) satisfies the equation [17],

\[
k \frac{\partial^2 T}{\partial x^2} - \frac{\partial}{\partial x} q_r = \rho_p C_p \frac{\partial}{\partial \xi} T, \quad 0 \leq x \leq L.
\]  

(3 - 1)

Boundary conditions are the same as described by equations (2-6 and 2-7), i.e.
The temperature at \( x = 0 \) is taken to be constant, and to represent the diurnal variation of temperature at the cold boundary we again use

\[
T_c = C_1 + C_2 \sin\left(\frac{\pi \xi}{12}\right), \quad 0 < \xi \leq 12 \text{ hours} \quad (3 - 2c)
\]

and

\[
T_c = C_1 + C_3 \sin\left(\frac{\pi \xi}{12}\right), \quad 12 < \xi \leq 24 \text{ hours} \quad (3 - 2d)
\]

The radiative flux is computed from the radiation intensity, \( I(x, \mu) \), according to [17]

\[
q_r(x, \xi) = 2\pi \int_{-1}^{1} I(x, \mu) \mu d\mu \quad (3 - 3)
\]

The radiation intensity satisfies the integrodifferential equation of radiative transfer [17],

\[
\mu \frac{d}{dx} I + \sigma_s I(x, \mu) = \frac{\sigma_s}{2} \int_{-1}^{1} p(\mu, \mu') I(x, \mu') d\mu' + \kappa \frac{\sigma}{\pi} T^4(x, \xi) \quad (3 - 4)
\]

where \( \kappa \) and \( \sigma_s \) represent the absorption and scattering coefficients, respectively; \( \sigma_s = \kappa + \sigma \) is the extinction coefficient; \( \sigma \) is the Stefan-Boltzmann constant, and

\text{Solution by the } P_n \text{ Method.}
$p(\mu, \mu')$ is the phase function [17]. The phase function, which represents the fraction of the radiation originally travelling in the direction $\mu$ which is scattered into directions within $d\mu'$ about $\mu'$ , is usually represented by an expansion in Legendre polynomials [17]:

$$p(\mu, \mu') = \sum_{m=0}^{L} \beta_m P_m(\mu)P_m(\mu').$$  \hspace{1cm} (3 - 5)

Following Tong [16], the equations are nondimensionalized as follows:

the optical length variables $\tau$ and $\tau_o$ are defined as

$$\tau = \sigma_e x , \hspace{1cm} (3 - 6a)$$

$$\tau_o = \sigma_e L ; \hspace{1cm} (3 - 6b)$$

the nondimensional time as

$$t = \frac{k \xi}{\rho_p C_p L^2} , \hspace{1cm} (3 - 6c)$$

and temperature as

$$\theta = \frac{T}{T_h} . \hspace{1cm} (3 - 6d)$$

The heat generation rate within the insulation (zero for our case) is nondimensionalized as

Solution by the $P_n$ Method.
\[ \dot{Q} = \frac{\dot{q}_l}{4\sigma T_n^4}. \quad \text{(3-6e)} \]

It is convenient to define a nondimensional radiation intensity,

\[ \psi(\tau, \mu) = \frac{I(\tau, \mu)}{\sigma T_n^4}, \quad \text{(3-6f)} \]

and heat flux,

\[ Q_r = \frac{q_r}{4\sigma T_n^4}. \quad \text{(3-6g)} \]

For convenience, we also use the conduction-to-radiation parameter, defined as

\[ N = \frac{k\sigma_e}{4\sigma T_n^3}. \quad \text{(3-6h)} \]

The equations (3-1 to 3-6) can thus be rewritten as follows. The equation for conservation of energy is

\[ \frac{\partial^2 \theta}{\partial \tau^2} - \frac{1}{N} \frac{\partial}{\partial \tau} Q_r = \frac{1}{\tau_o^2} \frac{\partial}{\partial t} \theta, \quad \text{(3-7a)} \]

with the initial and boundary conditions,

\[ \theta = \theta_h = 1, \quad \tau = 0, \quad t > 0, \quad \text{(3-7b)} \]

Solution by the P_\infty Method.
\[ \theta = \theta_0 = \frac{T_c}{T_h}, \quad \tau = \tau_0, \quad t > 0 \text{ and} \]  \[ (3-7c) \]

\[ \theta = 1 - \frac{C_1}{T_h} \frac{\tau}{\tau_0}, \quad 0 < \tau < \tau_0, \quad t = 0. \]  \[ (3-7d) \]

Equation (3-7d) represents a linear initial temperature profile.

The equation for radiation transfer is

\[ \mu \frac{\partial \psi}{\partial \tau} + \psi(\tau, \mu) = \frac{\omega}{2} \sum_{m=0}^{L} \beta_m P_m(\mu) \int_{-1}^{1} P_m(\mu') \psi(\tau, \mu') d\mu' + (1 - \omega) \theta^4, \]  \[ (3-8) \]

with the boundary conditions

\[ \psi(0, \mu) = \frac{\epsilon_r \theta_1^4}{\pi} + \rho_s \psi(0, -\mu) + 2\rho_d \int_{0}^{1} \psi(0, -\mu') \mu' d\mu', \quad \mu > 0, \]  \[ (3-9a) \]

\[ \psi(\tau_o, \mu) = \frac{\epsilon_r \theta_2^4}{\pi} + \rho_s \psi(\tau_o, \mu) + 2\rho_d \int_{0}^{1} \psi(\tau_o, \mu') \mu' d\mu', \quad \mu > 0. \]  \[ (3-9b) \]

where \( \rho_s \) and \( \rho_d \) are the specular and diffuse components of reflectivity of the boundaries.

The heat flux is determined from

\[ Q_f(\tau) = \frac{1}{2} \int_{-1}^{1} \psi(\tau, \mu) \mu d\mu. \]  \[ (3-10) \]

Solution by the \( P_n \) Method.
Equations (3-7)-(3-10) constitute the complete formulation of the combined mode problem.

3.2 Solution method.

The heat conduction equation (3-7) is solved by an implicit, (central) finite-difference technique: the interval \(0 < \tau < \tau_s\) is divided into \(M\) sub-intervals of (equal) length, \(\Delta \tau = \tau_s/M\). Let \(\theta_m^{(p)}\) be the nondimensional temperature at the \(m^{th}\) node and after \(p\) time increments. Then the equation can be written in the implicit, (central) finite-difference form as follows:

\[
\frac{\theta_m^{(p+1)} - 2\theta_m^{(p+1)} + \theta_m^{(p+1)}}{\Delta \tau^2} = -\frac{1}{N} \frac{\partial Q_r}{\partial \tau} |_{\tau = \tau_m} \\
= \frac{1}{\tau_0^2} \frac{\theta_m^{(p+1)} - \theta_m^{(p)}}{\Delta \tau}, 1 < m \leq M + 1.
\]

When the values of \(\frac{\partial Q_r}{\partial \tau}\) at each node are known, and equation (3-8) is evaluated at the \(M + 1\) points, we have a system of \(M + 1\) linear algebraic equations for the \(\theta_m^{(p+1)}\), \(m = 1, 2, ... , M + 1\).

The solution for the nondimensionalized radiation intensity, \(\psi(\tau, \mu)\), is composed of two terms, one of which is the solution to homogenous form of the equation and the other the particular solution corresponding to the nonhomogenous source term, \((1 - \omega)\theta^4\), i.e.,

Solution by the \(P_N\) Method.
\[
\psi(\tau, \mu) = \psi_n(\tau, \mu) + \psi_p(\tau, \mu). \quad (3-12)
\]

Following Benassi et al. [25], the \(P_N\) approximation for the homogenous solution of the radiative transfer equation is written in the form.

\[
\psi_n(\tau, \mu) = \sum_{\ell=0}^{N} \frac{2\ell + 1}{2} P_{\ell}(\mu) \sum_{j=1}^{J} [A_j e^{-\tau_j/\ell} + (-1)^{\ell} B_j e^{-(\tau_j - \tau)/j}] g_{\ell}(\xi_j). \quad (3-13)
\]

The polynomials \(g_{\ell}(\xi)\) were introduced by Chandrashekhar [26], and obey the recurrence

\[
(\ell + 1) g_{\ell+1} = h_{\ell+1} g_{\ell}(\xi) - \ell g_{\ell-1}(\xi), \quad (3-14)
\]

with \(g_0(\xi) = 1\) and \(h_\ell = 2\ell + 1 - \omega_\ell^2\). We assume \(N\) is odd so that the eigenvalues \(\xi_j, j = 1, 2, ..., J\) (where \(J = (N + 1)/2\)), are the \(J\) positive zeros of \(g_{N+1}(\xi)\). Assume that the particular solution \(\psi_p(\tau, \mu)\) is known. Then we must simply determine the arbitrary constants \(\{A_j\}\) and \(\{B_j\}\) so that the approximate solutions given by equations (3-12) and (3-13) satisfy the boundary conditions (3-9). Therefore, substituting equations (3-12) and (3-13) into equations (3-9), multiplying the resulting equations by \(P_{N+1}(\mu)\) [21,26] and integrating over \(\mu\) from zero to one, the following system of linear algebraic equations is obtained [25].

\[
\sum_{\ell=0}^{N} \frac{2\ell + 1}{2} \sum_{j=1}^{J} \{(1 - (-1)^{\ell} \rho^j s_{a, \ell}) - 2(-1)^{\ell} \rho_0 s_{a, \ell} s_{a, \ell}\}
\times [A_j + (-1)^{\ell} B_j e^{-\tau_j/j}] g_{\ell}(\xi_j) = R_{1, \alpha}. \quad (3-15a)
\]
To compute the eigenvalues \{\xi_i\}, multiply equation (3-14) by \xi and rewrite the resulting equation, for \ell = 0, 2, 4, ..., as [25]

\[
\frac{\ell(\ell - 1)}{h_{\ell-1}h_{\ell+1}} g_{\ell-2}(\xi) + \frac{1}{h_{\ell}} \left[ \frac{(\ell + 1)^2}{h_{\ell+1}} + \frac{\xi^2}{h_{\ell-1}} \right] g_\ell(\xi)
\]

\[
+ \frac{(\ell + 2)(\ell + 1)}{h_{\ell+1}h_{\ell}} g_{\ell+2}(\xi) = \xi^2 g_\ell(\xi)
\]  

(3 - 16)
It follows that the squares of the $J$ positive zeros of $g_{n+1}(\xi)$ are the $J$ eigenvalues $\xi^2$ of the $J \times J$ tridiagonal matrix obtained from this equation, for $l = 0, 2, 4, \ldots, (N-1)$, and the truncation condition $g_{n+1}(\xi) = 0$. Having found the eigenvalues, the polynomials $g_{n}(\xi)$ can be computed.

Thus the system of linear algebraic equations (3-15) can be solved for $A_j$ and $B_j$.

Consistent with the $P_n$ approximation for the homogenous solution, a particular solution of the following form is assumed:

$$
\psi_{\mu}(\tau, \mu) = \sum_{n=0}^{N} \left( \frac{2n + 1}{2} \right) C_n(\tau) P_n(\mu). \quad (3 - 17)
$$

The coefficients $C_n(\tau)$ are evaluated as in Appendix A.

