MARTIN MARIETTA

HEATING 7.1 USER'S MANUAL

K. W. CHILDS

JULY 1991

HEATING 7.1 USER'S MANUAL

K. W. Childs

Computing and Telecommunications Division
at
Oak Ridge K-25 Site
Post Office Box 2003
Oak Ridge, Tennessee 37831-7039

July 1991

MASTER

CONTENTS

CONTENTS (continued)

CONTENTS (continued)

CONTENTS (continued)

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

HEATING is a FORTRAN program designed to solve steady-state and/or transient heat conduction problems in one-, two-, or three-dimensional Cartesian, cylindrical, or spherical coordinates. A model may include multiple materials, and the thermal conductivity, density, and specific heat of each material may be both time- and temperature-dependent. The thermal conductivity may be anisotropic. Materials may undergo change of phase. Thermal properties of materials may be input or may be extracted from a material properties library. Heat-generation rates may be dependent on time, temperature, and position, and boundary temperatures may be time- and position-dependent. The boundary conditions, which may be surface-to-boundary or surface-to-surface, may be specified temperatures or any combination of prescribed heat flux, forced convection, natural convection, and radiation. The boundary condition parameters may be time- and/or temperature-dependent. General graybody radiation problems may be modeled with user-defined factors for radiant exchange. The mesh spacing may be variable along each axis. HEATING is variably dimensioned and utilizes free-form input.

Three steady-state solution techniques are available: point-successive-overrelaxation iterative method with extrapolation, direct-solution (for one-dimensional or two-dimensional problems), and conjugate gradient.

Transient problems may be solved using any one of several finite-difference schemes: Crank-Nicolson implicit, Classical Implicit Procedure (CIP), Classical Explicit Procedure (CEP), or Levy explicit method (which for some circumstances allows a time step greater than the CEP stability criterion.) The solution of the system of equations arising from the implicit techniques is accomplished by point-successive-overrelaxation iteration and includes procedures to estimate the optimum acceleration parameter.

1. INTRODUCTION

HEATING is a multi-dimensional, general-purpose heat transfer code written in FORTRAN 77. The name HEATING is an acronym for Heat Engineering and Transfer In Nine Geometries (although with modifications there are now 12 geometries). Earlier versions of HEATING have been reported previously.[1-8] HEATING solves steady-state and/or transient problems in one-, two-, or three-dimensional Cartesian, cylindrical, or spherical coordinates. The mesh spacing may be variable along each axis. A model may include multiple materials, and the thermal conductivity, density, and specific heat of each material may be both time- and temperature-dependent. Thermal conductivity may be anisotropic. Materials may undergo up to five changes of phase for transient calculations involving one of the explicit procedures. The heat-generation rates may be dependent on time, temperature, and position. Boundary temperatures may be dependent on time and position. Boundary conditions include specified temperatures or any combination of prescribed heat flux, forced convection, natural convection, and radiation. The boundary condition parameters may be time- and/or temperature-dependent. In addition, one may model either one-dimensional radiative heat transfer directly across gaps embedded in the model or multidirectional radiation within enclosures using externally calculated radiation exchange factors.

The numerical techniques utilized for the steady-state and transient calculations are discussed in Sect. 2. For steady-state problems there are three solution techniques available: a point-successive-overrelaxation iterative method with a modified Aitken δ^2 extrapolation process, a direct-solution method for one- and two-dimensional problems, and a conjugate gradient method. Several numerical schemes are available in HEATING to solve transient heat-conduction problems. In the implicit technique, which can range from Crank-Nicolson to the Classical Implicit Procedure, the system of equations is solved by point-successive-overrelaxation iteration. The technique includes procedures to estimate the optimum acceleration parameters for both the case where the coefficient matrix remains unchanged during the calculations and the case where it varies during the solution because of dependence on time, temperature, or time-step size. The time-step size for the implicit transient calculations may be controlled so that either the maximum temperature change or the maximum relative temperature change over a time step does not exceed specified values. The time-step size may also be controlled by specifying an initial time step, a multiplication factor multiplied by the time-step size after each time level, a maximum allowable time-step size, and/or the time period for which it applies. Transient problems may also be solved either with the Classical Explicit Procedure, whose time step is stability-limited, or with the Levy explicit method, which allows a larger time step for some cases.

HEATING does not have to be recompiled with each execution since computer memory allocation is handled at execution time. Any array whose size is a function of the input data is variably dimensioned. Most array dimensions are determined by scanning the input data, but a few of the dimensions cannot be determined until the model is actually generated. The array dimensions that cannot be determined until model generation have default values which may be explicitly overridden by values supplied by the user in the input data.

HEATING uses free-form reading subroutines[9] to read the input data file which is subdivided into data blocks identified by keywords. HEATING will read the input data for a case until it encounters an end-of-case indicator. It checks the input data for errors and inconsistencies and issues messages identifying any problems encountered. Some data errors cause processing to be terminated immediately. However, most errors will allow the processing of input data to be completed, but will not allow the case to be executed. If no input data errors are encountered, HEATING will proceed with the specified calculations for the case.

Section 3 discusses modeling concepts and gives advice on the use of HEATING. It includes a number of suggestions concerning the use of special features of the code. A comprehensive discussion of the input data is presented in Sect. 4. An outline of the input data is included in the section to serve as a guide for the user who is somewhat familiar with the use of the code. A discussion outlining the output features of HEATING is presented in Sect. 5. The appendices include sample problems, instructions for running the code on the UNICOS CRAY computer at Martin Marietta Energy Systems, Inc., a discussion of codes available for postprocessing HEATING results, documentation of a material properties library for use with HEATING, and information on some utility codes for use with HEATING.

HEATING is not a black box that automatically yields the correct solution to the physical problem. Care must be exercised in correctly simulating the physical problem as well as in interpreting the results from the code. This version of HEATING has undergone the same extensive verification documented for a previous version[10] of the code. The verification process is discussed further in Appendix B.

2. NUMERICAL TECHNIQUE

2.1 STATEMENT OF THE PROBLEM

The HEATING computer program solves steady-state or transient heat-conduction problems in one, two, or three dimensions. Three coordinate systems are available in HEATING: Cartesian, cylindrical, and spherical. For each problem only one of the three coordinate systems can be used, and the entire problem geometry must be consistent with the chosen coordinate system. For an arbitrary orthogonal, curvilinear coordinate system with coordinates u, v, and w, the surfaces defined by the equations $u = constant$, $v = constant$, and $w = constant$ are referred to as coordinate surfaces. Any number of coordinate surfaces can be defined along each of the axes. In the Cartesian coordinate system the coordinates are x, y, and z; and the coordinate surfaces are planes perpendicular to the respective axis. In the cylindrical coordinate system the coordinates are r, θ, and z. An r coordinate surface is a surface of an infinitely long cylinder. A θ coordinate surface is a semi-infinite plane whose edge coincides with the axis of the cylinder. A z coordinate surface is an infinite plane perpendicular to the axis of the cylinder. In the spherical coordinate system the coordinates are r, θ, and ϕ. An r coordinate surface is the surface of a sphere. A θ coordinate surface is a semi-infinite plane whose edge coincides with the axis of the sphere (the line connecting the poles of the sphere). The θ coordinate surfaces in a spherical geometry are the same as those in a cylindrical geometry. The ϕ coordinate surfaces are the surfaces of cones with their vertices at the center of the sphere and their axes coinciding with the axis of the sphere. The angle ϕ is measured from the equatorial plane of the sphere. The $\phi = 0$ coordinate surface is the equatorial plane.

In order to define the nodes, a system of orthogonal coordinate surfaces is superimposed on the problem. The planes may be unequally spaced, but they must extend to the outer boundaries of the problem. An internal node is defined by the intersection of three coordinate surfaces. For illustrative purposes, the equations and the discussion that follow are written for a three-dimensional problem in Cartesian coordinates.

A portion of the grid surrounding an internal node (denoted by o) is depicted in Fig. 2.1. Heat may flow from this node to each adjacent node along paths that are parallel to each axis. Thus for a three-dimensional problem, heat may flow from an internal node to each of its six neighboring nodes. The nodes with which node o exchanges heat are denoted by the numbers 1 through 6 in Fig. 2.1. The system of equations describing the temperature distribution is derived by performing a heat balance about each node. The volume associated with a node is bounded by coordinate surfaces lying midway between the coordinate surfaces defining the node locations. The volume associated with node o is shown in Fig. 2.2.
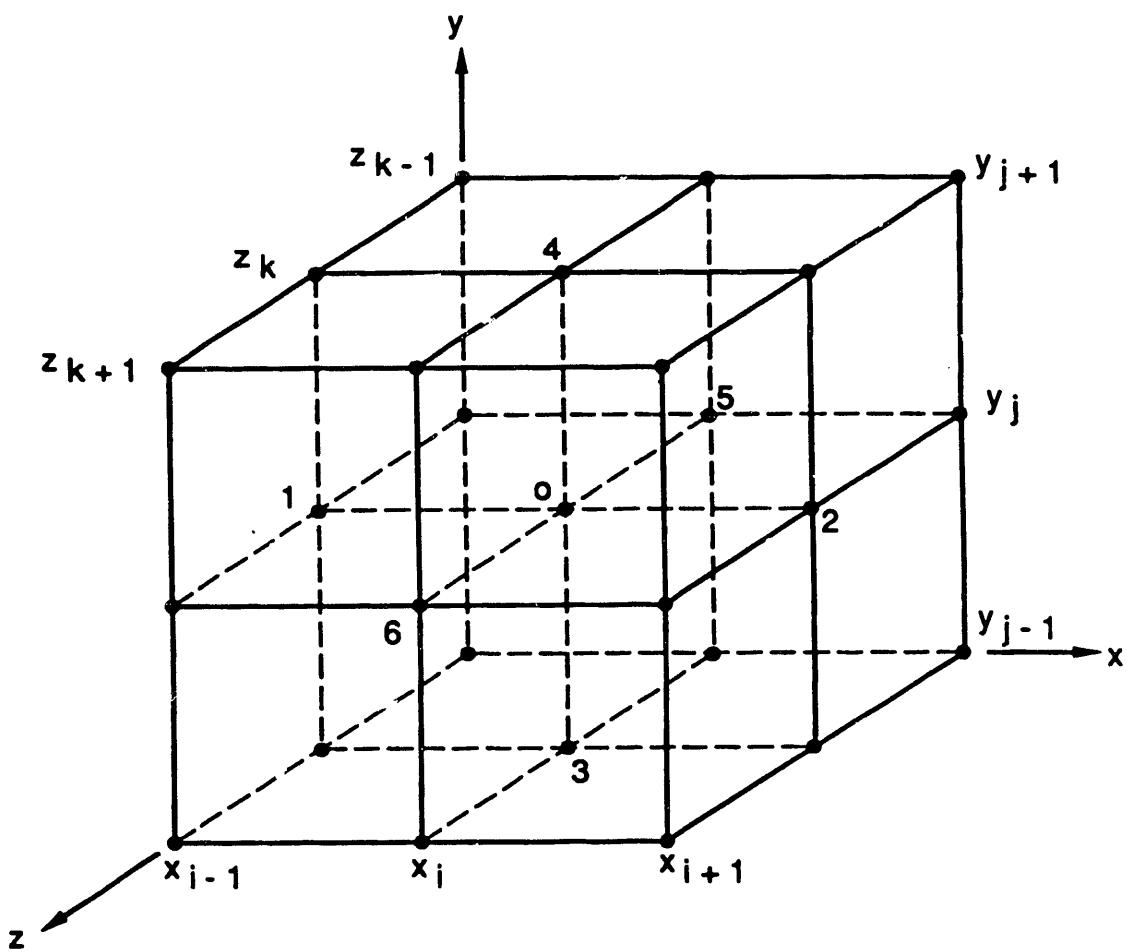
Fig. 2.1. Grid structure surrounding node of interest.
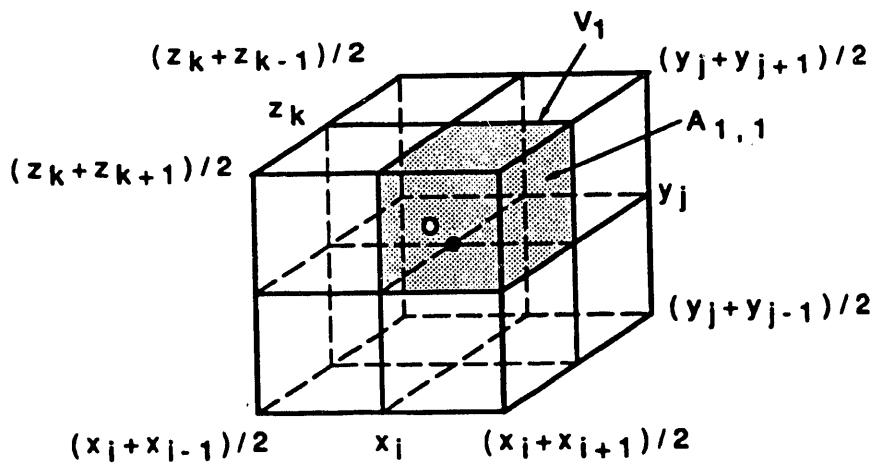
Fig. 2.2. Volume associated with node of interest (one of octants comprising volume is shaded).

The finite-volume heat balance equation for node o is

$$C_o \frac{T_o^{n+1} - T_o^n}{\Delta t} = P_o + \sum_{m=1}^{6} {}_o K_m (T_m^n - T_o^n) \, , \tag{2.1}$$

where T_m^n is the temperature of the mth node adjacent to node o at time t_n, ${}_o K_m$ is the conductance between nodes o and m, C_o is the heat capacitance associated with node o, P_o is the heat generation rate, and Δt is the time step $(t_{n+1} - t_n)$. Equation (2.1) approximates the transient heat conduction equation using a first-order accurate, explicit, forward Euler integration over time. For a uniform grid spacing, the spatial approximation is equivalent to a central difference which is second-order accurate.

The shaded volume in Fig. 2.2 indicates one of the eight subvolumes (or octants) which make up the total nodal volume. Each of the octants may contain a different material. Thus, a node may be composed of as many as eight different materials, and the heat flow path between adjacent nodes may be composed of as many as four different materials positioned in parallel. For a three-dimensional problem, one C, one P, and six Ks will be associated with each internal node. These parameters are calculated as follows for node o:

$$C_o = \sum_{\ell=1}^{8} c_{p\ell} \rho_\ell V_\ell \, , \tag{2.2}$$

$$P_o = \sum_{\ell=1}^{8} Q_\ell V_\ell \, , \tag{2.3}$$

$${}_o K_m = \frac{1}{L_m} \sum_{\gamma=1}^{4} k_{m,\gamma} A_{m,\gamma} \, , \tag{2.4}$$

where

$c_{p\ell}$	=	specific heat of material in the ℓth octant,
ρ_ℓ	=	density of material in the ℓth octant,
V_ℓ	=	volume of the ℓth octant of node o,
$Q\ell$	=	heat generation rate per unit volume in ℓth octant,
L_m	=	distance between node o and adjacent node m,
$k_{m,\gamma}$	=	thermal conductivity of material in the γth of four heat-flow paths between nodes o and m, and
$A_{m,\gamma}$	=	cross-sectional area of the γth heat flow path between nodes o and m.

With reference to Fig. 2.2, the V_ℓ's and $A_{m,y}$'s are further defined, by using examples, as follows:

$$V_1 = \left(\frac{x_{i+1} - x_i}{2}\right) \times \left(\frac{y_{j+1} - y_j}{2}\right) \times \left(\frac{z_{k+1} - z_k}{2}\right) ; \tag{2.5}$$

$$A_{1,1} = \frac{y_{j+1} - y_j}{2} \times \frac{z_{k+1} - z_k}{2} . \tag{2.6}$$

Since nodes lying on the surface of a model or nodes from one- or two-dimensional problems will not have six neighbors, the general heat balance equation for node i having M_i neighbors can be written as

$$C_i \frac{T_i^{n+1} - T_i^n}{\Delta t} = P_i + \sum_{m=1}^{M_i} {}_i K_{\alpha_m}(T_{\alpha_m}^n - T_i^n) , \tag{2.7}$$

where α_m is the mth neighbor of the ith node. By choosing the increments between grid lines small enough, the solution to the system of equations yields an acceptable approximation to the governing differential equation.

2.2 STEADY-STATE HEAT CONDUCTION

For a steady-state heat conduction problem, the heat balance equation reduces to

$$P_i + \sum_{m=1}^{M_i} {}_i K_{\alpha_m}(T_{\alpha_m} - T_i) = 0 . \tag{2.8}$$

If there are N nodal points, Eq. (2.8) will yield a system of N equations with N unknowns. HEATING contains two techniques for solving this system of equations. The first technique involves point-successive-overrelaxation (SOR) iteration with a modified Aitken δ^2 extrapolation process. This method can be used for any steady-state problem. However, for certain classes of problems convergence may be slow or the convergence criterion may be unreliable since it only requires that the maximum relative change in temperature from one iteration to the next be less than the specified value. Such difficulties can arise in problems with widely varying parameters such as thermal conductivity, grid spacing along an axis, or nonlinear boundary conditions. Thus, a direct-solution technique is also available to solve the steady-state system of equations. The direct-solution technique is available only for one- and two-dimensional problems, but it is not always more efficient than the successive-overrelaxation method since it can require a large amount of computer memory. This can occur for two-dimensional calculations with a large number of lattice lines along the x (or r) axis, or with heat transfer across a gap along the y (or θ) or z (or ϕ) axes.

2.2.1 Point-Successive-Overrelaxation Iteration

Solving Eq. (2.8) for T_i yields

$$T_i = \frac{P_i + \sum_{m=1}^{M_i} {}_iK_{\alpha_m}T_{\alpha_m}}{\sum_{m=1}^{M_i} {}_iK_{\alpha_m}} \quad . \qquad (2.9)$$

Since the values of T_α are unknown, the temperature at node i cannot be calculated directly. However, an iterative procedure can be used to estimate the steady-state temperature distribution. If an estimate to the temperature distribution exists, then Eq. (2.9) can be applied at each node, and a better estimate to the temperature distribution may be obtained. This new estimate can be used in Eq. (2.9) to produce a better estimate. This procedure is known as Jacobi iteration or the method of simultaneous iterations and can be written as

$$T_i^{(n+1)} = \frac{P_i + \sum_{m=1}^{M_i} {}_iK_{\alpha_m}T_{\alpha_m}^{(n)}}{\sum_{m=1}^{M_i} {}_iK_{\alpha_m}} \quad , \qquad (2.10)$$

where the superscript (n) implies the nth iterate. It can be observed in Eq. (2.10) that some components of $T_i^{(n+1)}$ are known during the iterative sweep but are not used. The Gauss-Seidel method represents a modification of Jacobi iteration by using the most recent values of T_i during the iteration. The term "successive iteration" is commonly employed in conjunction with Gauss-Seidel iteration to denote the fact that new components of the unknown vector T_i are successively used as they are obtained. Gauss-Seidel iteration is defined as

$$T_i^{(n+1)} = \frac{P_i + \sum_{m=1}^{L_i} {}_iK_{\alpha_m}T_{\alpha_m}^{(n+1)} + \sum_{m=L_i+1}^{M_i} {}_iK_{\alpha_m}T_{\alpha_m}^{(n)}}{\sum_{m=1}^{M_i} {}_iK_{\alpha_m}} \quad , \qquad (2.11)$$

where L_i is defined so that $\alpha_m < i$ for $m \leq L_i$; and $\alpha_m > i$ for $m > L_i$. Instead of using the results of Eq. (2.11) as the $(n + 1)$st iterate, assume that it only yields an intermediate estimate and denote it as $\hat{T}_i^{(n+1)}$. Then define the $(n + 1)$st iterate to be

$$T_i^{(n+1)} = T_i^{(n)} + \beta[\hat{T}_i^{(n+1)} - T_i^{(n)}] \quad . \qquad (2.12)$$

The accelerated Gauss-Seidel technique can then be expressed as

$$T_i^{(n+1)} = (1 - \beta)T_i^{(n)} + \beta \left[\frac{P_i + \sum_{m=1}^{L_i} {}_iK_{\alpha_m}T_{\alpha_m}^{(n+1)} + \sum_{m=L_i+1}^{M_i} {}_iK_{\alpha_m}T_{\alpha_m}^{(n)}}{\sum_{m=1}^{M_i} {}_iK_{\alpha_m}} \right] . \qquad (2.13)$$

Varga[11] refers to this method as the point-successive-overrelaxation method or SOR. To increase the rate of convergence, an exponential approximation for Eq. (2.13) is made based on the temperature change from one iteration to the next. The algorithm based on this approximation is used instead of Eq. (2.13) to calculate the new temperatures for nodes having relative temperature changes exceeding 10^{-3}. However, to prohibit the technique from diverging because of a bad estimate of the initial temperature distribution, the algorithm is designed to bound the temperature change so that the new temperature cannot be more than twice the old temperature. The exponential approximation reduces to Eq. (2.13) for small temperature changes. Successive iterations are carried out until

$$\left| \frac{T_i^{(n)} - T_i^{(n+1)}}{T_i^{(n+1)}} \right|_{maximum \ for \ all \ nodes} \leq \epsilon , \qquad (2.14)$$

where ϵ is the specified convergence criterion.

Since the coefficient matrix produced by the heat balance equation is symmetric and positive definite, the Ostrowski-Reich theorem assures convergence if the acceleration factor β is limited to $0 < \beta < 2$. An optimum value of β in Eq. (2.13) is difficult to obtain. If an input value is not supplied for β, the default value is 1.5. If the rate of convergence is slow, β is reduced by 0.1. During an extrapolation cycle, the relative temperature change over an iteration is monitored over ten consecutive iterations. If the relative temperature changes do not decrease monotonically over these ten iterations, then the current relative temperature change is compared with the one arising ten iterations earlier. If the current relative temperature change is greater than two-thirds of the old one, then the SOR technique is assumed to be converging slowly. This process may be repeated until $\beta = 1.0$. However, the code will not increase β.

Another extrapolation procedure used to increase the rate of convergence in an iterative solution to a system of equations is the "Aitken δ^2 extrapolation procedure." If $T^{(n-1)}$, $T^{(n)}$, and $T^{(n+1)}$ are the temperatures at a certain point at the $(n-1)$st, nth and $(n+1)$st iterations, respectively, and if

$$\left| T^{(n)} - T^{(n-1)} \right| > \left| T^{(n+1)} - T^{(n)} \right| \qquad (2.15)$$

and

$$\left[T^{(n+1)} - T^{(n)} \right] \left[T^{(n)} - T^{(n-1)} \right] > 0 , \qquad (2.16)$$

then a better estimate of the temperature is

$$T_{new} = T^{(n+1)} + \frac{\left[T^{(n+1)} - T^{(n)}\right]^2}{\left[T^{(n)} - T^{(n-1)}\right] - \left[T^{(n+1)} - T^{(n)}\right]} \quad . \tag{2.17}$$

HEATING uses a modification of the Aitken δ^2 method by calculating an extrapolation factor, B, and approximating Eq. (2.17) at each node with

$$T_i^{(n+1)} = \tilde{T}_i^{(n+1)} + B\left[\tilde{T}_i^{(n+1)} - T_i^{(n)}\right] \quad , \tag{2.18}$$

where $\tilde{T}^{(n+1)}$ represents the $(n+1)$st iterate at node i before extrapolation.

An extrapolation cycle is defined as follows: The code completes 20 iterations and checks to see if the maximum of the absolute values (i.e. the sub-norm) of the relative temperature changes over an iteration has decreased monotonically over the last ten iterations. If not, the cycle starts over. If so, the code will extrapolate only if the relative change in extrapolation factors over two consecutive iterations is less than 5% and the sub-norm of the relative temperature changes decreases monotonically over the same two iterations. The extrapolation factor, B, which is the same for each node, is based on the maximum relative temperature change for two consecutive iterations.

2.2.2 Direct-Solution Technique

If Eq. (2.8) is rewritten with the heat-generation term P_i and the terms associated with the boundary temperatures on the right-hand side of the equal sign, the system of equations can be represented in matrix form as

$$A\vec{t} = \vec{b} \quad , \tag{2.19}$$

where A is the square coefficient matrix, $\vec{t}$ is the column vector of unknown temperatures, and $\vec{b}$ is the column vector containing the forcing functions, consisting of terms related to heat-generation rates and boundary conditions. Matrix A is banded, symmetric, and positive-definite. When the bandwidth is not large, the system of equations can be solved efficiently using a direct-solution technique. HEATING contains a direct-solution technique to solve one- and two-dimensional steady-state problems. The bandwidth on most three-dimensional problems will be quite large, leading to excessive computer memory and computing time requirements. Thus the direct-solution technique has not been implemented for three-dimensional problems.

After building the banded coefficient matrix, HEATING determines the bandwidth and calls the appropriate subroutine from a library of simultaneous linear algebraic equation solvers.[12] Most one-dimensional problems have a tridiagonal coefficient matrix and are solved using a symmetric, positive-definite, tridiagonal matrix subroutine. Otherwise, systems of equations are solved using general symmetric, positive-definite, banded matrix subroutines.

For nonlinear problems, the terms P_i and $_iK_{\alpha_m}$ in Eq. (2.8) may be a function of the unknown temperatures of the associated nodes. This system of nonlinear equations may be

reduced to a system of linear equations by evaluating P_i and $_iK_{\alpha_m}$ at estimates of the nodal temperatures. New estimates for nodal temperatures are then determined using one of the techniques mentioned above. In an iterative process the terms P_i and $_iK_{\alpha_m}$ are reevaluated using the newest nodal temperature distribution, and the new system of linear equations is solved. This iterative procedure can be represented in matrix form by denoting the nth iterate of the banded coefficient matrix, the temperature vector, and the vector of forcing functions as $A^{(n)}, \vec{t}$, and $\vec{b}$, respectively. Then,

$$A^{(n)} = A\left(\vec{t}^{(n)}\right) \tag{2.20}$$

and

$$\vec{b}^{(n)} = \vec{b}\left(\vec{t}^{(n)}\right) . \tag{2.21}$$

The coefficients in matrix A and vector $\vec{b}$ are evaluated using the nth iterate of the temperature vector. The system of equations

$$A^{(n)}\vec{t} = \vec{b}^{(n)} \tag{2.22}$$

is solved to yield $\vec{t}^{(n+1)}$. This process, referred to as functional or direct iteration, is repeated until the convergence criterion has been satisfied.

The convergence criterion was developed by Becker.[13] The following discussion was taken from that reference. Define a heat residual vector, $\vec{r}$, at the nth iteration as

$$\vec{r}^{(n)} = A^{(n)}\vec{t}^{(n)} - \vec{b}^{(n)} . \tag{2.23}$$

This vector contains the heat residual, $\vec{r}^{(n)}$, at each node. From Eqs. (2.8) and (2.23), the ith component of vector $\vec{r}$ is

$$r_i^{(n)} = \sum_{m=1}^{M_i} {}_iK_{\alpha_m}^{(n)}\left(T_{\alpha_m}^{(n)} - T_i^{(n)}\right) + P_i^{(n)} . \tag{2.24}$$

Making this residual sufficiently small at each node is assumed equivalent to approaching the correct temperature at each node. To scale this nodal residual for problems involving large nodal heat flows, an average heat flow at each node is calculated as the average of the absolute values of all the heat flows into and out of the node, including heat generation. A relative residual is then determined at each node as the residual at that node divided by the average heat flow at that node. Thus, at each node two measures of accuracy are calculated: a residual and a relative residual. The error at each node is determined as the lesser of either the residual at that node or the relative residual at that node. Then the maximum error is determined over all nodes in the model. If this maximum error is less than the convergence criterion, the calculation is considered converged.

Experience has shown that for a nonlinear problem a residual on the order of 10^{-10} to 10^{-12} is the minimum error that can be achieved because of machine imprecision in the calculations using eight-byte real numbers. For a nonlinear problem it is recommended that the user run a problem with two or three different convergence criteria (such as 10^{-6}, 10^{-8}, 10^{-10}) to gain confidence in the answers given by the code.

2.2.3 Conjugate Gradient Method

A conjugate gradient technique is implemented in HEATING.[14] For a well-posed problem the linear conjugate gradient technique is "guaranteed" to converge in N iterations, where N is the number of unknowns. For the conjugate gradient technique, the system of equations is written in the same matrix form as used with the direct-solution technique:

$$A\vec{t} = \vec{b} . \tag{2.25}$$

Initially the residual vector, $\vec{r}$, is calculated based on the temperature vector, $\vec{t}$, and the direction vector, $\vec{p}$, is set equal to the residual vector:

$$\vec{r}^0 = \vec{b} - A\vec{t}^0 , \tag{2.26}$$

$$\vec{p}^0 = \vec{r}^0 . \tag{2.27}$$

The iteration steps for $k = 0,1,...$ are given below:

1. determine distance to move along direction vector based on a one-dimensional minimization and calculate new solution estimate,

$$\alpha_k = -(\vec{r}^k, \vec{r}^k)/(\vec{p}^k, A\vec{p}^k) , \tag{2.28}$$

$$\vec{t}^{k+1} = \vec{t}^k - \alpha_k \vec{p}^k ; \tag{2.29}$$

2. calculate a new residual vector,

$$\vec{r}^{k+1} = \vec{r}^k - \alpha_k A\vec{p}^k ; \tag{2.30}$$

3. check for convergence, if

$$\max \left| r_i^{k+1} \right|_{1 \le i \le N} \ge \epsilon ; \tag{2.31}$$

continue,

4. calculate a new direction vector,

$$\beta_k = \frac{(\vec{r}^{k+1}, \vec{r}^{k+1})}{(\vec{r}^k, \vec{r}^k)} \ , \tag{2.32}$$

$$\vec{p}^{k+1} = \vec{r}^{k+1} + \beta_k \vec{p}^k \ . \tag{2.33}$$

2.3 TRANSIENT HEAT CONDUCTION

HEATING is designed to solve a transient problem by any one of several numerical schemes. The first is the Classical Explicit Procedure (CEP), which involves the first forward difference with respect to time and is stable only when the time step is smaller than the stability criterion. Levy's modification to the CEP is the second scheme, and it requires the temperature distribution at two times to calculate the temperatures at the new time level. The technique is stable for a time step of any size. The third procedure, which is written generally, contains several implicit techniques that are stable for a time step of any size. One can use the Crank-Nicolson technique, the Classical Implicit Procedure (CIP) (also referred to as backwards Euler), or a linear combination of the two. The resulting system of equations is solved by point-successive-overrelaxation iteration. Techniques have been included in the code to approximate the optimum acceleration parameter. Change-of-phase problems can be modeled with one of the explicit solution techniques. A change-of-phase modeling procedure has not been implemented in the implicit procedures in HEATING.

Equation (2.7) is the basic heat balance equation for transient problems. However, the right-hand side is modified for all but the CEP. The Crank-Nicolson implicit procedure is the recommended technique for solving transient problems.

2.3.1 Classical Explicit Procedure

For a transient heat-conduction problem, the heat balance equation, Eq. (2.7), can be solved for T^{n+1} to give

$$T_i^{n+1} = T_i^n + \frac{\Delta t}{C_i} \left[P_i^n + \sum_{m=1}^{M_i} {}_i K_{\alpha_m} (T_{\alpha_m}^n - T_i^n) \right]. \tag{2.34}$$

Since Eq. (2.34) expresses the temperature of node i at the $(n+1)$st time level in terms of temperatures at the nth time level, it is an explicit technique, and the algorithm is known as the CEP or the forward Euler time integration. The numerical solution obtained by using this technique is stable, provided the time step satisfies the following inequality.[15]

The stability criterion is also a function of heat generation and heat flux, but this is not accounted for when calculating a limiting time step in HEATING.

$$\Delta t \leq \left[\frac{C_i}{\displaystyle\sum_{m=1}^{M_i} {}_iK_{\alpha_m}} \right]_{\text{minimum for all nodes}}$$
(2.35)

2.3.2 Levy's Modification to the Classical Explicit Procedure

The limitation on the size of the time step in Eq. (2.35) results in some computation costs becoming so high that the use of the algorithm defined by Eq. (2.34) becomes impractical. Levy[16] proposed an explicit method that is stable for any time step. The basic equation is

$$T_i^{n+1} = T_i^n + \frac{1}{1+Z_i}\left\{ \frac{\Delta t}{C_i}\left[P_i^n + \sum_{m=1}^{M_i} {}_iK_{\alpha_m}(T_{\alpha_m}^n - T_i^n) \right] + Z_i[T_i^n - T_i^{n-1}] \right\},$$
(2.36)

where

$$Z_i = \begin{cases} 0 & , \dfrac{\Delta t}{(\Delta t_{max})_i} \leq 1 \\[2ex] 0.5\left[\dfrac{\Delta t}{(\Delta t_{max})_i} - 1 \right] & , \dfrac{\Delta t}{(\Delta t_{max})_i} > 1 \end{cases}$$
(2.37)

$(\Delta t_{max})_i$ is the maximum time step, from Eq. (2.35), allowed at node i for a stable solution in the CEP method.

According to Levy, the accuracy is good if Z_i is zero for somewhat over half of the nodes (i.e., the time step should be smaller than the CEP stability criterion for somewhat over half the nodes). Although it is not immediately obvious from Eq. 2.36, the Levy technique actually calculates the nodal temperature change over a time step as the weighted average of the nodal temperature change calculated from an energy balance for the current time step and the nodal temperature change over the previous time step. The weighting factors are one for the current time step and Z_i for the previous time step. Thus, the Levy technique does not rigorously enforce conservation of energy at nodes where Z_i is non-zero. The user-supplied subroutine QTOTAL (discussed in Appendix E) can be used to check for overall conservation of energy during a transient if this is a concern.

The code prints out a message giving the minimum, median, and maximum nodal stability criteria as well as a warning if the entered Levy factor results in a time step larger than the median nodal stability criterion. Of course, one must experiment with the size of the time step in order to obtain an acceptable solution even when the time step is less than the median stability criterion.

2.3.3 Implicit Procedure

2.3.3.1 Heat-balance equation

If the right-hand side of Eq. (2.7) is evaluated at t_{n+1} instead of t_n, then the technique is the CIP or backward Euler time integration. If the right-hand side of Eq. (2.7) is evaluated at $t_{n+\frac{1}{2}}$, then the algorithm is the Crank-Nicolson (CN) procedure or trapezoidal rule. A general algorithm that includes both the CN technique and the CIP is

$$C_i^{n+\Theta} \frac{T_i^{n+1} - T_i^n}{\Delta t} = P_i^{n+\Theta} + \Theta \left[\sum_{m=1}^{M_i} {}_i K_{\alpha_m}^{n+\Theta} (T_{\alpha_m}^{n+1} - T_i^{n+1}) \right] + (1 - \Theta) \left[\sum_{m=1}^{M_i} {}_i K_{\alpha_m}^{n+\Theta} (T_{\alpha_m}^n - T_i^n) \right], \quad (2.38)$$

where $0 \leq \Theta \leq 1$ and where the superscript $n + \Theta$ implies that the parameter is evaluated at time $t_{n+\Theta}$. If $\Theta = 0.5$, then Eq. (2.38) becomes the CN technique and if $\Theta = 1.0$, the algorithm is the CIP. When Θ is less than 0.5, the technique is no longer stable. Note that Eq. (2.38) reverts to Eq. (2.7) when $\Theta = 0$. This algorithm has been incorporated into HEATING for $0.5 \leq \Theta \leq 1.0$.

2.3.3.2 Numerical technique

If there are N nodes in the problem, Eq. (2.38) yields N equations and N unknowns, and the resulting system of equations can be solved iteratively. If Eq. (2.38) is rewritten so that the temperatures at t_{n+1} are on the left-hand side, then the equation becomes

$$-\Theta \left[\sum_{m=1}^{M_i} {}_i K_{\alpha_m}^{n+\Theta} T_{\alpha_m}^{n+1} \right] + \left[\frac{C_i^{n+\Theta}}{\Delta t} + \Theta \sum_{m=1}^{M_i} {}_i K_{\alpha_m}^{n+\Theta} \right] T_i^{n+1} = H_i , \quad (2.39)$$

where

$$H_i = \frac{C_i^{n+\Theta}}{\Delta t} T_i^n + P_i^{n+\Theta} + (1 - \Theta) \left[\sum_{m=1}^{M_i} {}_i K_{\alpha_m}^{n+\Theta} (T_{\alpha_m}^n - T_i^n) \right]. \quad (2.40)$$

Defining

$$D_i = \frac{C_i^{n+\Theta}}{\Delta t} + \Theta \sum_{m=1}^{M_i} {}_i K_{\alpha_m}^{n+\Theta} , \quad (2.41)$$

and omitting the superscript, $n+1$, on the temperature, T, Eq. (2.39) can be rewritten as

$$-\Theta \left[\sum_{m=1}^{M_i} {}_i K_{\alpha_m}^{n+\Theta} T_{\alpha_m} \right] + D_i T_i = H_i , \quad (2.42)$$

where T_i represents the temperature of node i at the new time level. Solving for T_i,

$$T_i = \frac{\Theta\left[\sum_{m=1}^{M_i} {}_iK_{\alpha_m}^{n+\Theta} T_{\alpha_m}\right] + H_i}{D_i} \; .$$

(2.43)

Since the values of T_{α_m} are unknown, one cannot solve directly for the temperature at node i. However, if an initial estimate of the temperature distribution at the new time level exists, then Eq. (2.43) can be solved at each node to produce a new estimate for the temperature distribution. The procedure can be repeated using this new estimate. This process can be continued until the estimates have converged to the approximation of the temperature distribution at the new time level. This algorithm can be written as

$$T_i^{(n+1)} = \frac{\Theta\left[\sum_{m=1}^{M_i} {}_iK_{\alpha_m}^{n+\Theta} T_{\alpha_m}^{(n)}\right] + H_i}{D_i} \; ,$$

(2.44)

where the superscript in parentheses refers to the iteration level.

The iterative method resulting from applying Eq. (2.44) is known as Jacobi iteration or the method of simultaneous substitutions, as discussed in 2.2.1. Instead of using Eq. (2.44) in the iterative process which exhibits a relatively slow convergence rate, the technique can be refined further. Some components of $T_i^{(n)}$ are known during the Jacobi sweep but are not used. The Gauss-Seidel method represents a modification of Jacobi iteration by using the most recent values of T_i during the iteration. The term "successive iteration" is commonly employed in conjunction with Gauss-Seidel iteration to denote the fact that new components of the unknown vector T_i are successively used as they are obtained. Gauss-Seidel iteration is depicted in Eq. (2.45).

$$T_i^{(n+1)} = \frac{\Theta\left[\sum_{m=1}^{L_i} {}_iK_{\alpha_m}^{n+\Theta} T_{\alpha_m}^{(n+1)} + \sum_{m=L_{i+1}}^{M_i} {}_iK_{\alpha_m}^{n+\Theta} T_{\alpha_m}^{(n)}\right] + H_i}{D_i} \; ,$$

(2.45)

where L_i is defined so that $\alpha_m < i$ for $m \le L_i$; and $\alpha_m > i$ for $m > L_i$. Acceleration of the Gauss-Seidel procedure can produce improved convergence rates. Instead of using the result of Eq. (2.45) as the $(n + 1)$st iterate, it is treated as an intermediate estimate denote by $\hat{T}_i^{(n+1)}$

$$\hat{T}_i^{(n+1)} = \frac{\Theta\left[\sum_{m=1}^{L_i} {}_iK_{\alpha_m}^{n+\Theta} T_{\alpha_m}^{(n+1)} + \sum_{m=L_{i+1}}^{M_i} {}_iK_{\alpha_m}^{n+\Theta} T_{\alpha_m}^{(n)}\right] + H_i}{D_i} \; .$$

(2.46)

The $(n+1)$st iterate is defined as

$$T_i^{(n+1)} = T_i^{(n)} + \omega[\hat{T}_i^{(n+1)} - T_i^{(n)}] \tag{2.47}$$

or

$$T_i^{(n+1)} = (1 - \omega)T_i^{(n)} + \omega\hat{T}_i^{(n+1)} , \tag{2.48}$$

where ω is an acceleration factor. Since the coefficient matrix produced by the heat balance equation is symmetric, positive-definite, the Ostrowski-Reich theorem assures convergence if the acceleration factor is limited to $0 < \omega < 2$. The change in the estimated temperature distribution is greater than ($\omega > 1$) or less than ($\omega < 1$) the Gauss-Seidel update. The technique is referred to as underrelaxation and overrelaxation for $\omega < 1$ and $\omega > 1$, respectively. For $\omega = 1$, Gauss-Seidel iteration is recovered. Combining Eqs. (2.46) and (2.48) yields

$$T_i^{(n+1)} = (1 - \omega)T_i^{(n)} + \omega\frac{\Theta\left[\sum_{m=1}^{L_i} {}_iK_{\alpha_m}^{n+\Theta}T_{\alpha_m}^{(n+1)} + \sum_{m=L_{i+1}}^{M_i} {}_iK_{\alpha_m}^{n+\Theta}T_{\alpha_m}^{(n)}\right] + H_i}{D_i} . \tag{2.49}$$

The method described by Eq. (2.49) is referred to as the point-successive-overrelaxation iterative method or SOR. HEATING applies Eq. (2.49) to all nodes until the convergence criterion has been met.

The convergence criterion is derived as follows. When the nth iteration has been completed, substitute the nth iterate into Eq. (2.42) and denote the heat residual as

$$R_i^{(n)} = H_i + \Theta\left[\sum_{m=1}^{M_i} {}_iK_{\alpha_m}^{n+\Theta}T_{\alpha_m}^{(n)}\right] - D_iT_i^{(n)} . \tag{2.50}$$

Normalizing the heat residual by dividing by the right-hand side of Eq. (2.39) yields

$$\frac{R_i^{(n)}}{H_i} = \frac{H_i + \Theta\left[\sum_{m=1}^{M_i} {}_iK_{\alpha_m}^{n+\Theta}T_{\alpha_m}^{(n)}\right] - D_iT_i^{(n)}}{H_i} . \tag{2.51}$$

The convergence criterion based on the maximum normalized heat residual occurring at any node is

$$\left|\frac{R_i^{(n)}}{H_i}\right|_{maximum\ for\ all\ nodes} \leq \epsilon_1 . \tag{2.52}$$

If H_i is zero for a particular node, then the denominator is calculated as the L_1 norm of the numerator (the sum of the absolute value of the terms in the numerator divided by the total number of terms). If this is also zero, the node is not used in the test for convergence.

For the first time step, the starting estimate is equal to the initial temperature distribution. Thereafter, the starting estimate at t_{n+1} is determined by

$$T_i^{n+1} = T_i^n + (T_i^n - T_i^{n-1})\frac{\Delta t_{n+1}}{\Delta t_n} .$$

(2.53)

2.3.3.3 Temperature-dependent properties

For problems involving temperature-dependent thermal properties, the thermal properties are initially evaluated at the initial temperatures. Then, the point-successive-overrelaxation iterative method is applied to obtain the temperature distribution at the new time level. However, since none of the thermal properties are updated during this procedure, the converged temperatures are only an estimate to the temperature distribution. Thus, the thermal properties are reevaluated and the entire procedure is repeated until the technique converges to the temperature distribution at the new time level. This process contains two levels of iteration. The inner loop is the basic iterative process in the point-successive-overrelaxation iterative method; the outer loop iterates on the thermal properties. Let $T_i^{n,m}$ denote the temperature of node i after the mth iteration on the outer loop at time t_n. For the $(m+1)$st iteration on the outer loop, the temperature at which the thermal properties are evaluated is calculated as

$$T_i^{n+\Theta} = (1 - \Theta)T_i^n + \Theta T_i^{n+1,m} .$$

(2.54)

The temperature distribution has converged at time t_{n+1} when the relative temperature change averaged over all nodes is less than a prescribed value for successive iterations or

$$\frac{1}{N} \sum_{i=1}^{N} \frac{|T_i^{n+1,m} - T_i^{n+1,m-1}|}{|T_i^{n+1,m} - T_i^n|} \leq \epsilon_2 .$$

(2.55)

To avoid division by zero or very small values, the actual test used in the code is

$$\frac{1}{N} \sum_{i=1}^{N} \frac{|T_i^{n+1,m} - T_i^{n+1,m-1}|}{\max\left(|T_i^{n+1,m} - T_i^n|, 1.0\right)} \leq \epsilon_2 .$$

(2.56)

2.3.3.4 Estimation of the optimum acceleration parameter

The rate of convergence of the point-successive-overrelaxation iterative method is strongly dependent on the value of the acceleration parameter ω. The optimum value of the parameter, denoted as ω_o, is a function of time for problems whose effective thermal conductances and capacitances vary with time or temperature or whose time step changes during the calculations. Several techniques have been developed to estimate ω_o for transient problems with constant thermal properties. The method developed by Carre[17] has been incorporated into HEATING. This method consists of estimating ω_o based on the behavior of a norm of the residual vector

during the iterative procedure. The estimates are computed as a function of the iteration number until the process converges to a best estimate of the optimum value. Thereafter, the code uses this converged value as the acceleration parameter.

This process was not satisfactory for problems involving temperature- and time-dependent conductances and capacitances or for problems whose time step changes during the calculations, so an empirical process was developed and added to HEATING to estimate ω_o. For the initial time step, ω is equal to unity. The code will attempt to update ω every N_ω time steps. The criteria which are applied to determine whether or not the current value of ω is a good estimate to ω_o are based on the number of iterations required for the inner iterative loop to converge on the first pass through the outer iterative loop at some time step. When the code determines that an attempt to update ω should be made after completion of a particular time step, then the number of iterations for this time step is compared to the number for the time step immediately following the last modification to ω. If the change in the number of iterations is equal to or exceeds the criterion I_ω (an input value), then ω is increased according to

$$\omega^{n+1} = \omega^n + 0.1\left(2.0 - \omega^n\right),$$

(2.57)

where the superscript n refers to the value of ω at time t_n. On each subsequent time step a new estimate is made for ω_o using an algorithm similar to Eq. (2.57). However, ω may be either increased or decreased according to Table 2.1. When the change in the number of iterations for two consecutive time steps is less than the criterion J_ω (an input value), the code assumes that it has a good estimate for ω_o and uses this value for the subsequent time steps until it is time to attempt another ω update.

Table 2.1. Logic to determine whether ω should be increased or decreased

Last update resulted in ω being	Number of iterations compared to previous time step	
	Increased	Decreased
Increased	Decrease	Increase
Decreased	Increase	Decrease

2.3.3.5 Variable time-step size

The time-step size can be varied several ways during the implicit transient calculations. It can be controlled explicitly by specifying it as a constant size during the entire calculation or during prescribed time intervals throughout the transient. The time-step size can also be varied explicitly by multiplying it by a prescribed factor after each time step subject to maximum and minimum constraints.

The time-step size can be varied implicitly in two ways. One can specify the maximum temperature change and the maximum percentage change in temperature allowed at a node over a time step. The time-step size is decreased if one of the calculated maximum values exceeds its respective criterion. It is increased if both of the calculated maximum values are less than their respective criteria. If one of the calculated values exceeds its respective criterion, then

$$f_{\Delta t} = \min\left[0.95 \times \frac{criterion}{calculated\ value}, f_{input}\right], \qquad (2.58)$$

where f_{input} is an input value representing the factor that is multiplied by the old time-step size to obtain the new time-step size. The factor 0.95 ensures the decrease is large enough since the maximum temperature change is not linear with the changing time-step size. The new time-step size is determined as

$$\Delta t_{new} = \max\left(\Delta t_{min}, f_{\Delta t} \times \Delta t_{old}\right), \qquad (2.59)$$

where Δt_{min} is an input value representing the smallest time-step size allowed and Δt_{old} is the current time-step size. If one of the calculated values is less than its respective criterion, then

$$f_{\Delta t} = \min\left[0.95 \times \frac{criterion}{calculated\ value}, 1.0\right], \qquad (2.60)$$

where the factor 0.95 ensures the increase is not too large since the maximum temperature change is not a linear function of the time-step size. Then

$$f_{\Delta t} = \min\left(f_{\Delta t}, f_{input}, 2.0\right), \qquad (2.61)$$

where f_{input} was described above. The new time-step size is then calculated as

$$\Delta t_{new} = \min\left(\Delta t_{max}, f_{\Delta t} \times \Delta t_{old}\right), \qquad (2.62)$$

where Δt_{max} is an input value representing the largest time-step size allowed. If both options are specified, the code uses the smaller of the two new time steps.

If the new time-step size is greater than or equal to the old one, the code accepts the temperature distribution it has just calculated and moves on to the next time level. If the new time-step size is smaller than the old one, the code rejects the temperature distribution it has just calculated and returns to the old time level. It then calculates a new temperature distribution at the new time level using the smaller time-step size. If the code reduces the time-step size ten times, it writes out a warning message, reduces the time-step size to the minimum value, recalculates the temperature distribution using the new time-step size and moves ahead to the next time level. If a time-step size has been reduced, the code will not allow it to be increased again until five time steps have lapsed.

When the time-step size is varied implicitly, the time-step size and the associated criteria can be greatly affected by discontinuities in the boundary conditions, time-dependent functions, temperature-dependent functions, and the initial conditions.

As the transient calculations approach a printout time, the time-step size is automatically reduced to allow the temperature distribution to be printed at the specified time. The old time-step size is saved so that calculations can resume using the old time step after the printout.

2.4 TEMPERATURE-DEPENDENT THERMAL PROPERTIES

HEATING allows the thermal conductivity, k, the specific heat, c_p, and the density, ρ, to vary with temperature. A temperature-dependent density as implemented in HEATING does not satisfy conservation of mass. The code determines the conductivity of the material between two nodes, i and j, by evaluating the temperature-dependent conductivity at the average temperature of the connected nodes. This temperature is calculated as

$$\bar{T}^{(n)} = \frac{T_i^{(n-1)} + T_j^{(n-1)}}{2} \tag{2.63}$$

after completion of the nth iteration for steady-state problems, as

$$\bar{T}^n = \frac{T_i^{n-1} + T_j^{n-1}}{2} \tag{2.64}$$

after completion of the nth time step for transient problems involving one of the explicit techniques, and as

$$\bar{T}^{n+\Theta} = \frac{T_i^{n+\Theta} + T_j^{n+\Theta}}{2} \tag{2.65}$$

during the calculation of the temperature distribution at time t_{n+1} for transient problems involving the implicit procedure. The temperatures denoted as $T_i^{n+\Theta}$ in Eq. (2.65) are evaluated according to Eq. (2.54). For transient problems, the specific heat and density are determined for node i by evaluating the respective temperature-dependent function at T_i^n after completing the nth time step using one of the explicit techniques and at $T_i^{n+\Theta}$ as determined by Eq. (2.54) during the calculation of the temperature distribution at time t_{n+1} using the implicit procedure. In addition, the thermal conductivity of a material can be anisotropic.