With these coefficients determined, the heat flux $Q_\mu(\tau)$ and its derivative $\frac{\partial Q_\mu(\tau)}{\partial \tau}$ are determined from equations (3-10 and 3-13)

$$
Q_\mu(\tau) = \frac{(1 - \omega)}{2} \sum_{j=1}^{J} A_j e^{-\tau/\xi_j} - B_j e^{-(\tau - \tau_0)/\xi_j} + \frac{1}{2} \int_{-1}^{1} \psi_{\mu}(\tau, \mu) \mu d\mu. \quad (3 - 18)
$$

The calculation procedure may now be summarized.

1. The initial temperature distribution, i.e. $\theta_m$, $1 \leq m < M + 1$, is known.

2. Find the eigenvalues, $\{\xi_j\}$, and the polynomials, $g_{n}(\xi_j)$ from equations (3-16) and (3-14), respectively.
3. Approximate $\theta^k(\tau)$ by a polynomial, $\theta^k(\tau) = \sum_{k=0}^{K} a_k \tau^k$.

4. Evaluate the constants $C_n(\tau)$ as shown in Appendix A.

5. Determine the particular solution, $\psi(\tau, \mu)$, from equation (3-17).

6. Determine the $P_n$ constants $\{A_j\}$ and $\{B_j\}$ by solving the equations (3-15).

7. Find the heat flux $Q_n(\tau)$ and its derivative $\frac{\partial Q_n(\tau)}{\partial \tau}$ from equation (3-18).

8. Use equation (3-11) to determine the new temperature distribution.

9. Repeat steps 3 to 6 until the end of the period of simulation.

The computation described above was performed using a computer program written in FORTRAN. A copy of the program and instructions for running it are provided in Appendix C. Sample runs and outputs can be found in Appendix D.

Test cases were run for various numbers of spatial increments, $M$. The effects on results are discussed in the next sub-section. Similarly, various orders of $P_n$ approximation were tried ($L$, the order of kernel-approximation for the phase function, should be less than $N$ of $P_n$). Also observed was the effect of varying the time increment. For the polynomial fit for the temperature profile, both a fourth-degree polynomial and hermite cubic interpolation were tried.
3.3 Results and Discussion

The results to be presented are for the case where $T_h = 300$ K and the constants in the equations (3-10) are as follows: $C_1 = 273$ K, $C_2 = 5$ K, $C_3 = 11$ K. The variation of $T_c$ over a 24 hour period is shown in Fig. 3. It represents the temperature change of the exterior surface of a wall during wintertime. The input properties used in the calculation are given in Table 3. The values specified in Table 3 are intended to be representative of those for the 0.0889 m (3.5 in.) thick commercially available R-11 fiberglass building insulations [16].

Table 3. Properties of the Building Insulation Used in Calculations

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L$ (m)</td>
<td>0.0889</td>
</tr>
<tr>
<td>$\varepsilon_h$</td>
<td>see tables 2 to 7</td>
</tr>
<tr>
<td>$\varepsilon_c$</td>
<td>see tables 2 to 7</td>
</tr>
<tr>
<td>$\rho_e$ (kg/m$^3$)</td>
<td>10</td>
</tr>
<tr>
<td>$C_e$ (J/kg K)</td>
<td>1000</td>
</tr>
<tr>
<td>$k$ (W/mK)</td>
<td>0.027</td>
</tr>
<tr>
<td>$\sigma_e$ (m$^{-1}$)</td>
<td>see tables 2 to 7</td>
</tr>
<tr>
<td>$\omega$</td>
<td>0.5</td>
</tr>
</tbody>
</table>

The definition of accumulated heat per unit area, ACQ, is

Solution by the $P_N$ Method.
\[ ACQ = \int_0^{t'} q dt, \quad \text{where} \quad q = -k \frac{\partial T}{\partial x} + q', \quad (3-20) \]

and \( t' \) in equation (3-20) is the end of the period in the calculation. Consistent with the finite-difference method used in the computations, the derivative in equation (3-20) was approximated as in equation (3-11).

Another equation that the system must satisfy is:

Heat in = Heat out + Heat stored

So, the heat storage could also account for a part of the difference between accumulated heats at the boundaries. Net heat stored in the insulation, until time \( t' \) would be given by:

\[ \text{Heat stored per unit area} = \int_{x=0}^{L} C_p \left[ \rho T(\xi = t', x) - T(\xi = 0, x) \right] dx \quad (3 - 21) \]

Tables 4 and 5 list the effects of varying \( N \) (the order of \( P_n \) approximation), \( L \) (the order of kernel approximation) and \( M \) (the number of spatial increments). The column under the title "Maximum Difference" is the maximum magnitude of \( \frac{ACQ_{\text{ACC}} - ACQ_{\text{ACH}}}{ACQ_{\text{ACC}}} \times 100 \) obtained for a simulation of 48 hours. When the order of approximation is increased, the results in general improve in that the differences between the accumulated heats decrease. For the cases of both \( \sigma_n = 400 \text{m}^{-1} \) and \( \varepsilon_n = \varepsilon_c = 1.0 \), and \( \sigma_n = 10,000 \text{m}^{-1} \) and \( \varepsilon_n = \varepsilon_c = 1.0 \), improvements occur until \( M \) and \( N \) reach a value of 99; after that the results do not change. However, the change in results, from the lowest order of approximation to the highest order, is not as much for the case
of $\sigma_e = 10,000\text{m}^{-1}$ as it is for $\sigma_e = 400\text{m}^{-1}$; "Maximum Difference" comes down from 3.92% to 0.74% for $\sigma_e = 400\text{m}^{-1}$, whereas it decreases from -1.05% to -0.83% for $\sigma_e = 10,000\text{m}^{-1}$. This is plausible since a higher extinction coefficient implies larger optical thickness, and lower orders of approximation should suffice [17].

The percent differences do not become zero, even on increasing the various orders of approximation after a certain point; this is because the temperature profile is approximated by a polynomial, which is used for determining the particular solution of intensity.
Table 4. Effect of Various Parameters in $P_N$ Model on the Results, $\sigma = 400m^{-1}, \epsilon_n = \epsilon_c = 1.0$

<table>
<thead>
<tr>
<th>N</th>
<th>L</th>
<th>M</th>
<th>Maximum Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>01</td>
<td>10</td>
<td>3.92%</td>
</tr>
<tr>
<td>01</td>
<td>01</td>
<td>20</td>
<td>2.85%</td>
</tr>
<tr>
<td>01</td>
<td>01</td>
<td>50</td>
<td>1.49%</td>
</tr>
<tr>
<td>01</td>
<td>01</td>
<td>99</td>
<td>0.80%</td>
</tr>
<tr>
<td>05</td>
<td>04</td>
<td>10</td>
<td>3.91%</td>
</tr>
<tr>
<td>05</td>
<td>04</td>
<td>20</td>
<td>2.82%</td>
</tr>
<tr>
<td>05</td>
<td>04</td>
<td>50</td>
<td>1.43%</td>
</tr>
<tr>
<td>05</td>
<td>04</td>
<td>99</td>
<td>0.78%</td>
</tr>
<tr>
<td>11</td>
<td>10</td>
<td>10</td>
<td>3.90%</td>
</tr>
<tr>
<td>11</td>
<td>10</td>
<td>20</td>
<td>2.78%</td>
</tr>
<tr>
<td>11</td>
<td>10</td>
<td>50</td>
<td>3.87%</td>
</tr>
<tr>
<td>11</td>
<td>10</td>
<td>99</td>
<td>0.76%</td>
</tr>
<tr>
<td>21</td>
<td>20</td>
<td>10</td>
<td>3.87%</td>
</tr>
<tr>
<td>21</td>
<td>20</td>
<td>20</td>
<td>2.78%</td>
</tr>
<tr>
<td>21</td>
<td>20</td>
<td>50</td>
<td>1.40%</td>
</tr>
<tr>
<td>21</td>
<td>20</td>
<td>99</td>
<td>0.75%</td>
</tr>
<tr>
<td>49</td>
<td>20</td>
<td>50</td>
<td>1.39%</td>
</tr>
<tr>
<td>49</td>
<td>20</td>
<td>99</td>
<td>0.74%</td>
</tr>
<tr>
<td>99</td>
<td>20</td>
<td>50</td>
<td>1.39%</td>
</tr>
<tr>
<td>99</td>
<td>20</td>
<td>99</td>
<td>0.74%</td>
</tr>
</tbody>
</table>

Solution by the $P_N$ Method.
Table 5. Effect of Various Parameters in $P_N$ Model on the Results, $\sigma_i = 10,000 \text{m}^{-1}$, $\epsilon_h = \epsilon_c = 0.1$

<table>
<thead>
<tr>
<th>$N$</th>
<th>$L$</th>
<th>$M$</th>
<th>Maximum Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>01</td>
<td>10</td>
<td>-1.05%</td>
</tr>
<tr>
<td>01</td>
<td>01</td>
<td>20</td>
<td>-0.99%</td>
</tr>
<tr>
<td>01</td>
<td>01</td>
<td>50</td>
<td>-0.96%</td>
</tr>
<tr>
<td>01</td>
<td>01</td>
<td>99</td>
<td>-0.93%</td>
</tr>
<tr>
<td>05</td>
<td>04</td>
<td>10</td>
<td>-1.03%</td>
</tr>
<tr>
<td>05</td>
<td>04</td>
<td>20</td>
<td>-0.98%</td>
</tr>
<tr>
<td>05</td>
<td>04</td>
<td>50</td>
<td>-0.95%</td>
</tr>
<tr>
<td>05</td>
<td>04</td>
<td>99</td>
<td>-0.92%</td>
</tr>
<tr>
<td>21</td>
<td>20</td>
<td>10</td>
<td>-1.00%</td>
</tr>
<tr>
<td>21</td>
<td>20</td>
<td>20</td>
<td>-0.95%</td>
</tr>
<tr>
<td>21</td>
<td>20</td>
<td>50</td>
<td>-0.92%</td>
</tr>
<tr>
<td>21</td>
<td>20</td>
<td>99</td>
<td>-0.90%</td>
</tr>
<tr>
<td>49</td>
<td>20</td>
<td>10</td>
<td>-0.98%</td>
</tr>
<tr>
<td>49</td>
<td>20</td>
<td>20</td>
<td>-0.92%</td>
</tr>
<tr>
<td>49</td>
<td>20</td>
<td>50</td>
<td>-0.89%</td>
</tr>
<tr>
<td>49</td>
<td>20</td>
<td>99</td>
<td>-0.87%</td>
</tr>
<tr>
<td>99</td>
<td>20</td>
<td>10</td>
<td>-0.94%</td>
</tr>
<tr>
<td>99</td>
<td>20</td>
<td>20</td>
<td>-0.89%</td>
</tr>
<tr>
<td>99</td>
<td>20</td>
<td>50</td>
<td>-0.87%</td>
</tr>
<tr>
<td>99</td>
<td>20</td>
<td>99</td>
<td>-0.83%</td>
</tr>
</tbody>
</table>

Solution by the $P_N$ Method.
The results from the $P_n$ model (for the best orders of approximation) are given in Tables 6, 7 and 8. Tables 6 and 7 are obtained using a fourth degree polynomial fit, whereas Table 8 used the hermite cubic interpolation. The results in these tables are given in terms of the following quantities:

Percent Difference \[ \frac{\text{ACC} - \text{ACH}}{\text{ACC}} \times 100 \]

Percent Storage \[ \frac{\text{Heat stored per unit area}}{\text{ACC}} \times 100 \]

On comparing the results for $\varepsilon_r = \varepsilon_e = 1.0$, the $P_n$ model seems to work appreciably better; for the case of a high extinction coefficient, even the $P_n$ model does not yield better results after a low order of approximation. Heat storage accounts for a part of the difference between accumulated heats only for the initial period of simulation, after which its contribution to the difference becomes negligible. This is because accumulated heat is a cumulative term but the boundary temperatures vary about some mean value, so the average temperature of insulation also remains in the neighbourhood of some quantity. This implies that the denominator in the definition of percent storage keeps increasing while the numerator does not increase beyond a certain value. Hence the percent storage becomes negligible for long periods of simulation.

### 3.4 Conclusion and Recommendations

In conclusion it can be stated that the $P_n$ method of solution is more accurate than the two-flux model, and that the large differences between heats accumulated at the
Table 6. Results from the $P_n$ Model for Accumulated Heat at the Boundaries, $a_t = 400 \text{m}^{-1}$, $\epsilon_h = \epsilon_c = 1.0$

<table>
<thead>
<tr>
<th>t (hrs.)</th>
<th>$ACH(J/m^2)$</th>
<th>$ACC(J/m^2)$</th>
<th>Percent Difference</th>
<th>Percent Stored</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$0.5287E+05$</td>
<td>$0.5252E+05$</td>
<td>-0.67</td>
<td>0.15</td>
</tr>
<tr>
<td>2</td>
<td>$0.1033E+06$</td>
<td>$0.1028E+06$</td>
<td>-0.46</td>
<td>0.13</td>
</tr>
<tr>
<td>3</td>
<td>$0.1516E+06$</td>
<td>$0.1510E+06$</td>
<td>-0.37</td>
<td>0.12</td>
</tr>
<tr>
<td>4</td>
<td>$0.1980E+06$</td>
<td>$0.1975E+06$</td>
<td>-0.28</td>
<td>0.11</td>
</tr>
<tr>
<td>5</td>
<td>$0.2433E+06$</td>
<td>$0.2428E+06$</td>
<td>-0.19</td>
<td>0.10</td>
</tr>
<tr>
<td>6</td>
<td>$0.2878E+06$</td>
<td>$0.2876E+06$</td>
<td>0.08</td>
<td>0.08</td>
</tr>
<tr>
<td>7</td>
<td>$0.3323E+06$</td>
<td>$0.3324E+06$</td>
<td>0.05</td>
<td>0.07</td>
</tr>
<tr>
<td>8</td>
<td>$0.3774E+06$</td>
<td>$0.3780E+06$</td>
<td>0.18</td>
<td>0.06</td>
</tr>
<tr>
<td>9</td>
<td>$0.4236E+06$</td>
<td>$0.4249E+06$</td>
<td>0.31</td>
<td>0.04</td>
</tr>
<tr>
<td>10</td>
<td>$0.4716E+06$</td>
<td>$0.4737E+06$</td>
<td>0.44</td>
<td>0.03</td>
</tr>
<tr>
<td>11</td>
<td>$0.5217E+06$</td>
<td>$0.5247E+06$</td>
<td>0.56</td>
<td>0.02</td>
</tr>
<tr>
<td>12</td>
<td>$0.5742E+06$</td>
<td>$0.5780E+06$</td>
<td>0.66</td>
<td>0.00</td>
</tr>
<tr>
<td>18</td>
<td>$0.9049E+06$</td>
<td>$0.9117E+06$</td>
<td>0.75</td>
<td>0.00</td>
</tr>
<tr>
<td>24</td>
<td>$0.1263E+07$</td>
<td>$0.1271E+07$</td>
<td>0.70</td>
<td>0.00</td>
</tr>
<tr>
<td>36</td>
<td>$0.1810E+07$</td>
<td>$0.1823E+07$</td>
<td>0.71</td>
<td>0.00</td>
</tr>
<tr>
<td>48</td>
<td>$0.2472E+07$</td>
<td>$0.2490E+07$</td>
<td>0.73</td>
<td>0.00</td>
</tr>
<tr>
<td>60</td>
<td>$0.3046E+07$</td>
<td>$0.3068E+07$</td>
<td>0.72</td>
<td>0.00</td>
</tr>
<tr>
<td>72</td>
<td>$0.3707E+07$</td>
<td>$0.3735E+07$</td>
<td>0.73</td>
<td>0.00</td>
</tr>
<tr>
<td>96</td>
<td>$0.4943E+07$</td>
<td>$0.4980E+07$</td>
<td>0.73</td>
<td>0.00</td>
</tr>
<tr>
<td>120</td>
<td>$0.6179E+07$</td>
<td>$0.6224E+07$</td>
<td>0.73</td>
<td>0.00</td>
</tr>
<tr>
<td>144</td>
<td>$0.7415E+07$</td>
<td>$0.7469E+07$</td>
<td>0.73</td>
<td>0.00</td>
</tr>
<tr>
<td>168</td>
<td>$0.8650E+07$</td>
<td>$0.8714E+07$</td>
<td>0.73</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Solution by the $P_n$ Method.
Table 7. Results from the \( P_n \) Model for Accumulated Heat at the Boundaries, \( \tau_s = 10.000 \text{m}^{-1}, \epsilon_n = \epsilon_c = 0.1 \)

<table>
<thead>
<tr>
<th>t (hrs.)</th>
<th>ACH (J/m²)</th>
<th>ACC (J/m²)</th>
<th>Percent Difference</th>
<th>Percent Stored</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.2890E+05</td>
<td>0.2860E+05</td>
<td>-0.83</td>
<td>0.19</td>
</tr>
<tr>
<td>2</td>
<td>0.5643E+05</td>
<td>0.5584E+05</td>
<td>-0.81</td>
<td>0.20</td>
</tr>
<tr>
<td>3</td>
<td>0.8271E+05</td>
<td>0.8187E+05</td>
<td>-0.79</td>
<td>0.19</td>
</tr>
<tr>
<td>4</td>
<td>0.1080E+06</td>
<td>0.1070E+06</td>
<td>-0.75</td>
<td>0.18</td>
</tr>
<tr>
<td>5</td>
<td>0.1325E+06</td>
<td>0.1314E+06</td>
<td>-0.68</td>
<td>0.16</td>
</tr>
<tr>
<td>6</td>
<td>0.1565E+06</td>
<td>0.1556E+06</td>
<td>-0.59</td>
<td>0.14</td>
</tr>
<tr>
<td>7</td>
<td>0.1806E+06</td>
<td>0.1799E+06</td>
<td>-0.37</td>
<td>0.12</td>
</tr>
<tr>
<td>8</td>
<td>0.2049E+06</td>
<td>0.2047E+06</td>
<td>-0.13</td>
<td>0.10</td>
</tr>
<tr>
<td>9</td>
<td>0.2300E+06</td>
<td>0.2302E+06</td>
<td>0.11</td>
<td>0.07</td>
</tr>
<tr>
<td>10</td>
<td>0.2560E+06</td>
<td>0.2568E+06</td>
<td>0.35</td>
<td>0.05</td>
</tr>
<tr>
<td>11</td>
<td>0.2831E+06</td>
<td>0.2848E+06</td>
<td>0.57</td>
<td>0.02</td>
</tr>
<tr>
<td>12</td>
<td>0.3117E+06</td>
<td>0.3141E+06</td>
<td>0.77</td>
<td>0.00</td>
</tr>
<tr>
<td>18</td>
<td>0.4927E+06</td>
<td>0.4972E+06</td>
<td>0.78</td>
<td>0.01</td>
</tr>
<tr>
<td>24</td>
<td>0.6740E+06</td>
<td>0.6796E+06</td>
<td>0.82</td>
<td>0.00</td>
</tr>
<tr>
<td>36</td>
<td>0.9857E+06</td>
<td>0.9937E+06</td>
<td>0.80</td>
<td>0.00</td>
</tr>
<tr>
<td>48</td>
<td>0.1348E+07</td>
<td>0.1359E+07</td>
<td>0.82</td>
<td>0.00</td>
</tr>
<tr>
<td>60</td>
<td>0.1660E+07</td>
<td>0.1673E+07</td>
<td>0.81</td>
<td>0.00</td>
</tr>
<tr>
<td>72</td>
<td>0.2022E+07</td>
<td>0.2039E+07</td>
<td>0.82</td>
<td>0.00</td>
</tr>
<tr>
<td>96</td>
<td>0.2696E+07</td>
<td>0.2718E+07</td>
<td>0.82</td>
<td>0.00</td>
</tr>
<tr>
<td>120</td>
<td>0.3370E+07</td>
<td>0.3398E+07</td>
<td>0.82</td>
<td>0.00</td>
</tr>
<tr>
<td>144</td>
<td>0.4044E+07</td>
<td>0.4078E+07</td>
<td>0.82</td>
<td>0.00</td>
</tr>
<tr>
<td>168</td>
<td>0.4718E+07</td>
<td>0.4757E+07</td>
<td>0.82</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Solution by the \( P_n \) Method.
Table 8. Results from the $P_n$ Model for Accumulated Heat at the Boundaries, using Hermite Cubic Interpolation.

\[ \sigma = 400 \text{m}^{-1}, \quad \varepsilon_n = \varepsilon_c = 1.0 \]

<table>
<thead>
<tr>
<th>$t$ (hrs.)</th>
<th>$\text{ACH (J/m}^2\text{)}$</th>
<th>$\text{ACC (J/m}^2\text{)}$</th>
<th>Percent Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4.4760E04</td>
<td>4.5428E04</td>
<td>-1.47</td>
</tr>
<tr>
<td>2</td>
<td>8.7550E04</td>
<td>8.9087E04</td>
<td>-1.73</td>
</tr>
<tr>
<td>3</td>
<td>1.2838E05</td>
<td>1.3113E05</td>
<td>-2.09</td>
</tr>
<tr>
<td>4</td>
<td>1.6759E05</td>
<td>1.7188E05</td>
<td>-2.50</td>
</tr>
<tr>
<td>5</td>
<td>2.0561E05</td>
<td>2.1177E05</td>
<td>-2.91</td>
</tr>
<tr>
<td>6</td>
<td>2.4299E05</td>
<td>2.5132E05</td>
<td>-3.33</td>
</tr>
<tr>
<td>7</td>
<td>2.8028E05</td>
<td>2.9105E05</td>
<td>-3.70</td>
</tr>
<tr>
<td>8</td>
<td>3.1806E05</td>
<td>3.3147E05</td>
<td>-4.04</td>
</tr>
<tr>
<td>9</td>
<td>3.5688E05</td>
<td>3.7304E05</td>
<td>-4.33</td>
</tr>
<tr>
<td>10</td>
<td>3.9720E05</td>
<td>4.1615E05</td>
<td>-4.55</td>
</tr>
<tr>
<td>11</td>
<td>4.3940E05</td>
<td>4.6106E05</td>
<td>-4.70</td>
</tr>
<tr>
<td>12</td>
<td>4.8372E05</td>
<td>5.0788E05</td>
<td>-4.76</td>
</tr>
<tr>
<td>18</td>
<td>8.2722E05</td>
<td>8.5546E05</td>
<td>-3.30</td>
</tr>
<tr>
<td>24</td>
<td>1.1761E06</td>
<td>1.1921E06</td>
<td>-1.35</td>
</tr>
</tbody>
</table>

Solution by the $P_n$ Method.
boundaries obtained by using the two-flux model are due to its low order of approximation.