2.5 BOUNDARY CONDITIONS

HEATING can accommodate a variety of boundary conditions. A boundary condition is applied along a surface of a region, and heat is transferred from a surface node to a boundary node or from a node on one face of a region to a node on the opposing face of the region. Surface nodes are nodes on the face of a region that are not covered by another region containing a material. Boundary nodes are dummy nodes used to represent the temperature of the environment to which a surface is exposed. Boundary temperatures are specified as input to the code. These temperatures are only used to calculate the heat flow across a boundary surface. The boundary condition types are the following:

1. the temperature on the surface of a region can be constant or a function of time and/or position;

2. the heat flux across the surface of a region can be constant or a function of time, position, and/or surface temperature;

3. the surface heat flux for a region can be specified indirectly by defining the heat transfer mechanism to be forced convection, radiation and/or natural convection; or

4. a combination of 2 and 3.

The temperatures of nodes on surfaces whose temperatures are specified are not calculated from Eq. (2.7) but are set equal to the specified value. The specified heat flux is multiplied by each node's surface area and the result is added to the heat generation term, P_i'', in Eq. (2.7).

The boundary condition types are surface-to-boundary (type 1), specified surface temperature (type 2), or surface-to-surface (type 3). Boundary conditions of the surface-to-boundary type are used to define heat transfer between a surface node and a boundary node. Surface-to-surface boundary conditions are used to define heat transfer between opposing surfaces. In this case, heat is transferred between a node on one surface to the corresponding node on the opposing surface. In Fig. 2.3, surface-to-surface boundary conditions could describe the heat transfer between nodes 1 and 2, nodes 3 and 4, and nodes 5 and 6.
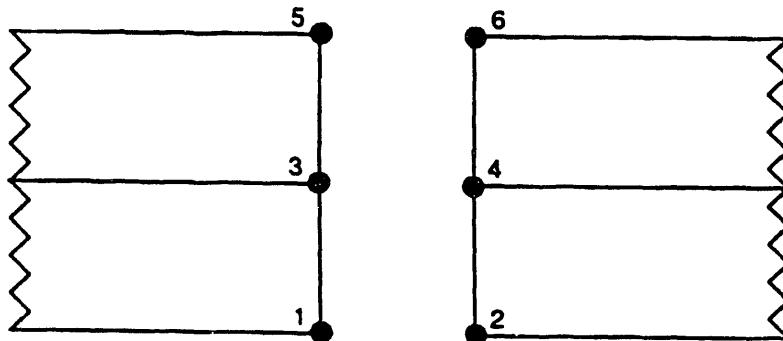
Fig. 2.3. Surface-to-surface heat transfer

For both surface-to-boundary and surface-to-surface boundary conditions, the heat flow term in Eq. (2.7) is

$$\left[{}_iK_{\alpha_m}(T_{\alpha_m}^n - T_i^n) \right] = {}_iK_b(T_b^n - T_i^n) , \qquad (2.66)$$

where ${}_iK_b$ is the effective conductance from surface node i to boundary node b or the opposing surface node b. T_b^n is either the temperature of boundary node b or the opposing surface node b at time t_n. The effective conductance is

$$_iK_{\alpha_m} = hA , \qquad (2.67)$$

where h is the effective heat transfer coefficient, and A is the surface area of node i associated with the boundary condition.

The effective heat transfer coefficient is

$$h = h_c + h_r\left[(T_i^n)^2 + (T_b^n)^2\right]\left[T_i^n + T_b^n\right] + h_n\left|T_i^n - T_b^n\right|^{h_e} , \qquad (2.68)$$

where

$$
\begin{aligned}
h_c &= \text{forced convection heat transfer coefficient,} \\
h_r &= \text{radiative heat transfer coefficient,} \\
h_n &= \text{coefficient for natural convection, and} \\
h_e &= \text{exponent for natural convection.}
\end{aligned}
$$

The parameters h_c, h_r, h_n, and h_e must be specified by the user and can be time- and/or temperature-dependent. When h_c, h_r, h_n, and h_e are temperature-dependent, they are evaluated at the average temperature of the opposing surface-nodes for surface-to-surface boundary conditions. For surface-to-boundary type boundary conditions, h_c, h_n, and h_e are evaluated at the average temperature of the related surface node and boundary node, whereas h_r is evaluated at the temperature of the surface node. For surface-to-boundary-type boundary conditions, the boundary temperature, T_b^n, must also be supplied by the user and may be a function of time and/or position. If the temperatures are entered in either °F or °C, the code converts them to absolute degrees when calculating the effective thermal conductance due to radiation. In computing the effective conductance for a surface-to-surface boundary condition across a radial gap, the code uses the surface area at the smaller radius bounding the gap. One may simultaneously model surface-to-surface heat transfer across a region as well as conduction through the region.

2.6 CHANGE OF PHASE

Selected materials are allowed to undergo up to five phase changes during an explicit transient calculation. Change of phase problems cannot be solved using implicit transient solution techniques in this version of HEATING. The temperature of a node that can change phase is calculated normally until a transition temperature is reached. The progress of the phase change at the node is indicated by X_i, which is the ratio of heat that has been absorbed after the transition temperature has been reached to the total heat needed to complete the phase change for a material in node i. If a job is restarted the initial melting ratio will be read from the restart file, otherwise the initial melting ratio is calculated as

$$X_i = \begin{cases} 0.0 \ , \ T_i^o < T_{melt} \\ 1.0 \ , \ T_i^o \geq T_{melt} \end{cases} , \qquad (2.69)$$

where

T_i^o $\quad$ = initial temperature of node i, and

T_{melt} $\quad$ = phase-change or transition temperature associated with node i.

If the melting ratio of a node is zero, its temperature is allowed to increase. When the temperature reaches the transition temperature the temperature of the node is held at the transition temperature, and the material is allowed to change phase in the following manner. The incremental change in melting ratio over the nth time step is calculated as

$$\Delta X_i^n = \frac{\Delta q_i^n}{\rho_{i,m}^n H_m V_{i,m}} , \qquad (2.70)$$

where

Δq_i^n $\quad$ = net heat input to node i during the nth time step,

$\rho_{i,m}^n$ $\quad$ = density of material m evaluated at T_i^n ,

H_m $\quad$ = latent heat of material m, and

$V_{i,m}$ $\quad$ = volume of material m associated with node i.

This incremental change is added to the current value of X_i at each time step until X_i exceeds unity. Any excess heat remaining if the ratio exceeds 1.0 is used to change the temperature of the node as follows:

$$\Delta T_i^n = (X_i^n - 1.0)\frac{\rho_{i,m}^n H_m V_{i,m}}{C_i^n} ,$$
(2.71)

where C_i^n is the heat capacitance of node i during the nth time step. After this temperature adjustment, the melting ratio is set to unity, and the temperature of the node is allowed to increase normally during future time steps.

Conversely, if the melting ratio of a node is unity, its temperature is allowed to decrease until it reaches the transition temperature of the material associated with it. Then, the temperature of the node is held at the transition temperature, and the material is allowed to change phase. The incremental change in melting ratio, Eq. (2.70), is added to the current value of X_i at each time step until X_i is less than zero. Any excess heat remaining after the ratio decreases below zero is used to change the temperature of the node as follows

$$\Delta T_i^n = X_i^n \frac{\rho_{i,m}^n H_m V_{i,m}}{C_i^n} .$$
(2.72)

Then, the melting ratio is set to zero, and the temperature of the node is allowed to decrease for future time steps.

If a node is associated with more than one material which can change phase or with a material with multiple phase changes, then each phase change is handled independently. The phase changes are ordered by increasing transition temperature. If the temperature of the node is increasing, then its temperature is not allowed to exceed the lowest transition temperature until the melting ratio increases from zero to unity. Once the melting ratio reaches unity, it is fixed there, and the temperature of the node is allowed to increase until it reaches the second transition temperature. Then, the melting ratio is reset to zero, and the temperature of the node is not allowed to increase until the melting ratio increases from zero to unity. Once the melting ratio reaches unity, the temperature of the node is allowed to increase again. This process is repeated until all phase changes associated with the node have occurred. The logic is similar when the temperature of the node is decreasing. However, the melting ratio of a node is set to zero when its temperature is between transition temperatures of the materials associated with it. In order to identify the actual phase(s) present in a nodal volume a phase indicator value is defined as

$$Y_i = (n_i - 1) + X_i ,$$
(2.73)

where X_i is the melting ratio for the phase change currently under way (or the last phase change that occurred) for node i and n_i is the ordinal number of the phase change occurring at node i. For problems involving phase change, this value is written to the plot data file for each output time.

2.7 NODE-TO-NODE CONNECTORS

HEATING allows the user to explicitly specify a thermal connection between any two nodes in the mesh or between a node and a boundary node. Any number of these connections can be specified. These connectors may be defined either in a separate file (indicated on Parameter Card 2) or in the CONNECTOR data block on the HEATING input file. In formulating the heat balance equation for a node having node-to-node connectors, additional heat flow terms in Eq. (2.7) are calculated as

$$\left[_iK_{\alpha_m}(T_{\alpha_m}^n - T_i^n) \right] = {}_iK_b(T_b^n - T_i^n) \ , \tag{2.74}$$

where $_iK_b$ is the effective conductance from node i to node b (node b is either a node in the mesh or a boundary node). Equation (2.74) is the same as Eq. (2.66) for surface-to-boundary or surface-to-surface boundary conditions. However, the effective conductance for node-to-node connectors is calculated as

$$_iK_b = {}_iM_b \left\{ h_c + h_r \left[(T_i^n)^2 + (T_b^n)^2 \right] \left[T_i^n + T_b^n \right] + h_n \left| T_i^n - T_b^n \right|^{h_e} \right\}$$

$$= {}_iM_b h \ , \tag{2.75}$$

where $_iM_b$ is a constant multiplier input by the user and the h terms are defined the same as for a boundary condition. There is a separate $_iM_b$ value for every node-to-node connector. The effective heat transfer coefficient, h, is calculated from information supplied on a BOUNDARY CONDITION card in the input. A heat-flow area must be included in the constant multiplier since HEATING does not compute an area for the calculation of the effective conductance.

3. SUGGESTIONS ON THE USE OF HEATING

3.1 GENERAL

This section provides information to aid in the development of models. This section is not intended to stand alone, but, rather, is to be used in conjunction with Sect. 4, which provides a detailed input description. The basic steps involved in developing a HEATING model are the following:

1. select a geometry type compatible with the physical problem;

2. define the model with a set of regions;

3. define all materials referenced in the region definitions;

4. define all initial temperature definitions referenced in the region definitions;

5. define all heat generation referenced in the region definitions;

6. define all boundary conditions referenced in the region definitions;

7. define a grid structure overlaying the entire model;

8. define any analytical or tabular functions used in the definition of materials, initial temperatures, heat generation, or boundary conditions;

9. if the problem involves a transient calculation, specify the times at which temperature distributions are to be stored; and

10. specify the solution type (steady-state, transient, or a sequence of steady-state and transient solutions.)

Simple problems will not require all of these steps, whereas the use of some special features (e.g., enclosure radiation modeling) will require additional steps. A set of instructions and observations are presented in this section to aid the user in developing HEATING models.

3.2 GEOMETRY SELECTION

The first step in developing a model is to choose the HEATING geometry type that is the most representative of the geometry of the physical problem. The available geometries are listed in Table 3.1. In the HEATING model, all surfaces must coincide with a coordinate surface in the chosen geometry. This includes all surfaces at the interface between materials within the model, as well as the exterior surfaces of the model. If all surfaces in the physical geometry do not coincide with coordinate surfaces for any of the available geometries, then an approximation of the true geometry will have to be made. An example of a problem whose surfaces do not all coincide with coordinate surfaces is the axisymmetric pressure vessel shown in Fig. 3.1. The

surfaces bounding the bottom and side of the vessel coincide with coordinate surfaces in the r-θ geometry, but the surfaces bounding the top of the vessel do not. Two possible modeling alternatives for the top are shown in Fig. 3.2. All model surfaces now coincide with coordinate surfaces. Whenever an approximation of the true geometry is made, it will not be possible to accurately approximate the heat capacity (model volume), surface area, and conduction paths at the same time. The user will have to determine which of these aspects to try to match. If the surface area does not match, boundary conditions definitions (e.g., forced convection heat transfer coefficient) may need to be adjusted based on the ratio of actual-to-modeled surface area. Of the two choices illustrated in Fig. 3.2, the one on the left is probably the better choice. The stairstep approximation on the right can be improved by using more, finer stair steps, but doing this will require more effort on the part of the user to define the model and more nodes will be generated.

Table 3.1. Geometry types implemented in HEATING

Cylindrical		Cartesian		Spherical	
1	r-θ-z	6	x-y-z	10	r
2	r-θ	7	x-y	11	r-ϕ
3	r-z	8	x-z	12	r-θ-ϕ
4	r	9	x		
5	z				

3.3 REGIONS

Regions are used to describe the geometry of the model, the distribution of various materials within the model, the location of boundary conditions on the model surfaces, zones of heat generation within the model, and initial temperature distributions. Complex models are created by assembling several regions.

A region is defined by specifying lower and upper bounding coordinate surfaces along each of the coordinate axes in the model. In three dimensions, six surfaces are required to define a region; in two dimensions, four surfaces (or lines) are required; and in one dimension, two surfaces (or points) are required. All other characteristics of one-, two-, and three-dimensional regions are the same. The characteristics of a region are described below.

1. A region may contain, at most, one material. A gap region that does not contain a material may be included in the model to allow the application of a surface-to-surface boundary condition between parallel surfaces.

2. A heat generation may be specified for a region. The volumetric heat-generation rate may be time-, position-, and/or temperature-dependent within the region.

3. An initial temperature may be specified for a region (see Sect. 2.8). This initial temperature may be position-dependent within the region.
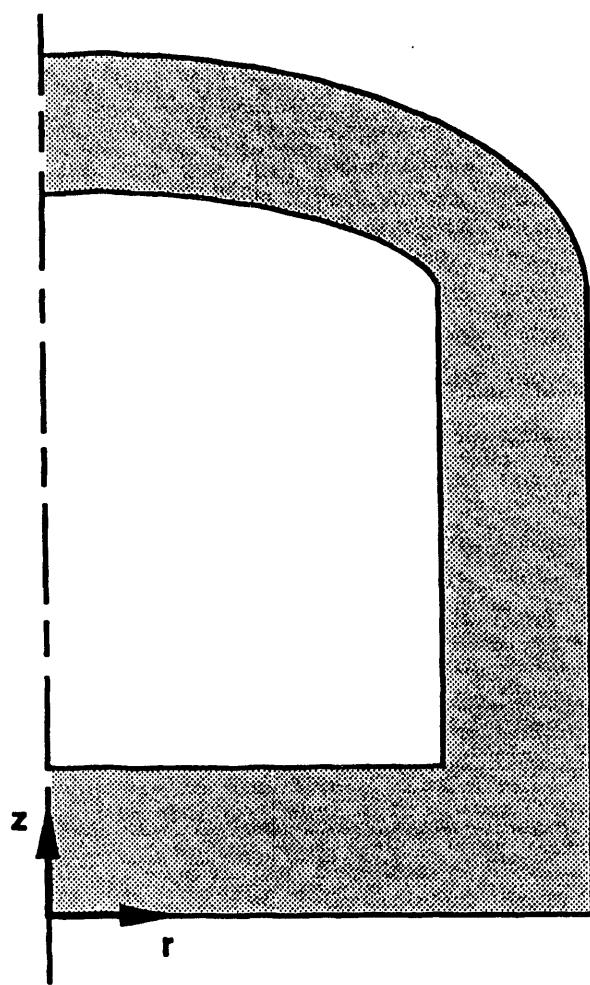
Fig. 3.1. Physical geometry with surfaces that do not coincide with coordinate surfaces.

— Modeled Shape

----- Actual Shape
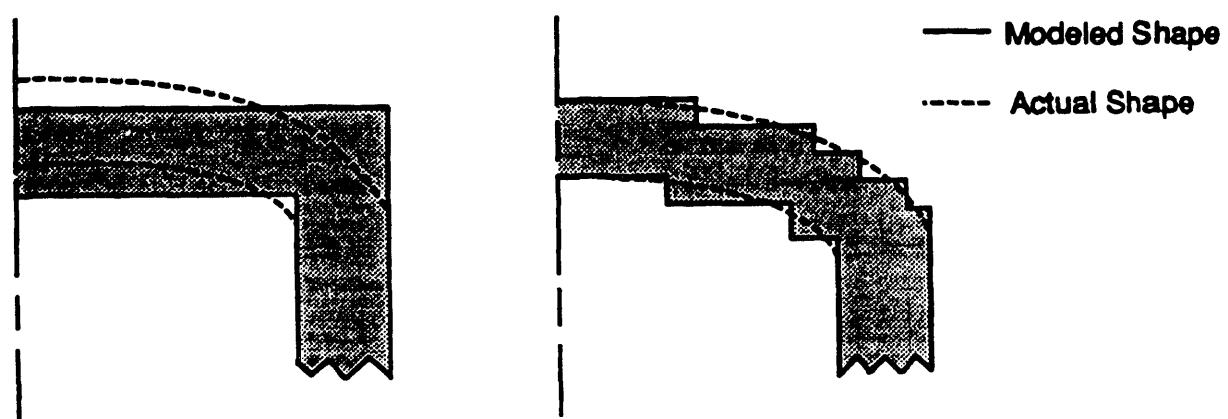
Fig. 3.2. Alternative approximations of physical geometry.

4. A surface-to-boundary-type boundary condition may be defined on any surface of a region if that surface is not covered by an adjacent region containing a material. The parameters specifying the boundary condition may vary with time, position, and/or temperature.

5. A surface-to-surface-type boundary condition may be defined on the two opposing surfaces of a region to model heat transfer between these surfaces (e.g., radiation or natural convection). The parameters specifying the boundary condition may vary with time, position, or temperature.

The geometric information about a region is input in the REGION data block along with reference numbers that identify the material, initial temperature, heat generation, and boundary conditions which are subsequently defined in the MATERIAL, INITIAL TEMPERATURE, HEAT GENERATION, and BOUNDARY CONDITION data blocks, respectively.

Consider, for example, a case consisting of a simple rectangle in x-y geometry, half of which contains one material, and the other half, a second material, as depicted in Fig. 3.3. This elementary case requires two regions (as indicated), one for each material. If the upper right corner of the rectangle is omitted as in Fig. 3.4, three regions are required. Regions 2 and 3 of Fig. 3.4 contain the same material.

Introducing boundary conditions as in Fig. 3.5, the left boundary of the left-most rectangle now contains two different boundary types. Thus, an additional region is required to account for the different boundary conditions. Any uncovered region surface that does not have a boundary condition specified is modeled as an adiabatic (or insulated) surface.

To specify heat transfer between parallel surfaces, a region is defined with a surface-to-surface boundary condition applied to the opposing surfaces. The boundary condition describing the heat transfer process (type 3, see Sect. 2.5) is applied along both of the surfaces of this region. The regions adjoining the parallel surfaces involving the surface-to-surface heat transfer must be defined and must contain a material. The region itself may or may not contain a material.

In Fig. 3.6, surface-to-surface heat transfer cannot be defined between the left and right boundaries of Region 3 since part of the area adjoining the right boundary is undefined. In Fig. 3.7 surface-to-surface boundary conditions can be applied along the left and right sides of Region 3.

If a surface-to-surface (type 3) boundary condition has been defined along a surface of a region and a surface-to-boundary (type 1) boundary condition is desired along the same surface, then the type-1 boundary condition must be applied along the surface of the adjoining region. In Fig. 3.7, surface-to-surface boundary conditions can be applied along the left and right sides of Region 3, and a surface-to-boundary boundary condition can be applied along the left side of Region 4. This can be done only if Region 3 is a gap region (i.e., it does not contain a material).
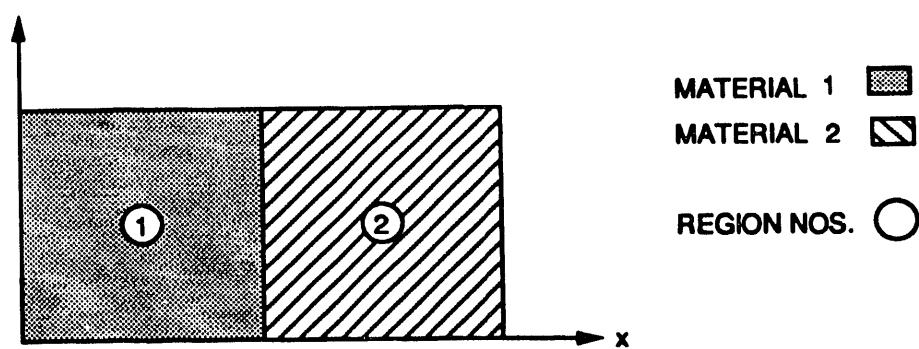
Fig. 3.3. Regions defining a two-dimensional rectangular model composed of two materials.
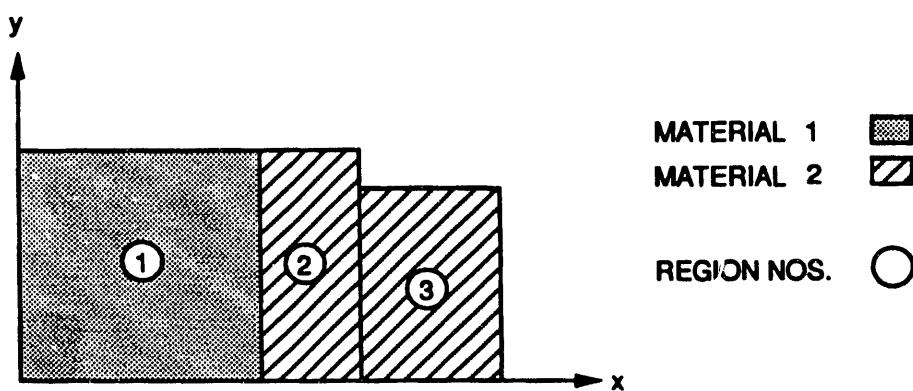
Fig. 3.4. Regions defining a two-dimensional rectangular model with indentation.
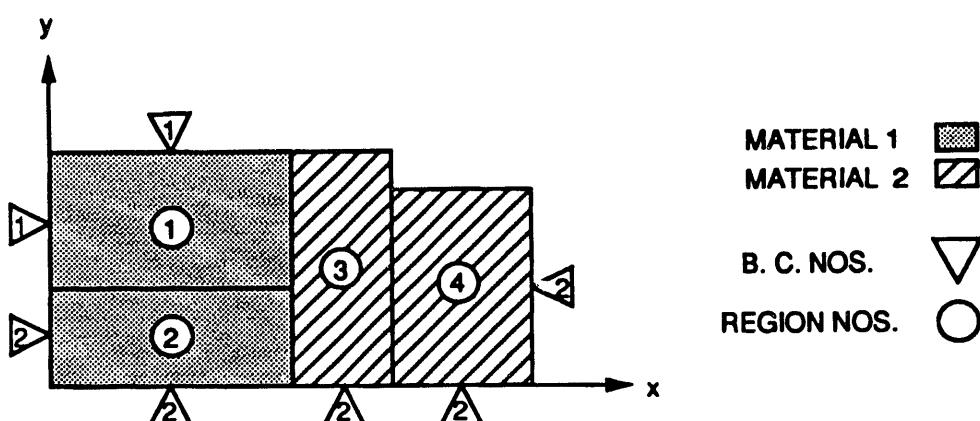
Fig. 3.5. Regions defining a two-dimensional rectangular model with boundary conditions.
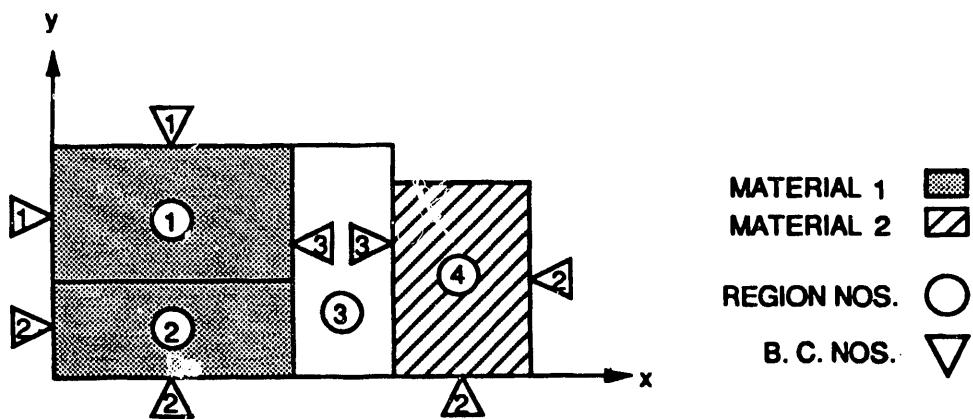
Fig. 3.6. Regions defining a two-dimensional rectangular model involving surface-to-surface boundary conditions, incompatible with HEATING.
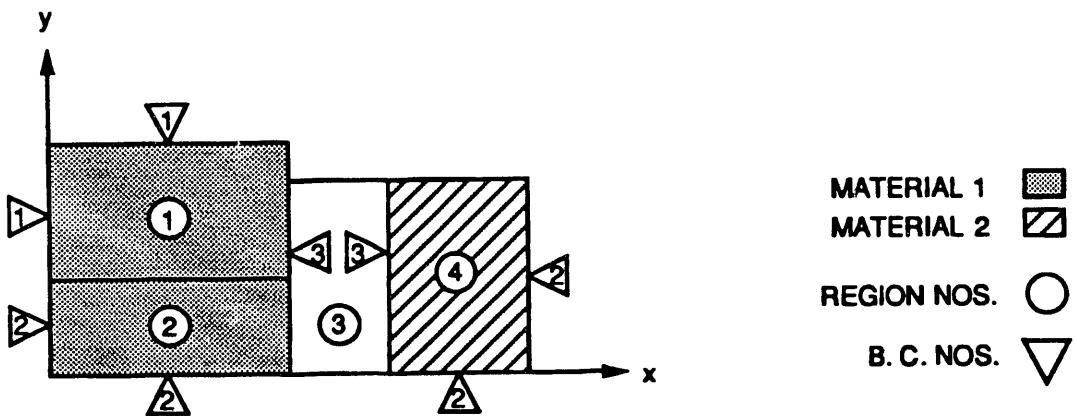
Fig. 3.7. Regions defining a two-dimensional rectangular model involving surface-to-surface boundary conditions.

Typically, a user generates a model by stacking regions as if they were physical building blocks; however, it is permissible to overlap or overlay regions (i.e., two or more regions are defined such that a portion of each occupies the same space.) When overlaying occurs, the material, boundary conditions, and heat generation for the last region defined in the input are used to generate the numerical model for the overlay zone. A summary of any overlaid regions is included in the output from a run.

3.4 DEFINITION OF REGION ATTRIBUTES

In defining the regions in the REGION data block, reference numbers are included to identify the material, initial temperature, heat generation, and any boundary conditions pertaining to the region. The input parameters that define these quantities are supplied in additional data blocks. Supplied in the MATERIAL data block are the thermal conductivity, specific heat, and density. The initial temperature and heat generation rate are given in the INITIAL TEMPERATURE and HEAT GENERATION data blocks, respectively. The parameters that define an effective heat transfer coefficient in Eq. (2.68) and any boundary heat fluxes are given in the BOUNDARY CONDITION data block.

If any of these parameters are a function of position, time, or temperature, reference numbers need to be supplied to indicate the tabular or analytical function defining this dependence. The tabular and analytical functions are supplied in the TABULAR FUNCTION and ANALYTICAL FUNCTION data blocks, respectively. If the desired dependence cannot be achieved with the built-in analytical function format, the user may supply his own as a user-supplied subroutine. Tabular functions, analytical functions, and user-supplied subroutines are discussed in subsequent sections.

The input parameters may be defined by a function having the following form:

$$P(x,y,z,t,T) = P_0 \cdot f(x,y,z,t,T) , \tag{3.1}$$

where P_0 is a constant factor and $f(x,y,z,t,T)$ may be a product of analytical and tabular functions, such as

$$f(x,y,z,t,T) = F_i(x) \cdot F_j(y) \cdot F_k(z) \cdot F_l(t) \cdot F_m(T) . \tag{3.2}$$

Table 3.2 shows the variable dependence of each parameter. If any variable is omitted from the definition of the parameter, then the corresponding factor is set equal to unity. The constant factor, P_0, is part of the input data, and its value appears on the data card that is used to define the parameter. If one of the input parameters is to be a function of position, time, or temperature, then the appropriate index or indices, i, j, k, l, or m, from Eq. (3.2) are entered on a data card, too. If an index is positive, then it defines the number of the analytical function for the respective variable. If it is negative, then the absolute value of the index defines the number of the tabular function for the respective variable. If any of the defined factors for a parameter are omitted from the input data, then that particular factor is set equal to unity in Eq. (3.2). If none of the factors are defined in the input, then that particular parameter is treated as a constant. If the value of P_0 is zero or is left blank on the data card and if the data indicate that the parameter is to be a function of position, time or temperature, then P_0 is set equal to unity in Eq. (3.1).

When setting up a tabular or analytical function defining the temperature dependence of any of the parameters associated with a boundary condition, it is necessary to know at what temperature the function is to be evaluated. In some situations the actual nodal temperature on the surface is used, and sometimes an average temperature (average of two nodal temperatures or average of a nodal and boundary temperature) is used. Table 3.3 defines which temperature

will be used to evaluate temperature-dependent boundary condition parameters in different situations.

If the parameter cannot be defined by a product of analytical and tabular functions as indicated in Eq. (3.2), then the user may supply a subroutine to evaluate $P(x,y,z,t,T)$. Table 3.2 contains each input parameter and the name of the corresponding subroutine that must be supplied if the user wishes to create a function. Table 3.2 also includes the independent variables that may be used to define each parameter. For further details involving user-supplied subroutines, see Sect. 3.8.1.

Table 3.2 Dependence of input parameters

Input Data Block	Parameter	Function of					User-supplied Subroutine
		x	y	z	t	T	
MATERIAL	k				$(\checkmark)$	$\checkmark$	CONDTN
	ρ				$(\checkmark)$	$\checkmark$	DNSITY
	c_p				$(\checkmark)$	$\checkmark$	CPHEAT
INITIAL TEMPERATURE	T_o	$\checkmark$	$\checkmark$	$\checkmark$	$(\checkmark)$		INITTP
HEAT GENERATION	Q	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	HEATGN
BOUNDARY CONDITIONS	T_b	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$		BNDTMP
	h_c	$(\checkmark)$	$(\checkmark)$	$(\checkmark)$	$\checkmark$	$\checkmark$	CONVTN
	h_r	$(\checkmark)$	$(\checkmark)$	$(\checkmark)$	$\checkmark$	$\checkmark$	RADITN
	h_n	$(\checkmark)$	$(\checkmark)$	$(\checkmark)$	$\checkmark$	$\checkmark$	NATCON
	h_e	$(\checkmark)$	$(\checkmark)$	$(\checkmark)$	$\checkmark$	$\checkmark$	NCONEX
	h_f	$(\checkmark)$	$(\checkmark)$	$(\checkmark)$	$\checkmark$	$\checkmark$	BNFLUX

NOTE: An $\checkmark$ indicates the parameter can be a function of the indicated variable using tabular or analytical functions. An $\checkmark$ enclosed in parentheses, $(\checkmark)$, indicates the parameter can only be a function of the indicated variable using the related user-supplied subroutine.

Table 3.3. Temperature used to evaluate temperature-dependent boundary condition parameters

Parameter	Boundary Condition Type		
	1 Surface-to-boundary	3 Surface-to-surface	1 or 3 Node-to-node
h_c	Average	Average	Average
h_r	Surface	Average	Average
h_n	Average	Average	Average
h_e	Average	Average	Average
h_f	Surface	Surface	Not Allowed

3.5 ANALYTICAL AND TABULAR FUNCTIONS

The analytical and tabular functions are built-in functions that may be used to aid in the description of the input parameters. An analytical function is defined by

$$F(v) = A_1 + A_2 v + A_3 v^2 + A_4 \cos(A_5 v) + A_6 \exp(A_7 v) + A_8 \sin(A_9 v) + A_{10} \ln(A_{11} v) . \quad (3.3)$$

A tabular function is defined by a set of ordered pairs, $(v_1, G(v_1)), (v_2, G(v_2)), \ldots (v_n, G(v_n))$, where the first element is the independent variable and the second is the corresponding function value. In order to evaluate the tabular function at some point, the program uses linear interpolation in the interval containing the point. The set of ordered pairs must have the independent variables arranged in ascending order, or

$$v_1 < v_2 < v_3 < \cdots < v_{n-1} < v_n . \quad (3.4)$$

If the function must be evaluated at some point outside of the defined domain of the function, then the value of the function will be

$$G(v) = \begin{cases} G(v_1) , & v < v_1 \\ G(v_n) , & v > v_n \end{cases} , \quad (3.5)$$

and a warning message is printed to inform the user.

3.6 MESH DEFINITION

Once the basic model has been constructed using regions, a lattice of nodal points at which temperatures are to be calculated is defined. The location of nodal points is determined by constructing a one-, two-, or three-dimensional mesh (or grid) in the applicable coordinate system to encompass the model. In a three-dimensional problem, the location of a nodal point is defined by the intersection of three coordinate surfaces (one for each of the coordinate axes). In a two-dimensional problem the intersection of two coordinate surfaces (or lines) define a nodal location, and in a one-dimensional problem only one coordinate surface (or point) is required. The values of coordinate surfaces intersecting at a nodal point give the coordinates of that point. HEATING constructs the grid with an internal mesh generator.

The definition of the grid is accomplished in two steps: First, a gross grid is defined along an axis by specifying two or more coordinate surfaces in ascending order. This gross grid is then selectively refined by specifying a number of equally spaced subdivisions to be made between each pair of consecutive gross grids. Specifying one subdivision does not insert any additional coordinate surfaces in the grid between those specified in the gross grid; specifying two subdivisions places one additional coordinate surface halfway between those specified in the gross grid, etc. This process is repeated for each coordinate axis appearing in the problem. In preparing input for HEATING this information is entered in the XGRID, YGRID, and/or ZGRID data blocks.

Two basic considerations influence the selection of the coordinate surfaces defining the gross grid:

1. All coordinate surfaces that bound regions should be specified in the appropriate gross grid definitions.

2. Additional coordinate surfaces may be specified between those bounding regions to allow localized refinement of the mesh.

In deciding the number of divisions to include between gross grid lines, the following points should be considered:

1. The mesh spacing needs to be fine enough to accurately represent the spatial variation of temperature. Stated another way, this means that the mesh needs to be fine enough so that the piecewise linear curve produced by plotting the temperatures vs nodal location along an axis matches the "true" continuous temperature distribution. It may not be possible to accurately judge this until after an initial execution. Sharp corners (large discontinuity in the slope of adjacent linear segments connecting nodal points) in a temperature-vs-distance plot that are not readily explainable in terms of the physical problem may be an indication of a grid that is too coarse.

2. Transitions from zones of fine-grid spacing to zones of coarser-grid spacing should be done gradually. Large step changes in the grid spacing can result in numerical difficulties and inaccurate solutions.

Once the grid has been established, the nodes are numbered starting with the grid location where all of the coordinates are at the minimum value occurring in the grid. If there is a material-containing region (not a void region) which includes this point, it is assigned node number one; otherwise, it is not assigned a node number, and the next point is checked. The order in which the grid points are checked is arrived at by varying the x (or r) coordinate most rapidly, the y (or θ) next most rapidly, and the z (or ϕ) least rapidly. Each point in the grid is checked, and those containing a material are numbered consecutively.

After the grid has been established and the nodes numbered, there is additional checking required for two- and three-dimensional cylindrical and spherical problems before the solution begins. In these geometries it is possible for more than one node to occupy the same physical location in the model. There can be numerous nodes sharing the same nodal locations whenever there is an inner radius of zero in an r-θ (cylindrical), r-θ-z (cylindrical), r-ϕ (spherical), or r-θ-ϕ (spherical) geometry. There can be two nodes at each of several nodal locations whenever the model subtends an angle of 2π in the θ direction in an r-θ (cylindrical), r-θ-z (cylindrical), or r-θ-ϕ (spherical) geometry. There can also be numerous nodes sharing nodal locations whenever a model is bounded by $\pm\pi/2$ in the ϕ direction in an r-θ-ϕ (spherical) geometry. In the output, all nodes sharing the same location have the same temperature. Information concerning shared nodal locations (if any exist) is summarized in the output and should be checked to confirm that the intended model is being used.

A model is assumed to subtend an angle of 2π in the θ direction if

$$6.2800 < \Delta\theta < 6.2836 , \tag{3.6}$$

where $\Delta\theta$ is the difference between the first and last θ coordinate surface defining the model. For a model in an r-θ-ϕ (spherical) geometry, the model is assumed to extend to $\pi/2$ in the ϕ direction if

$$1.5700 < \phi_{max} < 1.5709 , \tag{3.7}$$

where ϕ_{max} is the maximum ϕ coordinate surface defining the model. A similar check is made to see if the model extends to $-\pi/2$ in the ϕ direction.

3.7 SOLUTION TECHNIQUES

The STEADY-STATE and/or TRANSIENT data blocks are used to specify a solution or sequence of solutions to be performed. The order in which these data blocks occur in the input data specifies the order in which the solution will be carried out. Each of these data blocks may occur as many times as desired by the user. The following techniques are available in HEATING: three steady-state solution techniques (SOR, direct, and conjugate gradient), two explicit transient solution techniques (classical explicit and Levy modified explicit), and two implicit transient solution techniques (Crank-Nicolson and fully implicit). HEATING should not be treated as a black box that automatically yields the correct solution. The user must decide upon an appropriate solution technique for a particular problem. Care must be exercised in correctly simulating the physical problem as well as in interpreting the results.

3.7.1 Steady-State Solution Techniques

For steady-state problems, the mesh spacing can significantly affect the solution. (This is also true for transient problems.) Until a user has had enough experience to develop a feel for what constitutes an adequate mesh spacing for a particular problem, solutions should be obtained for several, progressively finer mesh spacings and compared. In addition, one must pay particular attention to the convergence criterion.

3.7.1.1 Successive-overrelaxation

In the SOR technique the convergence check is based on the relative temperature change between two successive iterations, not on an energy balance. It is possible to have a problem that is converging so slowly that the convergence criterion is satisfied even though the last iterate is a poor estimate of the true solution. To ensure that this is not the case, the user should check the overall energy balance on the problem by summing the energy flow rates due to boundary conditions and heat generations. This information is printed in the output at each specified printout time. The SOR solution technique can generally be sped up by providing a good estimate of the initial temperature distribution. A poor estimate of the initial temperature has been observed to prevent convergence to the true solution for some highly nonlinear problems (e.g., a two-part model coupled only by radiation connectors). For nonlinear problems, a faster solution can often be obtained by not reevaluating the temperature-dependent properties every iteration. However, for some problems, convergence problems occur if temperature-dependent properties are not evaluated every iteration. Convergence problems have also been observed in the SOR technique with problems whose temperature range spans zero (i.e., both positive and negative temperatures occur in the solution). Division by zero can occur or erratic behavior of the extrapolation technique may hinder or even prevent convergence. This situation can be avoided by using an absolute temperature scale for these problems.

Some situations have been observed where, because of loss of significance in the calculations, the solution does not continue to move closer to the solution with additional iterations. This situation can occur when the range of conductances (kA/l or Ha) from a node to its neighbors spans "several" orders of magnitude because of a large change in mesh spacing and/or large differences in thermal conductivity of adjacent regions. Ideally the model should be redefined to minimize these difficulties, but this is not always possible. These SOR convergence problems prompted the inclusion in HEATING of the direct-solution and conjugate gradient techniques discussed below.

3.7.1.2 Direct-solution

The direct-solution technique is implemented for one- and two-dimensional problems, and, unless the bandwidth is excessively large, it is the preferred steady-state-solution technique. For a linear problem using the direct-solution technique the initial temperature distribution has no impact on the time required to obtain the solution since only one iteration is ever required. The initial temperature distribution can significantly impact the solution time for highly nonlinear problems. The choice of initial temperature can preclude obtaining a solution when a natural convection boundary condition is specified. If the initial temperature is the same as the boundary temperature, the initial conductance for heat transfer to that boundary is zero which can lead to an oscillation in the solution. Some highly nonlinear problems may tend to oscillate

regardless of the initial temperature distribution chosen.

The coefficient matrix used in the steady-state direct-solution technique is a banded matrix. The matrix used by the code is dimensioned MWIDTH by MAXPTS, where MWIDTH is the maximum bandwidth and MAXPTS is the maximum number of nodes. In order to minimize the computer memory and the computing time required to solve a specific problem, the user should set MAXPTS to be exactly the number of nodes in the model and MWIDTH to be exactly the bandwidth of the model. However, there is no advantage to making MWIDTH smaller than 7.

For a one-dimensional problem with no surface-to-surface or node-to-node connections, the bandwidth, MWIDTH, is three. For two-dimensional problems with no surface-to-surface or node-to-node connections, the bandwidth is twice the number of fine lattice lines along the x (or r) axis plus one. For one- or two-dimensional problems with surface-to-surface or node-to-node connections, the bandwidth will be twice the maximum difference between the node numbers of connected nodes plus one.

Since the bandwidth for a two-dimensional problem is a function of the number of fine lattice lines along the x (or r) axis, the user should set up the geometry of his model to minimize this number if possible. For example, if one is running a two-dimensional x-y problem with no surface-to-surface connections, it would be better to have two fine lattice lines along the x axis and ten fine lattice lines along the y axis, giving a bandwidth of 5, rather than having ten fine lattice lines along the x axis and two fine lattice lines along the y axis, giving a bandwidth of 21.

3.7.1.3 Conjugate grad. ent

A three-dimensional direct-solution technique is not implemented in HEATING due to the inherently large bandwidths associated with these problems. Therefore, the conjugate gradient technique is included to give the user an alternative to the SOR technique. The conjugate gradient technique is more tolerant of a poorly conditioned coefficient matrix than the SOR and will often converge when the SOR will not. Without roundoff the conjugate gradient procedure converges to the "exact" solution in n iterations for a linear problem, where n is the number of unknowns. Since poorly conditioned matrices and roundoff are concerns, HEATING allows a maximum of $2n$ iterations to obtain a solution. A better estimate of the initial temperature may reduce the required number of iterations slightly, but testing has shown that the technique is fairly insensitive to the initial temperature distribution.

An outer iterative loop is used to solve nonlinear problems. The linear conjugate gradient method is used to solve the system of equations, the coefficient matrix and right-hand-side vector are reevaluated based on the new temperature estimates, and the new linear system of equations is solved. This process is continued until the newest temperatures create an energy balance that satisfies the convergence criterion. Since a linear solution is obtained before the properties are updated, the same precautions on choosing the initial temperature distribution discussed for the direct-solution technique apply here. Similarly, the solution may tend to oscillate for highly nonlinear problems.

3.7.2 Transient Solution Techniques

As with the steady-state solutions, the mesh spacing can significantly impact the accuracy

of transient solutions. Once again, the user should experiment with the mesh spacing to ensure that an acceptable solution is obtained. Using an appropriate time step in the calculations is also crucial.

3.7.2.1 Explicit

With the classical explicit solution, the time step must be less than or equal to the stability criterion in order to yield a stable solution. The code calculates the stability criterion and will use it for the time step if the user does not specify that a smaller time step be used. The Levy modification of the classical explicit technique removes the restriction that the time step satisfy the stability criterion in order to produce a stable solution. This technique is stable for any size time step, but an accurate solution can generally only be obtained if the time step is less than the CEP stability criterion for a significant portion of the nodes (Levy suggests "somewhat over half"). However, for either of these techniques, particularly the Levy method, a stable solution does not necessarily mean an accurate solution.

3.7.2.2 Implicit

In general, an implicit solution technique can make use of a time step that is significantly larger than the explicit stability criterion, but considerable care must still be taken in choosing the time step that produces an accurate solution. A solution produced with the Crank-Nicolson technique tends to oscillate with time − particularly when a transient has a step change in a boundary condition. Although this oscillation eventually dies out, it can have a severe impact on the solution. This oscillation can be minimized by using small time steps with the Crank-Nicolson technique early in the transient and larger time steps later, or the transient can begin with an explicit solution and later switch to the Crank-Nicolson technique. The user needs to experiment with the time step to ensure that an accurate solution is obtained.

3.8 USER-SUPPLIED SUBROUTINES

3.8.1 Defining Input Parameters

Subroutines may be supplied by the user to evaluate any of the parameters listed in Table 3.2. Thus, if an input parameter cannot be defined as the product of tabular and/or analytical functions as described in Sect. 3.6, the user may add his own computational technique for evaluating the parameter. All dependence on time, temperature, and/or position must be included in the user-supplied subroutine (i.e., tabular or analytical functions cannot be used in conjunction with a user-supplied subroutine to define part of the dependence). However, a user-supplied subroutine can access any of the tabular functions defined in the input. The user-supplied subroutine is referenced by specifying one of the factors of the parameter as an analytical function and by specifying no coefficients for the corresponding analytical function (i.e., leave the A2 card blank). Since this analytical function is only a flag to tell the code to call the appropriate user-supplied subroutine, the same analytical function can be specified for more than one parameter. The computational technique is then supplied in the subroutine associated with the parameter of interest (see Table 3.2).

HEATING contains dummy subroutines for each of the parameters listed in Table 3.2. If the user references one of the routines but fails to supply his own, then the code will write out an error message. Each user-supplied subroutine has the same argument list. However, all five independent variables are not initialized for each subroutine. Only the independent variables marked in Table 3.2 are initialized when each respective subroutine is called. The unit system in the user-supplied subroutines must be consistent with the input data unit system.

Subroutine BNDTMP, a typical, dummy user-supplied subroutine, is shown in Fig. 3.8. With the exception of HEATGN, all the other user-supplied subroutines are basically the same as BNDTMP, the only differences being the variables which are initialized (see Table 3.2) and the parameter which is being evaluated. Subroutine HEATGN is shown in Fig. 3.9. For the heat generation rate parameter, the time-dependent factor is evaluated independently of the other factors.

If the thermal conductivity of a material is anisotropic (i.e., directionally dependent), then it must be defined in user-supplied CONDTN (see Sect. 4.6).

A user-supplied subroutine may access any of the tabular functions that have been entered by calling subroutine TABLE. The calling sequence is

CALL TABLE(ARG,VAL,NTBPRS,NTAB,HIVAL,LOVAL,NTABLE,XIN,YOUT) ,

where

ARG	=	array containing independent variables for all tables,
VAL	=	HEATING array containing dependent variables for all tables,
NTBPRS	=	number of entries in table,
NTAB	=	translation table that relates the internal table numbers to the external (user-defined) numbers,
HIVAL	=	logical variable used to indicate whether table has ever been evaluated above the range of independent variables,
LOVAL	=	logical variable used to indicate whether table has ever been evaluated below the range of independent variables,
NTABLE	=	table number (internal numbering system) to be evaluated
XIN	=	value of independent variable, and
YOUT	=	value of dependent variable returned by TABLE.

The user should not change the values of any of these arguments except NTABLE, XIN, and YOUT. The value for NTABLE must be supplied in the numbering system used internal to the code, which is not necessarily the number entered in the TABULAR FUNCTIONS data block when defining the table. The first table defined is given the internal number of one, the second defined is two, et cetera. However, the internal numbering is less obvious when the material properties library is referenced. Each material referenced in the library can cause zero, one, or two tables to be defined. If the TABULAR FUNCTION data block precedes the

MATERIAL data block in the input, the first table defined by the user will always be table one in the internal numbering system.

```
      subroutine bndtmp(rvalue,r,th,z,tim,tsn,value,number,n,arg,val,
     . ntbprs,ntab,hival,loval)
c *****************************************************************
c when it is referenced, this subroutine must be supplied by the user to
c evaluate the user-supplied function defining the boundary temperature.
c this parameter may be time-dependent and/or position-dependent.
c the variables in the argument list are defined in the following table.
c (y) indicates that the variable has been defined when this subroutine
c is called.  (n) indicates it has not been defined.
c
c     variable  type ·        definition
c     rvalue    real          value of the user-supplied function
c                             evaluating the boundary temperature (this
c                             value must be computed by this routine). (n)
c                             name of variable in calling sequence --
c                             tdum(i).
c     r         real          x or r coordinate of node n. (y)
c     th        real          y or theta coordinate of node n. (y)
c     z         real          z or phi coordinate of node n. (y)
c     tim       real          time at which parameter is to be
c                             evaluated. (y)
c     tsn       real          temperature at which parameter is to be
c                             evaluated. (n)
c     value     real          constant value of the parameter which
c                             appears on its respective input card (b1) if
c                             it is non-zero.
c                             if it is zero, value contains 1.0. (y)
c                             name of variable in calling sequence --
c                             bytemp(nbdtp).
c     number    integer       index of parameter being evaluated. (y)
c                             name of variable in calling sequence - nbdtp
c     n         integer       node number. (n)
c     arg,val   real          arrays containing the independent and
c                             dependent values for tabular functions,
c                             respectively.  only used in calls to table.
c *****************************************************************
c
      double precision rvalue, r    , th   , z    , tim  , tsn  ,
     .                  value , arg  , val
c
      dimension        arg(1), val(1)

      (Insert the algorithm to compute the boundary temperature here.  If
      more than one boundary temperature is defined by this user-supplied
      subroutine, NUMBER defines the boundary condition whose temperature
      is to be calculated for the current call.)

      return
      end
```

Fig. 3.8. Dummy user-supplied subroutine BMDTMP.

```
      subroutine heatgn(rvalue,r,th,z,tim,tsn,value,number,n,arg,val,
     . ntbprs,ntab,nival,loval)
c *********************************************************************
c when it is referenced, this subroutine must be supplied by the user
c to evaluate the user-supplied function defining the volumetric heat
c generation rate.  this parameter may be time-, temperature- and/or
c position-dependent.
c
c the heat generation rate is calculated as
c          q = f(r,th,z,tsn) * g(tim)
c this subroutine may be called first to calculate the factor f.  then,
c this routine is called from the solution algorithm to update the
c  time-dependent factor g of the heat generation rate.  when this
c factor is updated, only tim, number, and value are defined where
c value = 1.0.  the two calls may be identified by the value of n.
c in the first call, n contains the number of the node whose heat
c generation rate is being evaluated.  in the second call, n is set
c to zero.
c
c the variables in the argument list are defined in the following table.
c (y) indicates that the variable has been defined when this subroutine
c is called.  (n) indicates it has not been defined.
c
c     variable   type        definition
c     rvalue     real        value of the user-supplied function
c                            evaluating the volumetric heat generation
c                            rate (this value must be computed by this
c                            routine). (n)
c                            name of variable in calling sequence - gen.
c     r          real        x or r coordinate of node n. (y)
c     th         real        y or theta coordinate of node n. (y)
c     z          real        z or phi coordinate of node n. (y)
c     tim        real        time at which parameter is to be
c                            evaluated. (y)
c     tsn        real        temperature at which parameter is to be
c                            evaluated. (y)
c     value      real        constant value of the parameter which
c                            appears on its respective input card (g) if
c                            it is non-zero.
c                            if it is zero, value contains 1.0. (y)
c                            name of variable in calling sequence --
c                            gen0(ngn).
c     number     integer     index of parameter being evaluated. (y)
c                            name of variable in calling sequence - ngn.
c     n          integer     node number. (y)
c     arg,val    real        arrays containing the independent and
c                            dependent values for tabular functions,
c                            respectively.  only used in calls to table.
c *********************************************************************
c
      double precision rvalue, r     , th    , z     , tim   , tsn   ,
     .                    value , arg   , val
c
      dimension       arg(1), val(1)
      if(n.eq.0) then
          (Insert algorithm to compute the time dependent portion of the
           volumetric heat generation rate here, if required.)
      else
          (Insert algorithm to compute the position and temperature dependent
           portion of the volumetric heat generation rate here, if required.)
      endif
      return
      end
```

Fig. 3.9. Dummy user-supplied subroutine HEATGN.