The $P_n$ model for transient heat transfer through an absorbing, emitting, and anisotropically scattering medium, has been developed in this report. Now, calculations from the model can be compared with experimental results to test the model's validity.
References


23. Federeghi, F.D., "Vacuum Boundary Conditions for Spherical Harmonics Meth­

24. Pellaud, B., "Numerical Comparison of Different Types of Vacuum Boundary
Conditions for the $P_n$ Approximation," Transactions of the American Nuclear So­


Appendix A

Particular Solution for Radiation Intensity

Equation (3-9) is

\[ \mu \frac{\partial \psi}{\partial \tau} + \psi(\tau, \mu) = \frac{\omega}{2} \sum_{m=0}^{L} \beta_m \int_{-1}^{1} P_m(\mu') \psi(\tau, \mu') d\mu' + (1 - \omega) \theta^4. \]  

Representing \( \theta^4(\tau) \) by a \( K \)th degree polynomial, i.e. \( \theta^4(\tau) = \sum_{k=0}^{K} a_k \tau^k \), for \( K = 4 \) we get

\[ \theta^4(\tau) = a_0 + a_1 \tau + a_2 \tau^2 + a_3 \tau^3 + a_4 \tau^4 \]  

Also from equation (3-15),

\[ \psi(\tau, \mu) = \sum_{n=0}^{N} \left( \frac{2n + 1}{2} \right) C_n(\tau) P_n(\mu). \]  

Substituting equations (ii and iii) in equation (i) and using the recurrence relation for Legendre polynomials,
\[ (2n + 1)\mu P_n(\mu) = nP_{n-1}(\mu) + (n + 1)P_{n+1}(\mu) \]

the following equation is obtained.

\[
\sum_{n=0}^{N} \frac{dC_n}{d\tau} [nP_{n-1}(\mu) + (n + 1)P_{n+1}(\mu)]
\]

\[+ \sum_{n=0}^{N} (2n + 1)C_n(\tau)P_n(\mu) \]

\[= \omega \sum_{\ell=0}^{L} \beta_{\ell} P_{\ell}(\mu)C_{\ell}(\tau) + 2(1 - \omega) \sum_{m=0}^{4} a_m \tau^m \]

Transforming the indices and collecting terms we get,

\[
\sum_{n=0}^{N} \left( (n + 1) \frac{dC_{n+1}}{d\tau} + n \frac{dC_{n-1}}{d\tau} + (2n + 1)C_n(\tau) \right)P_n(\mu)
\]

\[+ \frac{dC_N}{d\tau}P_{N+1} - (N + 1) \frac{dC_{N+1}}{d\tau}P_N \]

\[= \omega \sum_{\ell=0}^{L} \beta_{\ell} P_{\ell}(\mu)C_{\ell}(\tau) + 2(1 - \omega) \sum_{m=0}^{4} a_m \tau^m \quad (\text{iv}) \]

Assume \( N \geq L \); define \( \beta_{\ell} = 0 \), \( \ell > L \) and operate on equation (iv) with \( \int_{-1}^{1} d\mu P_{k}(\mu) \), \( k < N \). Using the conditions of orthogonality for Legendre polynomials, i.e.
\[ \int_{-1}^{1} P_n(\mu')P_m(\mu')d\mu' = \begin{cases} 0 & \text{here, } m \neq n \\ \frac{2}{2m+1} & \text{here, } m = n \end{cases} \]

Equation (iv) becomes

\[
\frac{dC_k(\tau)}{d\tau} + (1 - \omega \beta_0)C_0(\tau) = (1 - \omega) \sum_{m=0}^{4} a_m \tau^m, \quad k = 0
\]

\[
(k + 1)\frac{dC_{k+1}}{d\tau} + k \frac{dC_{k-1}}{d\tau} + (2k + 1 - \omega \beta_k)C_k(\tau) = 0, \quad k = 1,2, \ldots
\]

\[ C_k(\tau) \text{ can be represented by a fourth degree polynomial } [2], \quad C_k(\tau) = \sum_{m=0}^{4} c_{km} \tau^m. \]

Varying \( k \) from 0 to 4 the following equations are obtained:

**For \( k = 0 \)**

\[ c_{10} + (1 - \omega \beta_0)c_{00} = 2(1 - \omega)a_0, \]

\[ 2c_{12} + (1 - \omega \beta_0)c_{01} = 2(1 - \omega)a_1, \]

\[ 3c_{13} + (1 - \omega \beta_0)c_{02} = 2(1 - \omega)a_2, \]

\[ 4c_{14} + (1 - \omega \beta_0)c_{03} = 2(1 - \omega)a_3, \]

\[ (1 - \omega \beta_0)c_{04} = 2(1 - \omega)a_4. \]

**For \( k = 1 \)**

\[ 2c_{21} + c_{01} + (3 - \omega \beta_1)c_{10} = 0, \]

\[ 4c_{22} + 2c_{02} + (3 - \omega \beta_1)c_{11} = 0, \]

\[ 6c_{23} + 3c_{03} + (3 - \omega \beta_1)c_{12} = 0. \]
\begin{align*}
8c_{23} + 4c_{04} + (3 - \omega_2) c_{13} &= 0, \\
5c_{14} &= 0.
\end{align*}

\textbf{k = 2}
\begin{align*}
3c_{31} + 2c_{11} + (5 - \omega_2) c_{20} &= 0, \\
6c_{32} + 4c_{12} + (5 - \omega_2) c_{21} &= 0, \\
9c_{33} + 6c_{13} + (5 - \omega_2) c_{22} &= 0, \\
12c_{34} + 8c_{14} + (5 - \omega_2) c_{23} &= 0, \\
c_{24} &= 0.
\end{align*}

\textbf{k = 3}
\begin{align*}
4c_{41} + 3c_{21} + (7 - \omega_3) c_{30} &= 0, \\
8c_{42} + 6c_{22} + (7 - \omega_3) c_{31} &= 0, \\
12c_{43} + 9c_{23} + (7 - \omega_3) c_{32} &= 0, \\
16c_{44} + 12c_{24} + (7 - \omega_3) c_{33} &= 0, \\
c_{34} &= 0.
\end{align*}

\textbf{k = 4}
\begin{align*}
4c_{31} + (9 - \omega_4) c_{40} &= 0, \\
8c_{32} + (9 - \omega_4) c_{41} &= 0, \\
12c_{33} + (9 - \omega_4) c_{42} &= 0, \\
16c_{34} + (9 - \omega_4) c_{43} &= 0, \\
c_{44} &= 0.
\end{align*}
Solving the equations for $c_{km}$ the following results for $C_i(\tau)$ are obtained.

\begin{align*}
C_4(\tau) &= c_{40}, \\
C_3(\tau) &= c_{30} + c_{31}\tau, \\
C_2(\tau) &= c_{20} + c_{21}\tau + c_{22}\tau^2, \\
C_1(\tau) &= c_{10} + c_{11}\tau + c_{12}\tau^2 + c_{13}\tau^3, \\
C_0(\tau) &= c_{00} + c_{01}\tau + c_{02}\tau^2 + c_{03}\tau^3 + c_{04}\tau^4.
\end{align*}

The coefficients $c_{km}$ are solved for in the following way (and order):

\begin{align*}
c_{04} &= \frac{2(1 - \omega)a_4}{1 - \omega\beta_0}, \\
c_{03} &= \frac{2(1 - \omega)a_3}{1 - \omega\beta_0}, \\
c_{13} &= \frac{-4c_{04}}{3 - \omega\beta}, \\
c_{22} &= \frac{-6c_{13}}{5 - \omega\beta_2}, \\
c_{31} &= \frac{-6c_{22}}{7 - \omega\beta_3}, \\
c_{40} &= \frac{-4c_{31}}{9 - \omega\beta_4}, \\
c_{02} &= \frac{2(1 - \omega)a_2 - 3c_{13}}{1 - \omega\beta_0}, \\
c_{11} &= \frac{-4c_{22} - 2c_{02}}{3 - \omega\beta_1}.
\end{align*}
\[
\begin{align*}
\c_{00} &= \frac{2(1 - \omega) a_0 - c_{11}}{1 - \omega \beta_0}, \\
c_{12} &= \frac{-3c_{03}}{3 - \omega \beta_1}, \\
c_{21} &= \frac{-4c_{12}}{5 - \omega \beta_2}, \\
c_{21} &= \frac{-3c_{21}}{7 - \omega \beta_3}, \\
c_{20} &= \frac{-3c_{31} - 2c_{11}}{5 - \omega \beta_2}, \\
c_{01} &= \frac{2(1 - \omega) a_1 - 2c_{12}}{1 - \omega \beta_0}, \\
c_{10} &= \frac{-2c_{21} - c_{01}}{3 - \omega \beta_1}.
\end{align*}
\]
Appendix B

Polynomial Fit for Temperature Profile

\[ \theta^4(\tau) \] needs to be represented as a polynomial for obtaining the coefficients, \( C_\alpha(\tau) \) in the particular solution of \( \psi(\tau, \mu) \). Here we discuss the approximation of \( \theta^4(\tau) \) by a fourth degree polynomial, i.e.

\[ \theta^4(\tau) = \sum_{k=0}^{4} a_k \tau^k \]

Five constants \( a_0, a_1, \ldots, a_4 \) need to be evaluated. So five conditions ought to be satisfied. This can be done in different ways; the following two schemes were considered for this report.

1. Matching the values and derivatives at the end-points gives 4 conditions and the fifth condition is obtained by matching the value at the mid-point.
2. Matching the values at the end points and at three other points within the medium, including the mid-point gives five conditions.

As illustrated in figure 4, the second scheme provides a better approximation to the values obtained by solving equation (3-8); hence it is used in the computer program.
Figure 4. Comparison of schemes for temperature profile fit, at time = 0.5 hrs. and emissivities = 1.0
Appendix C

FORTRAN PROGRAM

IMPLICIT REAL*8(A-H,O-Z)
CHARACTER*4 GEOM,CORR
REAL*8 LINS,LSCR
INTEGER VCPU,TCPU
PARAMETER(NPM = 100)
PARAMETER(NTM = 100)
COMMON/CONST/OM,P1S,P2S,P1D,P2D,TAO,EPS1,EPS2,HR,T1,T2,T0,N,LL
COMMON/ARRAYS/BETA(100),H(100),A0(100),A1(100),A2(100),S(100),G(100)
COMMON/QQ/T(100,3),BOM(100,100),WK(100),RHS(100),JFVT(100),R(100),
SV(100),E(100,100),U(100,100),Y(100,100)
COMMON/ORN/LMTIME,QOOT,EPST,TINIT,RHOP,RHOH,CH,LSCR,SIGE,COND,
LINS,CP0,TFAC,NO
DIMENSION TAU(NTM),AA(0:5),THETAC(NTM),THETA(NTM),RT(NTM)
CT(NPM,NTM),THETAP(NTM),TEMPER(NTM)
DIMENSION C(0:5,0:5),DQR(NTM),QR1(NTM),QR2(NTM),CI(5)
DUM(NTM,NTM)
READ(3,*) N,LL,OM,P1S,P2S,P1D,P2D,TAO,EPS1,EPS2,T1,T2,T0
READ(3,*) GEOM,R1,R2,CRI,MM,NPTE,EPST,ITMAX
READ(3,*) DTIM0,TMAX

C

C**************************************************************************************************************************

C

DELTEMP = (T2-T1) /NPTS
TEMPER(1) = T1
DO 4691 I = 2, NPTS + 1
4691 TEMPER(I) = TEMPER(I-1) + DELTEMP
ZNITEM = (TEMPER(1) + TEMPER(NPTS + 1)) * 0.5
DO 3691 I = 2, NPTS
3691 ZNITEM = ZNITEM + TEMPER(I)
ZNITEM = ZNITEM / NPTS
WRITE(6,7691) ZNITEM
7691 FORMAT(,'AVG. INITIAL TEMPERATURE =',F6.2)

READ(3,*) RHOP,CP0,LINS,COND,LSCR,SIGE,RHOH,CH
TFIX = T1
WRITE(5,167)

C

C ....... start clock
C

CALL TIMEON
C

C.......TAO is nondimensional length-------------------
C

TAO = SIGE*LINS
SIG = 5.6586D0
RC = (COND*SIGE)/(4.00*SIG**1.0)**1**3;
TFAC = RHOP*CP0*LINS**2/COND
C

C....... convert DTIME to nondimensional form...................
C

DTIME = DTIM0/TFAC
JFAC = 1.8031*DTIM0
HR = RHOH*CH*LSCR/RHOP/CP0/LINS
C

IF(GEOM.EQ.'SPHR') WRITE(4,121)

FORTRAN PROGRAM
IF(GEOM.EQ.'SPHR') STOP
WRITE(4,112)
WRITE(4,100) N,LL,OM,P1S,P2S,P1D,P2D,TAO
10002 WRITE(4,131) QDOT,SIGE,RC,CP0
121 FORMAT('/120X,'NO COMBINED-MODE PROGRAM FOR SPHERES YET'/)
C
C......NN represents true order of PN approximation............
C-----must increment by one to avoid zero subscripts---------
C
C CALL ERRSET(208,256,-1,0,0)

NN = N + 1
NN1 = NN + 1
L1 = LL + 1
PI = 3.14159265358979300

C
C..... Set up tau mesh.....
C
TAU(1) = 0.000
CT = TAO/DBLE(NPTS)
DO 2 I = 2,NPTS + 1
2 TAU(I) = (I-1)*CT
WRITE(5,166)(TAU(I),I = 1,NPTS + 1,20)
WRITE(4,122) T1,T2,TO
DO 4 I = 1,LL
4 READ(3,*) BETA(I)
IF(LL.GE.NN1) GO TO 55
DO 5 I = L1,NN1
5 BETA(I) = 0.0D0
55 CONTINUE
DO 6 I = 1,NN1
6 WRITE(4,101) BETA(I)
C
C.......compute HL's..........................
C
H(1) = 1.00-OM*BETA(1)
DO 7 L = 2,NN1
P = L
H(L) = 2^P - 1.0 - 0.0 * BETA(L)
7 CONTINUE
C
C CALL EIGEN TO SET UP EIGENVALUE MATRIX
C
CALL EIGEN
N2 = NN/2
DO 11 I = 1, N2
  T(I,1) = A1(I)
  T(I,2) = A0(I)
  T(I,3) = A2(I)
11 CONTINUE
C
NM = NPM
CALL IMTQL1(N2, A0, A2, IERR)
WRITE(4, 106) IERR
WRITE(4, 107)
DO 17 I = 1, N2
  EIG = DSQRT(A0(I))
  A2(I) = EIG
  IF(EIG.GT.1.0D0) WRITE(1, 1020) EIG
17 WRITE(4, 102) EIG
1020 FORMAT(1X, 1PD16.8)
C
C 'SAL' COMPUTES THE S'S
C
CALL SAL
C
C COMPUTE RATIOS OF G'S BY BACKWARD RECURSION: R'S
C
K = 0
10 K = K + 1
R(NN1) = 0.0D0
XI = A2(K)
IF(XI.LT.1.0D0) GO TO 22
DO 19 I = 1, N
J = NN1 - 1
RJ = DBLE(J - 1)
R(J) = RJ / (H(J)*XI - DBLE(J)*R(J + 1))
19 CONTINUE

C
C FINALLY, USE RATIOS TO COMPUTE G'S BY FORWARD RECURSION
C

G(1,K) = 1.00
G(2,K) = (1.00-OM)*XI
DO 21 I = 3, NN
   G(I,K) = RJ*G(I-1,K)
21 CONTINUE
GO TO 25

XI = A2(K)
G(1,K) = 1.00
G(2,K) = (1.00-OM)*XI
DO 24 I = 2, NN
   EL = I-1
   M = I + 1
   G(M,K) = (M(I)*XI*G(I,K)-EL*G(I-1,K))/((EI+1.00)
24 CONTINUE

IF(K.LT.N2) GO TO 18

C
C...... Set initial temperature profile in non dimensional form
C
T2 = T2/T1
T20 = T2
T1 = 1.00
THETO = T2
DO 28 I = 1, NPTS + 1
   THETAP(I) = (THETO-1.00)/TAO
   THETA(I) = (THETO-1.00)*(TAU(I)/TAO) + 1.00
   THETA0(I) = 1.000
   DQ(I) = 0.000
28 CONTINUE
WRITE(4,136)

FORTRAN PROGRAM
DO 29 I = 1, NPTS + 1
RT(I) = THETA(I) * TFIX
29 WRITE(4,123) TAU(I), RT(I)
TIME = 0.0D0
DTH = 0.0D0
M = MM
CALL TEMFIT(NTM, M, NPTS, OM, THETA, THETAP, AA, TAU, TAO, CI)
CALL CTAU(NPM, NTM, M, NPTS, TAU, C, CT, AA)
ISYS = 0
CALL SYSTEM(NPM, NTM, M, NPTS, RC, BOM, C, CT, RHS, TAU, AA, ISYS)
ISYS = 1
DO 31 J = 1, NN
DO 31 I = 1, NN
31 DUM(I, J) = BOM(I, J)
LDA = NPM
IT = 1
CALL LIN(ID, IT, IPVT, U, Y, SV, E, WK, BOM, LDA, NN, RHS, CRIT)
ACHR = 0.0D0
ACCR = 0.0D0
ACHC = 0.0D0
ACCC = 0.0D0
ACC = 0.0D0
ACH = 0.0D0
CALL QSLAB(RHS, CT, C, NPM, NTM, NPTS, QM0, DQR, TAU, AA, QR1, QR2, Q)
DTAU = TAU / DBLE(NPTS)
C
C Start time loop
C
C
WRITE(13, (1109) N, NPTS)
1109 FORMAT(/, 20X: N, 'PTS = ', I3)
WRITE(13, (1107))
& 8X, ' % DIFF', 3X, 'TC')
JTIM = 0
CONTINUE

TIME = TIME + DTIME
JTIM = JTIM + 1
DO 288 I = 1, NPTS + 1

288 THETA(I) = THETA(I)
CALL RCT(T20, DQR, THETA, DTH, RC, TIME, EPST, NTM, NPTS)
T2 = THETA(NPTS + 1)
IF(JTIM/JFAC*JFAC.EQ.JTIM) THEN
WRITE(4, 137) TIME*TFAC/3.603
DO 229 I = 1, NPTS + 1
RT(I) = THETA(I)*TFIX
ENDIF
I = 0
292 I = I + 1
IF(I.GT.NPTS + 1) GO TO 45
DIFF = ABSTHETA(I)-THETA(I))
IF(DIFF.GT.EPSTIM) GO TO 293
GO TO 292
293 CONTINUE
C
C Compute new solution to radiative transfer equation:
C
M = MM
C
... Fit polynomial to temperature profile
C
CALL TEMFIT(NTM, M, NPTS, OM, THETA, THETAP, AA, TAU, TAO, CI)
C
... derivative of theta at tau = 0
C
I = NPTS + 1
DTAU = TAU(NPTS + 1)-TAU(NPTS)
DTH = (THETA(2)-THETA(1))/DTAU
DTHN = (THETA(NPTS + 1)-THETA(NPTS))/DTAU
C

FORTRAN PROGRAM
C.... QCONDH and QCONDC are the conduction components
C       of heat fluxes
C
QCONDC=-COND*SIGE*TFIX*DTHN
QCONDH=-COND*SIGE*TFIX*DTH
C
C.... solve for C(tau) -----equation (3-17)
   CALL CTAU(NPM,NTM,M,NPTS,TAU,C,CT,AA)
C
C ... SET UP SYSTEM OF EQUATIONS FOR Pn COEFFICIENTS
C
M==MM
   CALL SYSTEM(NPM,NTM,M,NPTS,RC,BOM,C,CT,RHS,TAU,AA,ISYS)
   DO 295 J=1,NN
   DO 295 I=1,NN
295 BOM(I,J)=DUM(I,J)
C
C *** SOLVE EQUATIONS ***
C
IM=1
LDA=NPM
IT=1
   CALL LIN(ID,IT,IPVT,U,V,SV,E,WK,BOM,LDA,NN,RHS,CRIT)
C
   CALL QSLAB(RHS,CT,C,NPM,NTM,NPTS,QM0,DQR,TAU,AA,QR1,QR2,Q)
   Q1=QR2(1) *.40*SIG*(TFIX/1.D2)**4
   Z1Q=QR2(NPTS/2)*.40*SIG*(TFIX/1.D2)**4
   Q2=QR2(NPTS+1)*.40*SIG*(TFIX/1.D2)**4
   WRITE(8,1) Q1,Z1Q,Q2
1 FORMAT(3(F6.4,5X))

ACHR=ACHR+DTIME*TFAC*Q1
ACCR=ACCR+DTIME*TFAC*Q2
ACHC=ACHC+DTIME*TFAC*QCONDH
ACCC=ACCC+DTIME*TFAC*QCONDC

FORTRAN PROGRAM
ACC = ACCR + ACCC
ACH = ACHR + ACHC

TTR = THETA(KPTS + 1)*TFIX
DIFF = (QCONDH + Q1 - QCOND - Q2)/(QCONDH + Q1)*1.D2
WRITE(7,2099) TIME*TFAC/6.1,TTR,QCONDH,Q1,QCOND,Q2,DIFF
2099 FORMAT(2F6.1,3(F10.5))
DIFF = (ACH - ACC)/ACC*1.D2
RTIME = TIME*TFAC/3.6D3
IF(JTIM/JFAC*JFAC.EQ.JTIM)
$WRITE(13,1108) RTIME,Q1,Q2,ACH,ACC,DIFF,TTR
1108 FORMAT(F7.2,4(D12.5,1X),F6.2,2X,F6.2)
C
AVGT = TFIX*(THETA(1) + THETA(NPTS + 1))*5D0
DO 5891 I = 2,NPTS
5891 AVGT = AVGT + TFIX*THETA(I)
AVGT = AVGT/NPTS
CUMINC = AVGT - ZNITEM
HEATIN = CUMINC*LINS*CP0

C...Write out the temperature profile

IF(JTIM/JFAC*JFAC.EQ.JTIM) THEN
WRITE(5,165) TIME*TFAC/3.6D3,(TFIX*THETA(I),I = 1,NPTS + 1,20)
WRITE(6,6691) TIME*TFAC/3.6D3,ACH,ACC, &
(ACC - ACH)/ACC*100, HEATIN/ACC*100
6691 FORMAT(2F6.2,'(',E10.4,2(' ',F1.2))
165 FORMAT(2X,F7.2,(2X,F10.4))
166 FORMAT(6X,'TIME,6(2X,F7.2)/2X,'HOURS',19X, 
$TEMPERATURE (K)')
167 FORMAT(9X,6(6X,'TAU'))
ENDIF
C...
IF(TIME*TFAC.LT.TMAX) GO TO 289
45 CONTINUE