3.8.2 Monitoring Solution

Subroutine UMONTR is a user-supplied subroutine that gives the user additional capability to monitor the solution. UMONTR is called at the following times:

1. just prior to the evaluation of thermophysical parameters during initial model setup,
2. at the beginning of each steady-state or transient calculation,
3. at the completion of calculations for each time level for transient problems,
4. at the end of each iteration for steady-state problems.

Entry USRPRT in UMONTR is called after each printout of the temperature distribution to allow the user to print any additional information desired. All variably dimensioned arrays are available to the subroutine through the master arrays, C, IC, and LC, and the array pointers in labeled commons /MONREG/, /MONMAT/, /MONHGN/, /MONBCD/, /MONGRD/, /MONSTR/, /MONRPT/, and /MONIPT/. For example, the array containing the volumetric heat capacity is SMRCV. Its pointer is ISMRCV in labeled common /MONRPT/ or SMRCV(1)=C(ISMRCV). Thus, the volumetric heat capacity [see Eq. (2.2)] for node 25 is obtained by adding labeled common /MONRPT/ to subroutine UMONTR and referencing the ISMRCV plus 24th element in master array C. A dummy subroutine UMONTR is depicted in Fig. 3.10. Variables such as the current time, current time step, iteration number, etc., can be made available to UMONTR by adding the appropriate labeled common blocks. After obtaining some familiarity with the code, a user can incorporate coding in subroutine UMONTR to monitor certain values at specified nodes and times or to perform additional calculations. The user should not alter the value of any variable passed from HEATING. A few of the array pointers are listed in Table 3.4. More pointers are not defined here because use of most other variables requires a more detailed understanding of the code. If further information is desired, please contact the developers.

Table 3.4. Pointers for variably dimensioned arrays

HEATING Array	Location in Master Array	Array Definition
T1	C(IT1)	Temperature at nodal points
X1	C(IX1)	Phase fraction for nodal volume
SMRCV	C(ISMRCV)	Sum of $\rho \cdot c_p \cdot V$ for nodal volume
R	C(IR)	Fine grid lines along first axis
TH	C(ITH)	Fine grid lines along second axis
Z	C(IZ)	Fine grid lines along third axis
NTPI	IC(NTPI)	Index of fine grid line (first axis) passing through node
NTPJ	IC(NTPJ)	Index of fine grid line (second axis) passing through node
NTPK	IC(NTPK)	Index of fine grid line (third axis) passing through node

```
      subroutine umontr(c,ic,lc)
c **********************************************************************
c *   subroutine umontr allows the user access to any of the variably
c *   dimensioned arrays at the end of each iteration for steady state
c *   calculations or at the end of each time level for transient
c *   calculations by adding one or more of the labeled commons /mon???/.
c *   umontr called from sssor, dirsol, tranex, and tranim at the end
c *   of each iteration or time step.
c *   usrprt called from sumary at output times.
c **********************************************************************
c
      implicit double precision (a-h,o-z)
c
      dimension       c(1)
      integer         ic(1)
      logical         lc(1)
c
c **********************************************************************
c
      common /monreg/ noreg,matl,mats,intem,its,ngen,ngens,nregbc,
     . nregdm,iregdi
      common /monmat/ imatna,icondu,idenst,isphea,ixd,ixc,ixk,ixt,
     . ixtp,mat,ncontp,ndentp,nsphtp,munch,mcp
      common /monint/ intm,nitpos,itempi
      common /monhgn/ igen0,iggen,iptmf,iqhgnu,iqhgnn,ngn,ngnfcn
      common /monbdc/ nbdtp,nbytyp,nbytfn,nbtpos,ibhflg,nbctim,nbctem,
     . ibytem,ibcdef,iqbndu,iqbndn,itdum,itdum0,itdum1,lpdbtp,npdbtb,
     . npdbti,npdbtn
      common /mongrd/ ir,ith,iz,irg,ithg,izg,ndrg,ndthg,ndzg
      common /monstr/ nanalt,nparm,ia,iarg,ival,ntbprs,ntab,lhival,
     . loval,iprtim,nds
      common /monrpt/ it1,it2,icdlis,ivglis,ismrcv,ismvg,id,iu,iaa,it3,
     . it4,idiag,ih,neqnod
      common /monipt/ nb,nnb,npt,nvg,ntpi,ntpj,ntpk,mreg,nt1,nblist,
     . nns,nslist,nglist,npdbtp,ludcap,ludcon,ludgen,ludpro,lfixtp
      common /moncon/ nnc,nclist,nbnnc,icnlis,n1tmp,n2tmp,nbctmp
      common /monpc / islcap,ix1,nphsch,npclst
c
c **********************************************************************
c
c     these common blocks supply the number of nodes in the problem
c     and the current time (nt and tim, respectively).
c
      common /prbtyp/ nt     , ngeom , ntype , nset  , kf    , idegre
      common /transt/ tim    , ftime , deltat, ktmfct, nstpex, ltrans
      logical         ltrans
c
c **********************************************************************

      (Enter Fortran for any calculations and printout to be done at the
       end of each steady-state iteration or transient time step here.)

      return
      entry usrprt(c,ic,lc)

      (Enter Fortran for any calculations and printout to be done at
       normal printout times here.)

      return
      end
```

Fig. 3.10. Dummy user-supplied subroutine UMONTR.

3.9 ENCLOSURE RADIATION MODELING

Two approaches are available in HEATING for modeling radiation in enclosures. The first, and simplest to use, is the gap radiation model accessed by specifying a surface-to-surface boundary condition across a region. With this approach, radiation is modeled as strictly one-dimensional (i.e., a node can exchange heat by radiation only with the node that is directly across a region from itself). This approximation is accurate only for narrow gaps. For diffuse, gray surfaces, the h_r term in Eq. (2.68) is typically defined as

$$h_r = \frac{\sigma}{1/\epsilon_1 + (A_1/A_2)(1/\epsilon_2 - 1)} , \tag{3.8}$$

where σ is the Stefan-Boltzmann constant, A is an area, and ϵ is an emissivity. The subscripts distinguish between the two nodes, with 1 indicating a node on the surface having the smaller area (if the areas are not the same) and 2, a node on the surface with the larger area. This model is input by specifying the surfaces defining the gap in the REGION data block and the surface-to-surface boundary condition in the BOUNDARY CONDITION data block.

When the narrow gap assumption is not valid, a more realistic model is needed that defines radiant energy exchange between each radiation node and all other nodes in the model that it can "see." This more general radiation modeling technique can be handled within HEATING by including user-defined heat flow paths (node-to-node connectors) in the model. This technique is not as automated as most of the other HEATING model building techniques in that it requires the user to have knowledge of the node numbers and associated areas when preparing the input. The user can determine the node numbers for a geometrically simple problem before running the model, but it will generally be easier to build the model without the node-to-node connectors, make a preliminary execution of the code to obtain node numbers, and then set up the node-to-node connector input data.

A HEATING boundary condition (either type 1 or 3) is used to specify the heat transfer mechanism(s) that will be applied between nodes connected with node-to-node connectors. For radiation modeling, the term h_r in Eq. (2.75) should be set equal to the Stefan-Boltzmann constant in the appropriate units and all other h terms set to zero. A constant multiplier is entered in the input for each of the node-to-node connectors. This constant multiplier is equal to $\mathscr{F}_{ij} \cdot A_i$, where $\mathscr{F}_{ij}$ is the fraction of the energy emitted by surface i (if it is assumed to be a black surface) absorbed by surface j, and A_i is the area associated with node i. The quantity $\mathscr{F}_{ij}$ depends on the geometric arrangement of all surfaces in the enclosure and their emittances, and calculating them is a nontrivial exercise requiring specialized, stand-alone computer programs. Nodes to be connected and corresponding constant multipliers are entered in the CONNECTOR data block (Sect. 4.13) in the input or in a separate node-to-node connector file (Sect. 4.4.2), and the boundary condition defining the heat transfer mechanism is entered in the BOUNDARY CONDITION data block. An example of radiation modeling using node-to-node connectors is included as Sample Problem 2 in Appendix C.

3.10 INITIAL TEMPERATURE DEFINITION

The initial temperatures of nodes are assigned at four separate stages in the input processing, with initial temperatures assigned at a later stage overriding any previously assigned value. The sequence involved in determining the initial temperature of a node is as follows:

1. All nodes have a default initial temperature of zero.

2. If a node is in a region for which an initial temperature has been specified, the initial temperature is determined from the information in the appropriate INITIAL TEMPERATURE data block in the input. If a node is partially contained in more than one region, the initial temperature is calculated as a heat-content-average (volume-average if a steady-state where density and specific heat are not defined) of the specified initial temperatures of the materials associated with the node.

3. If a problem is being restarted, nodal temperatures are read from the plot file created by a previous run.

4. Nodes on boundaries having a specified surface temperature are assigned the temperature given by the appropriate BOUNDARY CONDITION data in the input.

3.11 MATERIAL PROPERTIES LIBRARIES

The program can extract material properties from a material properties library. Documentation for this library is included as Appendix D of this document. The default units are cm-g-s-cal-°C, but HEATING has the capability to convert these to a units system consistent with the other input data (Sect. 4.6.4).

The material properties given in the libraries are density, thermal conductivity, specific heat, transition temperature, and latent heat. The density of the material is normally either the value at or near room temperature, or the lowest temperature for which specific heat or conductivity is tabulated, whichever is highest. A single value for the thermal conductivity is given if it is constant or if the temperature dependence is unknown. When a table of conductivity vs temperature is listed, the table will be stored and used as a tabular function. A single value for specific heat is given if it is constant or if the temperature dependence is unknown. When a table of specific heat vs temperature is listed in the library, the table will be stored and used as a tabular function. The transition temperature is the temperature at which either a phase change or a solid-state transition occurs. The latent heat is the amount of heat absorbed by the material when the temperature is increased past the transition temperature. The transition temperature and the latent heat of a material will be stored only if the ninth entry on the ML card (Sect. 4.6.4) is not equal to zero.

Material conductivity and specific heat temperature-dependent functions will be used whenever they are given in the material properties library. The function numbers assigned to tabular functions read from the material properties library for the thermal conductivity and the specific heat of a material are MATNO+1000 and MATNO+10000 where MATNO is the

material number as read from the material properties library. (See Appendix G for a list of the materials and associated number.) When the material properties library is being used, tabular function numbers in the range 1001−9007 or 11001−19007 should not be assigned to the user-supplied tabular functions. A maximum of five phase changes are allowed, including phase changes in the materials from the material properties library.

Users may wish to create their own material properties library for a specific project or problem. The format is described in Appendix G. This user-created library can simplify model development and minimize the possibility of entering property data incorrectly. Instructions for specifying that HEATING use the user-created library rather that the default library are given in Appendix A. A user-created library can be in any consistent units system, but care must be taken not to use any of the automatic unit conversion features built into HEATING if the default system is not used.

4. INPUT DESCRIPTION

4.1 GENERAL

The input data for a HEATING run consist of an OPTIONS data block, a title card, three parameter cards, data blocks defining the model, data blocks defining output, and data blocks defining the solution technique. The OPTIONS data block allows the user to customize the size of variably dimensioned arrays to minimize memory requirements for a particular problem. There are eleven data blocks that can be used to define the model, two data blocks to define output options, and two data blocks to define the solution sequence. A data block consists of a keyword card, followed by the data cards for that block. The keyword starts in column 1, and at least the first four characters of the keyword must be entered without any embedded blanks. The data block keywords are given in Table 4.1.

The data blocks that define the model may appear in any order. A case always contains a REGIONS data block, a MATERIALS data block, and one or more of the grid-definition data blocks - XGRID, YGRID, and ZGRID. The inclusion of any other data block is dependent on the particular problem being solved. Input data cards are read until either a STEADY-STATE or TRANSIENT keyword is encountered, at which point the input phase ceases, the data are processed, and the model generated. The solution sequence indicated by STEADY-STATE and/or TRANSIENT data blocks is then performed. The final data card for a case is an end-of-data indicator card that contains a percent sign (%) in column 1 and a blank in column 2.

The user-defined data identification numbers for the regions, materials, initial temperatures, heat generations, boundary conditions, analytical functions, and tabular functions may be any unique positive integer value accepted by the computer. They do not have to be numbered consecutively.

The input data are read using free-form reading subroutines that allow data to be entered in an unstructured manner. Data items are separated by one or more blanks or by a comma. If a comma is used, there can be no intervening blanks between the previous data item and the comma. All character data, such as the material names, must be delimited by a blank, not by a comma. With a few exceptions noted below, data may be entered anywhere in columns 1 through 72. Labels may be entered in columns 73 through 80 of the data cards to aid in identifying the data. If the data will not fit on one card, an "@" appears in column 1 of each continuation card. There is no limit to the number of continuation cards. A continuation card may not immediately follow a title card or a keyword card. Comment cards may appear anywhere in the input data deck following the title card. The comment cards are identified by an asterisk (*) in column 1.

Real numbers may be entered in several forms (e.g., 1.733×10^{-4} may be entered as 1.733-4, 1.733E-4, 1.733D-4, or 0.0001733). Multiple entries of the same data value can be achieved by entering the number of repeats, followed by an asterisk (*), followed by the data value to be repeated. For example, 5*2 enters five successive values of 2 in the input data. There must not be any blanks between the number of repeats and the repeat flag (*). Trailing zero entries at the end of a data card may be omitted.

Any consistent set of units may be used. For problems involving radiation, the default for temperature units is degrees Fahrenheit. If the temperature units are either degrees Celsius or absolute degrees, the temperature units flag must be set (see Sect. 4.4.1). The units associated with the algorithms that appear in user-supplied subroutines must be consistent with those of the input data.

In the sections that follow, the characters (A), (I), or (R) appearing after the heading for some of the entries indicate that the entry is alphanumeric, integer or real, respectively.

Table 4.1. Data blocks

Data block keyword	Required characters	Defining
OPTIONS	OPTI	Memory requirements
REGIONS	REGI	Model
MATERIALS	MATE	
INITIAL CONDITIONS	INIT	
HEAT GENERATION	HEAT	
BOUNDARY CONDITIONS	BOUN	
XGRID	XGRI	
YGRID	YGRI	
ZGRID	ZGRI	
CONNECTORS	CONN	
ANALYTICAL FUNCTIONS	ANAL	
TABULAR FUNCTIONS	TABU	
PRINTOUT TIMES	PRIN	Output
NODES MONITORED	NODE	
STEADY-STATE	STEA	Solution Sequence
TRANSIENT	TRAN	

4.2 OPTIONS DATA BLOCK

Every array in HEATING whose length is dependent on the input data is variably dimensioned. HEATING makes an initial scan of the input data file to determine the appropriate size for many of these arrays. However, some of the array dimensions cannot be determined until the model is actually generated, and their values must be specified by the user if they are expected to exceed the default value. Table 4.2 lists each variable whose value can be altered in the OPTIONS data block and the default values. If the user wishes to override the default values used to allocate computer memory for the variably dimensioned arrays within HEATING, the OPTIONS data block must be supplied.

To supply an OPTIONS data block, the first card in the input data contains the keyword OPTIONS beginning in the first column. Beginning on the next card, the name of the variable whose value is to be changed is entered, followed by an equal sign and the desired value. Either a comma or a blank may be used to separate entries when more than one value is being altered. If the data will not fit on one card, each additional card begins with an "@" in the first column.

Table 4.2. List of variables set in OPTIONS data block

Variable	Default Value	Definition
MAXCP	0	Maximum number of distinct nodal phase changes
MAXPBT	1	Maximum number of position-dependent boundary temperature nodes
MAXPTS	50	Maximum number of nodes
MWIDTH	7	Maximum bandwidth in coefficient matrix for direct steady-state

4.3 CASE TITLE

This card contains a descriptive title for the problem in the first 72 columns. The card itself cannot be omitted although it may be left blank. This title serves to identify the output associated with a case.

4.4 PARAMETER CARDS

4.4.1 Card P1

1. CPU Time for Problem (R)

   This entry is the amount of CPU time (in seconds) to be allocated for the problem to run. If the CPU time at the end of an iteration or time step is greater than or equal to this entry, the calculations are terminated.

2. Geometry Type (I)

   This entry specifies the geometry to be used for this problem. The available geometries are listed below:

Cylindrical		Cartesian		Spherical	
1	r-θ-z	6	x-y-z	10	r
2	r-θ	7	x-y	11	r-ϕ
3	r-z	8	x-z	12	r-θ-ϕ
4	r	9	x		
5	z				

3. Initial Time (R)

   If this is not a restart of a previous problem, this entry indicates the initial time for transient problems and the time at which the time-dependent functions are evaluated for steady-state problems.

   If this is a restart, the problem will be restarted at the last output time on the previous plot data file that is less than or equal to this entry. If this entry is zero or blank, the restart transient will resume at the time that the previous one terminated.

4. Temperature Units (I)

   This entry is only needed for problems involving radiation. It indicates the units of temperature:

Entry	Temperature Units
0	°F
1	°C
2	°R or °K

4.4.2 Card P2 - Optional Input/Output Files

1. Restart Flag (I)

   A non-zero value for this entry indicates a plot data file produced by a previous HEATING run is to be used to establish the initial temperature distribution for this run. The new plot data file produced by this run will contain the data from this file.

2. Node-to-Node Connectors Flag (I)

   A non-zero value for this entry indicates that node-to-node connector data are to be read from a file other than the normal input file. The connectors read from this file are in addition to any connectors entered in the CONNECTOR data block. The records in the node-to-node connector file contain the same information as the C1 and C2 cards in the CONNECTOR data block (Sect. 4.13), but this file is unformatted. All real variables are of length eight bytes, and integer variables are the default length for the particular computer being used.

3. Convergence Information Flag (I)

   A non-zero value for this entry indicates that convergence information is not to be written to the normal print file. The convergence information can be quite voluminous, particularly for implicit transient calculations. A positive entry directs this information to a separate output file,and a negative entry suppresses it.

4.4.3 Card P3 - Flags to Control Output

1. Output of Selected Information (I)

   The option to output selected information during the calculations exists to aid the user in the detection of input data errors and to assist in debugging and understanding cases that are not performing as anticipated. Several types of output are available using this feature: (1) a nodal description that consists of tables indicating the neighbors and regions corresponding to each node; (2) a list of nodal connectors for surface-to-surface boundary conditions; (3) thermal property tables containing the temperature, capacitance, power, and ordered pairs consisting of each neighbor and the effective conductance between the node and the neighbor for each node in the model along with information concerning the temperature dependence for the heat capacitance, effective conductance, and power associated with each node; tables containing phase change information at each node; and the temperature distribution after each iteration or time step. The thermal property tables may be output either for the initial time level only or for each time the thermal properties are reevaluated. The temperature distribution may be output at the end of each iteration in the steady-state techniques, at the end of each time step for the explicit transient methods, and at the end of each iteration in either the nonlinear or the linear loop for the implicit transient calculations. All output using this feature is generated on the standard output unit. The user should exercise caution in using this feature since it can generate excessive output. The value of the flag for each desired option is located in the following

list, and the sum of these values is the integer that defines this option:

Option	Value of Flag
Nodal Description	1
Capacitance, Power, Conductance at Initial Time Level	10
Capacitance, Power, Conductance at Every Reevaluation	20
Temperature Distribution Every Steady-State Iteration	100 or 200
Temperature Distribution Every Transient Explicit Time Step	100 or 200
Temperature Distribution Every Nonlinear, Transient Implicit Iteration	100
Temperature Distribution Every Linear, Transient Implicit Iteration	200

Only one option involving output of thermal properties may be specified for a case. Also, only one option involving output of the temperature distribution may be specified for a case.

As an example, if the user wants the thermal properties output at the initial time level (value of flag is 10) and the temperature distribution output at every linear iteration (value of flag is 200), then a value of 210 (10 + 200) is entered.

2. Node-to-Node Connector Table Flag (I)

This entry is a flag to control the printing of a table of all node-to-node connectors for the problem. For problems with a significant number of radiation connectors, this table can be quite voluminous. The default is for the table not to be printed. A value of 1 is entered to obtain the table.

4.5 REGION DATA BLOCK

Each region is described by a pair of cards. At least one region must be defined.

4.5.1 Keyword Card

<u>REGI</u>ONS

4.5.2 Card R1

1. Region Number (I)

   This entry contains a user-defined identification number for the region. Region numbers must be unique positive integers, but they need not be numbered consecutively.

2. Material in Region (I)

   This entry is the identification number for the material that occupies the region. An entry of zero indicates the region does not contain a material (gap region).

3. Smaller X (or R) Region Dimension (R)

4. Larger X (or R) Region Dimension (R)

5. Smaller Y (or Θ) Region Dimension (R)

6. Larger Y (or Θ) Region Dimension (R)

7. Smaller Z (or Φ) Region Dimension (R)

8. Larger Z (or Φ) Region Dimension (R)

   If the problem is one- or two-dimensional, then the region dimensions for the unused coordinate or coordinates should be zero. For cylindrical or spherical geometries, r must be greater than or equal to zero. The values for θ and ϕ must be in radians. For spherical geometries, ϕ must fall in the range $-\pi/2 \le \phi \le \pi/2$. All bounding coordinate surface values entered in the REGION data block must also appear as gross grid lines in appropriate XGRID, YGRID, or ZGRID data blocks.

4.5.3 Card R2

1. Initial Temperature of Region (I)

    This entry is the identification number for the initial temperature to be used for the region. If this entry is zero, then the initial temperature for the region is zero. This entry should be zero for a gap region.

2. Heat Generation of Region (I)

    This entry is the identification number for the heat generation to be used for the region. If this entry is zero, then the code assumes that there is no heat generation for this region. This entry should be zero for a gap region.

3. Boundary Condition on Smaller X (or R) Region Boundary (I)

4. Boundary Condition on Larger X (or R) Region Boundary (I)

5. Boundary Condition on Smaller Y (or Θ) Region Boundary (I)

6. Boundary Condition on Larger Y (or Θ) Region Boundary (I)

7. Boundary Condition on Smaller Z (or Φ) Region Boundary (I)

8. Boundary Condition on Larger Z (or Φ) Region Boundary (I)

    Entries 3 through 8 are the identification numbers for the boundary conditions to be applied to the six surfaces of the region. A boundary condition should not be specified on a surface shared by adjacent regions unless it is a type-3 boundary condition, or unless one of the regions is a gap region. An entry of zero indicates an insulated surface.

4.6 MATERIAL DATA BLOCK (CARDS M, PC, AND ML)

One material should be defined for each unique material identification number appearing in the region definitions. An M card (and possibly a PC card) is required to describe each material when the user furnishes the material properties data. Alternatively, an ML card is required when the material data are to be read from the Lawrence Livermore National Laboratory's material properties library.

The user can specify that materials be allowed to undergo phase changes when using one of the explicit solution techniques. Phase change is not implemented for the implicit transient solution techniques. Any combination of single- and multiple-phase-change materials can be modeled as long as the total number of phase changes does not exceed five (e.g., one material can undergo up to five phase changes or five materials can each undergo a single phase change.)

Anisotropic thermal conductivity can be modeled by specifying the conductivity as temperature-dependent and the associated temperature-dependent function as user-supplied. The user supplies the anisotropic algorithm for the material in subroutine CONDTN (Sect. 3.8). The following labeled COMMON is needed in subroutine CONDTN:

COMMON /SURBCD/ NDIR

NDIR is an INTEGER containing a value of 1, 2, or 3 to indicate that the thermal conductivity is to be evaluated along the x (or r), y (or θ), or z (or ϕ) axis, respectively.

4.6.1 Keyword Card

MATERIALS

4.6.2 Card M (User-defined Material Properties)

1. Material Number (I)

    This entry contains the user-defined identification number for this material. Each material has a unique, positive identification number.

2. Material Name (A)

    This entry contains the name of the material. This name, which may consist of up to eight alphanumeric characters, is used only to aid in identification. It is terminated by one or more blanks (not a comma) and, thus, cannot contain any embedded blanks.

3. Conductivity (R)

    This entry contains the thermal conductivity if it is a constant. If the conductivity is time and/or temperature-dependent, this entry contains a constant that is multiplied by the function value(s) for the corresponding time and/or temperature dependence.

4. Density (R)

    This entry contains the density if it is a constant. If the density is time- and/or temperature-dependent, this entry contains a constant that is multiplied by the function value(s) for the corresponding time and/or temperature dependence.

5. Specific Heat (R)

    This entry contains the specific heat if it is a constant. If the specific heat is time- and/or temperature-dependent, this entry contains a constant that is multiplied by the function value(s) for the corresponding time and/or temperature dependence.

6. Conductivity Temperature-Dependent Function (I)

    This entry identifies the analytical or tabular function defining the thermal conductivity dependence on temperature.

7. Density Temperature-Dependent Function (I)

    This entry identifies the analytical or tabular function defining the density dependence on temperature.

8. Specific Heat Temperature-Dependent Function (I)

    This entry identifies the analytical or tabular function defining the specific heat dependence on temperature.

9. Phase-Change Flag (I)

    A non-zero entry indicates the material can undergo a change of phase. A PC card immediately follows this M card for a phase-change material. Change of phase is only implemented in the explicit solution techniques – not in the implicit solution techniques.

4.6.3 Card PC (Phase Change Parameters)

This card is included only if a positive value is entered as the ninth entry on the preceding M card. If a material can undergo multiple phase changes, entries 1 and 2 are repeated as many times as necessary. There can be a maximum of five phase changes in all of the materials.

1. Phase Change or Transition Temperature (R)

2. Latent Heat or Heat of Transformation (R)

4.6.4 Card ML (Material Properties Library)

1. Material Number (I)

   This entry contains the user-defined identification number for this material. Each material has a unique, positive identification number.

2. Material Name (A)

   This entry contains the material number from the material properties library preceded by an asterisk (*). This five-character name is used to locate the related data in the library. It is terminated by one or more blanks (not a comma) and, thus, cannot contain any embedded blanks.

3. Data Unit Conversion Control (I)

   This entry provides for data conversion from the unit system of the material properties library to a unit system consistent with the rest of the input. The conversion will be performed in accordance with the following table:

Entry	Problem Units
0	cm-g-s-cal-°C (library units, no conversion)
1	m-kg-s-J-°C
2	m-kg-s-J-K
3	ft-lb-hr-Btu-°F
4	in.-lb-s-Btu-°F
5	user-defined (conversion based on entries 4 through 8)

4. **Unit Conversion Factor for Density (R)**

This entry contains the conversion factor X_d for the density where the conversion will be performed as

$$\rho = X_d \cdot \rho_{library} .$$

5. **Unit Conversion Factor for Specific Heat (R)**

This entry contains the conversion factor X_c for specific heat where the conversion will be performed as

$$c_p = X_c \cdot c_{p,library} .$$

6. **Unit Conversion Factor for Thermal Conductivity (R)**

This entry contains the conversion factor X_k for thermal conductivity where the conversion will be performed as

$$k = X_k \cdot k_{library} .$$

7. **Unit Conversion Factor for Temperature (R)**

This entry contains the conversion factor X_T for temperature where the conversion is described in 8 below.

8. **Unit Conversion Factor for Temperature (R)**

This entry contains the conversion factor X_{TP} for temperature where the conversion will be performed as

$$T = X_T \cdot T_{library} + X_{TP} .$$

9. **Phase Change Flag (I)**

A positive value for this entry indicates the material can undergo a change of phase. If this entry is zero, the latent heat and transition temperature for the material is not read from the material properties library. If the third entry on this card is 5, then the latent heat is converted from the unit system in the library to the unit system of the problem by

$$H = X_c \cdot X_T \cdot H_{library} ,$$

where X_c and X_T are defined above.

4.7 INITIAL TEMPERATURE DATA BLOCK

One initial temperature should be defined for each unique initial temperature identification number appearing in the region definitions for this problem.

4.7.1 Keyword Card

INITIAL TEMPERATURES

4.7.2 Card I

1. Initial Temperature Identification Number (I)

   This entry contains the user-defined identification number for this initial temperature. Each initial temperature has a unique, positive identification number.

2. Initial Temperature (R)

   This entry contains the initial temperature if it is constant. If the initial temperature is position-dependent, this entry contains a constant that is multiplied by the function value(s) for the corresponding position dependence.

3. X (or R) Position-Dependent Function Number (I)

4. Y (or Θ) Position-Dependent Function Number (I)

5. Z (or Φ) Position-Dependent Function Number (I)

   Entries 3, 4, and 5 identify analytical or tabular functions, defining the x (or r), y (or θ), and z (or ϕ) dependence. If the problem is one- or two-dimensional or if the initial temperature does not vary along a particular axis of the region, then the position-dependent function parameter associated with that coordinate must be zero. A positive integer indicates an analytical function, and a negative integer indicates a tabular function.

4.8 HEAT GENERATION DATA BLOCK

One heat generation should be defined for each unique heat generation identification number appearing in the region definitions for this problem.

4.8.1 Keyword Card

HEAT GENERATIONS

4.8.2 Card G

1. Heat Generation Identification Number (I)

   This entry contains the user-defined identification number for this heat generation. Each heat generation has a unique, positive identification number.

2. Volumetric Heat Generation Rate (R)

   This entry contains the volumetric heat generation rate if it is a constant. If the heat generation is position-, time-, and/or temperature-dependent, this entry contains a constant that is multiplied by the function value(s) for the corresponding position, time, and/or temperature dependence. The units on heat generation are energy per unit volume per unit time. The heat generation rate for a region may be positive (heat source) or negative (heat sink).

3. Time-dependent Function Number (I)

4. Temperature-dependent Function Number (I)

5. X (or R) Position-dependent Function Number (I)

6. Y (or Θ) Position-dependent Function Number (I)

7. Z (or Φ) Position-dependent Function Number (I)

   Entries 3 through 7 give the analytical or tabular function identification numbers defining the time, temperature, and position dependence of the heat generation. If the heat generation rate per unit volume does not vary with time or temperature, then the time- or temperature-dependent function parameter must be zero. If the heat generation rate per unit volume does not vary along an axis or if the problem is one- or two-dimensional, then the position-dependent function parameter corresponding to that coordinate must be zero.

4.9 BOUNDARY CONDITION DATA BLOCK

One boundary condition should be defined for each unique boundary condition identification number appearing in the region definitions for this problem. Every boundary condition must have a B1 and B2 card. The B3 and B4 cards are optional. The B3 and B4 cards, if any, follow the B1 and B2 cards.

4.9.1 Keyword Card

BOUNDARY CONDITIONS

4.9.2 Card B1

1. Boundary Condition Number (I)

    This entry contains the user-defined identification number for this boundary condition. Each boundary condition has a unique, positive identification number.

2. Boundary Condition Type (I)

    This entry defines the boundary condition type according to the following table:

Entry	Boundary Condition Type
1	Surface-to-boundary
2	Prescribed surface temperature
3	Surface-to-surface

    If this entry is less than 1 or greater than 3, then no heat transfer connections will be made and the boundary will be treated as an insulated boundary.

3. Boundary Temperature (R)

    This entry contains the boundary temperature if it is constant. If the boundary temperature is time- and/or position-dependent, this entry contains a constant that is multiplied by the function value(s) for the corresponding position dependence. This entry is read, but not used, for a type-3 boundary condition.

4. Time-Dependent Function Number (I)

    This parameter identifies an analytical or tabular function. If the boundary temperature is independent of time or if the boundary type is 3, then this entry may be zero. A positive integer indicates an analytical function, and a negative integer indicates a tabular function.

5. X (or R) Position-Dependent Function Numbers (I)

6. Y (or Θ) Position-Dependent Function Numbers (I)

7. Z (or Φ) Position-Dependent Function Numbers (I)

Entries 5, 6, and 7 refer to analytical or tabular functions that define the position dependence along each axis. If the boundary temperature does not vary along an axis or if the problem is one- or two-dimensional, then the position-dependent function parameter corresponding to that coordinate must be zero. If any of the entries 5, 6, or 7 are nonzero, the user must ensure that MAXPBT is large enough to accommodate the total number of position-dependent boundary temperature nodes that will be generated by invoking this option. The values of MAXPBT may be changed by using the OPTIONS data block. A positive integer indicates an analytical function, and a negative integer indicates a tabular function.

4.9.3 Card B2

Up to six entries may be specified on this card. The first four entries are used to determine an effective heat transfer coefficient as given by

$$h_{eff} = h_c + h_r[T_s^2 + T_b^2][T_s + T_b] + h_n|T_s - T_b|^{h_e},$$

where

T_s = surface temperature,
T_b = boundary temperature, and
$h_?$ = heat transfer terms as defined below.

This card is left blank for a type 2 boundary condition.

1. Forced Convection Heat Transfer Coefficient (R)

   This entry contains a value for h_c, if h_c is constant. If h_c is time- and/or position-dependent, this entry contains a constant which is multiplied by the function value(s) for the corresponding time and/or position dependence.

2. Coefficient for Radiation (R)

   This entry contains a value for h_r, if h_r is constant. If h_r is time- and/or position-dependent, this entry contains a constant which is multiplied by the function value(s) for the corresponding time and/or position dependence.

3. Natural Convection Multiplier Term (R)

   This entry contains a value for h_n, if h_n is constant. If h_n is time- and/or position-dependent, this entry contains a constant which is multiplied by the function value(s) for the corresponding time and/or position dependence.

4. **Natural Convection Exponent Term (R)**

This entry contains a value for h_e, if h_e is constant. If h_e is time- and/or position-dependent, this entry contains a constant which is multiplied by the function value(s) for the corresponding time and/or position dependence.

5. **Prescribed Heat Flux (R)**

This entry contains a value for surface heat flux. If the heat flux is time- and/or position-dependent, then this value is multiplied by the function value(s) for the corresponding time and/or position dependence. For a type 1 (surface-to-boundary) boundary condition a positive heat flux indicates heat gain to the model from the surroundings, and a negative heat flux represents heat loss from the model to the surroundings. For a type 3 (surface-to-surface) boundary condition a positive heat flux indicates heat flow from the surface with the smaller coordinate value to the opposing surface (i.e., in the positive coordinate direction); and a negative heat flux indicates heat flow in the negative coordinate direction. If the opposing surfaces do not have the same area (e.g., radial heat flow in a cylindrical or spherical problem), the surface having the smaller area is used to calculate the heat flow.

6. **Time and Temperature dependence Flag (I)**

If any of the five preceding parameters are functions of time or temperature, then additional information must be entered on a B3 and/or B4 card. The flag for time and temperature dependence indicates whether or not the B3 and B4 cards are present for this particular boundary condition. Its value is determined according to the following table:

Dependence	Entry	Additional Cards
None	0	None
Time	1	B3 only
Temperature	2	B4 only
Time and Temperature	3	B3 and B4

4.9.4 Card B3

This card is only supplied when entry 6 on card B2 is equal to one or three. Each entry identifies an analytical or tabular function that defines the time-dependent function associated with the respective parameter on Card B2. Each entry corresponds to subscript l in Eq. (3.2). If an entry is zero, then the associated parameter is not time-dependent. A positive integer indicates an analytical function, and a negative integer indicates a tabular function.

1. **Forced Convection Coefficient Time-Dependent Function Number (I)**

2. Radiation Coefficient Time-Dependent Function Number (I)

3. Natural Convection Multiplier Time-Dependent Function Number (I)

4. Natural Convection Exponent Time-Dependent Function Number (I)

5. Heat Flux Time-Dependent Function Number (I)

4.9.5 Card B4

This card is only supplied when entry 6 on card B2 is equal to 2 or 3. This card is just like Card B3 except each entry identifies the analytical or tabular function that defines the temperature-dependent function associated with the respective parameter on Card B2. Each entry corresponds to subscript m in Eq. (3.2). A positive integer indicates an analytical function, and a negative integer indicates a tabular function.

1. Forced Convection Coefficient Temperature-Dependent Function Number (I)

2. Radiation Coefficient Temperature-Dependent Function Number (I)

3. Natural Convection Multiplier Temperature-Dependent Function Number (I)

4. Natural Convection Exponent Temperature-Dependent Function Number (I)

5. Heat Flux Temperature-Dependent Function Number (I)

4.10 XGRID DATA BLOCK

The XGRID data block should be omitted for a geometry type of 5. The gross grid lines along the x (or r) axis are entered on the X1 card in ascending order. All bounding x (or r) coordinate surfaces entered in the REGION data block must also appear as gross grid lines on the X1 card.

The X2 card specifies the number of equally spaced fine grid lines to be included between two adjacent gross grid lines. In particular, an entry on an X2 card specifies the number of equal increments between the gross grid line in the corresponding entry on the X1 card and the gross grid line immediately following it. Any X1 or X2 card that is a continuation card contains an "@" in column 1.

4.10.1 Keyword Card

XGRID

4.10.2 Card X1

The X1 cards correspond to the x (or r) coordinate.

4.10.3 Card X2

The X2 cards correspond to the x (or r) coordinate. There will be one less entry on this card than there are on the X1 cards.

4.11 YGRID DATA BLOCK

The YGRID data block should be omitted for geometry types of 3, 4, 5, 8, 9, 10, and 11. The gross grid lines along the y (or θ) axis are entered on the Y1 card in ascending order. All bounding y (or θ) coordinate surfaces entered in the REGION data block must also appear as gross grid lines on the Y1 card.

The Y2 card specifies the number of equally spaced fine grid lines to be included between two adjacent gross grid lines. In particular, an entry on a Y2 card specifies the number of equal increments between the gross grid line in the corresponding entry on the Y2 card and the gross grid line immediately following it. Any Y1 or Y2 card that is a continuation card contains an "@" in column 1.

4.11.1 Keyword Card

YGRID

4.11.2 Card Y1

The Y1 cards correspond to the y (or θ) coordinate.

4.11.3 Card Y2

The Y2 cards correspond to the y (or θ) coordinate. There will be one less entry here than there are on the Y1 cards.

4.12 ZGRID DATA BLOCK

The ZGRID data block should be omitted for geometry types of 2, 4, 5, 7, 9, and 10. The gross grid lines along the z (or ϕ) axis are entered on the Z1 card in ascending order. All bounding z (or ϕ) coordinate surfaces entered in the REGION data block must also appear as gross grid lines on the Z1 card.

The Z2 card specifies the number of equally spaced fine grid lines to be included between two adjacent gross grid lines. In particular, an entry on a Z2 card specifies the number of equal increments between the gross grid line in the corresponding entry on the Z1 card and the gross grid line immediately following it. Any Z1 or Z2 card that is a continuation card contains an "@" in column 1.

4.12.1 Keyword Card

ZGRID

4.12.2 Card Z1

The Z1 cards correspond to the z (or ϕ) coordinate.

4.12.3 Card Z2

The Z2 cards correspond to the z (or ϕ) coordinate. There will be one less entry here than there are on the Z1 cards.

4.13 CONNECTOR DATA BLOCK

The node-to-node connector data are entered in pairs of cards, C1 and C2. These paired cards may be repeated as many times as necessary to input all of the connector data.

4.13.1 Keyword Card

CONNECTORS

4.13.2 Card C1

1. Base Node Number (I)

   When the energy equation is generated by HEATING for this base node, an additional term will be added for each node-to-node connector specified in the input. An additional term will not be added to the equation for a node connected to this base node unless reciprocity is specified in entry 4 of this card.

2. Number of Nodes Connected to the Base Node (I)

3. Boundary Condition Number (I)

   This entry is the identification number of the boundary condition defining an effective heat transfer coefficient for the node-to-node connectors.

4. Reciprocity Flag (I)

   If this entry is zero, reciprocal connections will not be generated. If the entry is not zero, reciprocal connections will be generated. In a radiation problem the use of reciprocity will reduce the number of connectors that must be input by almost one-half. The use of reciprocity is recommended to avoid accidentally producing an asymmetric conductance matrix.

4.13.3 Card C2

The C2 card contains a set of paired entries. The number of pairs is equal to the number of connections to be added to the base node (entry two of card C1). The first number of each pair is an integer and gives the number of the node to be connected to the base node. A negative number entered as the node number indicates the connection is to a boundary node rather than a normal node. The temperature of the indicated boundary is used in solving the energy equation for the base node. This boundary condition does not have to be the same boundary condition entered in position three of the C1 card. The second number of each pair is the constant multiplier for the effective heat transfer coefficient given by the boundary condition used for this connection (entry three of card C1). As many pairs may be entered on a single card as can be fit into columns 1-72. The C2 card can be continued on additional cards as necessary by placing an "@" in column 1 of the continuation cards.

4.14 ANALYTICAL FUNCTION DATA BLOCK

One analytical function should be defined for each unique analytical function identification number appearing in the definition of materials, initial temperatures, heat generations, and boundary conditions for this problem. Each analytical function is described by an A1 card and one or more A2 cards. An analytical function is defined by

$$F(v) = A_1 + A_2 v + A_3 v^2 + A_4 \cos(A_5 v) + A_6 \exp(A_7 v) + A_8 \sin(A_9 v) + A_{10} \ln(A_{11} v) .$$

4.14.1 Keyword Card

ANALYTICAL FUNCTIONS

4.14.2 Card A1

1. Analytical Function Number (I)

    This entry is the user-defined identification number for this analytical function. Each analytical function has a unique, positive identification number.

4.14.3 Card A2

The A2 card contains a set of ordered pairs of data. If this card is left blank or if the first entry is a zero, a user-supplied subroutine will be used to evaluate the appropriate function. A continuation card must contain an "@" in column 1. Each ordered pair is defined as follows:

1. Coefficient Identification Number (I)

    This entry is an integer between 1 and 11 that identifies the coefficient whose value is the next entry on this card.

2. Coefficient Value (R)

4.15 TABULAR FUNCTION DATA BLOCK

One tabular function should be defined for each unique tabular function identification number appearing in the definition of materials, initial temperatures, heat generations, and boundary conditions for this problem. Each tabular function is described by a T1 card and one or more T2 cards. A tabular function is defined by specifying a set of ordered pairs. Each ordered pair contains an independent variable value and its dependent functional value; viz., v_1, $F(v_1)$; v_2, $F(v_2)$; etc. The values of v_i are entered in ascending order. Linear interpolation is used to evaluate the function between the points. If the function is evaluated either below or above the range of independent variable values, the first or last dependent functional value, respectively, will be used. At least two data pairs are required.

4.15.1 Keyword Card

TABULAR FUNCTIONS

4.15.2 Card T1

1. Tabular Function Number (I)

    This entry contains the user-defined identification number for this tabular function. Each tabular function has a unique, positive identification number.

4.15.3 Card T2

The T2 card contains the ordered pairs defining the tabular function. A continuation card must contain an "@" in column 1. Each ordered pair is defined as follows:

1. Independent Variable (R)

2. Dependent Variable (R)

4.16 PRINTOUT TIMES DATA BLOCK

This data block defines the times at which a solution summary is written to the print file and the temperature and phase distributions are written to the plot data file. This data block is not used for steady-state problems.

4.16.1 Keyword Card

PRINTOUT TIMES

4.16.2 Card O

This card contains the times in ascending order at which the standard output is performed. Each entry is a floating-point number. If the data will not fit on one card, continuation cards must contain an "@" in column 1.

4.17 NODES MONITORED DATA BLOCK

This data block allows the user to monitor the temperature of a few nodes throughout a steady-state or transient calculation. A table of nodal temperatures is written to a separate print file at the requested frequency.

4.17.1 Keyword Card

NODES MONITORED

4.17.2 Card S

1. Frequency for Special Monitoring of Temperatures (I)

   This entry specifies the number of steady-state iterations or transient time steps between printouts of the temperature of the nodes specified on the remainder of this card.

2. Node(s) to be Monitored (I)

   Nodes whose temperatures will be printed as a function of the number of iterations for steady-state calculations or the number of time steps for transient calculations. Any continuation card must contain an "@" in column 1.

4.18 STEADY-STATE DATA BLOCK

This data block is included whenever a steady-state solution is desired.

4.18.1 Keyword Card

STEADY-STATE

4.18.2 Card SS

1. Steady-State Solution Technique (I)

   This entry specifies the solution technique to be used for this steady-state calculation according to the following table.

Entry	Solution Technique
1	SOR
2	direct-solution (one- or two-dimensional problems only)
3	conjugate gradient

2. Maximum Number of Steady-State Iterations Allowed (I)

   This entry defines the maximum number of iterations allowed before the steady-state calculations are terminated. For linear problems, the direct-solution technique requires only one iteration to obtain the solution. If this entry is zero, the code will use a default of 500 iterations for the SOR technique or 20 iterations for the direct technique. For the conjugate gradient technique, this entry is the number of iterations in the nonlinear (outer) loop. The default value is 20. For the linear (inner) loop the maximum number of iterations is twice the number of nodes in the problem.

3. Steady-State Convergence Criterion (R)

   For the SOR technique, this entry contains the value of ϵ in Eq. (2.14) which defines the steady-state convergence criterion. For the direct-solution technique, this entry contains the convergence criterion as discussed in Sect. 2.2.2. The steady-state calculation will continue until the convergence criterion is met. Since the criterion which ensures convergence varies from case to case, the user must rely on judgment and experience in determining the correct value for his particular problem. If this entry is zero, the code uses a default of 10^{-5}.

4. Steady-State Overrelaxation Factor (R)

    This entry is the value of β [see Eq. (2.13)]. Its value must be in the range $1 \leq \beta < 2$. If a zero is entered for steady-state problems using the SOR solution technique, the code uses a default of 1.9.

5. Updating Temperature-Dependent Properties (I)

    This entry specifies the number of iterations that are allowed before the temperature-dependent thermal properties are reevaluated for steady-state problems using the SOR solution technique. Once the convergence criterion has been satisfied, this value is set to one. The calculations are continued until the convergence criterion is satisfied a second time. Most nonlinear problems will require less cpu time to reach convergence if the thermal properties are not evaluated at each iteration. The default value is 1.

4.19 TRANSIENT DATA BLOCK

This data block is included whenever a transient solution is desired.

4.19.1 Keyword Card

TRANSIENT

4.19.2 Card TR

1. Transient Solution Technique (I)

   This entry indicates the solution technique that is to be used for this transient calculation.

Entry	Technique	Additional Cards Required
1	Explicit	TR1
2	Implicit	TR2 and one or more TR3

2. Final Time (R)

   This entry indicates the final time for the transient.

4.19.3 Card TR1

   This card must be included whenever an explicit transient solution is desired (i.e., when Entry 1 on the TR card is 1.)

1. Initial Time Step (R)

   This entry contains the initial time step for the explicit transient calculation. If this value is less than or equal to the stability criterion (Eq. 2.35), the input value is used. If the input value is larger than the stability criterion (but not more than a factor of 10 larger), the time step is set equal to the stability criterion. If the input value is more than a factor of 10 larger than the stability criterion, the case is terminated. If a value of zero is entered, the initial time step is set equal to the stability criterion.

2. Levy Technique Option (I)

   This entry is the factor by which the stable time increment is multiplied to form the time increment for Levy's explicit method. If the entry is zero or 1, Levy's method will not be used. If Levy's explicit method is specified (greater than 1), the initial time step in the previous entry will not be used. The time increment is initially set to the stability criterion.

3. **Updating Temperature-Dependent Properties (I)**

    This entry is the number of time steps between reevaluation of temperature-dependent properties. If the Levy technique is to be used or if this entry is zero or blank, the properties will be reevaluated every time step. The maximum value permitted for this entry is 10.

4.19.4 Card TR2

This card must be included whenever an implicit transient solution is desired (i.e., when Entry 1 on the TR card is 2.) This entire card may be left blank or any of its entries may be zero, and the default values will be used. Default values are based on previous experience with the code but are not necessarily the best values for a given problem. They are probably good starting points.

1. **Implicit Time-Differencing Scheme (R)**

    This entry defines the implicit technique that will be used to solve the transient problem. It refers to Θ in Eq. (2.38) and must in the range $0.5 \leq \Theta \leq 1.0$. The default is 0.5.

2. **Convergence Criterion for Solution of Linear Equations (R)**

    This is the convergence criterion that must be met in order for the iterative technique to terminate successfully at each time step. This convergence criterion corresponds to ϵ_1 in Eq. (2.52). The default is 10^{-5}.

3. **Number of Iterations Between Convergence Checks for Linear Loop (I)**

    This entry indicates the number of iterations between tests for convergence in the linear loop. Experience has shown that for problems requiring large numbers of iterations (over 50) for the linear loop to converge, a significant savings in computer time can be achieved by not performing the calculations necessary for convergence tests at every iteration. However, the number entered applies for all of the transient calculations. The default value is 1.

4. **Convergence Criterion for Problems Involving Temperature-Dependent Parameters (R)**

    This convergence criterion applies to the average relative change in temperature over all nodes between two consecutive nonlinear iterations. It corresponds to ϵ_2 in Eq. (2.55). The default is 10^{-5}.

5. **Additional Convergence Criterion for Problems Involving Temperature-Dependent Parameters (R)**

This convergence criterion applies to the maximum absolute change in temperature at any node between two nonlinear iterations. The default is zero, so this criteria is not used unless explicitly specified by the user. This option is useful for problems that are only slightly nonlinear. By entering an appropriate value for this entry the nonlinear iteration within a time step can be eliminated, and temperature-dependent properties will be reevaluated only once per time step.

6. **SOR Acceleration Parameter Initial Value (R)**

This defines the initial value of the point-successive-overrelaxation iteration acceleration parameter [ω in Eq. (2.57)]. It also defines the method that will be used to update the acceleration parameter. If this entry is positive, then the acceleration parameter will remain constant throughout the calculations and will be equal to the value of this entry. If it is zero, then the acceleration parameter will be optimized empirically as a function of time. This appears to be the best option for nonlinear problems. If it is negative, then the acceleration parameter will be calculated using Carre's technique.[17] The absolute value of this entry must be less than 2.0.

7. **Time Steps Between Acceleration Parameter Optimization (I)**

This entry defines the number of time steps between attempts to optimize the acceleration parameter empirically [referred to as N_ω in Sect. 2.3.3.4]. It is used only when Entry 6 is zero. The default value is 1.

8. **Number-of-Iterations Criteria for Acceleration Parameter Updates (I)**

For the case when the acceleration parameter will be updated empirically (Entry 6 is zero), then this entry, the seventh, defines the change-in-number-of-iterations criterion [referred to as I_ω in Sect. 2.3.3.4] which must be met before the acceleration parameter will be updated. The default is 5. For the case when the SOR acceleration parameter will be updated using Carre's technique, this entry defines the number of iterations between updates. The default is 12.

9. **Change-in-Number-of-Iterations Criteria (I)**

The change-in-number-of-iterations criterion [referred to as J_ω in Sect. 2.3.3.4] which is used to determine when a good estimate to the optimum acceleration parameter has been found. This entry is used only when the acceleration parameter will be updated empirically (Entry 6 on this card is zero). The default is 2.

4.19.5 Card TR3

This card must be included whenever an implicit transient solution is desired (i.e., when Entry one on the TR card is 2.) When an implicit scheme is used to solve a transient problem, the time step may be variable. This allows the time step to increase as the solution smooths out and to decrease when some parameter varies rapidly with time. The information controlling the value of the time step is contained on one or more TR3 cards. The size of the time step is automatically adjusted in order to get printouts of the temperature distribution at the specified time. If the size of the coefficients in the system of equations varies by orders of magnitude (10^5 or greater), it has been observed that point-successive-overrelaxation iteration may converge very slowly (it may not converge at all). This occurs when the grid spacing or thermal properties vary by orders of magnitude over the problem. It can be observed by examining the stability criterion table in the output. If this appears to be happening, either further subdivide some of the larger nodes or combine some of the smaller ones. In some cases, it may help to use a larger time step.

1. Initial Time Step (R)

    If this entry is zero for the initial TR3 card, the initial time step will be equal to the stability criterion for the Classical Explicit Procedure. If this entry is zero for any TR3 card after the first one, the time step will be equal to the last one used subject to any constraints following on this card.

2. Time Step Multiplication Factor (R)

    After the temperature distribution is calculated at a time level, the current time step is multiplied by a factor to determine a new time step. The default value is 1.0. For many problems whose parameters vary mildly with time and/or temperature, values between 1.0 and 1.1 have been acceptable.

3. Maximum Time Step (R)

    Once the time step reaches this value, it is no longer increased. The default is 10^{50}.

4. Time at which New TR3 Card is to be Read (R)

    This entry contains the maximum time at which the time step information on this card applies. When the problem time reaches this value, a new TR3 card is read. The default is 10^{50}.

5. Maximum Temperature Change per Time Step (R)

    This entry contains the maximum absolute temperature change allowed at a node from one time level to the next. The time step is adjusted to try to obtain this maximum temperature change per time step. This may not be achieved due to other constraints placed on the time step. If this entry is zero, then this feature is not invoked in calculating the time step.

6. Maximum Percentage Temperature Change per Time Step (R)

This entry contains the maximum percentage temperature change allowed at a node from one time level to the next. The time step is adjusted to try to obtain this maximum percent temperature change for each time step. This adjustment may not be achieved because of other constraints placed on the time step. If this entry is zero, then this feature is not invoked in calculating the time step.

7. Minimum Time Step (R)

This entry contains the minimum value of the time step. Once the time-step size reaches this value, it is no longer decreased. The default is one-tenth of the initial time-step size.

4.20 DATA-TERMINATION CARD

Each problem must be terminated with a data-termination card. This card consists of a percent sign (%) in column 1, followed by a blank in column 2.

4.21 INPUT SUMMARY

Table 4.3 contains a summary of the information required for all the data blocks. Each input parameter is contained within a box in the table. Since HEATING uses free-form input, the parameter boxes do not correspond to any particular columns in the input file. However, the last column of boxes (columns 73 through 80 of each card) are reserved for a card identification and are not read by HEATING. This last column contains an optional card identification and, in parentheses, the section in the report where additional information can be found. The first line in the parameter input boxes contains the variable name that is used in the program and, in parentheses, the type of that variable. The remainder of the box contains a short explanation of the input parameter. Where space permits, additional notes have been included to further describe the input.

Table 4.3. Summary of input data for HEATING 7.1

OPTIONS

Option Card (4.2): MAXCP=0,MAXPBT=1,MAXPTS=50,MWIDTH=7

NOTE: Default values are shown. It is only necessary to enter those variables whose values are to be changed from the default.

Title Card (4.3)

JOBDES: Job Description - Up to 72 Alphanumeric Characters.

P1 (4.4.1)

NGEOM (I)	TIM (I)	IDEGRE (I)
Geometry Type. Cylindrical Cartesian 1 r-θ-z 6 x-y-z, 2 r-θ 7 x-y, 3 r-z 8 x-z, 4 r 9 x, 5 z Spherical 10 r, 11 r-φ, 12 r-φ-z	Initial time. (See Sect. 4.4.1 for more information.)	Temperature units. (Only used for radiation) 0 = F, 1 = C, 2 = Absolute

MXCPU (I): CPU time in seconds for problem execution.

P2 (4.4.2)

JPLOT0 (I)	JCONN (I)	JCNVRG (I)
Restart flag. 0 = not a restart, 1 = restart of previous case	Flag to read node-to-node connector data file. 0 = do not read, 1 = read	Flag to redirect convergence info. 0 = standard output file, 1 = separate file, -1 = suppress

P3 (4.4.3)

IMONTR (I)	NPRTNN (I)
Flag for output of selected information during calculations. (See table in Sect. 4.4.3 for values.)	Print a table of all node-to-node connectors. 0 = no, 1 = yes

REGIONS

R1 (4.5.2)

NOREG(N) (I)	REGDIM(1,N) (R)	REGDIM(2,N) (R)	REGDIM(3,N) (R)	REGDIM(4,N) (R)	REGDIM(5,N) (R)	REGDIM(6,N) (R)
Region number.	Smaller x or r region dimension.	Larger x or r region dimension.	Smaller y or θ region dimension.	Larger y or θ region dimension.	Smaller z or φ region dimension.	Larger z or φ region dimension.

R2 (4.5.3)

MATS(N) (I)	NREGBC(1,N) (I)	NREGBC(2,N) (I)	NREGBC(3,N) (I)	NREGBC(4,N) (I)	NREGBC(5,N) (I)	NREGBC(6,N) (I)
Region material number. (Card M or ML)	Boundary condition on smaller x or r. (Card B1...)	Boundary condition on larger x or r. (Card B1...)	Boundary condition on smaller y or θ. (Card B1...)	Boundary condition on larger y or θ. (Card B1...)	Boundary condition on smaller z or φ. (Card B1...)	Boundary condition on larger z or φ. (Card B1...)

NGENS(N) (I): Region heat generation number. (Card G)

ITS(N) (I): Region initial temperature number. (Card I)

Table 4.3. Summary of input data for HEATING 7.1 (continued)

MATERIALS

M (4.6.2)

MAT(N) (I)	MATNAM(N) (A)	CONDUC(N) (R)	DENSTY(N) (R)	SPHEAT(N) (R)	NCONTP(N) (I)	NDENTP(N) (I)	NSPHTP(N) (I)	MCP(N) (I)
Material number.	Material name. (Maximum of 8 characters)	Material conductivity.	Material density. Can be zero for steady-state only.	Material specific heat. Can be zero for steady-state only.	Conductivity temperature-dependent function number.	Density temperature-dependent function number.	Specific heat temperature-dependent function number.	Phase change flag. 0 = no phase change 1 = phase change (PC card must be supplied)

NOTE: For the above entries, a positive integer references an analytical function and a negative integer a tabular function.

PC (4.6.3)

SLTM (R)	SLHM (R)	SLTM (R)	SLHM (R)
First phase-change or transition temperature.	Latent heat for first phase change.	Second phase-change or transition temperature.	Latent heat for second phase change.

NOTE: Change-of-phase modeling is implemented in the explicit solution techniques only.

ML (4.6.4)

MAT(N) (I)	MATNAM(N) (A)	MUNCH(N) (I)	XD(N) (R)	XC(N) (R)	XK(N) (R)	XT(N) (R)	XTP(N) (R)	MCP(N) (I)
Material number.	Material name. Library material number preceded by an asterisk (*).	Data units conversion flag.	Unit conversion factor for density.	Unit conversion factor for specific heat.	Unit conversion factor for conductivity.	Unit conversion factor (multiplier) for temperature.	Unit conversion factor (additive) for temperature.	Phase change flag. 0 = no phase change 1 = phase change if included in library data.

NOTE: The above entries are only used for units conversion if Entry 3 on this card has a value of 5.

INITIAL TEMPERATURES

I (4.7.2)

INT(N) (I)	TEMPIN(N) (R)	NTTPOS(1,N) (I)	NTTPOS(2,N) (I)	NTTPOS(3,N) (I)
Initial temperature number.	Initial temperature.	x- or r-dependent function number.	y- or θ-dependent function number.	z- or θ-dependent function number.

NOTE: For the above entries, a positive integer references an analytical function and a negative integer a tabular function.

HEAT GENERATIONS

G (4.8.2)

NGN(N) (I)	GEN0(N) (R)	NGNFCN(1,N) (I)	NGNFCN(2,N) (I)	NGNFCN(3,N) (I)	NGNFCN(4,N) (I)	NGNFCN(5,N) (I)
Heat generation number.	Volumetric heat generation rate.	Time-dependent function number.	Temperature-dependent function number.	x- or r-dependent function number.	y- or θ-dependent function number.	z- or θ-dependent function number.

NOTE: For the above entries, a positive integer references an analytical function and a negative integer a tabular function.

Table 4.3. Summary of input data for HEATING 7.1 (continued)

BOUNDARY CONDITIONS

NBDTP(N) (I) Boundary condition number.	NBYTYP(N) (I) Boundary condition type. 1 = surface-to-boundary 2 = prescribed surface temperature 3 = surface-to-surface	BYTEMP(N) (I) Boundary temperature. (Not used for NBYTYP=3)	NBTFN(N) (R) Time-dependent function number.	NBTPOS(1,N) (I) x- or r-dependent function number.	NBTPOS(2,N) (I) y- or θ-dependent function number.	NBTPOS(3,N) (I) z- or φ-dependent function number.	B1 (4.9.2)
			NOTE: For the above entries, a positive integer references an analytical function and a negative integer a tabular function. These entries are not used for surface-to-surface boundary conditions (NTYPE=3).				
BCDEF(1,N) (R) Forced convection heat transfer coefficient, h_c	BCDEF(2,N) (R) Radiation coefficient, h_r	BCDEF(3,N) (R) Natural convection multiplier term, h_{nc}	BCDEF(4,N) (R) Natural convection exponent term, h_{ce}	BCDEF(5,N) (R) Prescribed heat flux, h_q. (Positive for heat addition to the model.)	IBHFLF (R) Parameter flag. 0=no additional cards 1=B3 card only 2=B4 card only 3=B3 and B4 cards	NOTE: For the entries on this card, a positive integer references an analytical function and a negative integer a tabular function.	B2 (4.9.3)
NBCTIM(1,N) (I) Forced convection time-dependent function.	NBCTIM(2,N) (I) Radiation time-dependent function.	NBCTIM(3,N) (I) Natural convection multiplier time-dependent function.	NBCTIM(4,N) (I) Natural convection exponent time-dependent function.	NBCTIM(5,N) (I) Heat flux time-dependent function.			B3 (4.9.4)
NBCTEM(1,N) (I) Forced convection temperature-dependent function.	NBCTEM(2,N) (I) Radiation temperature-dependent function.	NBCTEM(3,N) (I) Natural convection multiplier temperature-dependent function.	NBCTEM(4,N) (I) Natural convection exponent temperature-dependent function.	NBCTEM(5,N) (I) Heat flux temperature-dependent function.	NOTE: For the entries on this card, a positive integer references an analytical function and a negative integer a tabular function.		B4 (4.9.5)

XGRID

RG(1) (R) Smallest x or r gross grid line.	RG(2) (R) Next larger x or r gross grid line.	RG(3) (R) Next larger x or r gross grid line.	...	RG(N) (R) Largest x or r gross grid line.	X1 (4.10.2)
NDRG(1) (I) Number of divisions between first and second gross grid lines.	NDRG(2) (I) Number of divisions between second and third gross grid lines.	...	NDRG(N-1) (I) Number of divisions between gross grid lines n-1 and n.		X2 (4.10.3)

Table 4.3. Summary of input data for HEATING 7.1 (continued)

YGRID

	Y1 (4.11.2)
THG(1) (R) Smallest y or θ gross grid line.	THG(2) (R) Next larger y or θ gross grid line.	THG(3) (R) Next larger y or θ gross grid line.	...	THG(N) (R) Largest y or θ gross grid line.

Y2 (4.11.3)
NDTHG(1) (I) Number of divisions between first and second gross grid lines.	NDTHG(2) (I) Number of divisions between second and third gross grid lines.	...	NDTHG(N-1) (I) Number of divisions between gross grid lines n-1 and n.

ZGRID

Z1 (4.12.2)
ZG(1) (R) Smallest z or φ gross grid line.	ZG(2) (R) Next larger z or φ gross grid line.	ZG(3) (R) Next larger z or φ gross grid line.	...	ZG(N) (R) Largest z or φ gross grid line.

Z2 (4.12.3)
NDZG(1) (I) Number of divisions between first and second gross grid lines.	NDZG(2) (I) Number of divisions between second and third gross grid lines.	...	NDZG(N-1) (I) Number of divisions between gross grid lines n-1 and n.

CONNECTORS

C1 (4.13.2)
N (I) Base node number to which connections are to be made.	NNC (I) Number of nodes connected to base node.	NBNNC (I) Boundary condition number defining heat transfer mechanism.	IRCPRO (I) Use reciprocity to generate connections. 0 = no 1 = yes

C2 (4.13.3)
NCLIST (I) First node connected to base node.	CNLIST (R) Constant multiplier portion of connectivity.	NCLIST (I) Second node connected to base node.	CNLIST (R) Constant multiplier portion of connectivity.	...

NOTE: Enter total of NNC pairs.

ANALYTICAL FUNCTIONS

A1 (4.14.2)
NANALT (I) Analytical function number.

NOTE: Entries on A2 Card define an analytical function of the following form. Only those terms having non-zero values need to be entered.

$$F(V) = A_1 + A_2 V + A_3 V^2 + A_4 \cos(A_5 V) + A_6 \exp(A_7 V) + A_8 \sin(A_9 V) + A_{10} \ln(A_{11} V)$$

A2 (4.14.3)
NPRM (I) Coefficient index, i.	A(NPRM) (R) Coefficient value, A_i.	NPRM (I) Coefficient index, i.	A(NPRM) (R) Coefficient value, A_i.	...

NOTE: To indicate a function is defined in a user-supplied subroutine, leave Card A2 blank.

Table 4.3. Summary of input data for HEATING 7.1 (continued)

TABULAR FUNCTIONS

					T1 (4.15.2)
NTABL (I) Tabular function number.					
ARG(1) (R) First independent value.	VAL(1) (R) First dependent value.	ARG(2) (R) Second independent value.	VAL(2) (R) Second dependent value.	. . .	T2 (4.15.3)

PRINTOUT TIMES

					O (4.16.2)
PRTIME (R) First printout time.	PRTIME (R) Second printout time.	PRTIME (R) Third printout time.	. . .		

NODES MONITORED

					S (4.17.2)
NTS (I) Number of iterations or time steps between output.	NDS (I) First node monitored.	NDS (I) Second node monitored.	. . .		

STEADY-STATE

					SS (4.18.2)
NTYPE (I) Solution technique. 1 = SOR 2 = Direct 3 = Conjugate gradient	NOITX (I) Maximum number of steady-state iterations. (Defaults: 500 for SOR, 20 nonlinear iterations for direct and conjugate gradient.)	EPI (I) Steady-state convergence criterion. (Default: 10^{-5})	BETA (R) SOR overrelaxation factor. $1.0 \le \beta < 2.0$ (Default: 1.9)	MCOUNT (I) Number of iterations between evaluation of temperature-dependent thermal properties for SOR. (Default: 1) Maximum = 10	

Table 4.3. Summary of input data for HEATING 7.1 (continued)

TRANSIENT

NTYPE (I) Solution Technique. 1 = Explicit 2 = Implicit	**FTIME** (R) Final time.		NOTE: For NTYPE = 1, supply TR1 Card. For NTYPE = 2, supply TR2 and one or more TR3 Cards.	**TR** (4.19.2)
DELTAT (R) Time step.	**KTMFCT** (I) Factor by which stable time step is increased between evaluation of temperature-dependent properties with Levy's method.	**NSTPEX** (I) Number of time steps between evaluation of temperature-dependent properties for CEP.		**TR1** (4.19.3)
THETA (R) Parameter defining differencing technique for transient equation. Corresponds to θ in Eq. 2.38. $0.5 \le \theta \le 1.0$ 0.5 = Crank-Nicolson 1.0 = Backwards Euler	**RESDUL** (R) Convergence criterion for implicit solution. Corresponds to ϵ_1 in Eq. 2.52. (Default: 10^{-5})	**NITREZ** (I) Number of iterations in linear loop between tests for convergence. (Default: 1)	**RELDIF** (R) Relative convergence criterion for nonlinear (temperature-dependent properties) loop. Corresponds to ϵ_2 in Eq. 2.55. (Default: 10^{-5}) · **ABSDIF** (R) Absolute convergence criterion for nonlinear (temperature-dependent properties) loop. (Default: 0) · **BETAT** (R) Initial value for SOR acceleration parameter. Corresponds to ω in Eq. 2.57. =0, optimize empirically <0, Carre's optimization >0, use constant value entered BETAT < 2 · **NUPBTA** (I) Number of time steps between acceleration parameter updates. Used when BETAT=0. Corresponds to N_ω in Sect. 2.3.4. (Default: 1) · **ITLRCO** (I) For BETAT=0, Number-of-iterations criterion to initiate acceleration parameter updates. Corresponds to I_ω in Sect. 2.3.4. (Default: 5) For BETAT<0, Number of time steps between SOR acceleration parameter updates. (Default: 12) · **ITLRCI** (I) Number-of-iterations criterion to terminate acceleration parameter updates. Corresponds to J_ω in Sect. 2.3.4. (Default: 2)	**TR2** (4.19.4)
DELTAT (R) Initial time step for implicit solution. (Default: previous time step if other that first TR3 card)	**TSFACT** (R) Factor by which the current time step is multiplied at each time step. (Default: 1.0)	**TSMAX** (R) Maximum time step size. (Default: 10^{50})	**TSCHGE** (R) Maximum time for which this series of time steps applies. When the current time exceeds this value, another TR3 card is read. (Default: 10^{50}) · **TPCGMX** (R) Maximum temperature change allowed at any node over a time step. · **PTPCGM** (R) Maximum percentage change in temperature allowed at any node over a time step. · **TSMIN** (R) Minimum time step size. (Default: DELTAT/10)	**TR3** (4.19.5)

Data termination card.	(4.20)

%

5. OUTPUT DESCRIPTION

5.1 GENERAL

Each execution of HEATING automatically produces two output files: a print file containing a processed version of the input data with a solution status summary and an unformatted plot data file containing the model geometry description, temperature distributions, and phase distributions. Output times at which solution information is written to the print and plot files are specified by the user in the PRINTOUT TIMES data block. A more detailed description of these files is given in the sections below.

There are two optional files that can be produced. If specified on Parameter Card 2 (Sect. 4.4.2), all of the convergence information will be written to the convergence information file rather than to the normal print file. If a NODES MONITORED data block is included in the input file, a file containing the temperature history of specified nodes will be created. On UNIX systems there is also a file produced by the shell script used to execute HEATING. This file lists the time and date of the run, the names of all input files supplied, and the names of all files generated by the HEATING run.

5.2 PRINT FILE

The print output generated by the HEATING computer code is best illustrated by example, and the reader is referred to Appendix C. This section identifies the types of output available and gives a brief description of some of the features. The code automatically lists the input data card images, tabulates the input data with descriptive headings, and lists some information generated by the code from the input data. The code also automatically prints information giving the progress of the calculations. Temperatures of selected nodes can be printed at a user-specified frequency to trace the progress of a transient- or steady-state calculation (see Sect. 4.17).

5.2.1 Input Return

HEATING automatically lists the card images with the card columns indicated every tenth card and with each card image numbered. Since this serves to document the input data exactly as it was supplied to the code, the feature assists the user in locating data errors identified later in the calculations. The output also contains information pertaining to the amount of computer memory required for the run, and whether the specified amount of computer memory must be increased to continue the calculations.

For each problem in a run, the standard output contains a summary of the capabilities of the current version of HEATING, along with a heading identifying the version of the code, the date and time of the run, the job name, and the computer on which the job was executed. This information is followed by tables that indicate the maximum number of parameters and the features selected for the problem. The values of parameters from the three parameter cards are then tabulated, along with the input/output logical unit numbers available for the problem. These values are followed by a tabulation of the remaining input data with descriptive headings.

89

The code then tabulates some of the data it generates from the input data. These data include the fine lattice lines along each axis, the total number of nodes, and the stability criterion at each node for transient problems.

5.2.2 Steady-State Convergence Information

For steady-state successive-overrelaxation iteration, a table is generated giving the status of the calculations. Every five iterations an entry is made in the table indicating the iteration number, the maximum relative change in temperature over the last iteration (along with its sign), the node where the above maximum occurred, the temperature of the node, and the extrapolation factor as computed by the code. Whenever the overrelaxation factor is modified or whenever an extrapolation occurs, a related message is written in the table.

For the nonlinear direct-solution technique, the following output is generated every iteration to indicate the status of the calculations: the iteration number, the maximum relative residual and the node where it occurred, along with the node's total heat flow, average heat flow, heat residual, and temperature; and the iteration number, the maximum residual and the node where it occurred, along with the node's total heat flow, average heat flow, and temperature.

For the conjugate gradient solution technique, a table is printed out that gives the status of the solution every fifth iteration. This table includes the number of iterations completed, the average residual for all nodes, the maximum residual, the node at which the maximum residual occurred, and the temperature of this node. For a nonlinear solution, a message is printed out indicating when the linear problem has converged and properties are being reevaluated. A new convergence table is then started with the linear iteration counter reset to zero.

5.2.3 Implicit Transient Convergence Information

For the transient implicit procedure, a table is generated giving the status of the calculations. This table appears between normal printouts and includes the time-level number, the time level, the number of iterations required for both the linear and nonlinear loops to converge, the values which must satisfy the convergence criterion for both the linear and nonlinear loops, the maximum absolute temperature change over the time increment and the node where it occurred, the maximum percent relative change in temperature over the time increment and the node where it occurred, and other information concerning the status of the numerical procedure being used.

5.2.4 Solution Summary at Printout Times

At specified printout times, as well as at the completion of a transient or steady-state solution, a summary of the solution status is printed. This summary consists of the current time-step size (transient cases only), the solution time, and the elapsed cpu time since the calculation was started. The maximum and minimum temperatures and the nodes where they occur in the problem are then written. If any heat generation functions are specified, the current rate of energy input to the model due to each function is listed in columns identified as modeled and neglected heat generation. Neglected heat generation is that which is associated with nodes that are on specified-surface-temperature boundaries. The neglected heat generation column will always be zero if there are no type-2 boundary conditions specified. A summary of boundary

temperatures and the heat flow rates on all boundaries follows. The column labeled "neglected heat flow" will only have a value in it if there is a node that has a specified surface temperature (type-2) boundary condition applied to one surface and a different boundary condition applied to another surface. The heat flow listed for each boundary does not include any specified fluxes. The net heat flow due to specified fluxes for all boundary conditions is listed separately. If the neglected heat generation or the neglected heat flow on a boundary is very significant compared with the modeled values, the user may need to reevaluate the model. For models having a boundary condition that has a position-dependent temperature, a table is output giving the node number and coordinates of each internal node on that boundary, along with the value for the position-dependent boundary temperature.

5.2.5 Special Monitoring of Temperatures

The temperatures at a few specified nodes may be tabulated as a function of the number of time steps for transient problems and as a function of the number of iterations for steady-state problems. This allows monitoring the temperatures at a few nodes of interest. See Sect. 4.17 for more details on how to use this option.

5.2.6 Output of Selected Information During Calculations

Selected information generated during the calculations may be output to assist the user in the detection of input data errors, in better understanding cases that are not performing as anticipated or that are suspected of being in error, and in debugging changes to the code.

This output consists of the following tables: the location of each node and a list of its neighbors; a list of nodal connectors for surface-to-surface boundary conditions; the regions that comprise each node; the temperature, heat capacitance, and power associated with each node and an indicator of whether or not the heat capacitance, effective conductance, and power at each node are dependent on temperature; each neighbor and related effective conductance of each connector for every node; phase-change information at each node; and the steady-state or transient temperature distributions at the specified times. See Sect. 4.4.3 for more details on the use of this feature.

5.3 UNFORMATTED PLOT DATA FILE

An unformatted plot data file is produced by every successful execution of HEATING. This file, which is used for graphical postprocessing and restarting calculations, contains the model geometry description, temperature distributions, and phase distributions. This file automatically contains the initial temperature and phase distributions for each transient calculation and the final distributions for all steady-state and transient calculations. Additional output times are specified by the user in the PRINTOUT TIMES data block.

Several post-processing options are available to produce graphical output from the HEATING solution stored in the unformatted plot data file. HEATPLOT reads this file directly and queries the user about the plots to be produced via an interactive menu selection system. Two interface codes, H7CONPLOT and H7PATRAN, are also available that read the unformatted plot data file and produce formatted files for postprocessing with CONPLOT and

PATRAN, respectively. If the user wishes to produce tabular output or if HEATPLOT, CONPLOT, or PATRAN do not produce the type of graphical output desired, the user may need to access this file in order to produce the desired output. For that reason the format of the file and a description of the information stored in it are given in Table 5.1. Even though HEATING is a finite-volume code, the plot file contains element definitions. These element definitions are used to describe the geometry to the postprocessor codes - HEATING does not use them internally. In Table 5.1, character, integer, and real variables are identified with C, I, and R, respectively. All real variables are eight bytes in length, and all integer variables are the default length for the computer on which HEATING is being run (eight bytes on a CRAY and four bytes on most other machines.)

Table 5.1. Contents of unformatted plot data file

Record 1: JOBDES, NGEOM, IT, JT, KT, NT, NBDTPT, MATSL, IERR, IWARN, CURTIM, VERSUN

JOBDES	C*72	Job description
NGEOM	I	Geometry type
IT	I	Number of grid lines along first axis
JT	I	Number of grid lines along second axis
KT	I	Number of grid lines along third axis
NT	I	Number of nodes
NBDTPT	I	Number of boundary conditions
MATSL	I	Number of unique phase changes
IERR	I	Number of errors produced during model generation
IWARN	I	Number of warnings produced during model generation
CURTIM	C*24	Time and date of HEATING run
VERSUN	C*15	Version used for analysis (e.g. HEATING 7.1b)

Record 2: (R(I),I=1,IT)

R	R	Array of grid line values along first axis

Record 3: (TH(I),I=1,JT)

TH	R	Array of grid line values along second axis

Record 4: (Z(I),I=1,KT)

Z	R	Array of grid line values along third axis

Table 5.1. Contents of unformatted plot data file (continued)

Record 5:   (N,NTPI(N),NTPJ(N),NTPK(N),I=1,NT)

N	I	Node number
NTPI	I	Grid line passing through node (first axis)
NTPJ	I	Grid line passing through node (second axis)
NTPK	I	Grid line passing through node (third axis)

Record 6:   NELEM,NODES

NELEM	I	Number of elements
NODES	I	Number of nodes defining each element

Records 7 through NELEM+6: MATL,(NODE(I),I=1,NODES)

MATL	I	Material in element
NODE	I	Array of nodes defining element

Record NELEM+7:   NSET, NOIT, DELTAT, TIM, IERR, IWARN

NSET	I	Solution type: -1=transient, +1=steady-state
NOIT	I	Number of steady-state iterations or transient time steps
DELTAT	R	Time step
TIM	R	Time
IERR	I	Number of errors produced by run to this point
IWARN	I	Number of warnings produced by run to this point

Record NELEM+8:
   No Phase Change   (T1(I),I=1,NT),(TDUM(I),I=1,NBDTPT)
   Phase Change   (T1(I),I=1,NT),(X1(I),I=1,NT),(TDUM(I),I=1,NBDTPT)

T1	R	Temperature array
X1	R	Phase fraction array
TDUM	R	Boundary temperature array

Additional Records: The previous two records are repeated for each temperature distribution that is output.

ACKNOWLEDGMENTS

REFERENCES

1. T. B. Fowler and E. R. Volk, *Generalized Heat Conduction Code for the IBM-704 Computer*, ORNL-2734, Union Carbide Corp., Nuclear Div., Oak Ridge National Laboratory, October 1959.

2. R. R. Liguori and J. W. Stephenson, *The HEATING Program*, ASTRA 417-5.0, ASTRA, Inc., Raleigh, NC, January 1961.

3. W. D. Turner and J. S. Crowell, *Notes on HEATING — An IBM 360 Heat Conduction Program*, CTC/INF-980, Union Carbide Corp., Nuclear Div., November 1969.

4. W. D. Turner and M. Siman-Tov, *HEATING3 — An IBM 360 Heat Conduction Program*, ORNL/TM-3208, Union Carbide Corp., Nuclear Div., Oak Ridge National Laboratory, February 1971.

5. W. D. Turner, D. C. Elrod, and I. I. Siman-Tov, *HEATING5 — An IBM 360 Heat Conduction Program*, ORNL/CSD/TM-15, Union Carbide Corp., Nuclear Div., Oak Ridge National Laboratory, March 1977.

6. D. C. Elrod, G. E. Giles, and W. D. Turner, *HEATING6: A Multidimensional Heat Conduction Analysis with the Finite-Difference Formulation*, ORNL/NUREG/CSD-2/V2, Union Carbide Corp., Nuclear Div., Oak Ridge National Laboratory, October 1981.

7. K. W. Childs, G. E. Giles, C. B. Bryan, and C. K. Cobb, *HEATING: A Computer Program for Multidimensional Heat Transfer Analysis (Version 6.1)*, ORNL/NUREG/CSD-2/V2/R3, Martin Marietta Energy Systems, Oak Ridge National Laboratory, to be published.

8. K. W. Childs, *HEATING 7.0 User's Manual*, K/CSD/INF/90-32, Martin Marietta Energy Systems, Inc., July 1990.

9. L. M. Petrie and N. F. Cross, *KENO IV — An Improved Monte Carlo Criticality Program*, ORNL-4938, Union Carbide Corp., Nuclear Div., Oak Ridge National Laboratory, November 1975.

10. C. B. Bryan, K. W. Childs, and G. E. Giles, *HEATING6 Verification*, K/CSD/TM-61, Martin Marietta Energy Systems, Inc., Oak Ridge National Laboratory, December 1986.

11. R. S. Varga, *Matrix Iterative Analysis*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1962.

12. J. Dongarra, C. B. Moler, J. R. Bunch, and G. W. Stewart, "LINPACK User's Guide," *SIAM*, Philadelphia, PA, 1979.

13. B. R. Becker, *A Direct Solution Technique for Solving Steady-State Problems Using the HEATING6 Heat Transfer Code*, K/CSD/TM-15, Union Carbide Corp. Nuclear Div., October 1977.

REFERENCES (continued)

14. G. H. Golub and C. F. Van Loan, *Matrix Computations*, 2nd Ed., The Johns Hopkins University Press, Baltimore, MD, 1989.

15. S. K. Hellman, G. Habetler, and H. Babrov, "Use of Numerical Analysis in the Transient Solution of Two-dimensional Heat-Transfer Problem with Natural and Forced Convection," *Trans. Am. Soc. Mech. Engrs.*, **78**, 1155–61 (1956).

16. S. Levy, *Use of "Explicit Method" in Heat Transfer Calculations with an Arbitrary Time Step*, 68-C-282, General Electric, Schenectady, NY, August 1968.

17. B. A. Carre, "The Determination of the Optimum Accelerating Factor for Successive Over-Relaxation," *Comp. J.*, **4**, 73–78 (1961).

Appendix A. RUNNING HEATING

A.1 RUNNING HEATING ON A UNICOS CRAY

The following instructions are for running HEATING 7.1 on the Martin Marietta Energy Systems UNICOS CRAY X-MP computer at the Oak Ridge K-25 Site. Some of the command line entries may differ for installations on other UNICOS CRAY systems (e.g., the batch queue names may be different.) Also the entire path name may need to be supplied rather that just the just the **h7** command at some sites. The command line for executing HEATING is

> **h7** [-i *ifile*] [-f *ffile*] [-r *rfile*] [-c *cfile*] [-m *mfile*] [-x *xfile*] [-q *queue*] [-lM *memory*]
> [-lT *time*] [-a *account*]

where the brackets around each of the options indicate that they are not required. The options are defined below.

Options:

-i *ifile*   This option specifies the name of a file in the current working directory that contains HEATING input. If this option is not specified, a file by the name of *input* must be present.

-f *ffile*   This option specifies the name of a file containing the Fortran source for any user-supplied subroutines.

-r *rfile*   This option specifies the name of a plot data file created by a previous execution of HEATING which is to be used to obtain restart information. The restart option must also be specified in the HEATING input file.

-c *cfile*   This option specifies the name of a file containing node-to-node connector information for this case. A flag must also be specified in the HEATING input file for this file to be read.

-m *mfiie*   This option specifies the name of a file containing a user-defined or modified material properties library. This option should not be included on the command line if the default material properties library is used.

-x *xfile*   This option specifies the name of an executable file for a case involving user-supplied subroutines. If the -f option was also specified, the executable created by this execution will be stored in *xfile* in the current working directory. If the -f option was not specified, an executable file named *xfile* must be present in the current working directory. This executable file would have been created by a previous execution of HEATING.

-q *queue*
This option specifies the type of execution requested. The choices are batch and background. The default is batch. If the default is taken or batch is entered, the job will be placed in one of the following batch queues based on the time specified on the -lT option.

small	-	Less than 20 minutes
medium	-	20 minutes to 2.5 hours
large	-	2.5 to 10 hours
huge	-	Over 10 hours

-lM *memory*
(minus-sign, lower-case-L, upper-case-M) This option specifies the maximum amount of memory that will be required for this execution. The default value is 2Mw. This only applies to batch jobs. Refer to the **qsub** command in the *UNICOS User Commands Reference Manual* for further details.

-lT *time*
(minus-sign, lower-case-L, upper-case-T) This option specifies the maximum amount of cpu time allowed for this execution. The default value is 1200s (20 min). This only applies to batch jobs. Refer to the **qsub** command in the *UNICOS User Commands Reference Manual* for further details. WARNING - If a batch job runs out of time, the results will be lost. To avoid this, enter a value for this option that is greater than the value specified on parameter card 1 in the HEATING input file.

-a *account*
This option specifies the account (five-digit computer charge number) to which this execution is to be charged. If this option is not specified, the user's default account will be charged. For background executions, the charge number specified becomes the default account for the remainder of the current login session.

Notes:

Every execution of HEATING creates a subdirectory off of the user's HOME directory. This directory is named h7_n, where n is an integer. All files required for or created by a particular HEATING run are stored in this directory until execution is complete. Typically, this directory will be removed when the execution of HEATING is complete, but there are circumstances where this will not happen (e.g., a system crash during execution or the user deletes a job from the batch queue before execution). If necessary the user can remove this directory with the following command:

    rm -r h7_n

Command Line Examples:

1.   HEATING input file named *input*, batch execution.

   h7

2.   HEATING input file named *myinput*, batch execution.

   h7 -i myinput

3.   HEATING input file named *myinput*, Fortran source for user-supplied subroutines
      in file *mysource*, executable created by this job to be stored in file *myheating7*,
      batch execution.

   h7 -i myinput -f mysource -x myheating7

4.   Use executable created in example 3 with new input file *myinput2*, batch execution
      with a maximum cpu time of 600 s.

   h7 -i myinput2 -x myheating7 -lT 600

5.   HEATING input file named *myinput*, background execution.

   h7 -i myfile -q back

HEATING OUTPUT FILES

File description	File name
System-generated messages and a list of names of all files used or generated by this execution of HEATING.	h7-*jid*[1]
Print file. This file is created for every execution of HEATING.	print*jid*[1]
Plot file. This file is created for every execution of HEATING.	plot*jid*[1]
Nodes monitored file. This file is created if a NODES MONITORED data block is included in the input.	h7node*jid*[1]
Convergence information file. This file is created if the fourth entry on Parameter Card 2 is greater than zero.	h7cvrg*jid*[1]

[1] *jid* is an integer number used to uniquely identify output from a particular job.

A.2 RUNNING HEATING ON A UNIX WORKSTATION

The instructions for running HEATING 7.1 on a UNIX workstation are essentially the same as for the UNICOS CRAY except that fewer options are available on the command line. The entire path name may need to be supplied rather that just the just the **h7** command depending on the particular installation. The command line for executing HEATING on a UNIX workstation is

h7 [**-i** *ifile*] [**-f** *ffile*] [**-r** *rfile*] [**-c** *cfile*] [**-m** *mfile*] [**-x** *xfile*] [**-at** *time/date*]

where the brackets around each of the options indicate that they are not required. The options are defined below.

Options:

-i *ifile*	This option specifies the name of a file in the current working directory that contains HEATING input. If this option is not specified, a file by the name of *input* must be present.
-f *ffile*	This option specifies the name of a file containing the Fortran source for any user-supplied subroutines.
-r *rfile*	This option specifies the name of a plot data file created by a previous execution of HEATING which is to be used to obtain restart information. The restart option must also be specified in the HEATING input file.
-c *cfile*	This option specifies the name of a file containing node-to-node connector information for this case. A flag must also be specified in the HEATING input file for this file to be read.
-m *mfile*	This option specifies the name of a file containing a user-defined or modified material properties library. This option should not be included on the command line if the default material properties library is used.
-x *xfile*	This option specifies the name of an executable file for a case involving user-supplied subroutines. If the -f option was also specified, the executable created by this execution will be stored in *xfile* in the current working directory. If the -f option was not specified, an executable file named *xfile* must be present in the current working directory. This executable file would have been created by a previous execution of HEATING.
-at *time/date*	This option allows a job to be executed at a later specified time in order to avoid overloading a workstation. The *time/date* can be specified in several formats. Refer to documentation on your specific workstation for more information on the **at** command.

Appendix B. VERIFICATION PROCEDURE

The HEATING series of general purpose, finite-difference, conduction heat transfer codes have been in use for many years. During this time the codes have been used extensively, and a general confidence has been developed in regard to their accuracy. In the development of each new version of HEATING, great care was taken to ensure that the code did produce accurate solutions to heat transfer problems. Unfortunately, for the most part, the checking that was done was not documented, nor was a formal verification ever presented in a citable document for any version of the code prior to HEATING6. In 1985 and 1986 a concerted effort was undertaken to formally verify and document HEATING6.[1] A rigorous verification was carried out in which HEATING temperature solutions were compared against analytical solutions obtained from the literature. Twenty-three analytical solutions were chosen in order to test various HEATING analysis options. The verification cases focused on one-dimensional Cartesian problems because of the abundance of available analytical solutions, but all of the HEATING geometry options were checked. There have been some minor modifications made to the verification cases as a result of changes in the capability of newer versions of HEATING.

The comparison-to-analytical-solution cases are not sufficient to adequately check all features of the code since they only exercise one option at a time. Any problems resulting as a consequence of the interaction of two or more features would not be discovered. Therefore, three additional reference cases were developed which exercise several analysis options simultaneously. These problems were intended to serve as a basis of comparison for any future modifications to the code or implementation of the code on other computer systems.

Once a version of HEATING had been formally verified against analytical solutions, then that version can be used to verify subsequent versions of the code. In this manner HEATING6 was used to verify HEATING 6.1 by manually comparing the printed output produced by the two codes for all of the verification and reference cases. While this approach is more expedient than comparing against the analytical solution, it is very laborious and potentially error prone. For these reasons a computer program, HEATCHEK,[2] was developed to automate the verification procedure. HEATCHEK compares the information written to the plot data file by the old and new versions of HEATING.

REFERENCES

1.   C. B. Bryan, K. W. Childs, and G. E. Giles, *HEATING6 Verification*, K/CSD/TM-61, Martin Marietta Energy Systems, Inc., Oak Ridge Gaseous Diffusion Plant, December 1986.

2.   W. Chu, *HEATCHEK: A Computer Program to Automate Verification of New Versions of HEATING*, K/CSD/INF-89/4, Martin Marietta Energy Systems, March 1989.

Appendix C. SAMPLE PROBLEMS

C.1 SAMPLE PROBLEM 1

A stainless steel wire with a radius of 1.5 mm has an electric current passing through it. The resulting internal heat generation rate per unit volume is 1.12×10^9 W/m^3. The surface of the wire loses heat by convection to the environment. The convective heat transfer coefficient and ambient temperature are 4000 W/m^2-K and 110°C, respectively. The conductivity of stainless steel is 19 W/m-K. The steady-state temperature distribution within the wire is to be determined. The centerline temperature is of particular interest since this is the hottest location.

This problem can be modeled in a one-dimensional, r-cylindrical geometry. The HEATING input file for this case, $ex1.dat$, is shown in Fig. C.1. The units given in the problem definition are not consistent since the wire radius is given in millimeters and other length units are in meters. In the input data millimeters are used for the length unit throughout.

```
Sample Problem 1
* 3 mm stainless steel wire with internal heat generation
* Units:  J, kg, s, mm, C
10  4                                              P1
                                                   P2
                                                   P3

REGIONS
1  1  0.0  1.5                                     R1
0  1  0   1                                        R2
MATERIALS
1  S-Steel  0.019                                  M
HEAT GENERATIONS
1  1.12                                            G
BOUNDARY CONDITIONS
1  1  110.0                                        B1
4.0e-3                                             B2
XGRID
0.0  1.5                                           X1
    8                                              X2
STEADY-STATE
2                                                  SS
X
```

Fig. C.1. Contents of input file *ex1.dat*.

This particular problem does not require that an OPTIONS Data Block be supplied since none of the default parameter limits are exceeded. Thus the first card in the data file is the Title Card. Immediately following the Title Card are some comment cards (indicated by the '*' in column 1). It is a good practice to include comment cards at this location to document what the file is for future reference. Parameter card 1 indicates that a maximum of 10 s of cpu time is allowed before terminating the solution and that the geometry is one-dimensional r-cylindrical. Since an initial time is not entered, it defaults to zero. Even though the temperature units for

105

this problem are °C, it is not necessary to enter a value for it on parameter card 1 since radiation is not being modeled. Since none of the options on parameter cards 2 and 3 are being exercised, both cards are left blank.

One region is defined in the REGIONS data block. This region definition references material number 1, heat generation number 1, and boundary condition number 1 which are all defined in their corresponding data blocks. The only material property required is the conductivity since only a steady-state solution is requested. The XGRID data block defines the mesh spacing. The radius of the wire is divided into eight equal divisions which results in nine nodes being defined. Since a steady-state solution is desired, a STEADY-STATE data block is included. The entry of 2 on the SS card indicates that the direct-solution technique is to be used. The card identifications contained in columns 73-80 are not required.

The command line for executing HEATING is given below.

```
h7 -i ex1.dat
```

HEATING responds with a line similar to the following.

```
System messages for this execution are in file h7-1
```

The contents of file *h7-1* after completion of the HEATING execution are shown in Fig. C.2.

```
Execution Begun on 07/17/91 at 10:16:29

STOP

Real      0.6
User      0.1
System    0.0

HEATING input data    :  /u/kch/heating/h7.1/example/ex1.dat

Files created         :  /u/kch/heating/h7.1/example/print1
                         /u/kch/heating/h7.1/example/plot1

Execution Completed on 07/17/91 at 10:16:31
```

Fig. C.2 Contents of file *h7-1*.

Every successful HEATING execution produces a print file and a plot file. Additional files may optionally be generated. The plot file is an unformatted file that is used as input for post-processing with one of the codes discussed in Appendix D. The print file for this case, *print1*, is given in Fig. C.3.

```
Current Time: Wed Jul 17 10:16:29 1991        Computer: IBM/AIX


    H   H  EEEEE   AAA   TTTTT III  N   N   GGG
    H   H  E      A   A    T    I   N   N  G   G
    H   H  E      A   A    T    I   NN  N  G
    HHHHH  EEE    AAAAA    T    I   N N N  G
    H   H  E      A   A    T    I   N  NN  G  GG
    H   H  E      A   A    T    I   N   N  G   G
    H   H  EEEEE  A   A    T   III  N   N   GGG


    Version      :  HEATING 7.1e
    Release date :  Apr. 26, 1991
    Serial Number:  0002


    Contacts:

    K. W. (Ken) Childs  (615) 576-1759 or FTS 626-1759
    G. E. (Gary) Giles  (615) 574-8667 or FTS 624-8667
    Heat Transfer and Fluid Flow Section
    Computing and Telecommunications Division
    Martin Marietta Energy Systems, Inc.
    Post Office Box 2003
    Oak Ridge, Tennessee 37831-7039


**************************** ECHO OF INPUT DATA ****************************

Record
     1     Sample Problem 1
     2     * 3 mm stainless steel wire with internal heat generation
     3     * Units:  J, s, mm, C
     4     10  4                                                        P1
     5                                                                  P2
     6                                                                  P3
     7     REGIONS
     8     1  1   0.0   1.5                                             R1
     9     0  1   0    1                                                R2
    10     MATERIALS
    11     1  S-Steel   0.019                                          M
    12     HEAT GENERATIONS
    13     1  1.12                                                      G
    14     BOUNDARY CONDITIONS
    15     1  1   110.0                                                 B1
    16     4.0e-3                                                       B2
    17     XGRID
    18     0.0  1.5                                                     X1
    19         8                                                       X2
    20     STEADY-STATE
    21     2                                                           SS
    22     %


**************************** CASE DESCRIPTION ****************************

  Sample Problem 1
```

Fig. C.3. Contents of file *print1*.

```
******************** SUMMARY OF PARAMETER CARDS DATA ********************

Maximum cpu time       -    10.00 seconds
Geometry type number  -     4 (or r-cy)
Initial time          -    0.0000000D+00
Temperature units     -    Fahrenheit (Significant only if radiation involved)

This is a restart of previous case           - No
Read node-to-node connector data file        - No
Redirect or suppress convergence information  - No

Output selected information during calculations - No


************************** SUMMARY OF REGION DATA **************************

Region   Material  Initial   Heat Gen.
Number   Number    Temp. No.  Number
   1        1         0          1


          ------------------- Dimensions / Boundary Numbers -------------------
Region      First Axis              Second Axis             Third Axis
Number   Smaller    Larger      Smaller    Larger      Smaller    Larger
   1   0.0000E+00  1.5000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
          0           1           0           0           0           0


************************** SUMMARY OF MATERIAL DATA **************************

Material   Material     ------------ Thermal Parameters ------------   Phase
Number     Name         -- Temperature-Dependent Function Numbers --   Change
                        Conductivity      Density      Specific Heat
   1       s-steel      1.900000D-02    0.0000000+00    0.0000000+00    No
                            0               0               0


******************** SUMMARY OF HEAT GENERATION RATE DATA ********************

Number    Power    Time-, Temperature-, and Position-Dependent Function Numbers
          Density      Time    Temperature  X or R    Y or Theta    Z or Phi
   1   1.120000D+00     0          0           0           0           0


******************** SUMMARY OF BOUNDARY CONDITION DATA ********************

Number:    1                      Type:  Surface-to-Boundary
   Temperature and Any Functions Used to Define Dependence:
              Temperature       :  1.100000E+02
   Heat Transfer Coefficients and Any Functions Used to Define Dependence:
              Forced Convection  :  4.000000E-03


************************** SUMMARY OF GRID STRUCTURE **************************

X (or R) Gross Grid Lines and Number of Divisions
   0.000000E+00   1.500000E+00
              8
```

Fig. C.3. Contents of file *print1* (continued).

```
X (or R) Fine Grid Lines Generated by HEATING
    1  0.00000E+00    2  1.87500E-01    3  3.75000E-01    4  5.62500E-01
    5  7.50000E-01    6  9.37500E-01    7  1.12500E+00    8  1.31250E+00
    9  1.50000E+00


***************i***** SOURCES OF NON-LINEARITY IN THE MODEL *********************

    The model is linear.


************* OPTION CARD PARAMETERS SPECIFYING ARRAY SIZE LIMITS ************

    Parameter      Current Value    Required Value
    MAXPTS             50               9
    MAXCP               1               0
    MAXPBT              1               0
    MWIDTH              7               3

The variably dimensioned arrays require      3311 single ( 4 byte) words
or    13K of memory.


*************** NUMBER OF PARAMETERS SPECIFIED BY THE INPUT DATA ***************

    Regions                                    1
    Materials                                  1
    Different phase changes                    0
    Initial temperatures                       0
    Heat generations                           1
    Different boundary conditions              1
    Gross grid lines along x or r axis         2
    Fine  grid lines along x or r axis         9
    Gross grid lines along y or theta axis     1
    Fine  grid lines along y or theta axis     1
    Gross grid lines along z or phi axis       1
    Fine  grid lines along z or phi axis       1
    Analytic functions                         0
    Tabular functions                          0
    Node-to-node connectors                    0
    Transient printout times                   0
    Nodes for monitoring of temperatures       0


**************************** INITIAL CONDITIONS ****************************

    Number of iterations completed =         0
    Current problem time        = C.000000000+00
    Elapsed cpu time (sec)      = 1.600000000-01

                ***** CURRENT TEMPERATURE SUMMARY *****

            Minimum Temperature = 0.00000E+00    at node    1
            Maximum Temperature = 0.00000E+00    at node    1

                ***** CURRENT HEAT GENERATION SUMMARY *****

    Number       Modeled Heat Generation     Neglected Heat Generation
                     (energy/time)                (energy/time)
       1             7.91681E+00                  0.00000E+00
```

Fig. C.3. Contents of file *print1* (continued).

```
                    ***** CURRENT BOUNDARY CONDITION SUMMARY *****

        Number        Temperature       Modeled Heat Flow  Neglected Heat Flow
                                          (energy/time)       (energy/time)
          1           1.10000E+02         4.14690E+00         0.00000E+00




    **************************************************************************

        BEGIN STEADY STATE CALCULATION - DIRECT SOLUTION TECHNIQUE

    **************************************************************************

            Maximum number of iterations    =    20
            Convergence criterion           =  1.0000000D-05


    ********************** STEADY-STATE SOLUTION OUTPUT ***********************

        Number of iterations completed =         1
        Current problem time           = 0.00000000D+00
        Elapsed cpu time (sec)         = 1.60000000D-01

                    ***** CURRENT TEMPERATURE SUMMARY *****

            Minimum Temperature = 3.20000E+02   at node      9
            Maximum Temperature = 3.53158E+02   at node      1

                    ***** CURRENT HEAT GENERATION SUMMARY *****

        Number        Modeled Heat Generation     Neglected Heat Generation
                          (energy/time)               (energy/time)
          1               7.91681E+00                 0.00000E+00

                    ***** CURRENT BOUNDARY CONDITION SUMMARY *****

        Number        Temperature       Modeled Heat Flow  Neglected Heat Flow
                                          (energy/time)       (energy/time)
          1           1.10000E+02        -7.91681E+00         0.00000E+00


    ************************** END OF HEATING EXECUTION ***********************

    Sample Problem 1

    ***** Number of warnings  --    0
    ***** Number of errors    --    0
```

Fig. C.3. Contents of file *print1* (continued).