FORTRAN PROGRAM
WRITE(4,124)
WRITE(9,124)
DO 50 I = 1,NPTS + 1
   TT = TFIX*THETA(I)
   IF(I.EQ.1) TMAX = TT
   WRITE(4,132) TAU(I),TT
   50 WRITE(9,132) TAU(I),TT
1106 FORMAT(/,8X,'COMPUTE TIME == ',F6.2,' CPU SECS')
C***************************************************************
C Write runtime:
C
C CALL TIMDAT(DATE,TCD,VCPU,NTCPU)
C TTIM = DBLE(NTCPU-TCPU)/1000.0D0
C
C CALL TIMECK(ITIM);
CTTIM = DBLE(ITIM)/100.0D0
WRITE(1,1106) TTIM
WRITE(9,1106) TTIM
WRITE(4,1106) TTIM
STOP
100 FORMAT(15,15,6(F8.3,2X))
101 FORMAT(D15.6)
102 FORMAT(1X,5(1PD16.8,1X))
106 FORMAT(/,3X,'ERROR PARAMETER FOR EIGENVALUE ROUTINE:',I3)
107 FORMAT(/,5X,'AND THE EIGENVALUES ARE...')
112 FORMAT(/,4X,'N','4X','L','4X','OM','8X','P15','7X','P2S','7X,
$','P1D','7X','P2D','5X','TAU0')
122 FORMAT(/,10X,'T1 == ','F9.2','T2 == ','F9.2','T0 == ','F9.2)
123 FORMAT(5X,F9.3,2X,F10.3,2X,D15.4,2X,D15.8)
124 FORMAT(/,12X,'TAU','17X','TEMPERATURE')
131 FORMAT(/,3X,'QOOT = ','D10.3','SIGE = ','F7.3','N == ','$','F6.2','CP = ','F9.2)
132 FORMAT(5X,F11.2,2X,F15.3)
136 FORMAT(/,3X,'Initial Temperature Profile')
137 FORMAT(/,3X,'Time == ','F8.2',' hours')
SUBROUTINE EIGEN
IMPLICIT REAL*8(A-H,O-Z)
COMMON/CONST/OM,P1S,P2S,P1D,P2D,TAO,EPS1,EPS2,HR.
$T1,T2,T0,N,L$
COMMON/ARRAYS/BETA(100),H(100),A0(10),A1(100),
$A2(100),S(100,100),G(100,100)$

N2=(N+1)/2
A1(1)=0.0D0
A0(1)=1.0D0/H(1)/H(2)
A2(1)=2.0D0/H(1)/H(2)
DO 10 I=2,N2
L=2**I-2
RL=L
LS=L+1
A0(I)=(RL+1)**2/H(LS+1)+RL**2/H(LS-1)/H(LS)
A2(I)=(RL+1)*(RL+2)/H(LS)/H(LS+1)
IF(I.EQ.1) GO TO 10
A1(I)=RL*(RL-1)/H(LS-1)/H(LS)
10 CONTINUE
RETURN
END

SUBROUTINE SAL
IMPLICIT REAL*8(A-H,O-Z)
COMMON/CONST/OM,P1S,P2S,P1D,P2D,TAO,EPS1,EPS2,HR,T1,T2,
$T0,N,L$
COMMON/ARRAYS/BETA(100),H(100),A0(100),A1(100),A2(100),
$S(100,100),G(100,100)$
N1 = N + 1
NN = N1/2

C WRITE(4,102)
DO 10 M = 1, NN
DO 10 L = 1, NN
10 S(M,L) = 0.00D0
S(1,1) = 0.50D0
S(1,2) = 1.00/3.00
DO 15 M = 1, NN
L = 2*M
ALPH = M - 1
15 S(M,L) = 1.00/(4*ALPH + 3.00)
DO 20 M = 2, NN
ALPH = M - 1
MN = M - 1
20 S(M,N) = (2*ALPH - 1.00)*/(ALPH + 1.00)*S(MN,1)/2.00
DO 30 M = 1, NN
DO 30 L = 1, NN
EL = L - 1
ALPH = M - 1
IN = L + 2
F1 = (1.00 - EL - 2*ALPH)/(4.00 + EL + 2*ALPH)
G1 = EL + 1.00 - 2*ALPH
IF(G1.EQ.0.00D0) GO TO 30
F2 = (EL + 1.00)/G1
F3 = (EL + 2.00 + 2*ALPH)/(EL + 2.00)
S(M,N) = F1*F2*F3*S(M,L)
30 CONTINUE
C DO 45 M = 1, NN
C 45 WRITE(4,101) (S(M,L), L = 1, NN)
RETURN
C 101 FORMAT(1X,7(F10.4,1X))
C 102 FORMAT(5X,'THE S-MATRIX')
END
C
C-----------------------------
SUBROUTINE QSLAB(RHS,CT,C,NPM,MP,NPTS,QM0,DQR,TT,AA,
$QR1,QR2,Q)
IMPLICIT REAL*A-H,O-Z
COMMON/CONST/OM,P1S,P2S,P1D,P2D,TA0,EPS1,EPS2,HR,T1,
$T2,T0,N,LL
COMMON/ARRAYS/BETA(100),H(100),A0(100),A1(100),
$A2(100),S(100,100),G(100,100)
DIMENSION RHS(100),CT(NPM,MP),AA(0:5),DQR(MP),TT(MP),
$QR1(MP),QR2(MP)
DIMENSION C(0:5,0:5)
SIG=5.669D0
PI = 3.14159265356979300
NN=N+1
N2 = NN/2
DT = TA0/DBLE(NPTS)
NP = NPTS
DO 40 I = 1,NPTS+1
TAU = DBLE(I-1)*DT
SUMJ=0.0D0
DO 10 J = 1,N2
XI = A2(J)
TERM1 = RHS(J)*EXP(-TAU/XI)
TERM2 = -RHS(N2 + J)*EXP(-(TAO-TAU)/XI)
10 SUMJ = SUMJ + (TERM1 + TERM2)/XI
C
C ... Compute contribution of particular solution
C
PART = CT(2,I)
HOMO = (1.0D0-OM)*SUMJ
QR2(I) = 2.0D0*PI/(HOMO+PART)
C*****************************************************************************
C 22 July '87 : Experiment to check normalization
C QR2(I) = QR2(I)/(4.0D0*PI)
C*****************************************************************************
40 CONTINUE
C
C ... compute the partial derivative of qr for use in RCT
C
DO 45 I = 1,NPTS + 1
TAU = TT(I)
SUM = 0.0D0
DO 43 J = 1,N2
XI = A2(J)
TERM1 = RHS(J)*DEXP(-TAU/XI)
TERM2 = RHS(N2 + J)*DEXP(-(TAU-TAU)/XI)
43 SUM = SUM + TERM1 + TERM2
DQR(I) = 2.0D0*PI*DUM + C(1,1) + 2.0D0*C(1,2)*TAU
& + 3.0D0*C(1,3)*TAU**2
C**************************************************************************
C 22 July '87 : Experiment to check normalization
DQR(I) = DQR(I)/(4.0D0*PI)
C**************************************************************************
45 CONTINUE
C
DO 47 I = 1,NPTS + 1
C 47 WRITE(7,118) I,TAU(I),DQR(I)
RETURN
END
C
C**************************************************************************
C Computes R1-alpha and R2-alpha for RHS of Pn eqs.
C
SUBROUTINE RALPHA(NPM,NTM,M,N,PNTS,J,RC,R1,R2,CT)
IMPLICIT REAL*8(A-H,O-Z)
COMMON/CONST/OM,P1S,P2S,P1D,P2D,TAO,EPS1,EPS2,HR,T1,
$ T2,T0,N,LL
COMMON/ARRAYS/BETA(100),H(100),A0(100),A1(100),
$ A2(100),S(100,100),G(100,100)
DIMENSION CT(NPM,NTM)
PI = 3.141592653589793D0
NP = N + 1

FORTRAN PROGRAM
MP = NPTS + 1
C MP = NPTS
C
C ... Compute first terms
C
SUM1 = 0.0D0
SUM2 = 0.0D0
DO 10 I = 1, NP
  K = I - 1
  TERM1 = DBLE((-1)**K*(2*K + 1))*CT(I,1)*S(I,J)/2.D0
  TERM2 = DBLE(2*K + 1)*CT(I,MP)*S(I,J)/2.D0
  SUM1 = SUM1 + TERM1
10 SUM2 = SUM2 + TERM2
C
IF(RCLT.1.0D0) GO TO 11
  R1 = (EPS1*T1**4 + 2.0D0*P1D*SUM1)*S(J,1)
  R2 = (EPS2*T2**4 + 2.0D0*P2D*SUM2)*S(J,1)
  GO TO 12
11 R1 = (EPS1*RC**4 + 2.0D0*P1D*SUM1)*S(J,1)
  R2 = (EPS2*(T2*RC)**4 + 2.0D0*P2D*SUM2)*S(J,1)
12 CONTINUE
C
C ... Compute second terms
C
SUM1 = 0.0D0
SUM2 = 0.0D0
DO 20 I = 1, NP
  K = I - 1
  MM = (-1)**K
  TERM1 = (1.0D0-MM*P1S)*DBLE(2*K + 1)*CT(I,1)*S(I,J)/2.D0
  TERM2 = (DBLE(MM)*P2S)*DBLE(2*K + 1)*CT(I,MP)*S(I,J)/2.D0
  SUM1 = SUM1 + TERM1
20 SUM2 = SUM2 + TERM2
C
C ... Combine
C
FORTRAN PROGRAM
R1 == R1-SUM1
R2 == R2-SUM2
C WRITE(7,1002) J,R1,R2
C1002 FORMAT(11X, J == ',12; R1 == ',020.13,' R2= ',020.13}
RETURN
END
C
C-------------------------------
C
C Un...Uses Gauss Elimination or SVD to Solve a Linear System...
C
SUBROUTINE LIN(ID,IT,IPVT,U,V,S,E,WORK,A,LDA,N,B,CRIT)
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION A(LDA,1),B(1)
DIMENSION IPVT(1)
DIMENSION U(LDA,1),V(LDA,1),S(1),E(1),WORK(1)
1001 FORMAT(25X:INFO not equal to zero in LIN')
1002 FORMAT(22X,'incorrect use at 10 and/or IT In LIN')
1003 FORMAT(14X,13,' SINGULAR VALUE(S) LESS THAN ',1PD8.1,
$'
1004 FORMAT(5X,13,1X,1PD12.5)
1005 FORMAT(5X,'THE SINGULAR VALUES',/)
MAX = N
IF((ID.NE.1).AND.(ID.NE.2).AND.(IT.NE.1).AND.(IT.NE.2))THEN
WRITE(*,1002)
STOP
ENDIF
IF(D.EQ.2)THEN
IF(IT.NE.2)THEN
JOB = 11
WRITE(2,1005)
CALL DSVDC(A,LDA,N,N,S,E,U,LDA,V,LDA,WORK,JOB,INFO)
DO 99 I = 1,N
99 WRITE(2,1004) I,S(I)
IF(INFO.NE.0)THEN
WRITE(*,1001)
STOP
ENDIF
INDEX = 0
VW = S(1)
DO 10 I = 1, N
   CN = S(I)/VW
   IF(CN.LT.CRIT) THEN
      INDEX = INDEX + 1
      S(I) = 0.0
   ELSE
      S(I) = 1.00/S(I)
   ENDIF
10 CONTINUE
IF(INDEX.GT.MAX) STOP
ENDIF
DO 10 I = 1, N
   WORK(I) = 0.D0
20 CONTINUE
DO 40 I = 1, N
   DO 30 J = 1, N
      WORK(I) = WORK(I) + U(J,J)*B(J)*S(I)
30 CONTINUE
40 CONTINUE
DO 50 I = 1, N
   B(I) = 0.D0
50 CONTINUE
DO 70 J = 1, N
   DO 60 I = 1, N
      B(I) = B(I) + V(I,J)*WORK(J)
60 CONTINUE
70 CONTINUE
ELSE
   JOB = 0
   IF(IT.EQ.1) THEN
CALL DGEF(A,LDA,N,IPVT)
C CALL DGECO(A,LDA,N,IPVT,RCOND,WORK)
C WRITE(1,1111) RCOND
C WRITE(4,1111) RCOND
C1111 FORMAT(/,5X,'CONDITION #=',D10.3,/)  
CALL DGES(A,LDA,N,IPVT,B)
C CALL DGESL(A,LDA,N,IPVT,B,JOB)
ELSE  
CALL DGES(A,LDA,N,IPVT,B)
C CALL DGESL(A,LDA,N,IPVT,B,JOB)
ENDIF
ENDIF
END