As is evident from the listing in Fig. C.3, the only nodal temperatures in the print file are the minimum and maximum temperatures occurring in the model. To obtain information about the temperature at other locations it is necessary to post-process the information in the plot file. There are several graphical and tabular forms of output available (see Appendix D for details.) As a example of tabular output, Fig. C.4 presents the temperature distribution obtained by post-processing with H7MAP.

```
Sample Problem 1                                                    Thu Jul 18 10:33:26 1991
                Steady-State Temperature Distribution at Time 0.0000E+00

        1     .00  |   353.16
        2     .19  |   352.64
        3     .38  |   351.09
        4     .56  |   348.50
        5     .75  |   344.87
        6     .94  |   340.21
        7    1.12  |   334.51
        8    1.31  |   327.77
        9    1.50  |   320.00
```

Fig. C.4. Contents of file created by H7MAP.

If the user decides that the temperature solution during the transient cool-down following a power shutoff is needed after completing the steady-state solution, a transient case can be run using the previous steady-state solution as the initial conditions. The original input file can be easily modified to produce the input file for the transient. The contents of this modified file, *ex1a.dat*, are shown in Fig. C.5.

```
Sample Problem 1a
* 3 mm stainless steel wire with internal heat generation
* Units:  J, kg, s, mm, C
10  4                                                              P1
1                                                                  P2
                                                                   P3

REGIONS
1  1  0.0  1.5                                                     R1
0  0  0   1                                                        R2
MATERIALS
1  S-Steel  0.019  7.865e-6  460.0                                M
BOUNDARY CONDITIONS
1  1  110.0                                                        B1
4.0e-3                                                             B2
XGRID
0.0  1.5                                                           X1
    8                                                              X2
PRINTOUT TIMES
0.5  1.0  1.5  2.0  2.5  3.0  3.5  4.0  4.5
TRANSIENT
1  5.0                                                             TR
0                                                                  TR1
%
```

Fig. C.5. Contents of input file *ex1a.dat*.

Several things should be noted about the modifications to the input file for the transient case. First, entry one on card P1 is a '1' to indicate that the initial temperature distribution is to be obtained from the plot file produced by a previous execution of HEATING. Second, the entry for heat generation number on the region definition (second entry on card R2) has been changed to a '0'. The HEAT GENERATION data block was also removed, but this was not actually necessary. Third, the density and specific heat for stainless steel must now be supplied. The values for density and specific heat are 7,865 kg/m³ and 460 J/kg-°C, respectively. Since the length unit in the input is mm, the value for density had to be converted. Fourth, a PRINTOUT TIMES data block has been added. If this data block had not been included the only information that would have been written to the print and plot files would be for times 0 s and 5 s (i.e. the beginning and ending times of the transient solution.) Fifth, the STEADY-STATE data block has been replaced with a TRANSIENT data block. The TR card indicates that an explicit transient solution is to be performed with a final time of 5 s. The command line for this execution of HEATING is given below.

```
h7 -i exla.dat -r plot1
```

Since one HEATING case has been run, and the output files have not been deleted; the files created by this execution will be *h7-2*, *plot2*, and *print2*. The contents of file *h7-2* are shown in Fig. C.6. The tail end of file *print2* is listed in Fig. C.7 (the beginning of the file is very similar to file *print1*.)

```
Execution Begun on 07/18/91 at 09:01:42

STOP

Real         1.2
User         0.4
System       0.1

HEATING input data    :  /u/kch/heating/h7.1/example/exla.dat
Restart data          :  /u/kch/heating/h7.1/example/plot1

Files created         :  /u/kch/heating/h7.1/example/print2
                         /u/kch/heating/h7.1/example/plot2

Execution Completed on 07/18/91 at 09:01:44
```

Fig. C.6. Contents of file *h7-2*.

```
****************************** INITIAL CONDITIONS ******************************

     Number of time steps completed =        0
     Current time step               = 0.000000000+00
     Current problem time            = 0.000000000+00
     Elapsed cpu time (sec)          = 1.500000000-01

               ***** CURRENT TEMPERATURE SUMMARY *****

           Minimum Temperature = 3.20000E+02   at node      9
           Maximum Temperature = 3.53158E+02   at node      1

             ***** CURRENT BOUNDARY CONDITION SUMMARY *****

     Number      Temperature      Modeled Heat Flow  Neglected Heat Flow
                                    (energy/time)       (energy/time)
       1        1.10000E+02         -7.91681E+00        0.00000E+00

                                       •


*******************************************************************************

               BEGIN TRANSIENT CALCULATION - EXPLICIT TECHNIQUE

*******************************************************************************

     Temperature-dependent material properties and boundary conditions will
     be reevaluated every time step.


Maximum of the stability criterion -  3.3471525D-03
Median of the stability criterion  -  3.3471525D-03
Minimum of the stability criterion -  1.6735763D-03 for point      1

The input time step size is  0.0000000D+00.
the time step size will be set to the stability criterion of  1.6735763D-03.


************************** TRANSIENT SOLUTION OUTPUT **************************

     Number of time steps completed =       299
     Current time step               = 1.6735627D-03
     Current problem time            = 5.003993060-01
     Elapsed cpu time (sec)          = 1.80000000D-01

               ***** CURRENT TEMPERATURE SUMMARY *****

           Minimum Temperature = 2.15800E+02   at node      9
           Maximum Temperature = 2.33167E+02   at node      1

             ***** CURRENT BOUNDARY CONDITION SUMMARY *****

     Number      Temperature      Modeled Heat Flow  Neglected Heat Flow
                                    (energy/time)       (energy/time)
       1        1.10000E+02         -3.98858E+00        0.00000E+00
```

Fig. C.7. Contents of file *print2*.

```
************************** TRANSIENT SOLUTION OUTPUT **************************

     Number of time steps completed =      598
     Current time step            = 1.67357627D-03
     Current problem time         = 1.00079861D+00
     Elapsed cpu time (sec)       = 2.20000000D-01

              ***** CURRENT TEMPERATURE SUMMARY *****

              Minimum Temperature = 1.63412E+02   at node    9
              Maximum Temperature = 1.72179E+02   at node    1

              ***** CURRENT BOUNDARY CONDITION SUMMARY *****

     Number      Temperature     Modeled Heat Flow  Neglected Heat Flow
                                 (energy/time)      (energy/time)
       1        1.10000E+02       -2.01358E+00        0.00000E+00


************************** TRANSIENT SOLUTION OUTPUT **************************

     Number of time steps completed =      896
     Current time step            = 1.67357627D-03
     Current problem time         = 1.49952434D+00
     Elapsed cpu time (sec)       = 2.40000000D-01

              ***** CURRENT TEMPERATURE SUMMARY *****

              Minimum Temperature = 1.37026E+02   at node    9
              Maximum Temperature = 1.41462E+02   at node    1

              ***** CURRENT BOUNDARY CONDITION SUMMARY *****

     Number      Temperature     Modeled Heat Flow  Neglected Heat Flow
                                 (energy/time)      (energy/time)
       1        1.10000E+02       -1.01886E+00        0.00000E+00


************************** TRANSIENT SOLUTION OUTPUT **************************

     Number of time steps completed =     1195
     Current time step            = 1.67357627D-03
     Current problem time         = 1.99992365D+00
     Elapsed cpu time (sec)       = 2.60000000D-01

              ***** CURRENT TEMPERATURE SUMMARY *****

              Minimum Temperature = 1.23644E+02   at node    9
              Maximum Temperature = 1.25883E+02   at node    1

              ***** CURRENT BOUNDARY CONDITION SUMMARY *****

     Number      Temperature     Modeled Heat Flow  Neglected Heat Flow
                                 (energy/time)      (energy/time)
       1        1.10000E+02       -5.14358E-01        0.00000E+00
```

Fig. C.7. Contents of file *print2* (continued).

```
************************* TRANSIENT SOLUTION OUTPUT ************************

    Number of time steps completed =      1494
    Current time step            = 1.67357627D-03
    Current problem time         = 2.50032295D+00
    Elapsed cpu time (sec)       = 2.90000000D-01

              ***** CURRENT TEMPERATURE SUMMARY *****

         Minimum Temperature = 1.16888E+02   at node     9
         Maximum Temperature = 1.18019E+02   at node     1

              ***** CURRENT BOUNDARY CONDITION SUMMARY *****

    Number      Temperature     Modeled Heat Flow  Neglected Heat Flow
                                (energy/time)      (energy/time)
      1         1.10000E+02     -2.59667E-01        0.00000E+00


************************* TRANSIENT SOLUTION OUTPUT ************************

    Number of time steps completed =      1793
    Current time step            = 1.67357627D-03
    Current problem time         = 3.00072226D+00
    Elapsed cpu time (sec)       = 3.10000000D-01

              ***** CURRENT TEMPERATURE SUMMARY *****

         Minimum Temperature = 1.13477E+02   at node     9
         Maximum Temperature = 1.14048E+02   at node     1

              ***** CURRENT BOUNDARY CONDITION SUMMARY *****

    Number      Temperature     Modeled Heat Flow  Neglected Heat Flow
                                (energy/time)      (energy/time)
      1         1.10000E+02     -1.31090E-01        0.00000E+00


************************* TRANSIENT SOLUTION OUTPUT ************************

    Number of time steps completed =      2091
    Current time step            = 1.67357627D-03
    Current problem time         = 3.49944799D+00
    Elapsed cpu time (sec)       = 3.40000000D-01

              ***** CURRENT TEMPERATURE SUMMARY *****

         Minimum Temperature = 1.11759E+02   at node     9
         Maximum Temperature = 1.12048E+02   at node     1

              ***** CURRENT BOUNDARY CONDITION SUMMARY *****

    Number      Temperature     Modeled Heat Flow  Neglected Heat Flow
                                (energy/time)      (energy/time)
      1         1.10000E+02     -6.63305E-02        0.00000E+00
```

Fig. C.7. Contents of file *print2* (continued).

```
************************* TRANSIENT SOLUTION OUTPUT *************************

   Number of time steps completed =     2390
   Current time step              = 1.67357627D-03
   Current problem time           = 3.999847300+00
   Elapsed cpu time (sec)         = 3.600000000-01

             ***** CURRENT TEMPERATURE SUMMARY *****

        Minimum Temperature = 1.10888E+02   at node      9
        Maximum Temperature = 1.11034E+02   at node      1

          ***** CURRENT BOUNDARY CONDITION SUMMARY *****

   Number      Temperature      Modeled Heat Flow  Neglected Heat Flow
                                 (energy/time)        (energy/time)
      1        1.10000E+02       -3.34861E-02        0.00000E+00


************************* TRANSIENT SOLUTION OUTPUT *************************

   Number of time steps completed =     2689
   Current time step              = 1.67357627D-03
   Current problem time           = 4.500246600+00
   Elapsed cpu time (sec)         = 3.800000000-01

             ***** CURRENT TEMPERATURE SUMMARY *****

        Minimum Temperature = 1.10448E+02   at node      9
        Maximum Temperature = 1.10522E+02   at node      1

          ***** CURRENT BOUNDARY CONDITION SUMMARY *****

   Number      Temperature      Modeled Heat Flow  Neglected Heat Flow
                                 (energy/time)        (energy/time)
      1        1.10000E+02       -1.69051E-02        0.00000E+00


************************* TRANSIENT SOLUTION OUTPUT *************************

   Number of time steps completed =     2988
   Current time step              = 1.67357627D-03
   Current problem time           = 5.000645910+00
   Elapsed cpu time (sec)         = 4.000000000-01

             ***** CURRENT TEMPERATURE SUMMARY *****

        Minimum Temperature = 1.10226E+02   at node      9
        Maximum Temperature = 1.10264E+02   at node      1

          ***** CURRENT BOUNDARY CONDITION SUMMARY *****

   Number      Temperature      Modeled Heat Flow  Neglected Heat Flow
                                 (energy/time)        (energy/time)
      1        1.10000E+02       -8.53431E-03        0.00000E+00
```

Fig. C.7. Contents of file *print2* (continued).

```
          The transient calculations have been completed.
          Final time is     5.00065D+00
          Number of time steps completed =  2988


************************** END OF HEATING EXECUTION **************************

Sample Problem 1a

***** Number of warnings  --    0
***** Number of errors    --    0
```

Fig. C.7. Contents of file *print2* (continued).

The steady-state and transient solutions could have both been performed in a single execution of HEATING. The input file for accomplishing this, *ex1b.dat*, is given in Fig. C.8. The time-dependence of the heat generation is defined by tabular function number 1. At time zero the function has a value of 1.0 which drops to 0.0 over the first 1.0×10^{-6} s of the transient and remains there for the remainder of the transient. This value is multiplied with the constant heat generation value (1.12 W/mm^3) to give the actual heat generation. There is a subtle difference between this approach and the two-step solution presented earlier. For the first transient time step the heat generation will be evaluated at a time of zero. Thus the heat generation will not be turned off until the beginning of the second time step. Since the time step is quite small this difference has no practical significance.

```
Sample Problem 1b
* 3 mm stainless steel wire with internal heat generation
* Units:  J, kg, s, mm, C
10  4                                                      P1
                                                           P2
                                                           P3

REGIONS
1  1  0.0  1.5                                             R1
0  1  0   1                                                R2
MATERIALS
1  S-Steel  0.019   7.865e-6  460.0                        M
HEAT GENERATIONS
1  1.12  -1                                                G
BOUNDARY CONDITIONS
1  1  110.0                                                B1
4.0e-3                                                     B2
XGRID
0.0  1.5                                                   X1
     8                                                     X2
TABULAR FUNCTIONS
1                                                          T1
0.0  1.0  1.0e-6  0.0  5.0  0.0                            T2
PRINTOUT TIMES
C.5  1.0  1.5  2.0  2.5  3.0  3.5  4.0  4.5
STEADY-STATE
2                                                          SS
TRANSIENT
1    5.0                                                   TR
0                                                          TR1
%
```

Fig. C.8. Contents of file *ex1b.dat*.

C.2 SAMPLE PROBLEM 2

This problem was selected to demonstrate the enclosure radiation-modeling capabilities of HEATING. The model consists of the finned surface, shown in Fig. C.9. A flat plate comprises the base, which has rectangular fins on its exterior surface. The base has a thickness of 100 mm. The fins are 10 mm thick, 150 mm long, and are spaced on a pitch of 60 mm. The base and fins are made of a mild steel with a conductivity of 50 W/m-K, a specific heat of 500 J/kg-K, and a density of 7800 kg/m^3. The finned surface is assumed to be a diffusely emitting and reflecting gray surface with an emissivity of 0.8. Heat transfer on the outside (finned) surface is by natural convection and radiation. The environmental emissivity is assumed to be 1.0. The external environment is at 38°C. The natural convection heat transfer coefficient is $2.0 \, | \, T_s - T_a \, |^{1/3}$ W/m^2-K, where T_s is the surface temperature and T_a is the ambient temperature. Heat transfer on the inside surface is by forced convection, with a heat transfer coefficient of 1000.0 W/m^2-K. The temperature of the fluid adjacent to the inside surface is 100°C. A steady-state solution is desired.
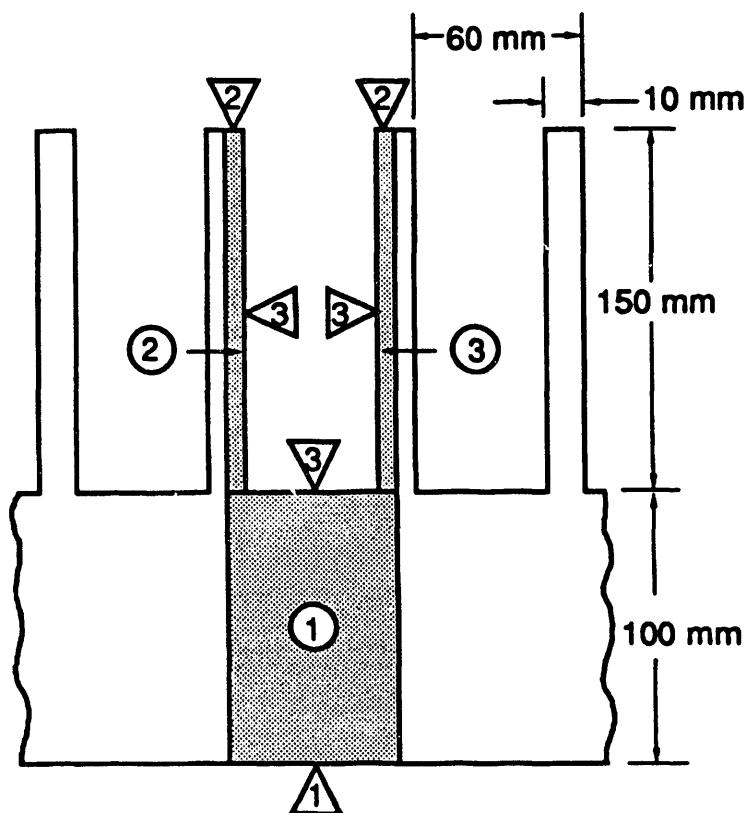
Fig. C.9. Sample problem 2 schematic.

The input data for this sample problem are given in Fig. C.10. In the input data the units are the following: energy, J; length, m; time, s; mass, kg; and temperature, °C. Five regions are used to model the problem. Boundary condition 1 models the forced convection on the inside surface. Boundary condition 2 models the combined radiation and natural convection heat transfer from the ends of the fins. Boundary condition 3 models the natural convection portion of the heat transfer from the surface of the cavity formed by adjacent fins. The radiation portion of the heat transfer from these surfaces is modeled with node-to-node connectors. Boundary condition 4 specifies that radiation is the only active heat transfer mechanism for the node-to-node connectors. A preliminary run of HEATING without the node-to-node connectors (i.e. entry 2 on the P1 card is zero) produces a plot data file that can be used to determine the node numbers on the surfaces of interest with H7MAP, CONPLOT, TECPLOT, or PATRAN (see Appendix D). The H7NODE program (Appendix E) could also be used to determine the node numbers. A node number map produced by H7MAP is presented in Fig. C.11.

```
options
 maxpts=150,mwidth=140
Sample Problem #2 for HEATING 7.1
100 7 0 1
0  1
0  1
regions
1  1  0.0    0.060  0.0   0.10
1  0  0    0      1     3
2  1  0.0    0.005  0.10  0.25
1  0  0    3      0     2
3  1  0.055  0.060  0.10  0.25
1  0  3    0      0     2
materials
1   mldsteel 50.0 7800.0  500.0
initial temperatures
1 1.0 0 -1
boundary conditions
1  1  100.0
1000.0
2  1  38.0
0  4.5359d-8  2.0  0.33
3  1  38.0
0  0  2.0  0.33
4  1  38.0
0  5.6699d-8
xgrid
0.00  0.005  0.055  0.060
      1     5     1
ygrid
0.0   0.10  0.25
     10    15
tabular functions
1
0.0  65.0   0.250  95.0
steady state
2  20  1.0-8
X
```

Fig. C.10. Contents of file *ex2.dat*.

```
Sample Problem #2 for HEATING 7.1                                    Thu Jul 18 12:39:19 1991
                                        Map of the Node Numbers

   26   .25 |  145   146                                              147   148
   25   .24 |  141   142                                              143   144
   24   .23 |  137   138                                              139   140
   23   .22 |  133   134                                              135   136
   22   .21 |  129   130                                              131   132
   21   .20 |  125   126                                              127   128
   20   .19 |  121   122                                              123   124
   19   .18 |  117   118                                              119   120
   18   .17 |  113   114                                              115   116
   17   .16 |  109   110                                              111   112
   16   .15 |  105   106                                              107   108
   15   .14 |  101   102                                              103   104
   14   .13 |   97    98                                               99   100
   13   .12 |   93    94                                               95    96
   12   .11 |   89    90                                               91    92
   11   .10 |   81    82    83    84    85    86                       87    88
   10   .09 |   73    74    75    76    77    78                       79    80
    9   .08 |   65    66    67    68    69    70                       71    72
    8   .07 |   57    58    59    60    61    62                       63    64
    7   .06 |   49    50    51    52    53    54                       55    56
    6   .05 |   41    42    43    44    45    46                       47    48
    5   .04 |   33    34    35    36    37    38                       39    40
    4   .03 |   25    26    27    28    29    30                       31    32
    3   .02 |   17    18    19    20    21    22                       23    24
    2   .01 |    9    10    11    12    13    14                       15    16
    1   .00 |    1     2     3     4     5     6                        7     8
           +--------------------------------------------------------------------
              .00   .00   .02   .02   .04   .05                      .06   .06
               1     2     3     4     5     6                        7     8
```

Fig. C.11. Node numbers on surface of fins.

Node-to-node connector data must be calculated external to HEATING and supplied either in an unformatted node-to-node connector file or in the CONNECTOR data block. Since connector data for a radiation problem can be quite voluminous, it is generally more convenient to supply an unformatted connector file to HEATING as is done in this sample problem. In the calculation of the radiation exchange factors, it is necessary to define a complete enclosure so that all of the radiation heat transfer is accounted for. Since there is not a physically defined enclosure in this problem, the plane extending between the tips of adjacent fins is included as a fictitious surface. A node number of -4 (indicating the external environment modeled by boundary condition 4) is used to indicate radiation to this fictitious surface. The total area associated with a corner node (nodes 82 and 87) includes an area on two separate surfaces: one horizontal and one vertical. Exchange factors were calculated separately for each of the two surfaces making up these corner nodes. Therefore, nodes 82 and 87 appear twice in the input of connectors for each node. A node-to-node connector is also included in the data for radiation from each node to itself. It is not necessary to include this information in the input, and, in fact, this connection will be eliminated by HEATING before calculations begin. It is included so that a quick check can be made to ensure that the sum of all connectors for each node equals the product of its area and emissivity. Reciprocity is used to reduce the size of the node-to-node connector file by almost one-half. The printed output for this sample problem is not presented

in this report. However, a portion of the summary of node-to-node connector data produced by the code (requested by entering a value of one for the second entry of the P2 card) is presented in Fig. C.12. In this printout, a negative sign on a connectivity is used to indicate that a particular connector was generated by reciprocity.

*********SUMMARY OF NODE-TO-NODE CONNECTOR DATA*********

```
          in the connector data a minus sign is a flag used as follows:
          *negative node number  -  connection is to a boundary
          *negative connectivity -  connection generated by reciprocity
```

connectors using heat transfer mechanism(s) specified by boundary condition   4.

base/   sum of   /-----------------------------------connections to base node-----------------------------------
node/connectivities/node connectivity/node connectivity/node connectivity/node connectivity/node connectivity

node	connectivity	node	connectivity	node	connectivity	node	connectivity	node	connectivity	node	connectivity
82	**8.0000D-03**										
-4	6.9391E-04	82	6.5080E-05	82	2.0482E-04	82	9.4746E-04	82	2.0220E-05	82	7.2269E-05
83	4.9385E-05	84	2.6471E-05	84	4.2011E-04	85	1.1916E-04	85	1.6535E-04	85	6.5912E-05
86	1.6905E-05	87	2.5076E-05	87	5.1734E-05	87	6.3553E-06	87	-1.1851E-04	87	2.5076E-05
90	-4.2019E-04	91	9.8728E-05	91	-6.5145E-05	94	3.2389E-04	94	1.5490E-04	94	-4.2843E-05
95	1.4111E-04	98	-6.2784E-05	98	2.8777E-04	99	-3.5797E-05	99	-3.4324E-05	99	2.3516E-04
102	-4.3543E-05	103	1.4900E-04	103	-3.1327E-05	106	1.8273E-04	106	1.1639E-04	106	-2.7426E-05
107	1.3389E-04	110	-2.8743E-05	110	1.3883E-04	111	-2.3759E-05	111	-2.1527E-05	111	1.0509E-04
114	-2.4735E-05	115	9.9792E-05	115	-2.0354E-05	118	8.0099E-05	118	7.2970E-05	118	-1.7283E-05
119	8.5257E-05	122	-1.8800E-05	122	6.1796E-05	123	-1.4581E-05	123	-1.4257E-05	123	4.8349E-05
126	-1.6405E-05	127	6.2728E-05	127	-1.2244E-05	130	3.8366E-05	130	4.7126E-05	130	-1.0239E-05
131	5.4217E-05	134	-1.2298E-05	134	3.0852E-05	135	-8.5218E-06	135	-8.8012E-06	135	2.5109E-05
138	-1.0485E-05	139	4.1184E-05	139	-7.0441E-06	142	2.0651E-05	142	1.6455E-05	142	-5.7691E-06
143	3.6176E-05	146	-3.8126E-06	143	1.7142E-05	146	-2.4638E-06			147	7.4856E-06
83	**8.0000D-03**										
-4	1.4164E-03	82	-4.2011E-04	82	-4.9385E-05	83	8.1818E-05	84	6.1879E-05	85	5.1643E-05
86	4.4448E-05	87	5.1734E-05	87	1.6905E-05	90	-1.1036E-03	91	2.3836E-04	94	-5.9538E-04
95	3.4245E-04	98	3.6053E-04	99	-3.3939E-04	102	-2.2163E-04	103	3.3090E-04	106	-1.5968E-04
107	2.8558E-04	110	2.4042E-04	111	-1.2264E-04	114	-9.8091E-05	115	2.0102E-04	118	-8.0501E-05
119	1.6837E-04	122	1.4183E-04	123	-6.7176E-05	126	-5.6663E-05	127	1.2032E-04	130	-4.8104E-05
131	1.0283E-04	134	8.8491E-05	135	-4.0951E-05	138	-3.4836E-05	139	7.6611E-05	142	-2.9528E-05
143	6.6679E-05	146	3.0119E-05	147	-1.2993E-05						
84	**8.0000D-03**										
-4	1.4374E-03	82	-2.6471E-05	82	6.5912E-05	83	-6.1879E-05	85	5.8106E-05	85	5.4936E-05
86	5.1643E-05	87	6.5912E-05	87	2.0220E-05	90	-5.9703E-04	94	3.4629E-04	94	-6.0016E-04
95	4.5564E-04	98	-4.5046E-04	99	4.2478E-04	102	-3.2891E-04	106	3.5370E-04	106	-2.4656E-04

Fig. C.12 Partial listing of table of node-to-node connectors.

node	value	node	value	node	value	node	value	node	value	node	value
107	2.8536E-04	110	-1.9075E-04	111	2.2972E-04	114	-1.5156E-04	115	1.8640E-04	118	-1.2296E-04
119	1.5291E-04	122	-1.0139E-04	123	1.2686E-04	126	-8.4658E-05	127	1.0635E-04	130	-7.1367E-05
131	8.9980E-05	134	-6.0575E-05	135	7.6705E-05	138	-5.1638E-05	139	6.5781E-05	142	-4.4121E-05
143	5.6675E-05	146	-1.9609E-05	147	2.5389E 05						

85 8.0000D-03

node	value	node	value	node	value	node	value	node	value	node	value
-4	1.4374E-03	82	-2.0220E-05	82	-6.5912E-05	83	-5.1643E-05	84	-5.4936E-05	85	5.8106E-05
86	6.1879E-05	87	1.1916E-04	87	2.6471E-05	90	-3.4629E-04	91	5.9703E-04	94	-4.5564E-04
95	6.0016E-04	98	-4.2478E-04	99	4.5046E-04	102	-3.5370E-04	103	3.2891E-04	106	-2.8536E-04
107	2.4656E-04	110	-2.2972E-04	111	1.9075E-04	114	-1.8640E-04	115	1.5156E-04	118	-1.5291E-04
119	1.2296E-04	122	-1.2686E-04	123	1.0139E-04	126	-1.0635E-04	127	8.4658E-05	130	-8.9980E-05
131	7.1367E-05	134	-7.6705E-05	135	6.0575E-05	138	-6.5781E-05	139	5.1638E-05	142	-5.6675E-05
143	4.4121E-05										

86 8.0000D-03

node	value	node	value	node	value	node	value	node	value	node	value
-4	1.4164E-03	82	-1.6905E-05	82	-5.1734E-05	83	-4.4448E-05	84	-5.1643E-05	85	-6.1879E-05
86	8.1818E-05	87	4.2011E-04	87	4.9385E-05	90	-2.3836E-04	91	1.1036E-03	94	-3.4245E-04
95	5.9538E-04	98	-3.6053E-04	99	3.3939E-04	102	-3.3090E-04	103	2.2163E-04	106	-2.8558E-04
107	1.5968E-04	110	-2.4042E-04	111	1.2264E-04	114	-2.0102E-04	115	9.8091E-05	118	-1.6837E-04
119	8.0501E-05	122	-1.4183E-04	123	6.7176E-05	126	-1.2032E-04	127	5.6663E-05	130	-1.0283E-04
131	4.8104E-05	134	-8.8491E-05	135	4.0951E-05	138	-7.6611E-05	139	3.4836E-05	142	-6.6679E-05
143	2.9528E-05	146	-3.0119E-05	147	1.2993E-05						

87 8.0000D-03

node	value	node	value	node	value	node	value	node	value	node	value
-4	2.0482E-04	82	6.9391E-04	82	-2.5076E-05	82	-6.3553E-06	82	-1.6535E-04	82	-2.5076E-05
83	-5.1734E-05	83	-1.6905E-05	84	-6.5912E-05	84	-2.0220E-05	85	-1.1916E-04	85	-2.6471E-05
86	-4.2011E-04	86	-4.9385E-05	87	7.2269E-05	87	9.4746E-04	87	9.4746E-04	87	6.5080E-05
90	-9.8728E-05	90	-3.2389E-04	91	6.5145E-05	91	4.2019E-04	94	-2.8777E-04	94	-1.4111E-04
95	1.1851E-04	95	4.2843E-05	98	-2.3516E-04	98	-1.5490E-04	99	3.5797E-05	99	6.2784E-05
102	-1.4900E-04	102	-1.8273E-04	103	3.1327E-05	103	4.3543E-05	106	-1.3883E-04	106	-1.3389E-04
107	3.4324E-05	107	2.7426E-05	110	-1.0509E-04	110	-1.1639E-04	111	2.3759E-05	111	2.8743E-05
114	-9.9792E-05	114	-8.0099E-05	115	2.0354E-05	115	2.4735E-05	118	-6.1796E-05	118	-8.5257E-05
119	2.1527E-05	119	-1.7283E-05	122	-4.8349E-05	122	-7.2970E-05	123	1.4581E-05	123	1.8800E-05
126	-6.2728E-05	126	-3.8366E-05	127	1.2244E-05	127	1.6405E-05	130	-3.0852E-05	130	-5.4217E-05
131	1.4257E-05	131	1.0239E-05	134	-2.5109E-05	134	-4.7126E-05	135	8.5218E-06	135	1.2298E-05
138	-4.1184E-05	138	-2.0651E-05	139	7.0441E-06	139	1.0485E-05	142	-1.7142E-05	142	-3.6176E-05
143	8.8012E-06	143	5.7691E-06	146	-7.4856E-06	146	-1.6455E-05	147	2.4638E-06	147	3.8126E-06

90 8.0000D-03

node	value	node	value	node	value	node	value	node	value	node	value
-4	4.2273E-04	82	4.2019E-04	82	6.5145E-05	83	1.1036E-03	84	5.9703E-04	85	3.4629E-04
86	2.3836E-04	87	3.2389E-04	87	9.8728E-05	90	1.1612E-04	91	6.7321E-04	94	-9.7334E-05
95	6.4020E-04	98	-8.5211E-05	99	5.4933E-04	102	-7.5439E-05	103	4.3836E-04	106	-6.6292E-05
107	3.3576E-04	110	-5.7514E-05	111	2.5308E-04	114	-4.9285E-05	115	1.9079E-04	118	-4.1818E-05
119	1.4513E-04	122	-3.5227E-05	123	1.1184E-04	126	-2.9522E-05	127	8.7437E-05	130	-2.4639E-05
131	6.9317E-05	134	-2.0473E-05	135	5.5661E-05	138	-1.6913E-05	139	4.5206E-05	142	-1.3866E-05
143	3.7078E-06	146	-5.9337E-06	147	1.6049E-05						

Fig. C.12 Partial listing of table of node-to-node connectors (continued).

C.3 SAMPLE PROBLEM 3

This sample problem was defined for instructive purposes and is not meant to represent a real engineering problem. The problem is two-dimensional in *x-y* coordinates and consists of three materials. Its configuration is shown in Fig. C.13. Numbers in circles identify regions, and numbers in triangles identify boundary conditions. The units used were Btu, °F, lb, in., and min.
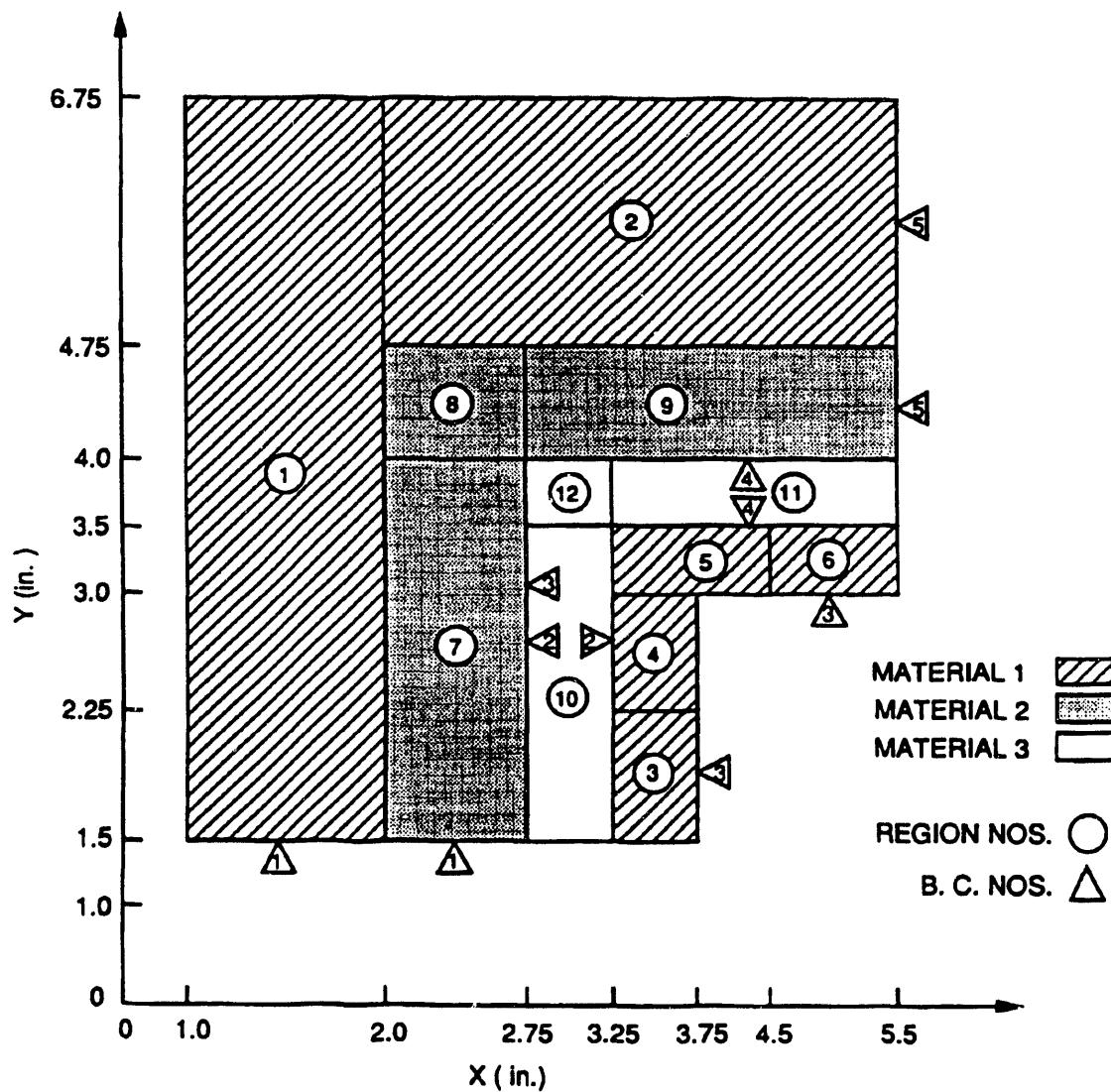
Fig. C.13. Sample problem 3 schematic.

Regions 1 to 6 contain material 1 (iron); regions 7 to 9 contain material 2 (stainless steel). An air gap between the two metals is modeled by regions 10, 11 and 12. The physical properties of these materials are given in Table C.1.

Table C.1. Material physical properties for sample problem 1

Property/Material	Iron (Material No. 1)	Stainless Steel (Material No. 2)	Air (Material No. 3)
Conductivity Btu/min-in-°F	0.0296 at 0°F 0.0264 at 752°F 0.0222 at 1832°F	0.013 at 0°F 0.0153 at 752°F 0.025 at 1832°F	1.82×10^{-5} at 0°F 3.41×10^{-5} at 500°F 4.68×10^{-5} at 1000°F 5.75×10^{-5} at 1500°F
Density lb/in^3	0.2801	0.2824	5.00×10^{-5} at 0°F 2.39×10^{-5} at 500°F 1.57×10^{-5} at 1000°F 1.17×10^{-5} at 1500°F
Specific Heat Btu/lb-°F	0.116	0.11	0.25

A spatially uniform heat generation exists in regions 1 and 2 at the rate of 1.0 Btu/(min-in^3), which varies according to time function 2 given in Fig. C.14. The initial temperature is a uniform 100°F. The boundary conditions on each of the faces are shown in Fig. C.13, and they are numbered in triangular frames. Surfaces with boundary condition 1 are in perfect thermal contact with a fluid. The fluid temperature is initially 200°F but varies with time according to time function 1 given in Fig. C.14. Boundary condition 2 is radiation across an air gap (region 10) between the two metals (emissivity $\epsilon = 0.8$). Conduction and natural convection are neglected. Boundary condition 3 is forced convection to a fluid at 68°F (one face of region 10 only). The heat transfer coefficient is 0.006 Btu/(min·in.^{2}·°F). Boundary condition 4 is combined heat transfer by radiation and natural convection across an air gap (region 11) between two metal surfaces (emissivity $\epsilon = 0.8$). Heat is also transferred by conduction through the air. The natural convection heat transfer coefficient is given by

$$h = 2.56 \times 10^{-5} \Delta T^{0.33} . \tag{C.1}$$

Boundary condition 5 has a time-dependent heat flux given by

$$h_f = 0.03 \cos\left[\frac{\pi}{360} t\right] , \tag{C.2}$$

and cooling by radiation and natural convection to the ambient at 100°F. The rest of the boundaries are insulated. Region 12 cannot be described for surface-to-surface radiation or natural convection because of the lack of opposing surfaces. Conduction through air could be taken into account but is neglected in this case. Therefore, the region definition is not included in the input file. The transient temperature distributions at 30 and 60 min are output, along with the steady-state temperature distribution, resulting from evaluating all time functions at 50 min. The temperatures at points (1.0,1.5) (3.75,3.0), (2.75,4.0), (5.5,4.0), and (5.5,6.75) are monitored every 10 time steps or iterations. Sixty seconds of CPU time are requested for problem execution.
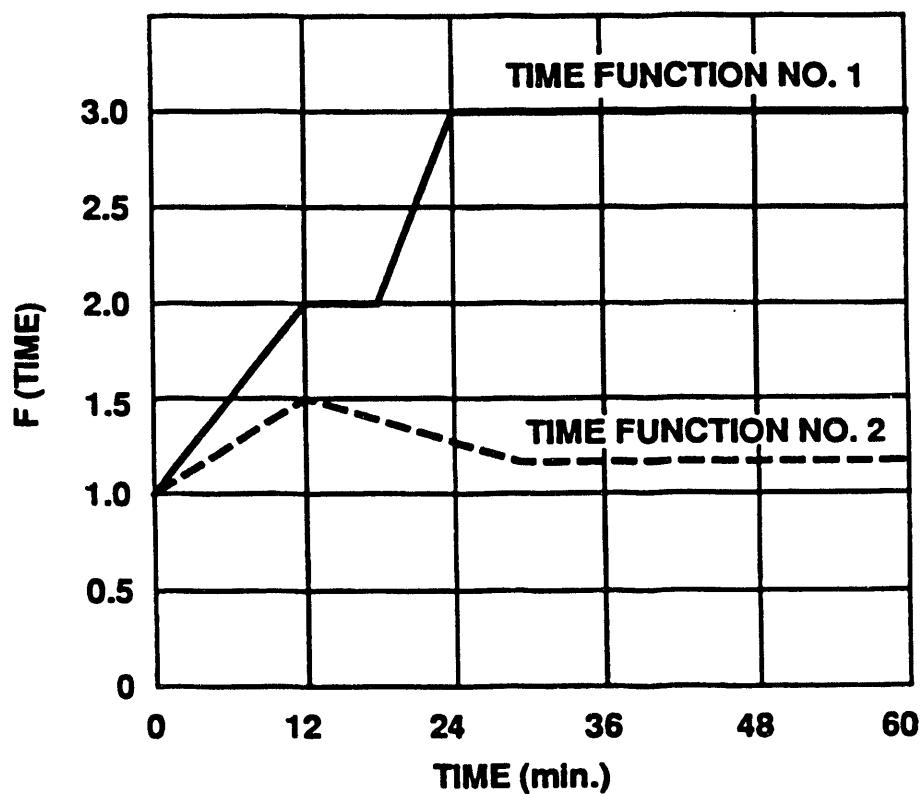
Fig. C.14. Time-dependent functions for sample problem 3.

The input data file for this problem is given in Fig. C.15. The transient calculations use the Crank-Nicolson Procedure, with an initial time-step size of 0.1 min. The time-step size is then allowed to vary by keeping the maximum relative temperature change at a node to 2.5% over a time step. The steady-state solution is obtained by using the direct-solution technique.

```
Sample Problem 3                                          title
60 7 0.0                                                  p1
0 0 1                                                     p2
0 0                                                       p3
regions
1 1 1.0 2.0 1.5 6.75                                      r1
1 1  0    0   1                                           r2
2 1 2.0 5.5 4.75 6.75                                     r1
1 1  0    5                                               r2
3 1 3.25 3.75 1.5 2.25                                    r1
1 0  0    3                                               r2
4 1 3.25 3.75 2.25 3.0                                    r1
1                                                         r2
5 1 3.25 4.5 3.0 3.5                                      r1
1                                                         r2
6 1 4.5 5.5 3.0 3.5                                       r1
1 0  0    0   3                                           r2
7 2 2.0 2.75 1.5 4.0                                      r1
1 0  0    3    1                                          r2
8 2 2.0 2.75 4.0 4.75                                     r1
1                                                         r2
9 2 2.75 5.5 4.0 4.75                                     r1
1 0   5                                                   r2
10 0 2.75 3.25 1.5 3.5                                    r1
1 0   2    2                                              r2
11 3 3.25 5.5 3.5 4.0                                     r1
1 0   0    4   4                                          r2
materials
1 iron 0 0.2801 0.116 -3                                  m
2 stainlss 0 0.2824 0.11 -4                               m
3 air 0 0 0.25 -5 -6                                      m
initial temperatures
1 100.0                                                   i
heat generations
1 1.0 -2                                                  g
boundary conditions
1 2 200.0 -1                                              b1
                                                          b2
2 3                                                       b1
0 1.58d-13                                                b2
3 1 68.0                                                  b1
6.0d-3                                                    b2
4 3                                                       b1
0 1.58d-13 2.56d-05 0.33                                  b2
5 1 100.0                                                 b1
0 1.58d-13 2.56d-5 0.33 0 1                               b2
0 0 0 0 1                                                 b3
xgrid
1.0 2.0 2.75 3.25 3.75 4.5 5.5                            l1
    2   1    1    1    1   1                              n1
ygrid
1.5 2.25 3.0 3.5 4.0 4.75 6.75                            l2
    1    1   1   1    1    4                              n2
analytical functions
1                                                         a1
4 0.03 5 0.0087266                                        a2
tabular functions
1                                                         t1
0.0 1.0, 12.0 2.0, 18.0 2.0, 24.0 3.0                     t2
```

Fig. C.15. Input data for sample problem 3.

```
2                                                              t1
0.0 1.0, 12.0 1.5, 30.0 1.125                                  t2
3                                                              t1
0.0 0.0296, 752.0 0.0264, 1832.0 0.0222                       t2
4                                                              t1
0.0 0.013, 752.0 0.0153, 1832.0 0.025                         t2
5                                                              t1
0.0 1.82d-5, 500.0 3.41d-5, 1000.0 4.68d-5, 1500.0 5.75d-5    t2
6                                                              t1
0.0 5.0d-5, 500.0 2.39d-5, 1000.0 1.57d-5, 1500.0 1.17d-5     t2
printout times
30.0 60.0                                                      o
nodes monitored
10 5 1 18 32 36 76                                             s
transient
2 60.0                                                         tp
                                                              tr2
0.1 0 5 0 0 2.5                                                tr3
steady state
2 20                                                           ss
%
```

Fig C.15. Input data for sample problem 3 (continued).

C.4 SAMPLE PROBLEM 4

Changes are made to sample problem 3 to demonstrate some additional capabilities of HEATING. The initial temperature varies as a function of y according to the following expression

$$T_o(y) = 235 - 20y \; , \tag{C.3}$$

and the heat generation rate in region 1 is a sum of exponentials defined by the expression

$$Q_1(t) = \sum_{i=1}^{3} C_{i1} e^{-\lambda_{1i} t} \; , \tag{C.4}$$

where the parameters are defined as

i	C_{i1}	λ_{1i}
1	0.5	0.0115525
2	0.3	0.0231049
3	0.2	0.0462098 ;

the heat generation rate in region 2 is a sum of exponentials defined by the expression

$$Q_2(t) = \sum_{i=1}^{2} C_{i2} e^{-\lambda_{2i} t} \; , \tag{C.5}$$

where the parameters are defined as

i	C_{i2}	λ_{2i}
1	0.6	0.0115525
2	0.4	0.0462098 .

Furthermore, the thermal conductivity for iron is assumed to be anisotropic, with the conductivity along the y-axis equal to twice that along the x-axis, as presented in Table C.1. The initial temperature is input to the code as an analytical function, but the two heat generation rates and the conductivity for iron have to be defined by user-supplied subroutines. The input data are presented in Fig. C.16. Tabular function numbers 2 and 3 are part of the input data but are not used. The user-supplied subroutines for the heat generation rate and the thermal conductivity for iron are presented in Fig. C.17 and C.18, respectively. The printed output for this sample problem is not presented in this report.

```
OPTIONS
MAXPTS=80,MWIDTH=17                                      OPTION
SAMPLE PROBLEM 4                                         TITLE
60 7 0.0                                                 P1
0 0 1                                                    P2
0 0                                                      P3
REGIONS
1 1 1.0 2.0 1.5 6.75                                     R1
1 1  0   0   1                                           R2
2 1 2.0 5.5 4.75 6.75                                    R1
1 1  0   5                                               R2
3 1 3.25 3.75 1.5 2.25                                   R1
1 0  0   3                                               R2
4 1 3.25 3.75 2.25 3.0                                   R1
1                                                        R2
5 1 3.25 4.5 3.0 3.5                                     R1
1                                                        R2
6 1 4.5 5.5 3.0 3.5                                      R1
1 0  0   0  3                                            R2
7 2 2.0 2.75 1.5 4.0                                     R1
1 0  0   3   1                                           R2
8 2 2.0 2.75 4.0 4.75                                    R1
1                                                        R2
9 2 2.75 5.5 4.0 4.75                                    R1
1 0  0   5                                               R2
10 0 2.75 3.25 1.5 3.5                                   R1
1 0   2   2                                              R2
11 3 3.25 5.5 3.5 4.0                                    R1
1 0   0   0  4  4                                        R2
MATERIALS
1 IRON 0 0.2801 0.116  3                                 M
2 STAINLSS 0 0.2824 0.11 -4                              M
3 AIR 0 0 0.25 -5 -6                                     M
INITIAL TEMPERATURES
1 0 0 2                                                  I
HEAT GENERATIONS
1 1.0 3                                                  G
BOUNDARY CONDITIONS
1 2 200.0 -1                                             B1
                                                        B2
2 3                                                      B1
0 1.58D-13                                               B2
3 1 68.0                                                 B1
6.0D-3                                                   B2
4 3                                                      B1
0 1.58D-13 2.56D-05 0.33                                 B2
5 1 100.0                                                B1
0 1.58D-13 2.56D-5 0.33 0 1                              B2
0 0 0 0 1                                                B3
XGRID
1.0 2.0 2.75 3.25 3.75 4.5 5.5                           X1
    2   1    1    1    1   1                             X2
YGRID
1.5 2.25 3.0 3.5 4.0 4.75 6.75                           Y1
    1    1   1   1    1    4                             Y2
ANALYTICAL FUNCTIONS
1                                                        A1
4 0.03 5 0.0087266                                       A2
2                                                        A1
1 235.0 2 -20.0                                          A2
3                                                        A1
                                                        A2
```

Fig. C.16. Input data for sample problem 4.

```
TARULAR FUNCTIONS
1                                                                    T1
0.0 1.0,  12.0 2.0,  18.0 2.0,  24.0 3.0                            T2
2                                                                    T1
0.0 1.0,  12.0 1.5,  30.0 1.125                                     T2
3                                                                    T1
0.0 0.0296,  752.0 0.0264,  1832.0 0.0222                           T2
4                                                                    T1
0.0 0.013,  752.0 0.015%,  1832.0 0.025                             T2
5                                                                    T1
0.0 1.82D-5,  500.0 3.41D-5,  1000.0 4.68D-5,  1500.0 5.75D-5       T2
6                                                                    T1
0.0 5.0D-5,  500.0 2.39D-5,  1000.0 1.57D-5,  1500.0 1.17D-5        T2
PRINTOUT TIMES
30.0 60.0                                                           O
NODES MONITORED
10 5 1 18 32 36 76                                                 S
TRANSIENT
2 60.0                                                              TR
                                                                    TR2
0.1 0 5 0 0 2.5                                                     TR3
STEADY STATE
2 20                                                                SS
%
```

Fig. C.16. Input data for sample problem 4 (continued).

```
      subroutine heatgn(rvalue,r,th,z,tim,tsn,value,number,n,arg,val,
    . ntbprs,ntab,hival,loval)
c ***********************************************************************
c      this user-supplied subroutine calculates the heat generation
c      rate for heat generation functions 1 and 2 for sample problem
c      number 4 in the heating manual.
c ***********************************************************************
      implicit double precision (a-h,o-z)
      common /iounit/ iecho , ihstry, imatlb, in    , io    , iplot ,
    .                 iplot0, icnvrg, iconn
      dimension        arg(1) , val(1)
      integer          ntbprs(1), ntab(1)
      logical          hival(1), loval(1)
      dimension        c1(3) , xlmda1(3), c2(2), xlmda2(2)
      data   c1        /0.5d0,0.3d0,0.2d0/
      data   xlmda1    /1.15525d-2,2.31049d-2,4.62098d-2/
      data   c2        /0.6d0,0.4d0/
      data   xlmda2    /1.15525d-2,4.62098d-2/
c ***********************************************************************
      rvalue = 0.0d0
      if(number.eq.1) then
        do 10 i=1,3
          rvalue = rvalue+c1(i)*dexp(-xlmda1(i)*tim)
  10    continue
      elseif(number.eq.2) then
        do 20 i=1,2
          rvalue = rvalue+c2(i)*dexp(-xlmda2(i)*tim)
  20    continue
      else
        write(io,1000) number
        stop
      endif
      return
 1000 format('0****** user supplied subroutine heatgn has been called to
    . evaluate the'/' ****** heat generation rate for heat generation f
    .unction number',i5,'.'/' ****** this function is not defined here,
    . so the calculations will be'/' ****** terminated.')
      end
```

Fig. C.17. User-supplied subroutine HEATGN for sample problem 4.

```
      subroutine condtn(rvalue,r,th,z,tim,tsn,value,number,n,arg,val,
     . ntbprs,ntab,hival,loval)
c ***********************************************************************
c      this user-supplied subroutine calculates the anisotropic,
c      temperature-dependent conductivity for material 1.  the
c      temperature-dependent conductivity in the x-direction is
c      given by the third tabular function in the input.  the
c      conductivity in the y-direction is twice that in the
c      x-direction.  used with sample problem 4 in heating manual.
c ***********************************************************************
      implicit double precision (a-h,o-z)
      common /surbcd/  ndir   , ndum   , cdty(24), condty(24), nsconn,
     .                 ndmno(24), nsnode(24)
      common /iounit/  iecho , ihstry, imatlb, in    , io    , iplot ,
     .                 iplot0, icnvrg, iconn
      dimension        arg(1) , val(1)
      integer          ntbprs(1), ntab(1)
      logical          hival(1), loval(1)
c ***********************************************************************
      if(number.eq.1) then
        call table(arg,val,ntbprs,ntab,hival,loval,3,tsn,rvalue)
        if(ndir.eq.2) rvalue = 2.0d0*rvalue
      else
        write(io,1000) number
        stop
      endif
      return
 1000 format('0****** user supplied subroutine condtn has been called to
     . evaluate the'/' ****** conductivity for material number',i5,'.
     .the conductivity for this'/' ****** material is not defined here,
     .so the calculations will be'/' ****** terminated.')
      end
```

Fig. C.18. User-supplied subroutine CONDTN for sample problem 4.

Appendix D. POSTPROCESSING

The normal printed output from a HEATING run only provides a summary of the solution since HEATING places a emphasis on postprocessing as the primary means of examining the results from an analysis. There are four computer programs available for graphical postprocessing of HEATING analyses — HEATPLOT,[1] CONPLOT, PATRAN,[2] and TECPLOT.[3] Additionally, two computer programs are available to produce tabular output - H7MAP and H7MONITOR. All of these approaches make use of the data stored in a plot data file produced by every execution of HEATING. The user may also write his own program that accesses HEATING plot data files to produce customized output. The procedure for reading this file is explained in Sect. 5.3.

D.1 HEATPLOT

HEATPLOT has the capability of generating temperature contours for any coordinate surface, temperature-vs-time plots, and temperature-vs-distance plots for two- or three-dimensional models. Temperature-vs-time and temperature-vs-distance plots may also be generated for one-dimensional models. It is best not to mix plot types during a single execution of HEATPLOT (e.g., if both a plot of temperature contours and a plot of temperature vs time are desired, they should be generated in two separate executions of HEATPLOT). HEATPLOT uses the CA-DISSPLA[4] graphics package and produces a compressed plot data file in dmf (DISSPLA meta file) format. Currently, the only computer at Martin Marietta Energy Systems on which HEATPLOT runs is the UNICOS CRAY. On this machine HEATPLOT produces a dmf file named *metfil*. The user can view results at a graphics terminal by use of the *dispop* command or route plots to various hardcopy devices by use of the *plot* command.

Typically, data defining the plots to be produced by HEATPLOT will be entered by the user in an interactive mode in response to prompts by the code. However, if a user anticipates producing similar plots for several executions of HEATPLOT, the input data can be saved in a file on the first execution. For subsequent executions, the code can read the data from this file and produce plots without the user reentering the data.

The command line for executing HEATPLOT on the UNICOS CRAY computer at Martin Marietta Energy Systems is

hp [-p *pfile*] [-i *ifile*]

where the brackets around each of the options indicate that they are not required. The options are defined below.

Options:

-p *pfile*   This option specifies the name of the file in the current working directory that contains plot data produced by a previous HEATING run. If this option is not specified, a file by the name of *ft01f001* must be present.

-i *ifile*   This option specifies the name of a file containing HEATPLOT input, defining the plots to be produced. HEATPLOT data specifying the plots to be produced can be entered interactively during execution of HEATPLOT or read from this file. This file would typically be saved from a previous interactive HEATPLOT execution, but it can be created by the user.

In an execution of HEATPLOT, the user is first asked if data defining the plots to be produced are to be entered interactively or are to be read from a data file. If data are to be read from a file, no additional user input is required. However, if the data are to be entered interactively the user is prompted for the length of the x and y axes. Next, the main selection menu (shown below) is displayed.

```
Choose an input option:

(0)   exit and generate disspla meta file
(1)   contour plots
(2)   temperature-time plots
(3)   temperature-distance plots
(4)   problem times
(5)   problem units
(6)   scaling parameters
(7)   plot region specification
(8)   change problem title
(9)   review problem summary
```
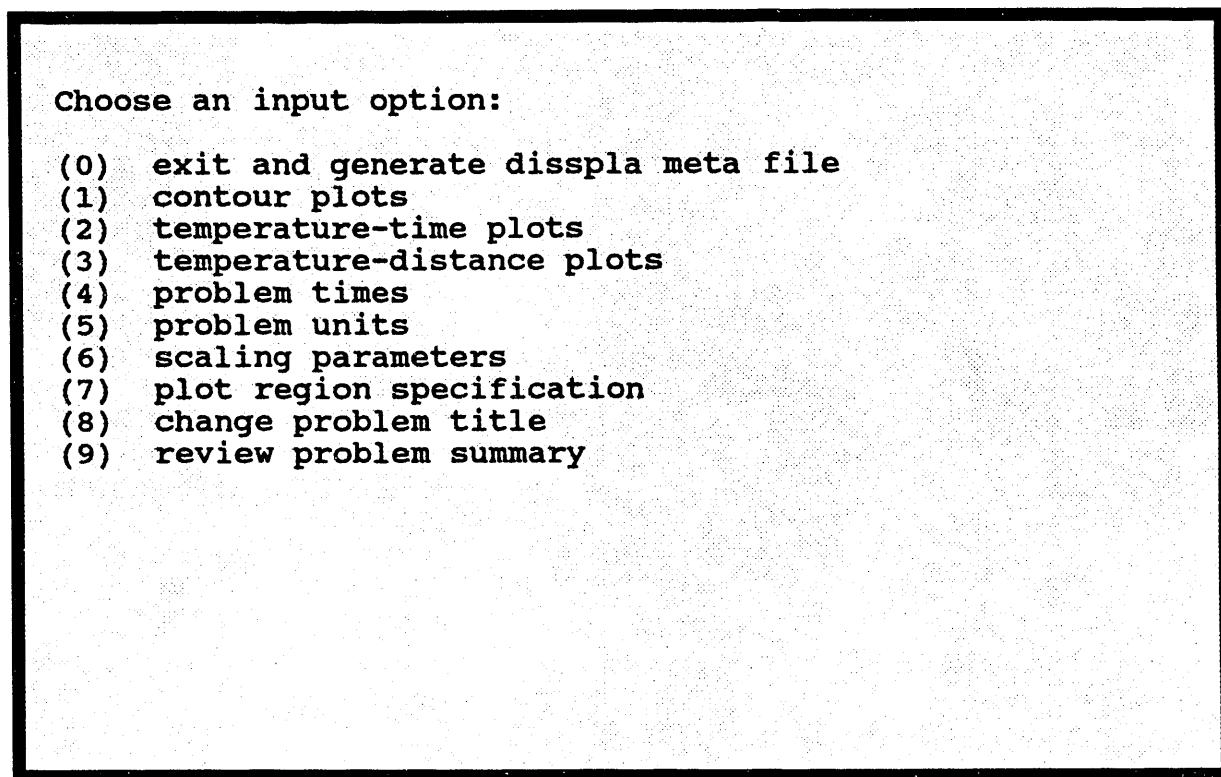
When an option is selected, the user will be prompted to answer a series of questions defining the selected option. The prompts are self-explanatory. Option 0 "exit and generate disspla meta file" is selected after the user has selected all other applicable options. If an option

---

In addition to plotting results, CONPLOT is also useful in model checking. A preliminary run can be made with HEATING in which there are no STEADY-STATE or TRANSIENT data blocks supplied. The plot data file resulting from this run can be used to verify the model geometry and material distribution.

D.3 H7PATRAN/PATRAN

PATRAN is a commercially available software package that, among many other capabilities, can be used to view meshes or postprocess results from engineering analysis codes. PATRAN cannot read the unformatted HEATING plot data file directly. The user must first run the HEATING-to-PATRAN interface code H7PATRAN to produce formatted data files for PATRAN. H7PATRAN is installed on the UNICOS CRAY at Martin Marietta Energy Systems and is available for most UNIX workstations. PATRAN is installed on two of the Scientific and Technical Computing mainframe computers - STC10 and STC06. The command line for executing H7PATRAN on the UNICOS CRAY or a UNIX workstation at Martin Marietta Energy Systems is

```
h7pat
```

The user is prompted for the name of an existing HEATING plot data file and names for PATRAN neutral and results files to be created.

D.4 H7TECPLOT/TECPLOT

TECPLOT is a commercially available interactive plotting program for visualizing engineering and scientific data. It can produce XY plots, mesh plots, contour plots, and vector plots. TECPLOT cannot read the unformatted HEATING plot data file directly. The user must first run the HEATING-to-TECPLOT interface code H7TECPLOT to produce formatted data files for TECPLOT. H7TECPLOT is installed on the UNICOS CRAY at Martin Marietta Energy Systems and is available for most UNIX workstations. TECPLOT is available only on specific workstations for which a license has been purchased from Amtec Engineering, Inc. The command line for executing H7TECPLOT on the UNICOS CRAY or UNIX workstations at Martin Marietta Energy Systems is

```
h7tec
```

The user is prompted for the name of an existing HEATING plot data file, the name for a TECPLOT ASCII input file to be created, and the selection of information to be stored in the TECPLOT ASCII file (temperature-time plots, temperature-distance plots, or contour plots).

D.5 H7MAP

H7MAP reads the plot data file generated by HEATING and produces node number and/or temperature maps similar to those generated during execution by earlier versions of HEATING (those versions prior to 7.0). H7MAP is installed on the UNICOS CRAY at Martin Marietta Energy Systems and is available for most UNIX workstations. The command line for executing H7MAP on the UNICOS CRAY or UNIX workstations at Martin Marietta Energy Systems is

```
h7map
```

The user is prompted for the name of an existing HEATING plot data file, the name for the output file to be created, and the selection of information to be written to the output file (node-number maps and/or nodal-temperature maps and output planes for 3-d problems).

D.6 H7MONITOR

H7MONITOR reads the plot data file generated by HEATING and produces a table of temperature vs time for selected nodes. This is a postprocessor version of the type of output that can be obtained during execution by using the NODES MONITORED data block although the output is formatted differently. H7MONITOR can only obtain the nodal temperatures at printout times stored in the plot file, whereas use of the NODES MONITORED data block allows more frequent monitoring. H7MONITOR is installed on the UNICOS CRAY at Martin Marietta Energy Systems and is available for most UNIX workstations. The command line for executing H7MONITOR on the UNICOS CRAY or UNIX workstations at Martin Marietta Energy Systems is

```
h7monitor
```

The user is prompted for the name of an existing HEATING plot data file, the name for the output file to be created, and the nodes whose temperatures are to be written to the output file.

D.7 USER-CREATED TABULAR OUTPUT

Sufficient information is found in the plot data file generated by HEATING to produce various other types of tabular output. Users can, of course, produce their own customized tabular output. Sect. 5.3 supplies information about the HEATING plot file for this purpose.

REFERENCES

1. C. K. Cobb and M. W. Wendel, "HEATPLOT-S: A Temperature Distribution Plotting Program for HEATING5/HEATING6," as described in Sect. F13 of *SCALE: A Modular Code System for Performing Standardized Computer Analyses for Licensing Evaluation*, NUREG/CR-0200 (Revision 4), Vol. 3, Martin Marietta Energy Systems, Inc., to be published.

2. *PATRAN Plus User's Manual (Release 2.4)*, PDA Engineering Software Products Div., Santa Ana, Calif., September 1989.

3. *Users Manual (Version 4.0), TECPLOT - Interactive Plotting for Scientists and Engineers*, Amtec Engineering, Inc., Bellevue, Wash., June 1990.

4. *CA-DISSPLA User's Manual*, Computer Associates International, Inc., Garden City, N.Y., 1988.

Appendix E. AUXILIARY PROGRAMS

Over a number of years special needs have arisen in developing HEATING models. These needs have often led to the development of auxiliary computer programs for use with HEATING. A few of these programs are identified in this appendix. The capabilities of these programs are summarized here, but sufficient information to make use of them is not included because they are not developed to the point that they can be released for general use. If there is a need to use any of them, the author of this document should be contacted.

E.1 QTOTAL

QTOTAL is actually a user-supplied subroutine that calculates and prints additional information on net energy flow into (or out of) the model during a transient calculation. HEATING, by default, calculates and prints the instantaneous rate of energy input to (or removal from) the model due to each boundary condition and heat generation function at normal print times. QTOTAL integrates these energy rates over time to give the net energy change for the model because of each boundary condition and heat generation function. It also calculates the changes in sensible heat and latent heat for the transient. This information is printed out at normal print times. A sample of this output is given in Fig. E.1. Two columns of numbers are printed — the first number is the energy actually used in energy balances in HEATING; the second number is energy not used in the energy balances. For boundary conditions, neglected energy only occurs when a node has a specified temperature (type 2) boundary condition on part of its surface and a type 1 boundary condition on another part of its surface. No heat flow is modeled from the type 1 boundary condition to the specified surface temperature node. Any energy resulting from a generation in the nodal volume associated with a specified temperature node is neglected. If a specified surface temperature boundary is time-dependent, the change in sensible heat for the nodes on the surface is neglected. In the example shown in Fig. E.1, the boundary conditions with zero net energy flow are surface-to-surface (type 3). This problem does not involve a phase change so information on latent heat is not printed. Net heat flow for surface-to-surface boundary conditions will always be zero.

E.2 H7NODE

H7NODE is an interactive program that gives the node number at any nodal location in the model. HEATING must be executed with the model of interest before running H7NODE since H7NODE makes use of the HEATING-produced plot data file. The user is first prompted for the name of the plot data file and then for the i, j, and k coordinate surface numbers passing through the node of interest. The code responds with the coordinate values and the node number of the node at this location.

```
        ***** net heat generation for transient *****

   number          modeled heat generation      neglected heat generation
                        (energy units)               (energy units)
      1                 -8.35406E+04                  0.00000E+00
    ---                 -----------                  -----------
    sum                 -8.35406E+04                  0.00000E+00




        ***** net boundary heat flow for transient *****

   number             modeled heat flow           neglected heat flow
                        (energy units)               (energy units)
      1                 -1.13633E+04                  2.42982E+04
      2                  0.00000E+00                  0.00000E+00
      3                 -4.30724E+03                  0.00000E+00
      4                  0.00000E+00                  0.00000E+00
      5                  0.00000E+00                  0.00000E+00
      6                  0.00000E+00                  0.00000E+00
      7                  8.93570E+04                  0.00000E+00
      8                  7.19095E+04                  0.00000E+00
      9                  3.59833E+04                  0.00000E+00
    ---                 -----------                  -----------
    sum                  1.81579E+05                  2.42982E+04




        ***** net change in sensible heat for transient *****

                       calculated temp. nodes       specified temp. nodes
                        (energy units)               (energy units)
                         9.80317E+04                  -8.60243E+02
```

Fig. E.1. Sample output from QTOTAL.

E.3 H7LINE

H7LINE produces a table of temperature vs distance along a line in a model. The line can be made up of a series of line segments, where each line segment lies along a coordinate axis. Each line segment is defined by entering the node number at each end of the line. In addition to the temperature and distance, the node numbers for all nodes lying on the line are given. H7LINE is currently implemented for steady-state problems only. H7LINE uses the plot data file produced by HEATING.

E.4 H7CONNECT

Very thin, highly conductive components can cause numerical difficulties in HEATING. It is sometimes possible to avoid these difficulties by modeling these components with node-to-node connectors rather than defining a region. H7CONNECT is an interactive program that generates a node-to-node connector file that models a thin, highly conductive layer in a two-dimensional, x-y geometry or a small-cross-section, highly conductive rod (wire) in a three-dimensional, x-y-z geometry. H7CONNECT makes use of the plot data file produced by a preliminary execution of HEATING without the highly conductive layer (or rod) modeled. The user is first prompted for the names of the HEATING plot file and the node-to-node connector file and then for the definition of the highly conductive component to be modeled. To define the connectors to be generated, the user enters (in response to prompts), the boundary condition to be used with the connectors, the nodes defining the two ends of the component, and the thickness (or cross-sectional area for a three-dimensional problems) of the component. H7CONNECT generates a node-to-node connector file that contains the geometric portion (A/l) of the conductance between nodes on the component. The conductivity of the layer is supplied as the first entry on the B2 card of the appropriate boundary condition definition.

E.5 H7PLATE

Very thin, highly conductive components can cause numerical difficulties in HEATING. To avoid these difficulties, it is sometimes possible to model these components with node-to-node connectors rather than defining a region. h7PLATE is an interactive program that generates a node-to-node connector file that models a thin, highly conductive layer in a three-dimensional, x-y-z geometry. H7PLATE makes use of the plot data file produced by a preliminary execution of HEATING without the highly conductive layer modeled. The user is first prompted for the names of the HEATING plot file and the node-to-node connector file and then for the definition of the highly conductive component to be modeled. To define the connectors to be generated, the user enters, (in response to prompts), the boundary condition to be used with the connectors, the nodes defining two diagonally opposed corners of the component, and the thickness of the component. H7PLATE generates a node-to-node connector file that contains the geometric portion (A/l) of the conductance between nodes on the component. The conductivity of the layer is supplied as the first entry on the B2 card of the appropriate boundary condition definition.

E.6 H7CONVERT

H7CONVERT is an interactive code that reads an unformatted HEATING plot data file and converts it to an equivalent formatted file, or vice versa. The user is first asked if the original file is formatted or unformatted and then is prompted for the names of the original plot file and the new file to be created. H7CONVERT allows the user to transfer HEATING plot data files between computers.

Appendix F. CONPLOT USER GUIDE

CONPLOT is a general-purpose contour plotting program written by Deryl A. Steinert of Martin Marietta Energy Systems, Computing and Telecommunications Division. Since there is not any published documentation for this program, the following set of user notes written by Mr. Steinert is supplied.

F.1 INTRODUCTION

CONPLOT is a general-purpose graphics program for generating contour plots of a user's mesh. User meshes can be any two- or three-dimensional grid system that can be defined by four-noded surfaces. Parameters to be contoured must be calculated at or mapped to the nodes. CONPLOT can produce mesh plots, deformed mesh plots, b/w and color contour plots, and color fringe plots of two- and three-dimensional meshes.

CONPLOT is written in FORTRAN-77 for execution on the STC10 VAX/VMS and UNICOS CRAY computer systems. The CA-DISSPLA graphics library is used to create plots on Tektronix terminals and emulators, and to create graphics metafiles. CONPLOT can also generate Hierarchical Data Files (HDF) for processing on a Macintosh using the National Center for Supercomputing Applications (NCSA) Image software.

This appendix describes the input structure and commands to CONPLOT.

F.2 EXECUTION OF CONPLOT

CONPLOT is currently available on the STC10 VAX/VMS and the UNICOS CRAY.

To execute CONPLOT on the STC10 VAX/VMS system type:

    RUN ALL:[DUS.PROGRAMS]CONPLOT1P0.EXE

To execute CONPLOT on the UNICOS CRAY system type:

    conplot

When the above execution lines are typed the following title screen will appear:

```
                    WELCOME TO
                    CONPLOT 1.0
                    ============

                 A COLOR  CONTOUR
                 PLOTTING PACKAGE

                   WRITTEN   BY
                   D.A. STEINERT

                 COMPILATION  DATE
                  MARCH 13,  1990




   ENTER NAMES FOR INPUT AND OUTPUT SESSION FILES

   INPUT < <CR>  FOR  TTY  INPUT)    >>
   OUTPUT   (DEFAULT=CONPLT.SES  )   >>
```
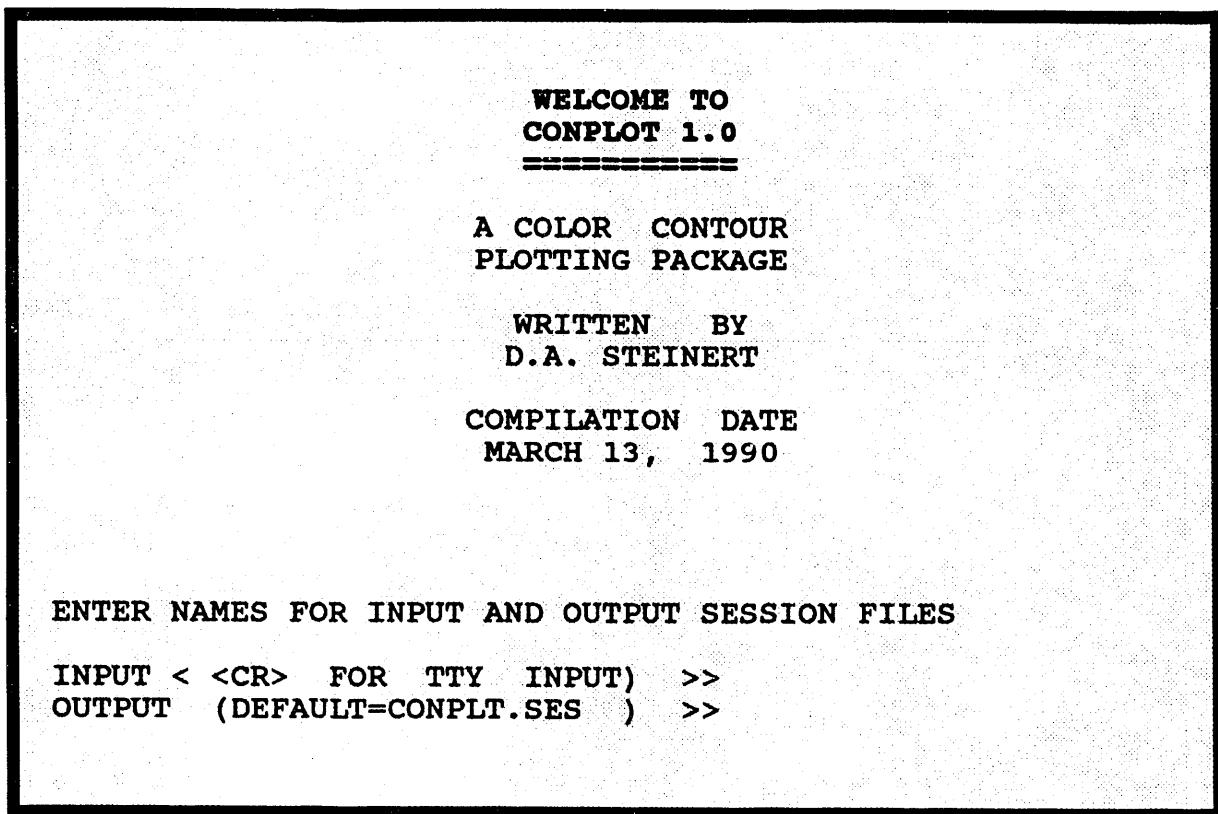
CONPLOT generates a session file which contains all of the commands entered to CONPLOT during an execution. By default, this file will be named *conplot.ses*. This session file can be edited and used as input to subsequent executions of CONPLOT. If an input session file is not supplied then CONPLOT will enter interactive menu mode otherwise, it will enter batch command mode and process all commands in the input session file.

A command to CONPLOT may be entered as a two-character **Menu Item** or as a **Full**

Command. **Menu Items** are only available from the appropriate menu screen. **Full Commands** may be entered from any menu screen, and only the first four characters are necessary to make the command unique. Not all **Menu Items** have an equivalent **Full Command.** In these cases, the **Menu Item** must be used, and it must be issued from the appropriate menu screen.

There are three types of commands in CONPLOT: parameter setting, toggles, and special. Parameter setting commands are those which enable the user to input the parameters that describe the plot to be generated (e.g., file names, scale factors, ranges). The user is prompted for the value or values of the parameters being set. Toggles are used to turn certain options on and off in CONPLOT (e.g., hidden line, color, shading). When using the **Full Command** for a toggle, the **Full Command** preceded by "no" will turn the option off and the **Full Command** alone will turn the option on. Special commands are those that allow the user to move from one menu screen to another, the help facility, and the commands for terminating CONPLOT.

F.3 FILE SET-UP MENU

When CONPLOT is in the interactive menu mode, meaning that a session file was not input, the following **File Set-Up Menu** will appear;

```
CONPLOT FILE SET-UP MENU

        COMMAND FILE          :  TTY INPUT
        OUTPUT COMMAND FILE   :  CONPLT.SES
GF      GEOMETRY FILE         :  ** NONE **
CF      COMPONENT FILES       :
                              :  ** NONE **
SF      SCALAR FILES          :   0

PM      PLOT MENU
HP      HELP
EX      EXIT CONPLOT




ENTER RESPONSE >>
```
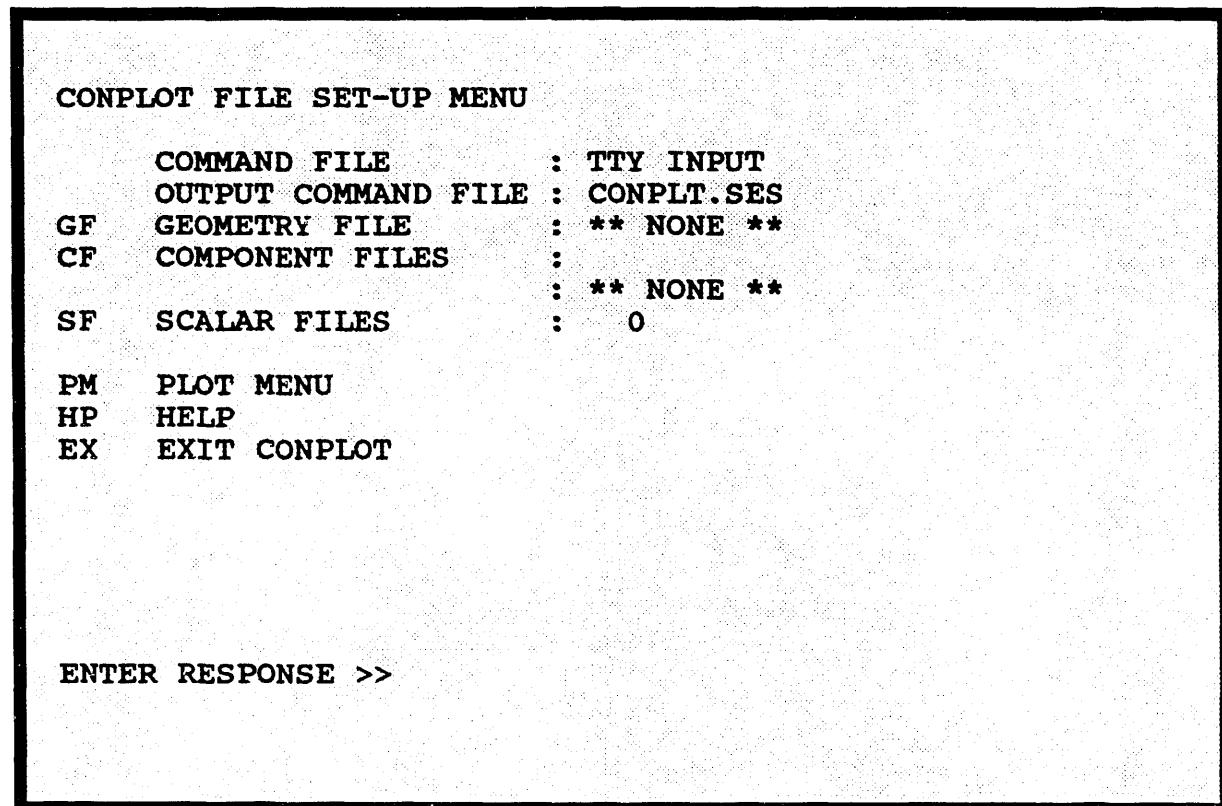
This menu is used to display and input the various files to be used by CONPLOT. The user must type one of the valid **Menu Items** for this menu. Note that the *geometry* file is required.

Menu Item	Full Command	Description
GF		The *geometry* file contains the description of the mesh to be plotted. This description consists of the nodal coordinate locations and surface connectivity information along with other information about each surface. To input the name of the *geometry* file, enter the command **GF** at the prompt. You will then be prompted for the name of the geometry file. **This file is required.**
CF		A *component* file contains the nodal response values for the model. These values can be temperature, stress, strain, displacement, or any other parameter that can be calculated at the nodes. Each *component*

file can contain more that 1 parameter and multiple time steps. CONPLOT can handle up to 5 *component* files at a time. To input the name of the *component* file(s), enter the command CF at the prompt. You will then be prompted for the names of each component file. To terminate *component* file name input, type a carriage return. The component file is not required if you are doing only mesh plots.

SF
The *scaler* file option is currently unavailable.

PM
The **PM** command takes the user to the **Plot Menu** screen. See the **Plot Menu** section for more details.

HP   HELP
The **HP** command gives a listing of all of the commands that CONPLOT knows about. See the **Help** section for more details.

EX
The **EX** command will cause CONPLOT to terminate.

NOTE: Any of the **Full Commands** may be entered at the prompt. See the **Full Commands** section for a listing of the commands.

For a detailed description of the *geometry* and *component* files, see the **File Formats** section.

F.4 PLOT MENU

The **Plot Menu** is entered by issuing the **PM** command from the **File Set-Up** Menu. This **Plot Menu** is used to move to other menus as well as to set particular options for CONPLOT.

```
PLOT MENU:

CC    CONTOUR COMPONENT          :  0 = NO CONTOUR
DF    DISPLACEMENT FACTOR        :  0.000
HL    HIDDEN LINE                :  OFF
MC    MIN. & MAX. FOR COMPONENT  :  ( 0.00000E+00,0.00000E+00)
NN    NODE NUMBERING             :  OFF

NC    NUMBER OF CONTOUR LEVELS   :   6              MUST BE <= 18
RX    ROTATE    X-AXIS
RY    ROTATE    Y-AXIS
RZ    ROTATE    Z-AXIS
TM    TIME                       :  0.00000E+00  (0.00000E+00->
                                                  0.87600E+04)

PL    GENERATE    PLOT
EX    RETURN

      ADDITIONAL MENUS

HO    HARDWARE OPTIONS MENU         SC    SCREEN MENU
MM    MOVIE MENU                    SO    SPECIAL OPTIONS MENU


ENTER RESPONSE >>
```

Menu Item	Full Command	Description
CC	COMPonent	The CC or COMPonent command is used to choose a particular nodal results parameter that is to be contoured on the mesh. Components are stored in *component* files and will vary from one *component* file to another. An input of <=0 or > the maximum number of components will turn the contour or fringe plot off. The user will be prompted for the component number to be plotted.
DF	DISPlacement	The DF or DISPlacement command is used to enter the factor to be used in scaling the x, y, and z displacements. The displacement component names must be X-DISP, Y-DISP, and Z-DISP in the *component* files. If these names are not found in any of the current *component* files the user will be prompted for a new *component* file.

Each node in the geometry will be displaced by the factor times the displacement at the node in each direction. A displacement factor of 0.0 will turn off displacements.

HL   **(no)HIDE**   The **HL** or **HIDE** command is a toggle switch that determines whether or not the hidden lines are to be drawn on the plot. Default is OFF.

MC   **CRANge**   The **MC** or **CRANge** command determines the minimum and maximum range to be contoured for the component. The minimum can be less than the default minimum, and the maximum can be greater than the default maximum. If the minimum and maximum are equal, the defaults are used. The user is prompted for the minimum first and then the maximum.

The defaults are set to the minimum and maximum for the component being plotted.

NN   **(no)NODE**   The **NN** or **NODE** command is a toggle switch that determines whether or not the node numbers will be displayed on the plot. When **NODE** is turned **ON**, the user is asked whether actual node numbers or CONPLOT node numbers are desired. Actual node numbers are those from the original mesh. Some codes that generate input to CONPLOT reduce the size of the input files by removing unused nodes. An array of actual node numbers is included in the *geometry* file.

The default for **NODE** is **OFF**.

NC   **NCONT**   The **NC** or **NCONT** command allows the user to change the number of contour or fringe levels of the plot.

The default for **NCONT** is 6.

RX   **ROTAte**   The **RX** command is used to rotate the mesh about the *x*-axis. The user will be prompted for the angle of rotation in degrees. When using the **ROTAte** command the user will be prompted first for the axis of rotation (*x*, *y*, or *z*) then for the angle of rotation in degrees. The angle of rotation can be plus or minus.

RY   **ROTAte**   The **RY** command is used to rotate the mesh about the *y*-axis. The user will be prompted for the angle of rotation in degrees. When using the **ROTAte** command the user will be prompted first for the axis of rotation (*x*, *y*, or *z*) then for the angle of rotation in degrees. The angle of rotation can be plus or minus.

RZ   **ROTAte**   The **RZ** command is used to rotate the mesh about the *z*-axis. The user will be prompted for the angle of rotation in degrees. When using the **ROTAte** command the user will be prompted first for the axis of rotation (*x*, *y*, or *z*) then for the angle of rotation in degrees. The angle of rotation can be plus or minus.

TM   TIME   The **TM** or **TIME** command is used to input the time step to be plotted. Linear interpolation will be performed between *component* file times steps. The user will be prompted for a time value between the minimum and maximum time values in the *component* files.

The default for **TIME** is 0.0.

PL   PLOT   The **PL** or **PLOT** command generates the current plot on the device that has been set. If a device has not been initialized, the user will then be asked to initialize one.

EX   The **EX** command returns CONPLOT to the **File Set-Up Menu.**

NOTE: Any of the **Full Commands** may be entered at the prompt. See **Full Commands** section for a listing of the commands.

F.5 HARDWARE OPTIONS MENU

The **Hardware Options Menu** displays and allows the user to change particular parameters associated with the plotting device being used.

```
HARDWARE OPTIONS MENU :

CI     COLOR INTENSITY          : 1.0
DV     DEVICE                   : TEKT
HC     HARDCOPY                 : OFF
IP     INPUT                    : TTY
SV     SAVE PLOT                : OFF
SM     SUMMARY
YR     Y-RESOLUTION             :  432
PL     GENERATE PLOT
EX     RETURN TO PLOT MENU



ENTER RESPONSE >>
```

Menu Item	Full Command	Description
CI	CINTens	The **CI** or **CINTens** command allows the user to change the color intensity or the color value in the HSV360 color call. The range of valid intensities is 0.0 (black) to 1.0 (full color intensity). Values between 0.0 and 1.0 give the colors a grayish look.
		The default for **CINTens** is 1.0 or full intensity.
DV	DEVIce	The **DV** or **DEVIce** command sets or changes the plotting device. The choices are TEKTRONIX and SAVE.

HC   (no)HARDcopy The **HC** or **HARDcopy** command is used to automatically produce hardcopies after CONPLOT produces a plot on the screen. For this option to work, a hardcopy unit must be connected to the terminal and it must support the TEKTRONIX hardcopy escape sequence. This works with a Macintosh running VersaTerm-Pro.

IP or TTY The **IP** or **TTY** command is used to switch between reading commands from the user's terminal and a file. This allows the user to store a long sequence of commands in a CONPLOT *session* file, begin execution of CONPLOT in batch command mode, and gain interactive control of CONPLOT when the **IP** or **TTY** command is input from the *session* file. This also allows the user to switch from interactive command mode to batch command mode by entering the name of the file containing the commands to process. All files to be used as *command* files must end with an **IP** or **TTY** command to switch to interactive control of CONPLOT or with one of the termination commands, **END, EXIT, QUIT,** or **STOP** to terminate CONPLOT.

SV   (no)SAVE The **SV** or **SAVE** command is a toggle switch that determines whether or not the plots are to be saved. When **SAVE** is turned **ON**, the user will be asked if the plots are to be saved as DISSPLA *metafile* files or .HDF files. The default is to save plots in a DISSPLA metafile. If the files are to be HDF files, the user will be asked to enter a prefix for the names of the files. This can be turned **ON/OFF** as needed.

SM   SUMMary The **SUMMary** command is currently not activated in this version of CONPLOT.

YR   YRES The **YR** or **YRES** command allows the user to determine the y resolution to be used in shading for the terminal. The user should not have to change this parameter. CONPLOT tries to set the correct resolution for the supported devices.

PL   PLOT The **PL** or **PLOT** command generates the current plot on the device that has been set. If a device has not been initialized, the user will then be asked to initialize one.

EX The **EX** command returns CONPLOT to the **File Set-Up Menu.**

NOTE: Any of the **Full Commands** may be entered at the prompt. See **Full Commands** section for a listing of the commands.

F.6 SCREEN MENU

The **Screen Menu** displays and allows the user to change particular parameters associated with the plot screen. These parameters include: page size, black/white or color, plot labels, etc.

```
SCREEN MENU :

BS   BACKGROUND SHADE      : OFF H - 0.00   S - 0.00   V - 1.00
BL   BOLD TEXT             : OFF
BR   BORDER
CN   CENTER                : OFF
CK   CLOCK                 : OFF
CL   COLOR                 : OFF
FR   FRAME                 : ON
HS   HARDWARE SHADING      : OFF
LG   LEGEND                : ON
PG   PAGE                  : XPAGE -> 11.000   YPAGE ->  8.500
LB   PLOT LABELS           : OFF
SH   SHADING               : OFF
TI   TITLE -> *** NONE ***
XR   X PLOT RANGE          : MIN -> 0.000    MAX ->  78.300
YR   Y PLOT RANGE          : MIN -> 0.000    MAX ->  35.000
ZR   Z PLOT RANGE          : MIN -> 0.000    MAX ->   0.000
PL   GENERATE PLOT
EX   RETURN TO PLOT MENU

ENTER RESPONSE >>
```

Menu Item	Full Command	Description
BS	(no)BSHAde	The **BS** or **BSHAde** command is used as a toggle switch to turn **ON** and **OFF** the background shading. If background shading is turned **ON**, the user will be prompted for the hue, saturation, and value for the background color. The area within the plot window that is not covered by the mesh will be shaded in the specified color. If background shading is turned **OFF**, no additional input is required.

The default for **BSHAde** is **OFF**.

BL   **(no)BOLD**   The **BL** or **BOLD** command is a toggle switch to determine whether the text on a plot is to be in bold letters or plain letters. When **BOLD** is turned **ON**, the user will be asked if the text is also to be shaded. If text is not shaded, then just the outlines of each character are plotted.

The default for **BOLD** is **OFF**.

BR   **BORDer**   The **BR** or **BORDer** command chooses the border type to be put around the plot frame. The types of borders are:

0 = No border
1 = Unclassified cyan border
2 = Confidential green/white candy stripe
3 = Secret red/white candy stripe

The default for **BORDer** is 0 (no border).

CN   **(no)CENTer**   The **CN** or **CENTer** command is a toggle switch for centering the mesh on the screen. If **CENTER** is **ON** then all subsequent plots will have the visible portion of the mesh centered. This also defines the center of rotation (xc, yc, zc) to be the average of the visible x, y, and z coordinates.

The default is for the mesh to be centered around the middle of the complete mesh.

CK   **(no)CLOCk**   The **CK** or **CLOCk** command is a toggle switch that turns **ON** and **OFF** the drawing of the clock in the upper right hand corner of the frame. If **CLOCk** is **ON**, then the clock will show the time of the current frame. The user will be prompted for two inputs. The first will be the number of time **UNITS** per revolution of the clock hand. The second input will be the clock **LABEL**.

Example: If time was in units of seconds, you might want 1 hand revolution on the clock to be 1 minute. Therefore, **UNITS** would be 60 (60 seconds in a minute), and **LABEL** would be **MINUTES**.

The default for **CLOCk** is **OFF**.

CL   **(no)COLOr**   The **CL** or **COLOr** command is a toggle switch from black/white to color contours.

The default is to generate black/white contours.

FR   **(no)FRAMe**   The **FR** or **FRAMe** command is a toggle switch that determines whether or not the frame is to be drawn on the plot. It is initially set to **ON**.

HS (no)HWSHd The **HS** or **HWSHd** command is a toggle switch that determines whether or not hardware shading is to be used for the plot. It is initially set to **OFF**.

LG (no)LEGEnd The **LG** or **LEGEnd** command is a toggle switch that determines whether or not the legend is to be printed on the plot. When **LEGEnd** is turned **ON**, the user will be asked to enter a legend heading. The **LEGEnd** switch is initially set to **ON**, with the heading taken from the component to be plotted.

PG PAGE The **PG** or **PAGE** command allows the user to change the size of the plot page. Both the x and y variables can be changed. Initially, x is set to 11.0, and y is set to 8.5.

LB (no)LABEls The **LB** or **LABEls** command is a toggle switch that determines whether or not the labels will be on the plot. Labels include title, displacement factor, and time.

The default for **LABEls** is **OFF**.

SH (no)SHADe The **SH** or **SHADe** command is a toggle switch that determines whether or not shading is to be used on the contour plot.

The default for **SHADe** is **OFF**.

TI TITLe The **TI** or **TITLe** command allows the user to change the plot title. The plot title is written across the top of the plot frame. The default plot title is set to the title in the *geometry* file. The **LABEls** command must be turned **ON** before the title will appear on the plot. The user will be prompted for a new plot title or allowed to choose the default.

XR XRANge The **XR** or **XRANge** command allows the user to specify a range of x values to plot. All surfaces within the viewing box created by **XRANge**, **YRANge**, and **ZRANge** are plotted. The surfaces must lie completely within the box.

YR YRANge The **YR** or **YRANge** command allows the user to specify a range of y values to plot. All surfaces within the viewing box created by **XRANge**, **YRANge**, and **ZRANge** are plotted. All surfaces must lie completely within the box.

ZR ZRANge The **ZR** or **ZRANge** command allows the user to specify a range of z values to plot. All surfaces within the viewing box created by **XRANge**, **YRANge**, and **ZRANge** are plotted. The surfaces must lie completely within the box.
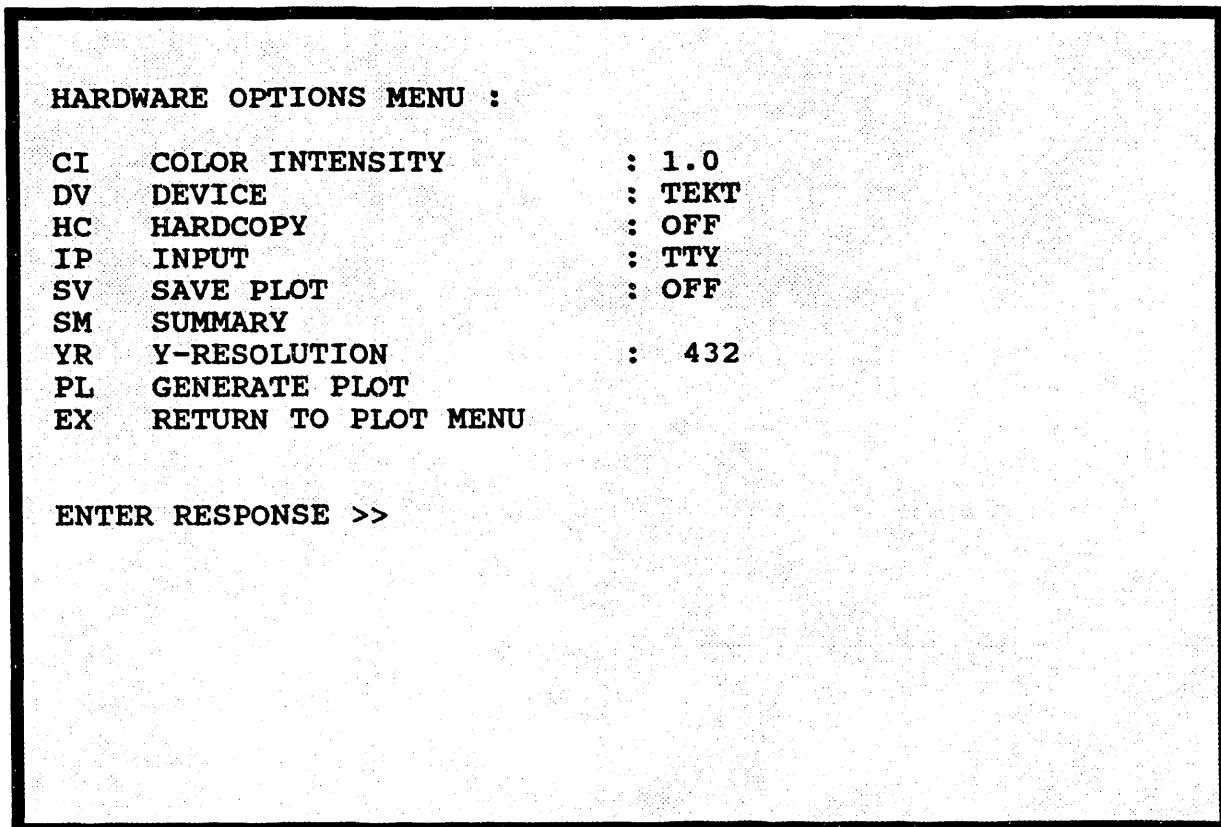
PL   **PLOT**   The **PL** or **PLOT** command generates the current plot on the device that has been set. If a device has not been initialized, the user will then be asked to initialize one.

EX   The **EX** command returns CONPLOT to the **Plot Menu.**

  NOTE: Any of the **Full Commands** may be entered at the prompt. See **Full Commancs** section for a listing of the commands.

F.7 MOVIE MENU

CONPLOT can perform linear interpolation between time steps to create movie or animation sequences. The **Movie Menu** is used to control the number of frames to generate, the number of times to repeat each frame, final rotations, and the time interval to animate.

The **Movie Menu** is **entered** from the **Plot Menu** by issuing the **MM** command or from anywhere in CONPLOT by issuing the **MOVIe** command.

```
MOVIE MENU

DV    DEVICE                : TEKT
MF    NUMBER OF FRAMES      :    1
RF    REPEAT FACTOR         :    1
RX    ROTATE X-AXIS
RY    ROTATE Y-AXIS
RZ    ROTATE Z-AXIS
ST    STARTING TIME         : 0.00000E+00   (MIN= 0.00000E+00)
ET    ENDING TIME           : 3.00000E+00   (MAX= 3.00000E+00)
PL    GENERATE MOVIE
EX    RETURN


ENTER RESPONSE >>
```

Menu Item	Full Command	Description
DV	DEVIce	The **DV** or **DEVIce** command sets or changes the plotting device. The choices of devices are TEKTRONIX and SAVE.
NF		The **NF** command sets the number of plot frames to be generated between the **START** and **END** times of the animation. CONPLOT will linearly interpolate between time steps. The more frames generated, the smoother the animation will be.

RF   The **RF** command sets the number of times each frame is to be duplicated.

RX   The **RX** is used to specify the rotation of the mesh about the x-axis. The user will be prompted for the angle of rotation in degrees.

RY   The **RX** is used to specify the rotation of the mesh about the y-axis. The user will be prompted for the angle of rotation in degrees.

RZ   The **RZ** is used to specify the rotation of the mesh about the z-axis. The user will be prompted for the angle of rotation in degrees.

ST   The **ST** command is used to set the time that the animation sequence will begin. The default is set to 0.0.

ET   The **ET** command is used to set the time that the animation sequence will end. The default is set to the maximum time step in all of the *component* files.