C
C-----------------------------------------------
C
SUBROUTINE RCT(T20,DQR,THETA,DTH,RC,T,EPST,NTM,NPTS)
IMPLICIT REAL*8(A-H,O-Z)
REAL*8 LSCR,LNS
COMMON,CONST/OM,P1S,P2S,P1D,P2D,TAO,EPST1,EPST2,
$HR,T1,T2,TO,N,LL
COMMON/ORNUOTIME,QOOT,EPST,TINIT,RHOP,RHOH,CH,
$LSCR,SIGE,COND,LINS,CPO,TFAC,NOP
DIMENSION TH(100),C1(100),C2(100),C3(100),RHS(100)
DIMENSION P(100)
DIMENSION THETA(NTM),DQR(NTM)

PI =3.141592653589793D0
DT =DTIME
C
***************
M =NPTS
DEL =TAO/DBLE(NPTS)
THNEW =1.0D0
DO 10 I =1,NPTS
   DQR(I)=DQR(I)**DE**2/RC
10 CONTINUE
DO 9 I=1,NPTS-1
RHS(I)=THETA(I+1)
9 TH(I)=RHS(I)

C
C ... set up tridiagonal coefficients
C
D1 = T20
D2 = 1.1D0/3.02
TT = T*TFAc/3.6D3
C JTIM = TT/2.00
C IF(JTIM.LT.1) GO TO 6
ITIM = TT/1.2D1
IF(ITIM/2.EQ.ITIM) D2 = 5.0D0/3.02
THMP1 = D1 + D2*DSIN(PI*TT/1.2D1)
C GO TO 6
C 6 CONTINUE
C
C ************************************************************
C*************** 29 JULY 87: EXPERIMENT ***************
C*************** TO DETERMINE SS PROFILE ***************
C THMP1 = T20
C 6 CONTINUE
C
C ************************************************************
C DO 12 I=1,NPTS-1
C
P(I)=(DEL**2)/(TAO**2*DT)
C1(I)=1.0D0
C2(I)=-(2.0D0+P(I))
12 C3(I)=1.0D0
C
C ... r. h. s.
C
RHS(1) = DQR(2)-P(1)*TH(1)-THNEW
RHS(M-1) = DQR(M)-P(M-1)*TH(M-1)-THMP1
DO 23 I=2,M-2
RHS(I) = DQR(I+1)-P(I)*TH(I)

FORTTRAN PROGRAM
23 CONTINUE
C  DO 99 I = 1, M - 1
C 99 WRITE(4,999) C1(I), C2(I), C3(I), RHS(I)
C 999 FORMAT(3X,3(D12.5,1X),5X,D12.5)
C
C ... solve
C
MSIZE = M - 1
CALL DGTSL(MSIZE, C1, C2, C3, RHS, INFO)
C
C Place TH in THETA and return to MAIN...
C
DO 24 K = 1, M - 1
THETA(K + 1) = RHS(K)
24 TH(K) = RHS(K)
THETA(1) = THNEW
THETA(M + 1) = THM1
RETURN
END
C\--------------------
C LINPACK ROUTINE FOR TRIDIAGONAL MATRICES
C
SUBROUTINE DGTSL(N, C, D, E, B, INFO)
IMPLICIT REAL*8(A-H,O-Z)
INTEGER N, INFO
DIMENSION C(100), D(100), E(100), B(100)
INTEGER K, KB, KP1, NM1, NM2
C BEGIN BLOCK PERMITTING ... EXITS TO 100
C
INFO = 0
C(1) = D(1)
NM1 = N - 1
IF (NM1 .LT. 1) GO TO 40
D(1) = E(1)
\( E(1) = 0.000 \)
\( E(N) = 0.000 \)

\[ \text{DO 30 } K = 1, NM1 \]
\[ KP1 = K + 1 \]

\[ \text{FIND THE LARGEST OF THE TWO ROWS} \]

\[ \text{IF (ABS(C(KP1)) .LT. ABS(C(K))) GO TO 10} \]

\[ \text{INTERCHANGE ROW} \]

\[ T = C(KP1) \]
\[ C(KP1) = C(K) \]
\[ C(K) = T \]
\[ T = D(KP1) \]
\[ D(KP1) = D(K) \]
\[ D(K) = T \]
\[ T = E(KP1) \]
\[ E(KP1) = E(K) \]
\[ E(K) = T \]
\[ T = B(KP1) \]
\[ B(KP1) = B(K) \]
\[ B(K) = T \]

\[ \text{10 CONTINUE} \]

\[ \text{ZERO ELEMENTS} \]

\[ \text{IF (C(K) .NE. 0.000) GO TO 20} \]
\[ \text{INFO} = K \]
\[ \text{............EXIT} \]
\[ \text{GO TO 100} \]

\[ \text{20 CONTINUE} \]
\[ T = -C(KP1)/C(K) \]
\[ C(KP1) = D(KP1) + T*D(K) \]
\[ D(KP1) = E(KP1) + T*E(K) \]
E(KP1) = 0.0D0
B(KP1) = B(KP1) + T*B(K)
30  CONTINUE
40  CONTINUE
IF (C(N) .NE. 0.0D0) GO TO 50
   INFO = N
   GO TO 90
50  CONTINUE
C
C BACK SOLVE
C
NM2 = N - 2
B(N) = B(N)/C(N)
IF (N .EQ. 1) GO TO 80
B(NM1) = (B(NM1) - D(NM1)*B(N))/C(NM1)
IF (NM2 .LT. 1) GO TO 70
DO 60 KB = 1, NM2
   K = NM2 - KB + 1
   B(K) = (B(K) - D(K)*B(K+1) - E(K)*B(K+2))/C(K)
60  CONTINUE
70  CONTINUE
80  CONTINUE
90  CONTINUE
100 CONTINUE
C
RETURN
END
C
C---------------------------------------------
C
SUBROUTINE SYSTEM(NPM,NTM,M,NPTS,RC,BOM,C,CT,RHS,STAU,A,ISYS)
IMPLICIT REAL*8(A-H,O-Z)
COMMON/CONST/OM,P1S,P2S,P1D,P2D,TAO,EPS1,EPS2,HR,$T1,T2,T0,N,LL
COMMON/ARRAYS/BETA(100),H(100),A0(100),A1(100),
$A_2(100),s(100,100),G(100,100)$
DIMENSION BOM(100,100),RHS(100)
DIMENSION CT(NPM,NTM),C(1:NTM,O:NPM,0:5),A(4,NPTS),
$TAU(NTM)$
IF(ISYS.NE.O) GO TO 88
SIG=5.669D0
PI=3.141592653589793D0
N1=N+1
NN=N1/2
DO 33 I=1,NN
DO 33 J=1,NN
SUM1=0.0D0
SUM2=0.0D0
XI=A2(J)
DO 29 L=1,N1
SW=DBLE((-1)**(L-1))
D1=(1.0D0-SW*PI5)*S(I,L)
D1=D1-2.0D0*SW*PI5*2.0D0*S(I,L)*S(I,1)
D1=D1*DBLE(2*L-1)/2.0D0
SUM1=SUM1+D1*G(I,J)
SUM2=SUM2+D1*G(I,J)*EXP(-TAU/XI)
29 CONTINUE
BOM(I,J)=SUM1
BOM(I,J+NN)=SUM2
SUM1=0.0D0
SUM2=0.0D0
DO 31 L=1,N1
SW=DBLE((-1)**(L-1))
D2=(1.0D0-SW*PI5)*S(I,L)
D2=D2-2.0D0*SW*PI5*2.0D0*S(I,L)*S(I,1)
D2=D2*DBLE(2*L-1)/2.0D0
SUM1=SUM1+D2*G(I,J)*S*EXP(-TAU/XI)
31 SUM2=SUM2+D2*G(I,J)
BOM(I+NN,J)=SUM1
BOM(I+NN,J+NN)=SUM2
33 CONTINUE

FORTRAN PROGRAM
DO 34 I = 1, NN
    CALL RALPHA(NPM, NTM, M, NPTS, I, RC, R1, R2, CT)
C    WRITE(4, 1029) I, R1, R2
C1029 FORMAT(3X, 'R1 = ', D24.16; ' R2 = ', D24.16)
    RHS(I) = R1
    RHS(I + NN) = R2
34 CONTINUE
C    WRITE(4, 100)
C    DO 35 I = 1, N1
C    35 WRITE(4, 101) (BOM(I, J), J = 1, N1), RHS(I)
C?????????????????????????????????????????????????C
C    WRITE(4, 4102)
C    DO 35 I = 1, N
C    DO 35 J = 1, N
C    35 IF(DABS(BOM(I, J)) GT 1.D1) WRITE(4, 4101) I, J, BOM(I, J)
C4102 FORMAT(I, 5X, 'THE COEFFICIENT MATRIX...')
C4101 FORMAT(6D10.3, 3X, D10.3)
C
RETURN
100 FORMAT(I, 5X, 'THE COEFFICIENT MATRIX...')
101 FORMAT(6D10.3, 3X, D10.3)
END

C-----------------------------------------------
CComputes polynomial expansion of Pn coefficients in particular solution.
C
SUBROUTINE CTAU(NP, MP, M, NPTS, TAU, C, CT, A)
IMPLICIT REAL*8(A-H, O-Z)
COMMON/CONST/OM, P1S, P2S, P1D, P2D, TAO, EPS1, EPS2, T1, T2, T0, RC, N, LL
COMMON/ARRAYS/BETA(100), H(100), A0(100), A1(100), A2(100), S(100, 100)
2, G(100, 100)
DIMENSION CT(NP, MP), C(0:5, 0:5), A(0:4), TAU(MP)
C
C... determines c(km) --- APPENDIX B, equation
C
FORTRAN PROGRAM
WM = 1.00-0M
CB0 = 1.00-0M*BETA(1)
CB1 = 3.00-0M*BETA(2)
CB2 = 5.00-0M*BETA(3)
CB3 = 7.00-0M*BETA(4)
CB4 = 9.00-0M*BETA(5)