PL   PLOT   The **PL** or **PLOT** command generates the current plot on the device that has been set. If a device has not been initialized, the user will then be asked to initialize one.

EX   The **EX** command returns CONPLOT to the **Plot Menu.**

NOTE: Any of the **Full Commands** may be entered at the prompt. See **Full Commands** section for a listing of the commands.

F.8 SPECIAL OPTIONS MENU

```
SPECIAL OPTIONS MENU :

ZF   ZOOM FACTOR         AX   AXIS                     : ON
EL   ELEMENTS            BK   BUCKETS                  : SMALL
MT   MATERIAL            CM   COLOR MIX                : CONT.
MR   MIRROR              CN   CONSOLIDATE RASTERS : OFF
OL   OUTLINE             IN   INTERIOR                 : OFF
PT   PART                NV   NODE VALUES              : OFF
PN   PLANE               OV   OVERLAY                  : OFF
RF   RELIEF              PP   PERSPECTIVE              : ON
RS   RESTORE             VT   VECTORS                  : OFF
SH   SHIFT
PL   GENERATE PLOT
EX   RETURN TO PLOT MENU




ENTER RESPONSE >>
```
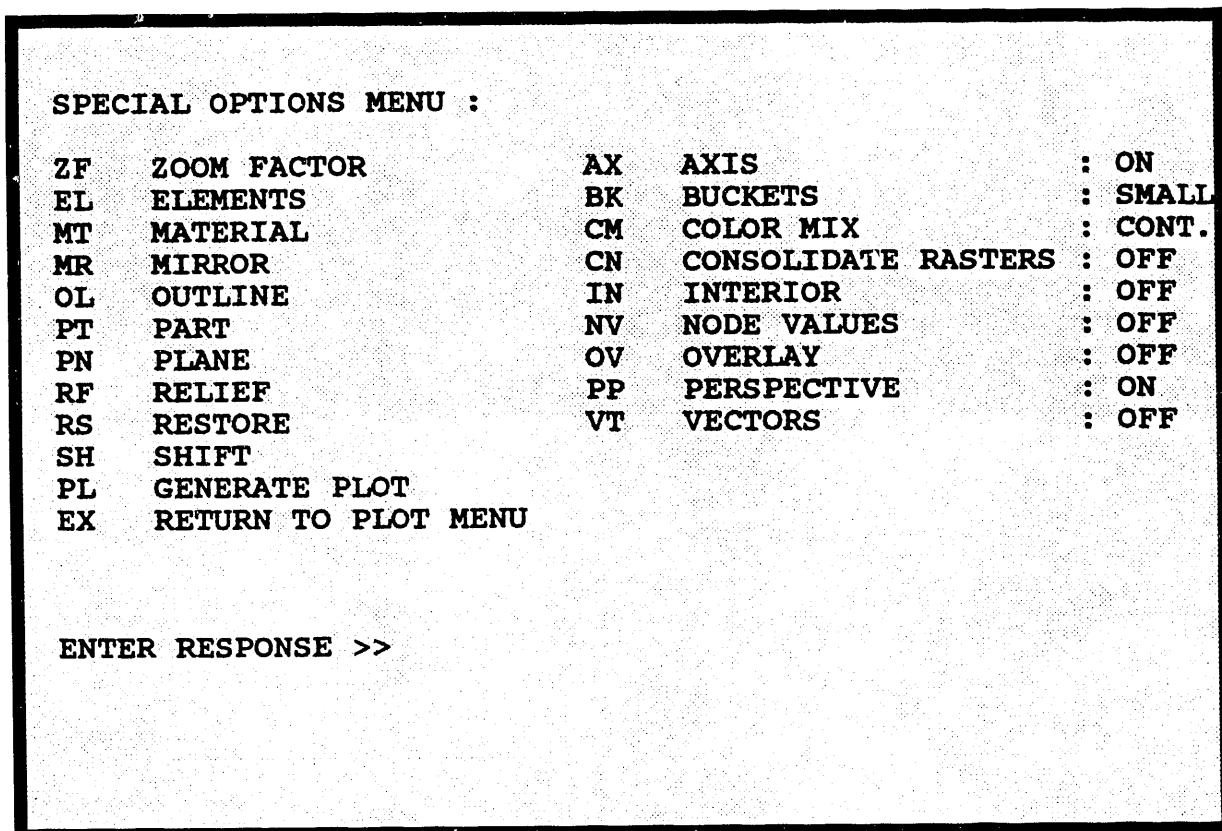
Menu Item	Full Command	Description
ZF	ZOOM	The ZF or ZOOM command is used to move the viewpoint closer to or further away from the center of the mesh. This has the effect of zooming in to or out from the mesh.
EL	ELEMents	The EL or ELEMents command allows the user to determine which elements are to be included in the plot. It is initially set to include all elements. An input of <0 will cause all elements to be visible. An input of 0 ends element input, allowing the program to continue.
MT	MATErial	The MT or MATErial command allows the user to choose specific material ranges to plot. Only elements that have those material numbers will be plotted.
		The default for MATErial is for all materials to be plotted.

MR **MIRRor**   The **MR** or **MIRRor** command allows the user to mirror the geometry about the x-, y-, or z- axis. This option can be used in conjunction with the **OVERlay** command. This option is useful for models which are symmetric about the x-, y-, or z- axis and only one half of the model has been analyzed.

The default for **MIRRor** is **OFF**.

OL **OUTLine**   The **OL** or **OUTLine** command allows the user to choose the type of mesh outline and the type of line to be drawn for the outlines.

The outline types include:

0 = No outlines
1 = Element outlines
2 = Part outlines
3 = Model outlines

The line drawing types include:

0 = Solid
1 = Dashed
2 = Dotted

The default for **OUTLine** is solid element outlines.

PT **PART**   The **PT** or **PART** command allows the user to determine which parts or surface groups of the mesh are to be visible in the plot. Parts are defined in the *geometry* file by grouping consecutively numbered surfaces together. An input of <0 will cause all parts to be visible. An input of 0 ends part input, allowing the program to continue.

The default for the **PART** command is for all parts to be plotted.

PN **PLANE**   The **PN** or **PLANE** command allows the user to choose constant x, y, and z planes to be plotted from the mesh. In order for a surface to be considered in a plane, each node of the surface must have the same x-, y-, or z-coordinate value. **CONPLOT** will display a list of coordinate planes and the number of surfaces in each plane. All surfaces that do not lie in a constant x, y, or z plane are grouped together under **MISCELLANEOUS**. In order to plot a particular plane, you must first turn off all surfaces (including **MISCELLANEOUS**) and then turn on the planes to be plotted.

RF	**RELIef**	The **RF** or **RELIef** plot option creates a carpet plot. This plot uses a component parameter that has been normalized as the z-coordinate value. This creates a plot where the displacement of the mesh out of the x-y coordinate plane is proportional to the component value at the nodes. This can be used with contouring and fringe plots of the same or different component parameters. This option is only valid for 2-d models.
RS	**RESTore**	The **RS** or **RESTore** command resets the **ROTAtion ZOOM**, and **SHIFt** options back to their respective default values.
SH	**SHIFt**	The **SH** or **SHIFt** command allows the user to shift the plot window up/down and left/right.
AX	**(no)AXIS**	The **AX** or **AXIS** command is a toggle switch that determines whether or not a 3-D axis is drawn in the lower left corner of the frame. It is initially set to **ON**.
BK	**BUCKets**	The **BK** or **BUCKets** command is a toggle switch that determines whether the bucket size is to be LARGE or SMALL. Buckets are used in hidden line removal and should be set to LARGE for models with many surfaces in a small region.
		The **BUCKets** switch is initially set to SMALL and should be adequate for most models.
CM	**(no)CMIX**	The **CM** or **CMIX** command is a toggle switch for causing the colors used in the contour and fringe plots to be MIXED (non-continuous) from magenta to blue.
		The default is for CONTINUOUS color from magenta to blue.
CN	**(no)CONSolidate**	The **CONSolidate** command is a toggle switch that determines whether or not the rasters are to be consolidated by color. This option minimizes the number of color changes and is useful for pen plotters. The **CONSolidate** option is set to **ON** when the **HWSHD** option is set to **ON**. The **CONSolidate** switch is initially set to **OFF**.
		The **CONSolidate** option can reduce metafile size and plotting time considerably.
IN	**(no)INTErior**	The **IN** or **INTErior** command is a toggle switch that turns on the visibility of all vectors interior to the model. This option may need to be used if portions of the model are displayed using the **ELEMents, PART,** or **MATErial** commands.
		The default for **INTErior** is for all interior vectors to be turned **OFF**.

NV (no)**NDVAlue** The **NV** or **NDVAlue** command is a toggle switch that determines whether or not the component values of each node are to be plotted at the visible nodes.

The default for **NDVAlue** is **OFF.**

OV (no)**OVERlay** The **OV** or **OVERlay** command is a toggle switch that determines whether or not to overlay the original mesh over a displacement and/or mirrored mesh.

The default for **OVERlay** is **OFF.**

PP (no)**PERSpect** The **PP** or **PERSpect** command is a toggle switch that turns perspective ON or OFF. Perspective is what gives a 3-dimensional plot its sense of depth; the farther away something is, the smaller it appears.

VT (no)**VECTors** The **VT** or **VECTors** command is a toggle switch that determines whether the vectors are turned ON or OFF. Since CONPLOT uses four-noded surfaces for plotting, it is sometimes necessary to create multiple surfaces to define one element face. This is especially true when midside nodes are retained in the model. When **VECTors** is turned ON, all vectors generated by CONPLOT are turned on. Some vectors are created but not plotted if they do not belong to the original mesh. This is mainly used for a debugging tool and may not have any other practical uses.

The default for **VECTors** is **OFF.**

PL   **PLOT** The **PL** or **PLOT** command generates the current plot on the device that has been set. If a device has not been initialized, the user will then be asked to initialize one.
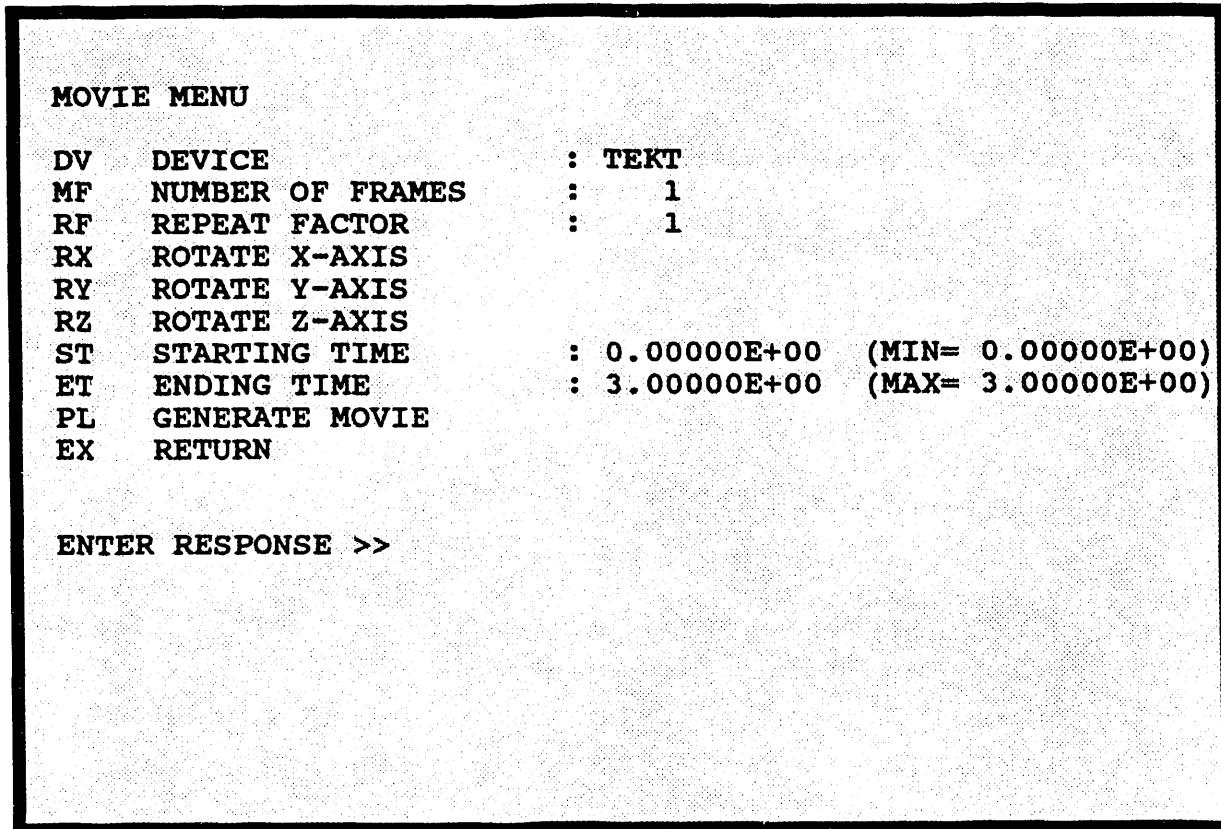
EX The **EX** command returns CONPLOT to the **File Set-Up Menu.**

    NOTE: Any of the **Full Commands** may be entered at the prompt. See **Full Commands** section for a listing of the commands.

F.9 HELP FACILITY

The **Help Facility** in CONPLOT is limited. At present it only gives a listing of the available commands in CONPLOT and can be used as a refresher when a command is forgotten.

```
                         COMMANDS LIST

    BORDER     BUCKETS       CINTENS     COMPONENT    CRANGE
    DEVICE     DISPLACEMENT  ELEMENTS    END          EXIT
    HELP       MATERIAL      MIRROR      MOVIE        NCONT
    NEW        OUTLINE       PAGE        PART         PLANE
    PLOT       QUIT          RELIEF      RESTORE      ROTATE
    SGAP       SHIFT         SPMRK       STOP         SUMMARY
    TIME       TITLE         TTY         XRANGE       YRANGE
    YRES       ZOOM          ZRANGE

                         TOGGLE SWITCHES    (NO)

    AXIS       BOLD          BSHADE      CENTER       CLOCK
    CMIX       COLOR         CONSOLIDATE DEBUG        FRAME
    HARDCOPY   HIDE          HWSHD       INTERIOR     LABELS
    LEGEND     NDVALUE       NODE        OVERLAY      PERSPECTIVE
    PRINT      SAVE          SHADE       VECTORS


    PRESS (CR) TO RETURN >>
```

F.10 FILE FORMATS

This section describes the formats of the *geometry* and *component* files.

Geometry File

The *geometry* file contains a description of the mesh to be plotted. A mesh is expressed in two or three dimensions and is described as a series of four-noded surfaces. Node locations are always given in the Cartesian coordinate system.

Card Type	Variable Name	Format	Description
1			Mesh description card (a80)
	TITLE	a80	TITLE is a name or short description of the mesh. The TITLE will be plotted at the top of the plot frame if the LABELs command is issued and the default title is chosen with the TITLe command.
2			Mesh dimensions card (3i5,15x,2i5)
	NPARTS	i5	The NPARTS variable tells how many surface groups there are. Surfaces can be grouped together as long as they are consecutive in the surface list. There must be at least one surface group and it can contain all the surfaces.
	NPTS	i5	The NPTS variable tells how many nodes there are in the mesh.
	NSURF	i5	The NSURF variable tells how many four-noded surfaces there are in the mesh.
	NDIMS	15x,i5	The NDIMS variable is the number of dimensions in the mesh. NDIMS must be either 2 or 3.
	NELEM	i5	The NELEM variable is the number of elements in the original mesh. This number may be the same as NSURF for 2-D meshes. It will most likely be different for 3-D meshes since a 3-D element (brick) is made up of at least six surfaces.
3			Part or surface group descriptions (16i5) ((i=1,2),j=1,NPARTS)

IGR(i,j)	16i5	**IGR(i,j)** is an array containing the beginning (i=1) and ending (i=2) surface numbers in surface group j.There must be as many cards as needed with 8 surface groups per card.

4   Node locations (6e12.5)((i= 1,**NDIMS**),j=1,**NPTS**)

PGRID(i,j)	6e12.5	**PGRID(i,j)** is an array containing the node coordinate locations. For **NDIMS**=2, **PGRID(1,j)** and **PGRID(2,j)** contain the x and y coordinate locations, respectively, for node j with 3 nodes per card. For **NDIMS**=3, **PGRID(1,j)**, **PGRID(2,j)**, and **PGRID(3,j)** contain the x, y, and z coordinate locations, respectively, for node j with 2 nodes per card.

5   Surface description (8i5)
One surface per card

ISRF	i5	**ISRF** is the surface number.
NEDGE	i5	**NEDGE** is the number of edges in the surface which are to be plotted. Some edges of a surface may have been generated when an eight-noded element face was broken into six four-noded surfaces. Not all of theses edges are from the original mesh and, therefore, may not be desired on the plot. **NEDGE** will be a number from 0 (no edges visible) to 4 (all edges visible). The first **NEDGE** edges are visible. Edges that are not part of the original mesh may still be plotted using the **VECT**ors command.
N1,N2,N3,N4	4i5	**N1, N2, N3,** and **N4** are the node numbers at the corners of the surface. Surface edges are formed between nodes **N1/N2, N2/N3, N3/N4,** and **N4/N1.** The first **NEDGE** edges are visible.
IEL1,IEL2	2i5	**IEL1** and **IEL2** are the elements that surface **ISRF** belongs to. A surface must belong to at least one element and can belong to at most two elements. If a surface belongs to only one element (**IEL1>0, IEL2=0**) then the surface is considered to be on the shell of the mesh. If the surface belongs to two elements (**IEL1>0, IEL2>0**) then the surface is considered to be on the interior of the mesh. Interior surfaces, by default, are not plotted. They may be plotted by using the **INTE**rior command.

6   Original node map numbers(16i5) (i=1,NPTS)

NODMAP(i)   16i5   The **NODMAP** array contains the actual node numbers from the mesh. Most codes that generate input to CONPLOT will try to reduce the size of the *geometry* file by removing unused nodes. For example, midside nodes may be removed from some meshes if they are not critical to the plotting. The **NODMAP** array is used so that the user can plot the actual node numbers on the plot rather than CONPLOT's node numbers when using the **NODE** command.

7   Element material numbers (16i5) (i=1,**NELEM**)

MATELM(i)   16i5   The **MATELM** array contains the material number for each element. This is used by the **MATE**rial command for determining which elements have a particular material type. From this information and the **IEL1** and **IEL2** variables of each surface the material type of a particular surface can be determined.

NOTE: Card types 6 and 7 are not required. If card type 6 is not present, CONPLOT will use its node numbers as the actual node numbers. If card type 6 is not present, then card type 7 must not be present either.

F.11 COMPONENT FILE

A *component* file consists of the number of components in the file, the names of the components, and the component values at each node for each time step. There may be as many as 5 *component* files associated with a particular mesh each with multiple and multiple components. Each *component* file may have a different set of time values. CONPLOT will handle the interpolation between time steps.

Card Type	Variable Name	Format	Description
1			Number of components (i5)
	NCOMP	i5	NCOMP is the number of components in the file.
2			Component names (8a10) (i=1,NCOMP)
	COMPON(i)	8a10	The COMPON array contains the names of the components in the file. These names will be such things as; DISP-X, DISP-Y, DISP-Z, STRESS X-X, etc.

The following cards are read for each time step until the end-of-file is reached.

Card Type	Variable Name	Format	Description
3			Time value (e12.5)
	TIME	e12.5	The TIME variable contains the time step value for which the following component values correspond.

The following card is read for each component (NCOMP).

Card Type	Variable Name	Format	Description
4			Component values (6e12.5) (i=1,NPTS)
	COMVAL(i)	6e12.5	The COMVAL array contains the component value for node i at TIME.

F.12 FULL COMMANDS

The following table lists alphabetically the full name of each CONPLOT command, and, if applicable, its two-character Menu Item and the menu(s) that the Menu Item may be issued from. For a description of each command, refer to the appropriate menu screen and item.

Full Command	Menu Item	Menu	Command Type
AXIS	AX	Special Options	toggle
BUCKets	BK	Special Options	parameter setting
BSHAde	BS	Screen	toggle
BOLD	BL	Screen	toggle
BORDer	BR	Screen	parameter setting
CENTer	CN	Screen	toggle
CINTens	CI	Hardware Options	parameter setting
CLOCk	CK	Screen	toggle
CMIX	CM	Special Options	toggle
COLOr	CL	Screen	toggle
COMPonent	CC	Plot	parameter setting
CONSolidate	CN	Special Options	toggle
CRANge	MC	Plot	parameter setting
DEVIce	DV	Hardware Options Movie	parameter setting
DISPlacement	DF	Plot	parameter setting
ELEMents	EL	Special Options	parameter setting
END			termination
EXIT			termination
FRAMe	FR	Screen	toggle
HARDcopy	HC	Hardware Options	toggle
HELP	HP	File Set-Up	
HIDE	HL	Plot	toggle
HWSHd	HS	Screen	toggle
INTErior	IN	Special Options	toggle
LABEls	LB	Screen	toggle
LEGEnd	LG	Screen	toggle
MATErial	MT	Special Options	parameter setting
MIRRor	MR	Special Options	parameter setting
NCONT	NC	Plot	parameter setting
NDVAlue	NV	Special Options	toggle
NODE	NN	Plot	toggle
OUTLine	OL	Special Options	parameter setting
OVERlay	OV	Special Options	toggle
PAGE	PG	Screen	parameter setting
PART	PT	Special Options	parameter setting
PERSpective	PP	Special Options	toggle
PLANe	PN	Special Options	parameter setting

PLOT	**PL**	Plot	generate plot
		Hardware Options	
		Screen	
		Movie	
QUIT			termination
RELIef	**RF**	Special Options	parameter setting
RESTore	**RS**	Special Options	parameter setting
ROTAte	**RX**	Plot	parameter setting
	RY	Plot	
	RZ	Plot	
SAVE	**SV**	Hardware Options	toggle
SHADe	**SH**	Screen	toggle
SHIFt	**SH**	Special Options	parameter setting
STOP			termination
SUMMary	**SM**	Hardware Options	
TIME	**TM**	Plot	parameter setting
TITLe	**TI**	Screen	parameter setting
VECTors	**VT**	Special Options	toggle
XRANge	**XR**	Screen	parameter setting
YRANge	**YR**	Screen	parameter setting
YRES	**YR**	Hardware Options	parameter setting
ZOOM	**ZF**	Special Options	parameter setting
ZRANge	**ZR**	Screen	parameter setting

Appendix G. MATERIAL PROPERTIES LIBRARY

This appendix contains documentation for the material properties library used by HEATING. It was extracted from the SCALE 4 document[1] and edited to be consistent with the latest version of HEATING. (Versions of HEATING subsequent to 6.1 have not been included as part of the SCALE system of codes.) This information is necessary for the user to identify materials in the library for use with HEATING. The format of the material properties library is also documented so that users can create their own project-specific libraries.

G.1 INTRODUCTION

The availability of computer programs for solving problems in transient and steady-state heat conduction makes it possible to rapidly and accurately solve large numbers of problems involving a wide variety of materials, provided that accurate input data are used. In particular, the thermal properties of the materials, including density, specific heat, and thermal conductivity and the dependence of these properties on temperature, temperatures and latent heats of transition, and the temperature ranges for specific applications, should be known as accurately as possible. Some of these properties are difficult to measure and may vary from sample to sample, depending on the method of fabrication and previous history. Therefore, when used in calculations, the possible inaccuracy or variability of thermal data should be accounted for in interpreting the results. For this reason, Dr. A. L. Edwards of the Lawrence Livermore National Laboratory (LLNL) compiled a library of thermal property data[2] in the late 1960s that is still being widely used in heat transfer calculations.

For many years, this LLNL library of thermal material properties has been used by the HEATING series of codes developed at Oak Ridge National Laboratory. Therefore, this library was selected for inclusion in, and subsequent distribution with, the SCALE computational system. The HTAS1 control module, which accesses HEATING, also uses the material properties library. The HEATING and HTAS1 programs give the user the option of inputting thermal data for a problem on the material input cards or of allowing the program to extract the properties from the LLNL material properties library. The material properties available in the library are density, thermal conductivity, specific heat, transition temperature, and latent heat. The density of the material is typically either the value at or near room temperature or the lowest temperature for which specific heat or conductivity is tabulated, whichever is highest. The thermal conductivity and specific heat of the material are given if they are constant or if the temperature dependence is unknown. When a table of thermal conductivity or specific heat vs temperature is listed in the library, the table will be stored and used as a tabular function by HEATING. The transition temperature is the temperature at which either a phase change or a solid-state transition occurs. The latent heat is the amount of heat absorbed by the material when the temperature is increased past the transition temperature. The material properties library is available in the centimeter-gram second-Calorie-degrees Celsius (cgs-cal-°C) system. Conversion to any other unit system can be controlled through HEATING input data. (See Sect. 4.6.4.)

The information and tables describing the format and contents of the LLNL thermal material properties library are taken directly from ref. 2 with the permission of the author, Dr. A. L. Edwards.

G.2 DESCRIPTION OF THE LIBRARY

Thermal property data for over 1000 materials have been compiled in the material properties library. The compilation includes (1) a numbered list of data sources, primarily those containing extensive data tabulations and bibliographies; (2) a numerical classification system used as a guide in assigning a permanent identification number to each material included in the system; (3) a system of quality indicators used to show the general reliability or accuracy of data included in the compilation; (4) thermal property data sheets for each material, including all tabular and graphical data collected or estimated by A. L. Edwards and converted to a standard system of units; (5) an alphabetical index of materials that includes the identification number assigned to each material, an alphanumeric designator for auxiliary identification, a list of data sources by number, and a quality indicator for each type of data given for the material; and (6) a list of material property data arranged according to the material identification numbers. These material properties data include the material identification number and alphanumeric designator, density, specific heat, thermal conductivity, phase-change or transition temperature, latent heat effect, and tables of specific heat and thermal conductivity vs temperature.

The data compilation described herein was completed about 1969 and, even at that time, no claim of completeness or comprehensiveness was made. However, developers of the HEATING codes never updated the library because it was determined to be sufficient as a general-purpose "backup" library for most standard materials. Analysts should check subsequent tables in this document for the source of the data for a given material to be used. Also, the analyst can disregard the material properties library and input original or updated data into HEATING. Thermal data for new or special materials not in the library should also be obtained by the analyst.

G.3 LIBRARY DATA SOURCES

The sources for most of the data included in the compilation and several other general sources are listed in abbreviated form in Table G.1. The numbers of the first column are used in the alphabetical material index to cross-reference the sources. Asterisks indicate the data sources actually used for the compilation. Classified data sources were omitted from ref. 2 and this report. The source used most extensively for the compilation is that edited by Touloukian (No. 20).

Reference 2 provides an estimate of the thermal properties for a large number of materials for which no reliable data could be found. These estimates have been included whenever the estimates appeared to be as accurate as available data for similar materials.

Many errors were found in the literature and corrected before including the data in the original compilation (ref. 2). Most of these errors were introduced by incorrect conversion from one unit system to another, incorrect labelling or interpretation of axes labels or scales on graphs, and confusion between units of mass or gram-molecular weight used for specific heats.

Table G.1. Data sources

1	*	PERRY J H	CHEMICAL ENGINEERS HANDBOOK 4TH ED	1963
2	*	MCADAMS W H	HEAT TRANSMISSION 3RD ED	1954
3	*		LIQUID METALS HANDBOOK 2ND ED	1952
4	*	MARKS L S	MECHANICAL ENGINEERS HANDBOOK 5TH ED	1951
5	*	FLEMING P	PRIVATE COLLECTION OF PLASTICS DATA	1968
6	*		MATERIALS ENGR MATL SELECTOR ISSUE	1967
7		PFEIFER H	PRIVATE COLLECTION OF MATERIALS DATA	1968
8	*	JAMES E	PROPERTIES OF CHEMICAL EXPLOSIVES UCRL 14592	1965
9			ASRE DATA BOOK	
10			AIRPLANE AIR CONDITIONING ENGR DATA	1952
11			INTERNATIONAL CRITICAL TABLES	1926
12		JOHNSON A I	THERMAL CONDUCTIVITY CHART FOR GASES	1954
13		HILSENRATH J	TABLES OF THERMAL PROPERTIES OF GASES NBS 564	1955
14		TIPTON C R JR	THE REACTOR HANDBOOK 2ND ED VOL 1	1960
15		KUONG J F	THERMAL PROPERTIES OF LIQUIDS	1963
16			THE REACTOR HANDBOOK VOL 3 ENGINEERING	1963
17			ENGLISH TRANSL OF HANDBUCH DER PHYSIC	1962
18		HOYT S L	METAL DATA	1952
19		AVDUYEVSKIY V S	FUNDLS OF HEAT TRANSFER IN AV AND RKT ENGR	1962
20	*	TOULOUKIAN Y S	THERMOPHYSICAL PROP OF HIGH TEMP SOLID MATLS	1967
21			PRELIM REPT ON THE PROP OF LI BE MG AL NBS6297	1959
22			TH PROP OF SELD LIGHT ELEMENT CPDS NBS7437	1962
23		KOWALCZYK L S	THERMAL COND AND ITS VARN WITH T AND P	1955
24		JOHNSON V J	A COMPDM OF PROP OF MATL AT LOW TEMP NBS	1960
25			JANAF THERMOCHEMICAL TABLES	
26		KELLEY K	ENTROPIES OF ELEMENTS AND INORG CPDS BM592	1961
27	*	WEAST R C	HANDBOOK OF CHEMISTRY AND PHYSICS 47TH ED	1966
28	*	JESTER M	PRIVATE COLLECTION OF LIQUID METAL PROPERTIES	1968
29	*	WICKS C E	THERMODYN PROP OF 65 ELEMENTS BU MINES 605	1963
30	*	PETERS R L	MATERIALS DATA MONOGRAPHS	1965
31	*	THOMPSON B	ENERGY CONTENT VS TEMP FOR D-PU A-PU AND U	
32	*	KELLEY K K	ENTHALPY PLOTS DWG K38745 BU MINES 584	1960
33	*	KELLEY K K	BU MINES 371	1939
34	*	LABER Ɔ	BMI FINAL REPT ON SANL TASKS 309/032A 401/027	1967
35	*		Y-12 DATA SHEETS	
36	*	SCHAUER D	WEAPONS MATL DATA BOOK	
37	*	SCHORSCH R H	ENGINEERING PROPERTIES OF SELECTED MATLS	1966
38	*		MODERN PLASTICS ENCYCLOPEDIA	1968
39	*		HANDBOOK OF MOLDED AND EXTR RUBBER GOODYEAR	
40	*		MACHINE DESIGN REFERENCE ISSUE PLASTICS	1964
41	*		METALS HANDBOOK 8TH ED	1961
42	*	STULL D R	THERMODYNAMIC PROPERTIES OF THE ELEMENTS ACS	1956
43	*	HAMPEL C A	RARE METALS HANDBOOK 2ND ED	1961
44	*	MOYER J	PRIVATE COLLECTION OF THERMAL DATA	1968
45	*	HODGE A W	PROP OF HIGH TEMP TI ALLOYS DMIC MEMO 230	1968
46	*	CARSLAW H JAEGER	J CONDUCTION OF HEAT IN SOLIDS 2ND ED OXFORD	1959
47	*	SMITHELLS C J	METALS REFERENCE BOOK 4TH ED VOL 3	1967
48	*	LYNCH J F RUDERER	ENGR PROP OF SEL CERAM MATLS BMI AM CER SOC	1966
49	*		PROP OF KENNAMETAL HARD CARBIDE ALLOYS	1963
50	*		PROP HANDBOOK CARBOLOY ETC GE MET PROD	1958
51	*	CLARK S P JR	HANDBOOK OF PHYSICAL CONSTANTS GEOL SOC AMER	1966
52	*	STEPHENS D R MAIMO	THE THERMAL COND OF ROCK SALT UCRL 6894-II	1964
53	*	SEDDON B J	PHYS PROP OF PU CERAMIC CPDS TRG1601	1968
54	*	STORMS E K	THE REFRACTORY CARBIDES	1967

G.4 MATERIAL CLASSIFICATION SYSTEM

The material classification system tabulated in Table G.2 was used to assign identification numbers to the materials included in this report. The classification of many materials was arbitrary. The range of numbers in each category of materials is sufficient to include most materials of interest and importance. Many unused numbers remain within each category for new materials that can be included in the compilation.

The alphanumeric identifiers assigned to each material aid in identifying the materials on the material data list. In the cases of the elements and most simple compounds, the atomic or molecular formulae are used. In most other cases, acronyms or mnemonics are used.

Table G.2. Material classification system

```
1000   ELEMENTS
       1001-1499   SOLIDS,     LIQUIDS
       1501-1599   GASES
       1601-1999   NOT  USED
2000   ALLOYS
       2001-2099   ALUMINUM    ALLOYS
       2101-2399   COPPER     ALLOYS
       2401-2499   MAGNESIUM    ALLOYS
       2501-2599   NICKEL     ALLOYS
       2601-2699   COBALT     ALLOYS
       2701-2799   TITANIUM    ALLOYS
       2801-2999   NOT  USED
3000   ALLOYS   (CONTINUED)
       3001-3099   IRON ALLOYS
       3101-3299   STEELS
       3301-3499   SUPER ALLOYS (CR-NI-FE)
       3501-3599   BERYLLIUM    ALLOYS
       3601-3649   LEAD, TIN, AND INDIUM ALLOYS
       3651-3679   MOLYBDENUM ALLOYS
       3681-3699   NIOBIUM (COLUMBIUM) ALLOYS
       3701-3749   PLUTONIUM    ALLOYS
       3751-3779   SILVER     ALLOYS
       3781-3799   TANTALUM    ALLOYS
       3801-3829   TUNGSTEN    ALLOYS
       3831-3859   URANIUM ALLOYS
       3861-3879   ZINC ALLOYS
       3881-3899   ZIRCONIUM    ALLOYS
       3901-3999   MISCELLANEOUS ALLOYS
4000   INORGANIC   COMPOUNDS
       4001-4199   OXIDES    (SINGLE)
       4201-4299   OXIDES (MIXED)
       4301-4399   SILICATES
       4401-4499   NITRIDES
       4501-4599   CARBIDES
       4601-4699   BORIDES
       4701-4739   BERYLLIDES
       4741-4759   SULFIDES
       4761-4769   PHOSPHIDES
       4771-4799   SILICIDES
       4801-4899   HALIDES
       4901-4949   INTERMETALLICS, METALLOIDS
       4951-4999   MISCELLANEOUS COMPOUNDS
5000   INORGANIC   MIXTURES
       5001-5199   GLASSES
       5201-5299   CERAMICS,    BRICKS
       5301-5399   NOT  USED
       5401-5499   CERMETS
       5501-5799   GAS MIXTURES
       5801-5999   NOT  USED
6000   INORGANIC   COMPOSITES
       6001-6099   ROCKS, MINERALS, SANDS, SOILS
       6101-6999   NOT  USED
7000   ORGANIC COMPOUNDS AND MIXTURES
       7001-7499 POLYMERS (INCLUDING FOAMS)
       7501-7599 HIGH EXPLOSIVES (INCLUDING MOCK H. E.)
       7601-7699   NOT  USED
8000   ORGANIC COMPOSITES AND NATURAL MATERIALS
       8001-8499    MISCELLANEOUS
       8501-8599 WOODS
       8601-8999   NOT  USED
9000   MISCELLANEOUS
       9001-9099 IDEALIZED MATERIALS
       9101-9999   NOT  USED
```

G.5 QUALITY INDICATOR SYSTEM

The data for each material were chosen as the best values. For cases in which conflicting or inaccurate data were found in the literature, ref. 2 provided the best values available for each material. The numerical quality indicators are necessarily crude and general but may be interpreted approximately as follows:

Quality Type 0 (or Blank)

Data were good or no statement was made in data source restricting their accuracy. No conflicting data were found in any other data sources. Most data in this category are probably accurate to within 1−2% for density, 5−10% for specific heat, and 10−20% for thermal conductivity. When tables or specific heat vs temperature or thermal conductivity vs temperature are included, the constant values are included only for information and generally apply only to temperatures ranging from 0−100°C or at the lower end of the indicated temperature range of the table, whichever is higher.

Quality Type 1

Good data but specific heat or thermal conductivity were reported only for a limited temperature range. The temperature range has been extended by extrapolation, and the resulting values are probably as accurate as reported values.

Quality Type 2

Either the accuracy of the data was poor or values reported by different sources disagreed. The best values were used whenever a judgment could be made, or average values were used with consideration given to data for similar materials.

Quality Type 3

No data could be found in the literature; however, the author (Edwards) made an estimate based on theoretical or empirical grounds or on data for similar materials. These estimates were included whenever they appeared to be about as accurate as reported data for similar materials.

Quality Type 4

No data could be found in the literature, and no accurate estimate could be made. The data, therefore, are considered to be a rough estimate based, in most cases, on data for similar materials.

Quality Type 5

No data could be found in the literature, and no estimate was made. A minimum requirement for inclusion of a material in the alphabetical index and material data list is the availability of at least estimated constant values of density, specific heat, and thermal conductivity.

G.6 MATERIAL INDEX

Table G.3 is an alphabetical index of material descriptions that are available in the library. The format used for the information on each card-image record is as follows:

Columns 1−5

Alphanumeric identifier that always starts in column 1 and begins with a letter. This identifier is provided as an aid in identifying the material on the material data list. The atomic or molecular formulae are used for all elements and most simple compounds; short names, abbreviations, acronyms, or mnemonics are used in other cases.

Column 6

Always left blank.

Columns 7−10

Material identification number, in the range from 1001−9999, and never exactly divisible by 1000. Each material is assigned its own identification number.

Column 11

Always left blank.

Columns 12−50

Material description, including as much information as necessary to identify the specific material. Many abbreviations are used, such as these examples: HP (hot pressed), SINT (sintered), 89 D (89% of theoretical density), 22 P (22% porosity), PRSD (pressed), FRD (fired), and HT (hot treated). A number of other standard abbreviations also are used for special alloys. Mass percentages follow component names. Atomic percentages precede the name of the molecular species.

Column 51

Normally left blank except when more than one card is used to list the data source numbers. In these cases, the additional cards have a + in column 51.

Columns 52−72

Data source numbers, specified in the list of data sources or other information identifying the sources of the compiled data. No classified data sources are listed. For all data estimated by the author or obtained from unknown data sources, a data source number is not indicated.

Columns 73—79

The numerical quality indicators for the data. One column is provided for each of the following properties: density (D), constant specific heat (C), constant thermal conductivity (K), temperature of transition or maximum service temperature (T), latent heat of transition (H), table of specific heat vs temperature (C), and table of thermal conductivity vs temperature (K). The one-letter abbreviation for each property is listed in Table G.3 over the column containing the corresponding quality indicator.

Column 80

Always left blank.

Many materials appear more than once in the index. The reason for this is that several common names exist for the same material or additional identifying information is included under separate entries.

Table G.3. Material index

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
ALOXD	4012	ALUMINUM OXIDE (AL2O3) (POLYXTAL 100 D)	1,20,27,32,47,48,51	200
ALOXP	4014	ALUMINUM OXIDE (AL2O3) (POLYXTAL, 55 D)	1,20,27,32,47,48,51	200
ALOXS	4016	ALUMINUM OXIDE (AL2O3) (SINGLE XTAL)	1,20,27,32,47,48,51	202
ALCR1	5412	ALUMINUM OXIDE + CR (AL2O3 23, CR 77)	20,48	302530
ALCR2	5413	ALUMINUM OXIDE + CR (AL2O3 70, CR 30)	20,48	302535
ANDAL	4312	ALUMINUM SILICATE (AL2O3.SIO2) (ORTHO)	20,27,47,51	20505
KYAN	4314	ALUMINUM SILICATE (AL2O3.SIO2) (TRICL)	20,27,47,51	500
MULL	4316	ALUMINUM SILICATE (3AL2O3.2SIO2) (100D)	20,27,47,48,51	500
AM	1430	AMERICIUM	20,27	342555
NH3	4881	AMMONIA (NH3) (GAS)	1,27,47	0
NH3LQ	4880	AMMONIA (NH3) (LIQUID UNDER PRESSURE)	1,27,47	500
NH4BR	4812	AMMONIUM BROMIDE (NH4BR) (PRSD 8 KB)	27,47	300530
NH4CL	4814	AMMONIUM CHLORIDE (NH4CL) (PRSD 8 KB)	27,47	10
ANALC	4329	ANALCITE (NA20.AL2O3.4SIO2.4H2O) (XTAL)	27,51	305555
ANDAL	4312	ANDALUSITE (AL2O3.SIO2) (ORTHO XTAL)	20,27,51	20505
SB	1006	ANTIMONY	20,1,32,42,27	0
SBLIQ	1007	ANTIMONY (LIQUID)	1,47,42,27,20,14	0
SBTEC	4913	ANTIMONY TELLURIDE (SB2TE3) (CPR, N DR)	20,27	300555
SBTEP	4912	ANTIMONY TELLURIDE (SB2TE3) (POLYXTAL)	20,27	300555
ARGAS	1501	ARGON (GAS)	27,2,42,41	0
AS	1008	ARSENIC, GREY	1,27,41,42	40005
ASTEX	4914	ARSENIC TELLURIDE (AS2TE3) (V ZONE MLT)	20	3305555
ART1	9001	ARTIFICAL MATERIAL 1 (D=C=K=0)		0
ART2	9002	ARTIFICAL MATERIAL 2 (D=C=0, K=10**9)		0
ART3	9003	ARTIFICAL MATERIAL 3 (D=C=K=l)		0
ART4	9004	ARTIFICAL MATERIAL 4 (D=C=l, K=10**9)		0
ART5	9005	ARTIFICAL MATERIAL 5 (D=l, C=K=10**9)		0
ART6	9006	ARTIFICAL MATERIAL 6 (D=C=K=TM=HM=l)		0
ART7	9007	ARTIFICAL MATERIAL 7 (DCKTM=l,HM=10**9)		0
AT	1437	ASTITINE	42	3343335
BRTOL	7501	BARATOL H.E. (TNT 26, BA NITRATE 76)	8	555
BA	1241	BARIUM	20,27,29,42,43	40005
BAB6	4612	BARIUM BORIDE (BAB6)	27,48	300555
BAF2	4816	BARIUM FLUORIDE (BAF2) (SINGLE CRYSTAL)	20,27,51	300230
BANO3	4952	BARIUM NITRATE (BA(NO3)2) (PRSD 8 K8)	1,27	300050
BASO4	4954	BARIUM SULFATE (BASO4) (CRYSTAL)	27,51	300550
BATIS	4213	BARIUM TITANATE (BAO.TIO2) (SINTERED)	20,27,48	500
BATIT	4212	BARIUM TITANATE (BAO.TIO2) (100 D)	20,27,48	500
BATIM	4214	BARIUM TITANATE (BAO.TIO2) (+MN,NB OX)	20,27,48	500
BK	1440	BERKELIUM	27	4445555
BERYL	4318	BERYL (3BEO.AL2O3.6SIO2)	20,27,51	320502
BE	1011	BERYLLIUM	20,6,1,32,34,27,36,42	22
BEBEO	5420	BERYLLIUM + BEO (BE, BEO 0.6-1.7)	20	3300
BE96A	3501	BERYLLIUM ALLOY (BE96.5) (AS RECEIVED)	20	303331
BE96H	3502	BERYLLIUM ALLOY (BE96.5) (ANNEALED)	20	303331
BE98A	3503	BERYLLIUM ALLOY (BE98.5) (AS RECEIVED)	20	303331
BE98H	3504	BERYLLIUM ALLOY (BE98.5) (ANNEALED)	20	303331
BE995	3505	BERYLLIUM ALLOY (BE99.5)	20	3033313
BERYL	4318	BERYLLIUM ALUMINOSILICATE (BERYL)	20,27,51	320502
BE2C	4512	BERYLLIUM CARBIDE (BE2C) (HP OR SNT)	14,20,28,47	2220022
CUBER	2181	BERYLLIUM COPPER (CU BAL, BE 0.38-0.55)	1	331
CUBEH	2182	BERYLLIUM COPPER (CU BAL, SE1.7-1.9)	1,6,41	2335
BE3N2	4414	BERYLLIUM NITRIDE (BE3N2) (PRSD 3.4 KB)	20,47	5
BEOXP	4024	BERYLLIUM OXIDE (BEO) (76 PC DENS)	14,20,27,36,43,44,	0
BEOXP	4024	BERYLLIUM OXIDE (BEO) (76 PC DENS)	+47.48,51	0
BEO	4022	BERYLLIUM OXIDE (BEO) (96 PC DENS)	14,20,27,36,43,44,	0
BEO	4022	BERYLLIUM OXIDE (BEO) (96 PC DENS)	+47,48,51	0
BEOBE	5414	BERYLLIUM OXIDE + BE (BEO, BE 3-12)	20	5500
BEOMO	5416	BERYLLIUM OXIDE + BE + NO (Be 7, MO 7)	20	5501
BEOSI	5418	BERYLLIUM OXIDE + BE + SI	20	305530
PORBE	5280	BERYLLIUM OXIDE PORCELAIN 4811	20	305530
BSPOD	4320	BETA-SPODUMENE (LI2O.AL2O3.4SIO2) (TET)	20,27,51	305550

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
BI	1016	BISMUTH	2,1,27,43,41,42,47	0
BILIQ	1017	BISMUTH (LIQUID)	1,14,27,43,47	0
BITEP	4916	BISMUTH TELLURIDE (BI2TE3-P) (PLANE DR)	20,47	220522
BITES	4918	BISMUTH TELLURIDE SULFIDE (BI2TE2S)	20	3305550
ZRB2Z	5498	BORIDE Z CERMET (ZRB2 81-87, MOSI2 13)	47,48	2000500
ZRB2B	5496	BOROLITE 101 CERMET (ZRB2 93-96, B 4-7)	47,48	505
BORON	1266	BORON	20,27,30,42,43,48	20205
B4C-D	4514	BORON CARBIDE (B4C) (DENSE)	20,27,48	2002500
B4C-P	4515	BORON CARBIDE (B4C) (POROUS)	20,27,48	2002500
BN	4416	BORON NITRIDE (BN) (PERP PR AXIS, 95 D)	20,27,36,47,48	500
BNPPA	4418	BORON NITRIDE (BN) (PRS AXIS, 94 D)	20,27,36,47,48	500
BNAC	4424	BORON NITRIDE (BN 80, C 20) (PRS AXIS)	20	5500
BNOX2	4420	BORON NITRIDE (BN 97, BN2O3 2) (PERP P)	20	5500
BNOXP	4422	BORON NITRIDE (BN 97, BN2O3 2) (PRS AX)	20	5500
SIB4	4614	BORON SILICIDE (B4SI)	27,48	300530
BRALM	2101	BRASS, ALUMINUM (CU76, ZN22, AL2)	1	555
BRCAR	2105	BRASS, CARTRIDGE (CU70, ZN30)	1,2,20,41,46	500
BRPBL	2111	BRASS, LEADED	6,41	3555
BRMUM	2116	BRASS, MUNTZ METAL	1,41	5555
BRRDC	2121	BRASS, RED, CAST (CU85, ZN5, PB5, SN3)	1,41	3555
BRRDW	2122	BRASS, RED, WROUGHT (CU85, ZN15)	1,41	20555
BRTIN	2126	BRASS, TIN (NAVAL AND ADMIRALTY)	1,2,41	2555
BRYEL	2131	BRASS, YELLOW (CU65, ZN35)	1,6,41	555
BRICR	5232	BRICK, CHROME (CR2O3 32)	1,27,41	5530
BRCRM	5234	BRICK, CHROME MAGNESITE (SEE REF 47)	20,47	330
DIATA	5246	BRICK, DIATOMACEOUS EARTH (ACCR STRATA)	4	305530
DIATH	5249	BRICK, DIATOMACEOUS EARTH (HIGH BURN)	4	305530
DIATM	5248	BRICK, DIATOMACEOUS EARTH (MOLDED, FRD)	4	305530
DIATP	5247	BRICK, DIATOMACEOUS EARTH (PRLL STRATA)	4	305530
DIATT	5251	BRICK, DIATOMACEOUS EARTH (USE TO 1100)	4	305530
DIATL	5250	BRICK, DIATOMACEOUS EARTH (USE TO 850C)	4	305530
BRIEF	5256	BRICK, EGYPTIAN FIRE (SIO2 64-71)	20	2323532
BRICF	5224	BRICK, FIRED CARBON	1,20,27,42,47	5505
BRIFR	5262	BRICK, FORSTERITE (MGO 58 SIO2 38) (20P)	20,47	335
BRALF	5212	BRICK, FUSED ALUMINA (AL2O3 96) (22 P)	1,20,47	2330
BRISI	5294	BRICK, HARD FIRED SILICA (SIO2 94-95)	20	330
BRALL	5214	BRICK, HIGH ALUMINA (AL2O3 53) (20 P)	47	322335
BRAL2	5215	BRICK, HIGH ALUMINA (AL2O3 83) (28 P)	47	322335
BRAL3	5216	BRICK, HIGH ALUMINA (AL2O3 87) (2L P)	20	303330
BRIKB	5265	BRICK, KAOLIN INSULATING (D = 0.30)	1,20	5510
BRIKA	5264	BRICK, KAOLIN INSULATING (D = 0.43)	1,20	5510
MAGNP	5267	BRICK, MAGNESITE (MGO 87)	1,20,27,47,48	330
MAGNA	5268	BRICK, MAGNESITE A (MGO 90) (14.5 P)	1,20,27,47,48	330
MAGNB	5269	BRICK, MAGNESITE B 93) (22.6 P)	1,20,27,47,48	330
MAGNC	5270	BRICK, MAGNESITE C 86) (17.8 P)	1,20,27,47,48	330
MAGNS	5266	BRICK, MAGNESITE SPALL RES (MGO 89)	1,20,27,47,48	335
BRIMM	5218	BRICK, MASONRY, MEDIUM	1,2,27,46	2025535
BRIFM	5259	BRICK, MISSOURI FIRECLAY	1,47	3300530
BRIFN	5258	BRICK, NORMAL FIRECLAY (22 P)	47	535
BRISL	5296	BRICK, SILICEOUS (SIO2 89 AL2O3 9) (25P)	47	335
BRIFS	5260	BRICK, SILICEOUS FIRECLAY (23 P)	47	535
BRISK	5298	BRICK, SILLIMANITE (22 PC POROSITY)	20	300530
BRIDS	5254	BRICK, STABILIZED DOLOMITE (22 P)	1,47	530
VERMI	4986	BRICK, VERMICULITE	20	305550
BRGAS	1546	BROMINE (GAS)	27,1,42	0
BRONZ	2141	BRONZE (CU75, SN25)	1,2	305555
BROAL	2146	BRONZE, ALUMINUM (CU92, AL8)	1,6,41	2000555
BROAR	2151	BRONZE, ARCHITECTURAL	1,41	2300555
BROCM	2156	BRONZE, COMMERCIAL	1,2,38,41	555
BROMN	2161	BRONZE, MANGANESE	1,41	555
BROPH	2166	BRONZE, PHOSPHER 10 PERCENT	1,41	5555
BROPL	2168	BRONZE, PHOSPHER 1.25 PERCENT	6,41	555
BROPM	2167	BRONZE, PHOSPHER 5 PERCENT	6,41	555