C----------------------------
C(1,3) = -8.00*WM*A(4)/CB0/CB1
C(1,2) = -6.00*WM*A(3)/CB0/CB1
C(1,1) = -1.92*D2*WM*A(4)/CB0/CB1
C(1,1) = -4.00*WM*A(2)/CB0/CB1
C(1,1) = -4.80*D1*WM*A(4)/(CB0+CB1)**2
C(0,0) = (2.00+WM*A(0)-C(1,1))/CB0
C(0,1) = (2.00*WM*A(1)-2.00*CB0/CB1
C(0,2) = (2.00*WM*A(2)-3.00*C(1,3))/CB0
C(0,3) = 2.00*WM*A(3)/CB0
C(0,4) = 2.00*WM*A(4)/CB0
C(2,2) = -6.00*CB0/CB2
C(3,1) = -6.00*CB1/CB3
C(4,0) = -3.00*CB1/CB4
C(2,1) = -4.00*C(1,2)/CB2
C(3,0) = -3.00*C(2,1)/CB3
C(2,0) = -(3.00*C(3,1)+2.00*C(1,1))/CB2
C(1,0) = (-2.00*C(2,1)-C(0,1))/CB1

DO 5 I = 1,NPTS+1
DO 5 J = 1,5
CT(J,1) = 0.00D0
DO 6 J = 1,5
   CT(J,1) = C(J-1,0)
CONTINUE
DO 10 I = 2,NPTS+1
   T = TAU(I)
   DO 8 K = 1,5
      DO 8 J = 0,4
         L = K-1
      CT(K,I) = CT(K,I) + C(L,J)*T**J

FORTRAN PROGRAM
CONTINUE
C WRITE(4,1000)
C DO 40 I = 1,NPTS + 1
C 40 WRITE(4,1001) (CT(L,I),L = 1,2)
C1001 FORMAT(4(D14.7))
C1000 FORMAT(/.10X:'THE CTAUS')
RETURN
END
C
C-----------------------
C TEM FIT . Fits polynomial to temperature profile
C
SUBROUTINE TEMFIT(NTM,M,NP,OM,THETA,THETAP,A,TAU,TAO,C)
IMPLICIT REAL*A-H,O-Z)
DIMENSION CI(5),A(0:4),THETA(NTM),THETAP(NTM),TAU(NTM)
DIMENSION COEF(5,5),RHS(5),IPVT(5)
DIMENSION U(5,1),V(5,1),S(5),E(5),WORK(5)
NPTS = NP
C
C ... set up system for polynomial coefficients
C
A(0) = THETA(1)**4
DT = TAO/DBLE(NP)
TH1P = (THETA(2)-THETA(1))/DT
THNP = (THETA(NP+1)-THETA(NP))/DT
C TH1P = (4.DO*THETA(2)-THETA(1))/2.DO*DT
C THNP = (3.DO*THETA(1)+THETA(1-2)-4.DO*THETA(1-1))/2.DO*DT
C A(1) = 4.DO*THETA(1)**3*TH1P
C
C!!!!!!!!!!!! OLD SCHEME BASED ON CHEBYSHEV POINTS !!!!!!!!!!
C DO 10 I = 1,5
C J = I
C 2 J = J + 1
C FRAC = CI(I)-TAU(J)
C IF(FRAC,GRT.0.000) GO TO 2
C J = J-1

FORTRAN PROGRAM
C FRAC = CI(J-I)*TAU(J)
C
C FF = THETA(J) + (THETA(J+1) - THETA(J))/(TAU(J+1) - TAU(J)) * FRAC
C
C WRITE(7,1099) I,J,FF
C
C1099 FORMAT(2X,13,2X,13,3X,F12.6)
C
C RHS(I) = FF**4
C
C DO 10 J = 1, S
C 10 COEF(I,J) = CI(J)*J
C
C##################################################################
C T = TAU
C ENP = DBLE(NP)
C DEET = T/ENP
C NN1 = NP/4
C NN2 = NP/2
C NN3 = 3*NP/4
C TEE1 = DEET*DBLE(NN1)
C TEE2 = DEET*DBLE(NN2)
C TEE3 = DEET*DBLE(NN3)
C DO 8 J = 1, 4
C 8 COEF(1,J) = TEE1**J
C 8 COEF(2,J) = TEE2**J
C 8 COEF(3,J) = TEE3**J
C 8 COEF(4,J) = TAO**J
C RHS(1) = THETA(NN1+1)**4 - A(0)
C RHS(2) = THETA(NN2+1)**4 - A(0)
C RHS(3) = THETA(NN3+1)**4 - A(0)
C RHS(4) = THETA(NP+1)**4-A(0)
C LDA = 5
C NN = 4
C IT = 1
C ID = 1
C CALL LIN(ID, IT, IPVT, U, V, S, E, WORK, COEF, LDA, NN, RHS, CRIT)
C
C WRITE(4,1001)
C DO 15 I = 1, 4
C 15 A(I) = RHS(I)
C
C WRITE(4,1004)

FORTRAN PROGRAM
C  
C... writes out actual temperatures and the ones from polynomial fit  
C  
C DO 20 I = 1,NPTS + 1  
C DT = TAO/DBLE(NPTS)  
C T = (I-1)*DT  
C TT = 0.0D0  
C IF(I.EQ.1) GO TO 18  
C DO 16 J = 0,4  
C 16 TT = TT + A(J)*T**J  
C GO TO 19  
C 18 TT = A(0)  
C 19 CONTINUE  
C DIFF = DABS(THETA(I)**4-TT)  
C 20 WRITE(10,1003) T,THETA(I)**4,TT  
    RETURN  
1001 FORMAT(I,' Polynomial coefficients ')  
1002 FORMAT(10X,D12.5,2X,D12.5)  
1004 FORMAT(/15X,'TEMP COMPUTED FROM POLYNOMIAL EXPANSION')  
1003 FORMAT(10X,F6.2,4X,2(F10.4,1X))  
    END
The following is the output from a sample run to obtain the temperature profile.

\[ \text{SIGMA E = } \frac{400}{M}, \text{ EPSH = EPSC = 1.0} \]

<table>
<thead>
<tr>
<th>TIME (HRS.)</th>
<th>TEMPERATURE (K)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>300.00 294.80 289.71 284.44 279.09</td>
</tr>
<tr>
<td>1.00</td>
<td>300.00 294.92 289.95 284.81 279.59</td>
</tr>
<tr>
<td>1.50</td>
<td>300.00 295.03 290.18 285.17 280.08</td>
</tr>
<tr>
<td>2.00</td>
<td>300.00 295.14 290.40 285.51 280.54</td>
</tr>
<tr>
<td>2.50</td>
<td>300.00 295.25 290.61 285.82 280.97</td>
</tr>
<tr>
<td>3.00</td>
<td>300.00 295.34 290.80 286.11 281.36</td>
</tr>
<tr>
<td>3.50</td>
<td>300.00 295.43 290.97 286.36 281.70</td>
</tr>
<tr>
<td>4.00</td>
<td>300.00 295.50 291.11 286.58 281.99</td>
</tr>
<tr>
<td>4.50</td>
<td>300.00 295.55 291.22 286.75 282.22</td>
</tr>
<tr>
<td>5.00</td>
<td>300.00 295.59 291.30 286.87 282.39</td>
</tr>
<tr>
<td>5.50</td>
<td>300.00 295.62 291.35 286.95 282.49</td>
</tr>
<tr>
<td>6.00</td>
<td>300.00 295.63 291.37 286.98 282.53</td>
</tr>
<tr>
<td>6.50</td>
<td>300.00 295.63 291.36 286.96 282.50</td>
</tr>
<tr>
<td>7.00</td>
<td>300.00 295.60 291.32 286.89 282.40</td>
</tr>
<tr>
<td>7.50</td>
<td>300.00 295.57 291.24 286.77 282.24</td>
</tr>
<tr>
<td>Value (°C)</td>
<td>Mass (g)</td>
</tr>
<tr>
<td>-----------</td>
<td>----------</td>
</tr>
<tr>
<td>5.00</td>
<td>200.00</td>
</tr>
<tr>
<td>5.50</td>
<td>200.00</td>
</tr>
<tr>
<td>6.00</td>
<td>200.00</td>
</tr>
<tr>
<td>6.50</td>
<td>200.00</td>
</tr>
<tr>
<td>7.00</td>
<td>200.00</td>
</tr>
<tr>
<td>7.50</td>
<td>200.00</td>
</tr>
<tr>
<td>8.00</td>
<td>200.00</td>
</tr>
<tr>
<td>8.50</td>
<td>200.00</td>
</tr>
<tr>
<td>9.00</td>
<td>200.00</td>
</tr>
<tr>
<td>9.50</td>
<td>200.00</td>
</tr>
<tr>
<td>10.00</td>
<td>200.00</td>
</tr>
<tr>
<td>10.50</td>
<td>200.00</td>
</tr>
<tr>
<td>11.00</td>
<td>200.00</td>
</tr>
<tr>
<td>11.50</td>
<td>200.00</td>
</tr>
<tr>
<td>12.00</td>
<td>200.00</td>
</tr>
<tr>
<td>12.50</td>
<td>200.00</td>
</tr>
<tr>
<td>13.00</td>
<td>200.00</td>
</tr>
<tr>
<td>13.50</td>
<td>200.00</td>
</tr>
<tr>
<td>14.00</td>
<td>200.00</td>
</tr>
<tr>
<td>14.50</td>
<td>200.00</td>
</tr>
<tr>
<td>15.00</td>
<td>200.00</td>
</tr>
<tr>
<td>15.50</td>
<td>200.00</td>
</tr>
<tr>
<td>16.00</td>
<td>200.00</td>
</tr>
<tr>
<td>16.50</td>
<td>200.00</td>
</tr>
<tr>
<td>17.00</td>
<td>200.00</td>
</tr>
<tr>
<td>17.50</td>
<td>200.00</td>
</tr>
<tr>
<td>18.00</td>
<td>200.00</td>
</tr>
<tr>
<td>18.50</td>
<td>200.00</td>
</tr>
<tr>
<td>19.00</td>
<td>200.00</td>
</tr>
<tr>
<td>19.50</td>
<td>200.00</td>
</tr>
<tr>
<td>20.00</td>
<td>200.00</td>
</tr>
<tr>
<td>20.50</td>
<td>200.00</td>
</tr>
<tr>
<td>21.00</td>
<td>200.00</td>
</tr>
<tr>
<td>21.50</td>
<td>200.00</td>
</tr>
<tr>
<td>22.00</td>
<td>200.00</td>
</tr>
<tr>
<td>22.50</td>
<td>200.00</td>
</tr>
<tr>
<td>23.00</td>
<td>200.00</td>
</tr>
<tr>
<td>23.50</td>
<td>200.00</td>
</tr>
<tr>
<td>24.00</td>
<td>200.00</td>
</tr>
</tbody>
</table>
It was obtained using the following data file.

```
99 20 0.465 0.0 0.0 0.0 0.0 3.556D1 0.10D1 0.10D1 3.0D2 2.73D2 1.0D0
'SLAB' 0.0D0 1.0D+01 1.0D-10 1 1 99 1.00-05 100
3.0D1 8.640D04
1.0 1000.0 0.0889 0.027 3.2D-04 4.000D2 2.115D3 4.35D2
1.0002
1.8908
2.8316
2.3493
2.0354
1.5964
1.02168
0.75243
0.386027
0.061976
-0.051829
-0.132697
-0.27966
-0.453012
-0.504900
-0.5337310
-0.411578
-0.1787070
0.0023427
0.1378430
```

Sample Output
The author was born on December 17, 1964 in New Delhi, India. In May, 1986 he completed his Bachelor of Technology degree at the Indian Institute of Technology, Delhi, India before coming to Virginia Polytechnic Institute and State University for graduate studies.

Sanjeev Gupta