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
BRSIH	2171	BRONZE, SILICON, HIGH	1,6,41	550
BRSIL	2172	BRONZE, SILICON, LOW	1,6,41	555
BRSNH	2176	BRONZE, TIN (CAST), HIGH LEADED	6,41	23555
BRSNL	2177	BRONZE, TIN (CAST), LEADED	6	3555
BUACR	7026	BUTADIENE-ACRYLONITRILE RUBBER + C	20	5555
BUTYR	7031	BUTYL RUBBER	6,39	555
CD	1041	CADMIUM	2,1,32,42,43,27,41,47	0
CDLIQ	1042	CADMIUM (LIQUID)	2,1,32,42,43,27,41,47	0
CACCP	4957	CALCITE (CACO3) (CRYSTAL) (C AXIS)	1,27,51	0
CACCS	4958	CALCITE (CACO3) (CRYSTAL) (A AXIS)	1,27,51	0
CA	1306	CALCIUM	27,1,42,43,41	5
CAB6	4616	CALCIUM BORIDE (CAB6)	27,48	300555
CACCP	4957	CALCIUM CARBONATE (CACO3) (CALCITE) (C)	1,27,51	0
CACCS	4958	CALCIUM CARBONATE (CACO3) (CALCITE) (A)	1,27,51	0
CACON	4956	CALCIUM CARBONATE (CACO3) (NATURAL)	1,27,51	5
CAF2A	4820	CALCIUM FLUORIDE (CAF2) (MINERAL AGGR)	1,20,27,47,51	205
CAF2	4818	CALCIUM FLUORIDE (CAF2) (SINGLE XTAL)	1,20,27,47,51	200
DOLOM	4960	CALCIUM MAGNESIUM CARBONATE (CAMGC206)	1,27,51	20515
CAO	4026	CALCIUM OXIDE (CAO) (PRESSED, 91 DENS)	1,20,27,47,48,51	200
CAOPP	4028	CALCIUM OXIDE (CAO) (PACKED PWD, 50 D)	1,20,27,47,48	200
GYPSM	4962	CALCIUM SULFATE DIHYDRATE (CASO4.4H2O)	1,27,51	555
CATIT	4216	CALCIUM TITANATE (CAO.TIO2)	20,27,48,51	500
CF	1443	CALIFORNIUM		4445555
CAMOR	1071	CARBON, AMORPHOUS (CARBON STOCK)	27,1	22
CGEM1	1061	CARBON, DIAMOND GEM QUALITY TYPE 1	1,20,27	500
CGRAF	1066	CARBON, GRAPHITE (TYPICAL K)	1,2,20,27,32,42,51	2000002
BRICF	5224	CARBON BRICK, FIRED	1,20,27,42,47	5505
CO2	4027	CARBON DIOXIDE	1,2,27	0
COGAS	4029	CARBON MONOXIDE	1,27	300030
CELA2	7037	CELLULOSE ACETATE (HIGH K)	6,38	20555
CELA1	7036	CELLULOSE ACETATE (LOW K)	6,38	20555
CEAB2	7042	CELLULOSE ACETATE BUTYRATE (HIGH K)	6,38	225555
CEAB1	7041	CELLULOSE ACETATE BUTYRATE (LOW K)	6,38	225555
CENIP	7046	CELLULOSE NITRATE (PYROXYLIN)	6,38	200555
CEPRL	7051	CELLULOSE PROPRIONATE (LOW K)	6,38	225555
CEPR2	7052	CELLULOSE PROPRIONATE (HIGH K)	6,38	225555
CETRA	7056	CELLULOSE TRIACETATE	38	5555
CE	1356	CERIUM	41,27,20,42	20205
CEB6	4626	CERIUM BORIDE (CEB6) (99.0 PC DENSE)	48	300555
CEO2	4030	CERIUM OXIDE (CEO2) (PRSD, SNTRD, 86 D)	1,20,47	201
CES	4742	CERIUM SULFIDE (CES)	20,48	22505
CE2S3	4744	CERIUM SULFIDE (CE2S3)	20,48	20512
CS	1231	CESIUM	1,14,27,28,32,43,47	40005
CSLIQ	1232	CESIUM (LIQUID)	14,27,28.32,43,47	22
CHALK	6018	CHALK (AV PROP)	1,46	25555
CLGAS	1506	CHLORINE (GAS)	27,1,42,2	0
CRALM	4973	CHROME ALUM (CR2(SO4)3.K2SO4.24H2O)	1,27	550
BRICR	5232	CHROME BRICK (CR2O3 32)	1,27,41	5530
BRCRM	5234	CHROME MAGNESITE BRICK (SEE REF 47)	20,47	330
CNISA	3312	CHROME-NICKEL-IRON SUPERALLOYS	6	555
CR	1201	CHROMIUM	20,1,27,32,43,42	0
CHCNI	5422	CHROMIUM CARBIDE + NI (CR(X)C(Y), NI)	48	2000555
CUCHR	2183	CHROMIUM COPPER (CU BAL, CRO.5)	6,20,47	2330
CRN	4426	CHROMIUM NITRIDE (CRN) (PRSD, 100 PC D)	27,47,48	505
CR2N	4428	CHROMIUM NITRIDE (CR2N) (PRSD, 100 D)	27,47,48	200505
CRS12	4772	CHROMIUM SILICIDE (CRSI2)	20,27,48	500
CO	1046	COBALT	20,27,1,32,43,30,41,	0
CO	1046	COBALT	+42	0
COCRW	2646	COBALT ALLOY (CO64,CR30,W6)	20	330333
COHEL	2612	COBALT ALLOY HE-1049	20	3300
COH21	2614	COBALT ALLOY HS-21 (AS CAST)	1,14,20,41	310
COH2A	2615	COBALT ALLOY HS-21 (AGED)	1,14,20,41	310
COH23	2616	COBALT ALLOY HS-23	14,20	303330

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
COH25	2618	COBALT ALLOY HS-25 (L-605) (WROUGHT)	1,14,41	330
COH27	2620	COBALT ALLOY HS-27 (AS CAST)	14,20	303330
COH30	2622	COBALT ALLOY HS-30 (422-19) (AS CAST)	14,20	333333
COH31	2624	COBALT ALLOY HS-31 (X-40) (AS CAST)	1,14,20,41	300330
COH36	2626	COBALT ALLOY HS-36 (CAST)	1,14,20	333
COJG3	2628	COBALT ALLOY JESSOP G-32	20	303330
COJ15	2630	COBALT ALLOY J-1570	20	3333330
COK42	2632	COBALT ALLOY K-42B	14	3323332
COMTL	2636	COBALT ALLOY MULTIMET (N-155) (LOW C)	1,14,20,41	330
COMTM	2634	COBALT ALLOY MULTIMET (N-155) (WROUGHT)	1,14,20,41	330
COS59	2638	COBALT ALLOY S-590 (WROUGHT)	14,41	330
COS81	2640	COBALT ALLOY S-816 (WROUGHT)	1,14,20,41	330
COV36	2642	COBALT ALLOY V-36 (WROUGHT)	41	333333
COW52	2644	COBALT ALLOY WI-52	20	3333330
COWSG	2645	COBALT ALLOY WI-52 (CR COATED SAMPLE)	20	3333330
CNLOX	4218	COBALT NICKEL OXIDE (46COO.46NIO.BLIO)	20,27	3305555
COBOX	4034	COBALT OXIDE (COO)	20,27,47,48	2005
COS	4774	COBALT SILICIDE (COS)	20,48	305555
NDTAW	3681	COLOMBIUM ALLOY (CB61,TA28,WL0,ZRO.5)	20	3300
NBWMO	3684	COLOMBIUM ALLOY (CB80, W15, MO5)	20	3300
NBTIZ	3682	COLOMBIUM ALLOY (CB85, TI10, ZR5)	20	3300
NBTA5	3683	COLOMBIUM ALLOY (CB95, TA5)	20	3303330
CMB-3	7502	COMPOSITION B-3 H.E. (RDX 60, TNT 40)	8	555
CMC-4	7503	COMPOSITION C-4 H.E. (RDX 90, BINDERS)	8	5555
CONCN	5240	CONCRETE, CINDER	1,2,20,27,46	25535
CONLW	5243	CONCRETE, LIGHTWEIGHT	1,2,20,27,46	5530
CONCS	5241	CONCRETE, STONE (1-2-4 MIX)	1,2,20,27,46	25535
CON14	5242	CONCRETE, 1-4 DRY	1,2,20,27,46	25535
CU	1051	COPPER	20,1,27,32,34,37,42,	0
CU	1051	COPPER	+47	0
CUWGT	2191	COPPER, WROUGHT (ETP, DHP, TEO.5, PBL)	41	3535
CUBER	2181	COPPER ALLOY (CU BAL, BE 0.38-0.55)	1	331
CUBEH	2182	COPPER ALLOY (CU BAL, BE 1.7-1.9)	1,6,41	2335
CUCHR	2183	COPPER ALLOY (CU SAL, CRO.5)	6,20,47	2330
CUPR3	2203	COPPER ALLOY (CU70, N130)	1,6,41	2555
CUPRL	2202	COPPER ALLOY (CU90, NI10)	1,6,41	2555
CUALL	2201	COPPER ALLOY (CU99.4, ALO.3, ZRO.27)	20	3303331
NICON	2506	COPPER ALLOY CONSTANTAN (CU55, NI45)	1,2,14,41,47	331
CUMAN	2195	COPPER ALLOY MANGANIN	2	3305550
CUNIS	2185	COPPER ALLOY NICKEL SILVER (NI 10-20)	1,2,6,41,47	2222555
CUGZN	2186	COPPER GILDING METAL (CU95, ZN5)	1,41	555
CUOLI	4220	COPPER LITHIUM OXIDE (96CUO.4LIO)	20,27	3305550
CUO	4036	COPPER OXIDE (CUO) (TENORITE)	1,20,27,47,51	2002220
CUSO4	4966	COPPER SULFATE (CUSO4) (CRYSTAL)	27	550
BLUEV	4967	COPPER SULFATE HYDRATE (CUSO4.5H2O)	1,27,51	555
CORDR	4322	CORDIERITE (2MGO.2AL2O3.5SIO2)	20,51	5510
CORKG	8514	CORK, GROUND	2,46	5555
CORKR	8515	CORK, GROUND, REGRANULATED	2,46	5555
CUPR3	2203	CUPRONICKEL (CU70, NI30)	1,6,41	2555
CUPRL	2202	CUPRONICKEL (CU90, NI10)	1,6,41	2555
CM	1446	CURIUM	20,27	4445555
DAN35	3901	DANDELION 35	34,36	5500
DATB	7504	DATB H.E. (DIAMINO TRINITROBENZENE)	8	300555
D-GAS	1551	DEUTERIUM	27	3305530
D20	4118	DEUTERIUM OXIDE (D20) (LIQUID)	14,27	300
DIABG	5020	DIABASIC GLASS (ARTIFICIAL)	27,51	3305530
DAPON	7061	DIALLYL PHTHALATE (DAPON)	35	2025555
DIATA	5246	DIATOMACEOUS EARTH BRICK (ACCR STRATA,)	4	305530
DIATH	5249	DIATOMACEOUS EARTH BRICK (HIGH BURN)	4	305530
DIATM	5248	DIATOMACEOUS EARTH BRICK (MOLDED, FRD)	4	305530
DIATP	5247	DIATOMACEOUS EARTH BRICK (PRLL STRATA)	4	305530
DIATT	5251	DIATOMACEOUS EARTH BRICK (USE TO 1100C)	4	305530
DIATL	5250	DIATOMACEOUS EARTH BRICK (USE TO 850C)	4	305530

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
DOLOM	4960	DOLOMITE (CAMG(CO3)2) (AVG PROP)	1,27,51	20555
BRIDS	5254	DOLOMITE BRICK, STABILIZED (22 P)	1,47	530
DY	1391	DYSPROSIUM	41,27,20,42	35
BRIEF	5256	EGYPTIAN FIRE BRICK (SIO2 64-71)	20	2323532
ES	1449	EINSTEINIUM		4445555
EPCR1	7066	EPOXY, DER 332 (C), HYSOL 6000-OP (K)	6,20,38	2220505
EPGFM	7071	EPOXY, GLASS FIBER FILLED (MOLDED)	20,38	25552
EPSFC	7076	EPOXY, SILICA FILLED, CAST	38	2225555
EPUNC	7081	EPOXY, UNFILLED, CAST	38	2025555
EPSOM	4971	EPSOMITE (MGSO4.7H2O) (CRYSTAL)	1,27,51	555
ER	1401	ERBIUM	41,27,20,42	5
ETHCE	7086	ETHYL CELLULOSE (WIDE RANGE OF C, K)	6,38	225555
ETHVA	7091	ETHYL VINYL ACETATE	38	25555
EU	1376	EUROPIUM	20,27,41,43	30005
EUB6	4634	EUROPIUM BORIDE (EUB6) (93.0 PC DENSE)	48	305555
FM	1452	FERMIUM		4445555
FBRFX	8201	FIBERFAX PAPER (CARBORUNDUM CO.)	44	5500
BRIFM	5259	FIRECLAY BRICK, MISSOURI	1,47	3300530
BRIFN	5258	FIRECLAY BRICK, NORMAL (22 P)	47	535
BRIFS	5260	FIRECLAY BRICK, SILICEOUS (23 P)	47	535
FLINT	5022	FLINT GLASS	1,27,46	25532
ART5	9005	FLUID, HEAT SINK (D=1, C=K=10**9)		0
ART4	9004	FLUID, WELL STIRRED (D=C=1, K=10**9)		0
F-GAS	1511	FLUORINE (GAS)	27,42	0
FCCTF	7101	FLUOROCARBONS, CFE AND CTFE	6,38	555
FCFEP	7106	FLUOROCARBONS, FEP	6,38	555
FCTFE	7111	FLUOROCARBONS, TFE (TEFLON)	1,6,20,38	2500
GLASF	5024	FOAMED GLASS (D = 0.144)	1	305532
FORST	4324	FORSTERITE (2MGO.SIO2) (100 PC DENSE)	1,20,27,51	0
BRIFR	5262	FORSTERITE BRICK (MGO 58 SIO2 38)(20 P)	20,47	335
FR	1455	FRANCIUM	42	4343335
SIO2	4062	FUSED SILICA GLASS	1,20,27,47,51	2200
SIO2	4062	FUSED QUARTZ GLASS (SIO2)	1,20,27,47,51	2200
GD	1381	GADOLINIUM	41,27,20,42	5
GDB6	4636	GADOLINIUM BORIDE (GDB6) (95.6 PC DENS)	48	300555
GD2O3	4038	GADOLINIUM OXIDE (GD2O3) (MONOC) (98 D)	20,47,48	2000505
GA	1311	GALLIUM	27,42,43,1	20005
GALIQ	1312	GALLIUM (LIQUID)	1,14,27,43,47	20005
GE-IP	1261	GERMANIUM (INTRINSIC, P-TYPE)	20,27,42,43,1,48	2
GE-N	1262	GERMANIUM (N-TYPE)	20,27,42,43,1,48	0
GLASS	5012	GLASS (SEE REF 27, PP E5-E8) (AVG PROP)	27	2225555
GLABC	5014	GLASS, BOROSILICATE CROWN	27,51	25502
PYRC6	5018	GLASS, CERAMIC, PYROCERAM 9606	20	5500
PYRC8	5019	GLASS, CERAMIC, PYROCERAM 9608	20	5500
DIABG	5020	GLASS, DIABASE (ARTIFICIAL)	27,51	3305530
FLINT	5022	GLASS, FLINT	1,27,46	25532
GLASF	5024	GLASS, FOAMED (D = 0.144)	1	305532
SIO2	4062	GLASS, FUSED QUARTZ (SIO2)	1,20,27,47,51	2200
SIO2	4062	GLASS, FUSED SILICA OR VITROUS SILICA	1,20,27,47,51	2200
GLASL	5026	GLASS, LEAD	20	305530
GLASW	5028	GLASS, LINE WINDOW	20	305530
OBSID	5030	GLASS, OBSIDIAN	27,51	305530
PYREX	5016	GLASS, PYREX	2,20,27,51	500
GLASP	5034	GLASS, SODA PLATE	20,51	305532
GLASC	5032	GLASS, SODA-LIME	20,27	3325532
VYCOR	5036	GLASS, VYCOR	20,27	500
AU	1056	GOLD	20,6,1,32,27,42,41,2	0
AULIQ	1057	GOLD (LIQUID)	14,20,27,42,43	40005
GRNTM	6014	GRANITE (AV PROP) (SEE REF 51)	1,2,27,46,51	25555
GRNTH	6013	GRANITE (HIGH K)	1,2,27,46,51	25555
GRNTL	6012	GRANITE (LOW K)	1,2,27,46,51	25555
GYPSM	4962	GYPSUM (CASO4.4H2O) (MINERAL) (HIGH D)	1,27,51	555
GYPSA	4964	GYPSUM (CASO4.4H2O) (ARTIFICIAL)	1,27	555

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
HF	1271	HAFNIUM	20,32,27,42,43,41	0
HFB2	4618	HAFNIUM BORIDE (HFB2)	20,27,48	20502
HFC	4518	HAFNIUM CARBIDE (HFC)	20,27,48,54	22502
HFN	4430	HAFNIUM NITRIDE (HFN) (HP STRD 78-92 D)	20,27,48	2020502
HFO2	4040	HAFNIUM OXIDE (HFO2) (MONOC) (94 D)	20,27,48	21
D2O	4118	HEAVY WATER (D2O) (LIQUID)	14,27	300
HEGAS	1516	HELIUM (GAS)	27,1,2,42	0
FE2O3	4042	HEMATITE (FE2O3)	1,27,32,47,48,51	20505
HO	1396	HOLMIUM	20,27,41,42,43	30005
H-GAS	1521	HYDROGEN (GAS)	27,2,42,1	0
ICE	4116	ICE (H2O) (SOLID)	27,46,51	0
IN	1211	INDIUM	1,27,43,30,42,41	20005
INLIQ	1212	INDIUM (LIQUID)	1,42,43,47	5
SOLD1	3611	INDIUM ALLOY (IN25, SN37.5, PB37.5)	36	3355
INSBP	4920	INDIUM ANTIMONIDE (INSB) (IMP 0.16)	20,27	3000000
INSBS	4921	INDIUM ANTIMONIDE (INSB) (IMP 0.33-1.2)	20,27	3000000
INASP	4922	INDIUM ARSENIDE (INAS) (POLYX, IMP 3.0)	20,27	3000500
INASS	4923	INDIUM ARSENIDE (INAS) (PURE, S-DOPED)	20,27	3000500
INTE	4924	INDIUM TELLURIDE (IN2TE3)	20,27	300550
I-SOL	1316	IODINE (SOLID)	27,42,41,1	5
IR	1276	IRIDIUM	27,41,20,42,43	20002
FEFUR	1083	IRON (0 TO 3000 DEG C)	20,1,32,27,42,44,51	0
FECRY	1082	IRON (-273 TO 763 DEG C)	20,1,32,27,42,44,51	0
FE	1081	IRON (-51 TO 1537 DEG C)	20,1,32,27,42,44,51	0
INDO2	3022	IRON, DUCTILE (MG CONTAINING)	1	300355
INDO3	3023	IRON, DUCTILE (MG CONTAINING, HEAT RES)	1	300355
INDO1	3021	IRON, DUCTILE (0.06 MG)	1,6	323333
GCIO1	3001	IRON, GREY CAST, FERRITIC (2.3-3.0 C)	1,2,6,20,27,33,46	3300
GCIO2	3002	IRON, GREY CAST, FERRITIC (3.2-3.8 C)	1,2,6,20,27,33,46	303330
GCIO3	3003	IRON, GREY CAST, PEARLITIC (2.3-3.0 C)	1,2,6,20,27,33,46	303330
GCIO4	3004	IRON, GREY CAST, PEARLITIC (3.0-3.2 C)	1,2,6,20,27,33,46	303330
GCIO5	3005	IRON, GREY CAST, PEARLITIC (3.4 C)	1,2,6,20,27,33,46	303330
GCIO6	3006	IRON, GREY CAST, PEARLITIC (3.7-3.8 C)	1,2,6,20,27,33,46	303330
GCIO7	3007	IRON, GREY CAST, PEARLITIC (4.12 C)	1,2,6,20,27,33,46	303330
FE	1081	IRON, INGOT (FE99.9+)	1,20,27,32,42,44	0
KOVAR	3031	IRON, NI-HARD TYPES 1 AND 2 (KOVAR)	1,20	200320
INNRD	3035	IRON, NI-RESIST, TYPE D2 (CAST)	1	300355
INNR3	3033	IRON, NI-RESIST, TYPE 3 (CAST)	1	355
INNR4	3034	IRON, NI-RESIST, TYPE 4 (CAST)	1	355
INNIR	3032	IRON, NI-RESIST, TYPES 1 AND 2 (CAST)	1,20	325
INNIT	3041	IRON, NI-TENSYLIRON (CAST, HEAT TREAT)	1	300355
INCFB	3026	IRON, NODULAR CAST, FERRITIC BASE	1,2,20	303330
INCPB	3027	IRON, NODULAR CAST, PEARLITIC BASE	1,2,20	303330
INMAL	3036	IRON, MALLEABLE (2.5 C)	1,6	5555
INWRT	3037	IRON, WROUGHT (VARIOUS)	1,2,6,27	2330
NINVR	2501	IRON ALLOY INVAR (FE64, NI36)	6,41,47	20355
FE2O3	4042	IRON OXIDE (FE2O3) (HEMATITE)	1,27,32,47,48,51	20505
FE3O4	4044	IRON OXIDE (FEO.FE2O3) (MAGNATITE)	20,27,47,51	2000005
FES2	4746	IRON SULFIDE (FES2) (SINGLE CRYSTAL)	27,47,51	505
BRIKB	5265	KAOLIN BRICK, INSULATING (D = 0.30)	1,20	5510
BRIKA	5264	KAOLIN BRICK, INSULATING (D = 0.43)	1,20	5510
KRGAS	1556	KRYPTON (GAS)	27,1,42	0
KYAN	4314	KYANITE (AL2O3.SIO2) (TRICL CRYSTAL)	20,27,51	500
LA	1351	LANTHANUM	41,27,1,20,42	5
LAB6	4624	LANTHANUM BORIDE (LAB6) (99.5 PC DENSE)	48	300555
LW	1491	LAWRENCIUM		4445555
PB	1091	LEAD	2,1,32,27,37,42	0
PBLIQ	1092	LEAD (LIQUID)	1,2,14,27,37,42,43,47	2
PB	1091	LEAD (0.07 CA)	36	0
PBANT	3601	LEAD, ANTIMONIAL (PB, SB 4-6) (HARD)	1,5,41	355
PBSBH	3602	LEAD, ANTIMONIAL (PB, SB 8-9)	1,6,41	355
PB	1091	LEAD, ANTIMONIAL (PB, SB1)	1,41	0
PB	1091	LEAD, TELLURIUM	1	0

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
SOLD1	3611	LEAD ALLOY (PB37.5, SN37.5, IN25)	36	3355
SOLD2	3612	LEAD ALLOY (PB39.2, SN60.8) (SOLDER)	34,41	5
SOLD4	3613	LEAD ALLOY (PB50, SN50) (SOLDER)	1,41	50
SOLD3	3614	LEAD ALLOY (PB60, SN40) (SOLDER)	1,41	55
GLASL	5026	LEAD GLASS	20	305530
PBO	4046	LEAD OXIDE (PBO) (YELLOW)	1,20,27,32,47	201
PBTEP	4927	LEAD TELLURIDE (PBTE) (POLYXTAL)	20,27	300550
PBTES	4926	LEAD TELLURIDE (PBTE) (SINGLE CRYSTAL)	20,27	300550
LIMST	6016	LIMESTONE (DENSE, DRY)	1,2,27,46,51	5555
LIMSW	6017	LIMESTONE (H2O 15.3)	1,2,27,46,51	5555
LI	1216	LITHIUM	1,27,28,41,43,42,47	0
LILIQ	1217	LITHIUM (LIQUID)	1,27,28.41,42,43,47	202
BSPOD	4320	LITHIUM ALUMINUM SILICATE (B-SPODUMENE)	20,27,51	305550
CNLOX	4218	LITHIUM COBALT NICKEL OXIDE (8-46-46 N)	20,27	3305555
CUOLI	4220	LITHIUM COPPER OXIDE (4LIO.96CUO)	20,27	3305550
LIH	4882	LITHIUM HYDRIDE (LIH) (CAST, VAC VOIDS)	20	200
LIH	4882	LITHIUM HYDRIDE (LIH) (POWDER COMPACT)	20	200
LIHCG	4884	LITHIUM HYDRIDE (LIH) (CAST, GAS VOIDS)	20	200
LIF	4822	LITHIUM FLUORIDE (LIF) (SINGLE CRYSTAL)	1,20,27,47,51	20202
LIFPB	4824	LITHIUM FLUORIDE (LIF 96) (PLASTIC BND)	1,20,27,47,51	200
MNSEL	4928	LITHIUM MANGANESE SELENIDE (97MN.3LI.-)	20,27	3305530
NIOLI	4222	LITHIUM NICKEL OXIDE (5LIO.95NIO)	20,27	3305550
LITBR	4968	LITHIUM TETRABORATE (LI2O.2B2O3) (EPOX)	27	555
LU	1416	LUTETIUM	20,27,41	230025
LX041	7505	LX-04-1 H.E. (HMX 85, VITON A 15)	8	200551
FE304	4044	MAGNATITE (FEO.FE2O3)	20,27,47,51	2000005
MAGNA	5268	MAGNESITE A BRICK (MGO 90) (14.5 P)	1,20,27,47,48	330
MAGNB	5269	MAGNESITE B BRICK (MGO 93) (22.6 P)	1,20,27,47,48	330
MAGNP	5267	MAGNESITE BRICK (MGO 87)	1,20,27,47,48	330
MAGNS	5266	MAGNESITE BRICK, SPALL RES (MGO 89)	1,20,27,47,48	335
MAGNC	5270	MAGNESITE C BRICK (MGO 86) (17.8 P)	1,20,27,47,48	330
MG	1096	MAGNESIUM	20,1,32,27,42	0
MGLIQ	1097	MAGNESIUM (LIQUID)	1,14,20,27,32,43	30005
MG042	2462	MAGNESIUM ALLOY (MG,AG2.5,CE2,ZRO.6)	47	333
MG001	2421	MAGNESIUM ALLOY AM100A (CASTING)	41	33
MG002	2422	MAGNESIUM ALLOY AZ31(X,S) (WROUGHT)	14,20,47	10
MG003	2423	MAGNESIUM ALLOY AZ31B(P,S) (WROUGHT)	1,20,41	0
MG004	2424	MAGNESIUM ALLOY AZ61A(X), AZM (WROUGHT)	14,41,47	33
MG005	2425	MAGNESIUM ALLOY AZ63A(AC,F) (CASTING)	14,41	33
MG006	2426	MAGNESIUM ALLOY AZSOA(X,FRGD) (WROUGHT)	1,20,37,41	303
MG007	2427	MAGNESIUM ALLOY AZ81A(T4) (CASTING)	41	300333
MG008	2428	MAGNESIUM ALLOY AZB55(X) (WROUGHT)	47	333
MG009	2429	MAGNESIUM ALLOY AZ91A,B (DC) (CASTING)	1,14,41,47	31
MG010	2430	MAGNESIUM ALLOY AZ91C(AC) (CASTING)	1,41	33
MG011	2431	MAGNESIUM ALLOY AZ92A(AC) (CASTING)	1,14,41	33
MG012	2432	MAGNESIUM ALLOY A3A (WROUGHT)	41	33333
MG013	2433	MAGNESIUM ALLOY AS(AC OR ST) (CASTING)	47	333
MG014	2434	MAGNESIUM ALLOY BZ33A(AC,AH) (CASTING)	1	333
MG015	2435	MAGNESIUM ALLOY EK30A, H812 (CASTING)	20,41	31
MG017	2437	MAGNESIUM ALLOY EK33A, H811 (CASTING)	20,41	32
MG016	2436	MAGNESIUM ALLOY EK41A(T5,T6) (CASTING)	41	333
MG018	2438	MAGNESIUM ALLOY HK31A(H24) (CASTING)	1,20,41	0
MG019	2439	MAGNESIUM ALLOY HK31A(O) (CASTING)	1,20,41	0
MG020	2440	MAGNESIUM ALLOY HK31A(T6) (CASTING)	1,20,41	0
MG021	2441	MAGNESIUM ALLOY HM21A(O,H24) (WROUGHT)	41	313
MG022	2442	MAGNESIUM ALLOY HM31A (WROUGHT)	20,41	3
MG023	2443	MAGNESIUM ALLOY HZ32A(AC),ZT1 (CASTING)	1,20,41,47	33
MG024	2444	MAGNESIUM ALLOY MAGNOX A12(X) (WROUGHT)	47	333
MG025	2445	MAGNESIUM ALLOY M1(AC) (CASTING)	14,47	333
MG026	2446	MAGNESIUM ALLOY M1(X,S) (WROUGHT)	14,47	333
MG027	2447	MAGNESIUM ALLOY M1A (WROUGHT)	41	33
MG028	2448	MAGNESIUM ALLOY PE (WROUGHT)	41	30033
MG029	2449	MAGNESIUM ALLOY ZA(AC) (CASTING)	47	3333

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
MG030	2450	MAGNESIUM ALLOY ZE10A(O,H24) (WROUGHT)	41	33
MG031	2451	MAGNESIUM ALLOY ZE41A(T5,HT) (CASTING)	20,41,47	330
MG032	2452	MAGNESIUM ALLOY ZH42 (CASTING)	41	33
MG033	2453	MAGNESIUM ALLOY ZH62A(AC),TZ6 (CASTING)	20,41,47	31
MG034	2454	MAGNESIUM ALLOY ZK20A (WROUGHT)	20,41	3303333
MG035	2455	MAGNESIUM ALLOY ZK51A, H807 (CASTING)	20,41,47	0
MG036	2456	MAGNESIUM ALLOY ZK60A,B, ZW6 (CASTING)	1,14,20,41,47	1
MG037	2457	MAGNESIUM ALLOY ZREO (EZ30) (CASTING)	20	303331
MG038	2458	MAGNESIUM ALLOY ZRE1(AA) (CASTING)	47	333
MG039	2459	MAGNESIUM ALLOY ZTY(X) (HK11) (WROUGHT)	20,47	331
MG040	2460	MAGNESIUM ALLOY ZW1(X) (ZK11) (WROUGHT)	47	333
MG041	2461	MAGNESIUM ALLOY ZW3(X) (ZK31) (WROUGHT)	47	333
MG043	2463	MAGNESIUM ALLOY 1959 (MG,CE4.33)	20	3303335
MG044	2464	MAGNESIUM ALLOY 1960 (NG,CE6.7)	20	3303330
MG045	2465	MAGNESIUM ALLOY 1961 (MG,CE11.85)	20	3303330
MG046	2466	MAGNESIUM ALLOY 1964 (MG,CE5,CO2,MNO.8)	20	3303330
MG047	2467	MAGNESIUM ALLOY 1992 (MG,CE4.45,CO3)	20	3303330
SPINX	4224	MAGNESIUM ALUMINATE (MGO.AL2O3) (XTAL)	20,27,47,48,51	500
SPINL	4225	MAGNESIUM ALUMINATE (MGO.AL2O3) (100 D)	20,27,47,48,51	500
CORDR	4322	MAGNESIUM ALUMINUM SILICATE (CORDIERIT)	20,51	5510
MGOPX	4050	MAGNESIUM OXIDE (MGO) (POLYXTAL, 100 D)	1,20,27,47,48,51	2000
MGOX	4048	MAGNESIUM OXIDE (MGO) (SINGLE CRYSTAL)	1,20,27,47,48	0
FORST	4324	MAGNESIUM SILICATE (2MGO.SIO2) (100 D)	1,20,27,51	0
STEAT	4326	MAGNESIUM SILICATE (MGO.SIO2) (COMMERC)	1,20,27,51	505
MG2SI	4776	MAGNESIUM SILICIDE (MG2SI)	20,27,48	300530
MGSO4	4970	MAGNESIUM SULFATE (MGSO4) (CRYSTAL)	1,27	50
EPSOM	4971	MAGNESIUM SULFATE HYDRATE (MGSO4.7H2O)	1,27,51	555
PORMT	5283	MAGNESIUM TITANATE PORCELAIN	20,48	305530
MNCRY	1101	MANGANESE	20,1,32,27,43,42	0
MNO	4052	MANGANESE OXIDE (MNO) (SINGLE XTAL)	1,27,48	5
MN3O4	4054	MANGANESE OXIDE (MN3O4) (87 PC DENSE)	20,27,48,51	22005
MNSEL	4928	MANGANESE SELENIDE (97MNSE.3LISE)	20,27	3305530
MNSI2	4778	MANGANESE SILICIDE (MNSI2)	20,27,48	300530
MNTEA	4930	MANGANESE TELLURIDE (95MNTE.5MNAS)	20	3305530
MNTEN	4931	MANGANESE TELLURIDE (99MNTE.NATE)	20	3305530
CUMAN	2195	MANGANIN (CU84, NI4, MN12)	2	3305530
MARBL	6021	MARBLE (AV PROP) (SEE REP 51)	1,27,51	25555
MARBD	6020	MARBLE DIELECTRIC (XTAL) (CACO3 99.99)	1,27,51	5550
MELM2	7122	MELAMINE (HIGH DENS, HIGH K)	6,38	2022555
MELM1	7121	MELAMINE (LOW DENS, LOW K)	6,38	2022555
MELAC	7123	MELAMINE, ALPHA CELLULOSE FILLED	38	22555
MELAS	7124	MELAMINE, ASBESTOS FILLED (MST 95-205)	38	2022555
MELAF	7125	MELAMINE, CELLULOSE FILLED (MST 95-205)	38	22555
MELFF	7127	MELAMINE, FABRIC OR FLOCK FILLED	6,38	555
MELGF	7126	MELAMINE, GLASS FIBER FILLED (MST 205)	6,38	555
MD	1461	MENDELEVIUM		4445555
HGLIQ	1207	MERCURY (LIQUID)	1,27,28,42,43,47,14	0
HGCL2	4826	MERCURY CHLORIDE (HGCL2) (PRSD 8 KB)	1,27,47	10
CH4	4890	METHANE (CH4) (GAS)	1,27	0
MEMAC	7131	METHYL METHACRYLATE	1,38	25550
MICAX	4982	MICA (SINGLE CRYSTAL) (A OR B AXES)	20,51	305555
MICAC	4983	MICA (SINGLE CRYSTAL) (C AXIS)	20,51	305555
MICAS	4984	MICA (SINGLE CRYSTAL) (SYNTHETIC) (98D)	20	305550
MICAB	4985	MICA BRICK (RED OR WHITE) (AVG PROP)	20	305552
MICAP	4987	MICA INSULATING POWDER	20	305550
HEM01	7514	MOCK H.E. LM-04-0 H4-048-A294-3	5,8	5530
HEM02	7515	MOCK H.E. LM-04-0 H7-048-A522.1	5,8	5500
HEM03	7506	MOCK H.E. RM-04-BG (LX-04-1 MECH MOCK)	8	5555
HEM04	7507	MOCK H.E. 90010 (PBX-9404 MECH MOCK)	8	5555
MO	1106	MOLYBDENUM	20,6,1,32,41,43,42,	0
MO	1106	MOLYBDENUM	+47,14	0
MOAFE	3672	MOLYBDENUM ALLOY (MO BAL, FE 0.25)	20	333311
MOW30	3673	MOLYBDENUM ALLOY (MO70, W30)	20	3300

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL			REFERENCES	QUALITY INDEX DCKTHCK[a]
MOATI	3671	MOLYBDENUM	ALLOY	(MO99.5, TIO.5)	1,6,41	300
MOBED	4712	MOLYBDENUM	BERYLLIDE (MOBE12)		20,48	500
MO2C	4520	MOLYBDENUM	CARBIDE (MO2C)		20,27,48	2022502
MO2N	4432	MOLYBDENUM	NITRIDE (MO2N) (PR SRD 100D)		47,48	2000505
MOSE2	4932	MOLYBDENUM	SELENIDE (MOSE2)		20	3303555
MOS12	4780	MOLYBDENUM	SILICIDE (MOS12)		20,27,48	2500
MOTE2	4934	MOLYBDENUM	TELLURIDE (MOTE2)		20	3305555
MULBY	3831	MULBERRY (U90, NB7.5, ZR2.5)			34,36,37	0
MULL	4316	MULLITE (3AL2O3.2SIO2) (100 PC DENSE)			20,27,48,51	500
ND	1366	NEODYMIUM			41,20,42	30225
NDB6	4630	NEODYMIUM	BORIDE (NDB6) (97.3 PC DENSE)		48	300555
NEGAS	1526	NEON (GAS)			27,2,1,42	20000
NEOPR	7136	NEOPRENE RUBBER			6,38,39	555
NP	1464	NEPTUNIUM			20,27	2440555
NI	1111	NICKEL			20,2,1,32,37,41,42,51	2
NIAL3	2588	NICKEL	ALLOY	(NI35,CR20,FE45)	41	5533
NIAL2	2586	NICKEL	ALLOY	(NI60,CR16,FE24)	41	5533
NIALL	2584	NICKEL	ALLOY	(NI62,CR12,FE26)	2,41	3305533
N1995	2590	NICKEL	ALLOY	(NI99.5) LOW C, 220, 225	1,14	320
NICKA	2503	NICKEL	ALLOY	A (NI99.4) (ANNEALED)	1,14,20	2
NICAN	2502	NICKEL	ALLOY	CALITE N	14	333
NICH3	2504	NICKEL	ALLOY	CHLORIMET 3	14	3333
NICON	2506	NICKEL	ALLOY	CONSTANTAN (NI45, CU55)	1,2,14,41,47	331
NICOD	2508	NICKEL	ALLOY	CORROSIST	14	333
NIDAE	2592	NICKEL	ALLOY	D AND E	1	30
NIDUH	2512	NICKEL	ALLOY	DURANICKEL (AND -R) (HARD)	1,14,47	333
NIDUS	2510	NICKEL	ALLOY	DURANICKEL (AND -R) (SOFT)	1,14,47	333
NID20	2514	NICKEL	ALLOY	DURIMET 20 (CAST)	1	303333
NIHSA	2516	NICKEL	ALLOY	HASTELLOY A (ANNEALED)	1,14,20	20332
NIHSB	2518	NICKEL	ALLOY	HASTELLOY B	1,14,20,41,47	300
NIHSC	2520	NICKEL	ALLOY	HASTELLOY C	1,14,41,47	302
NIHSD	2522	NICKEL	ALLOY	HASTELLOY D	1,14,41,47	333
NIHSN	2523	NICKEL	ALLOY	HASTELLOY N AND INOR-8	20,41	330
NIHSR	2524	NICKEL	ALLOY	HASTELLOY R-235	1,14,20	300
NIHSX	2526	NICKEL	ALLOY	HASTELLOY X	1,14,20,41	330
NIH80	2528	NICKEL	ALLOY	HY MU 80	14	200333
NILLG	2530	NICKEL	ALLOY	ILLIUM G	1,14,20,41,47	2002333
NILLR	2532	NICKEL	ALLOY	ILLIUM R	1,14,41,47	2000333
NINCL	2534	NICKEL	ALLOY	INCOLOY	1,14	323
NINC9	2536	NICKEL	ALLOY	INCOLOY 901	1,20	3310
NINCN	2540	NICKEL	ALLOY	INCONEL (CAST)	1,14,20,41,47	302
NINCA	2538	NICKEL	ALLOY	INCONEL (WROUGHT, ANNLD)	1,14,20,41,47	302
NINCW	2542	NICKEL	ALLOY	INCONEL W	1	333
NINCX	2544	NICKEL	ALLOY	INCONEL X AND X-750	1,20,41	301
NINC6	2545	NICKEL	ALLOY	INCONEL 600 (ANNEALED)	1,47	2003330
NINC7	2546	NICKEL	ALLOY	INCONEL 700	1,14,20,41	330
NINC2	2543	NICKEL	ALLOY	INCONEL 702 (ANNEALED)	1,20	332C
NIN7C	2547	NICKEL	ALLOY	INCONEL 713C (CAST)	1,20	200320
NINVR	2501	NICKEL	ALLOY	INVAR (HOT-ROLLED OR FRGD)	6,41,47	20355
COK42	2632	NICKEL	ALLOY	K-42B	14	3323332
NIMOC	2551	NICKEL	ALLOY	MONEL (CAST)	1,41	200333
NIMOU	2549	NICKEL	ALLOY	MONEL (COLD-DRAWN)	1,14,20,41	333
NIMON	2548	NICKEL	ALLOY	MONEL (HOT-ROLLED)	1,14,41,44,47	0
NIMLH	2552	NICKEL	ALLOY	MONEL, H (AS CAST)	1,41	220333
NIMHU	2555	NICKEL	ALLOY	MONEL, H (CAST, VAR COMP)	1,20,41	220333
NIMLK	2554	NICKEL	ALLOY	MONEL, K (ANNEALED)	1,14,20,41,44	200300
NIMKR	2556	NICKEL	ALLOY	MONEL, KR (ANNEALED)	1,14	200333
NIMLR	2558	NICKEL	ALLOY	MONEL, R (HOT-ROLLED)	1,20,41	200330
NIMLS	2560	NICKEL	ALLOY	MONEL, S (CAST, ALL COND)	1,14,20	200330
NIMOW	2553	NICKEL	ALLOY	MONEL WELDABLE ALLOY	1	200333
NIM4H	2557	NICKEL	ALLOY	MONEL 400	47	205533
NIML4	2550	NICKEL	ALLOY	MONEL 403 (HOT-ROLLED)	1	203333
COMTL	2636	NICKEL	ALLOY	MULTIMET (N-155) (LOW C)	1,14,20,41	330

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
COMTM	2634	NICKEL ALLOY MULTINET (N-155) (WROUGHT)	1,14,20,41	330
NIM25	2561	NICKEL ALLOY N-252 (GE J-1500)	20	3310
NICHR	2562	NICKEL ALLOY NICHROME V (NI80, CR 20)	1,47	23302
NINND	2564	NICKEL ALLOY HIMONIC DS	47	303333
NIM10	2577	NICKEL ALLOY NIMONIC 100	20	3303330
NIM15	2575	NICKEL ALLOY NIMONIC 105	47	3330
NIM75	2566	NICKEL ALLOY NIMONIC 75	1,14,20	330
NIM80	2568	NICKEL ALLOY NIMONIC 80	1,14,20,41,47	330
NIM8A	2570	NICKEL ALLOY NIMONIC 80A	1,20,41,47	330
NIM90	2572	NICKEL ALLOY NIMONIC 90	1,14,20,41	330
NIM95	2573	NICKEL ALLOY NIMONIC 95	20,47	303330
NIPER	2574	NICKEL ALLOY PERMANICKEL	1,14	335
NIREN	2576	NICKEL ALLOY RENE 41	20,41	303320
NIUDM	2578	NICKEL ALLOY UDIMET 500 (WROUGHT)	1,20,41	300330
NIWSP	2580	NICKEL ALLOY WASPALLOY	1,20,41	300330
NI330	2582	NICKEL ALLOY 330 (NI99.55) (ANNEALED)	1	335
CNLOX	4218	NICKEL COBALT OXIDE (46NIO.46COO.BLIO)	20,27	3305555
NIOLI	4222	NICKEL LITHIUM OXIDE (95NIO.5LIO)	20,27	3305550
NIOPL	4060	NICKEL OXIDE (NIO) (POLYXTAL, 68-74 D)	20,27,32,47,48,51	0
NIOPX	4058	NICKEL OXIDE (NIO) (POLYXTAL, 88-100 D)	20,27,32,47,48,51	0
NIO	4056	NICKEL OXIDE (NIO) (SINGLE XTAL)	20,27,32,47,48,51	5
CUNIS	2185	NICKEL SILVER (CU BAL, NI 10-20)	1,2,6,41,47	2222555
NIZNF	4226	NICKEL ZINC FERRITE (NI(ZN)O.FE2O3)	20	2025502
NB	1246	NIOBIUM	20,27,32,43,44,42	0
NBTAW	3681	NIOBIUM ALLOY (NB61, TA28, W10, ZR0.5)	20	3300
NBWMO	3684	NIOBIUM ALLOY (NB80, W15, MOS)	20	3300
NBTIZ	3682	NIOBIUM ALLOY (NBBS, TI10, ZR5)	20	3300
NBTA5	3683	NIOBIUM ALLOY (NB95, TA5)	20	3303330
NBBED	4714	NIOBIUM BERYLLIDE (NBBE12) (HP, 93-97D)	20,48	2000500
NBBE8	4716	NIOBIUM BERYLLIDE (NB2BE17)	20,48	300530
NBB2	4620	NIOBIUM BORIDE (NBB2) (PRESSED, SNTRD)	20,27,48	2500
NBC	4522	NIOBIUM CARBIDE (NBC)	20,27,48,54	20502
NBN	4434	NIOBIUM NITRIDE (NBN)	20,47,48	505
NB2N	4436	NIOBIUM NITRIDE (NB2N)	20,47,48	300535
NBSI2	4782	NIOBIUM SILICIDE (NBSI2)	20,27	300535
NOGAS	4061	NITRIC OXIDE (NO) (GAS)	1,27	500
NITRR	7141	NITRILE RUBBER	6,39	5555
NITRC	7508	NITROCELLULOSE H.E. (12.7 N)	8	305555
N-GAS	1531	NITROGEN (GAS)	27,2,42,1	0
NO	1467	NOBELIUM		4445555
NYLON	7151	NYLON 6, 11, 66, 610 (POLYCAPROLACTAM)	1,6,38	500
NYLGF	7156	NYLON, GLASS FILLED		20255
OBSID	5030	OBSIDIAN GLASS	27,51	305530
ORTHC	4328	ORTHOCLASE (K20.AL2O3.6SIO2) (CRYSTAL)	27,51	325555
OS	1114	OSMIUM	27,20,42,41	0
O-GAS	1536	OXYGEN (GAS)	27,42,1,2	0
PD	1236	PALLADIUM	20,41,42,27,43	0
E9011	7509	PBX-9011 H.E. (HMX 90, ESTANE 10)	8	555
E9404	7510	PBX-9404-03 H.E. (HMX 94, NC 3, BIND 3)	8,44	511
PETN	7511	PETN H.E.	8	555
PHCAF	7172	PHENOLIC, CAST, ASBESTOS FILLER	6,38	2555
PHCNF	7171	PHENOLIC, CAST, NO FILLER	6,38	222555
PHMHD	7177	PHENOLIC, MOLDED (HIGH DENSITY, K)	6	22555
PHMLD	7176	PHENOLIC, MOLDED (LOW DENSITY, K)	6	22555
PHRPD	7181	PHENOLIC RESIN, PRESSED, TYPES 40, 50	20	305555
PFPPF	7161	PHENOL-FORMALDEHYDE + PHENOL-FURFURAL	20	2225555
PHNXY	7191	PHENOXY	6,38	2000555
P	1321	PHOSPHORUS (WHITE)	27	40005
PLASL	4963	PLASTER, BUILDING (MOLDED, DRY)	1,27	555
PLAMT	7201	PLASTIC LAMINATE, VARIOUS TYPES	6,20	2222555
GLASP	5034	PLATE GLASS	20,51	305532
PT	1116	PLATINUM	20,1,32,27,41,43,42	0
PU	1121	PLUTONIUM	20,31,3,27,43,42,41,	0

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
PU	1121	PLUTONIUM	+36,44	0
PUSTD	3701	PLUTONIUM ALLOY (DELTA PHASE)	20,27,31,44	200
PUC	4524	PLUTONIUM CARBIDE (PUC) (ARCM OR CAST)	20,48,53,54	2025502
PUN	4438	PLUTONIUM NITRIDE (PUN)	27,48,53	300530
PUP	4762	PLUTONIUM PHOSPHIDE (PUP) (90 PC DENSE)	53	500
PUS	4747	PLUTONIUM SULFIDE	53	550
UPUOX	4242	PLUTONIUM URANIUM OXIDE (PUO2.4UO2) SEE	53	303530
PO	1473	POLONIUM	20,27	20005
PALLO	7211	POLYALLOMER	6,38	22555
PCARB	7221	POLYCARBONATE, VARIOUS FILLERS	6,38	2022555
PESTG	7231	POLYESTER, GLASS FIBER REINFORCED, TAC	20	2555
PETHH	7243	POLYETHYLENE, HIGH DENSITY	6,38	2555
PETHL	7241	POLYETHYLENE, LOW DENSITY	6,38	2555
PETHM	7242	POLYETHYLENE, MEDIUM DENSITY	6,38	2555
PIMHF	7251	POLYIMIDE, H-FILM, KAPTON	6,40	555
MEMAC	7131	POLYMETHYL METHACRYLATE	1,38	25550
PPROC	7262	POLYPROPYLENE, COPOLYMER	6,38	20555
PPROP	7263	POLYPROPYLENE, FILLED	6	20555
PPROM	7261	POLYPROPYLENE, HOPLIN	6,38	555
PSTYF	7271	POLYSTYRENE, FOAMED-IN-PLACE, RIGID	6	2020555
PSTYG	7276	POLYSTYRENE, GENERAL PURPOSE	6,20,38	2550
PSTYM	7281	POLYSTYRENE, MODIFIED	6	22555
PSTYP	7286	POLYSTYRENE, PREFOAMED, RIGID, DOW Q103	6,20	2002550
PSTYL	7287	POLYSTYRENE FOAM (D = 0.038) (1 ATM)	1	20552
PSTYV	7289	POLYSTYRENE FOAM (D = 0.046) (VACUUM)	1	20552
PSTYH	7288	POLYSTYRENE FOAM (D = 0.046) (1 ATM)	1	20552
PSULF	7291	POLYSULFONE	5,6,38	555
PURRF	7302	POLYURETHANE FOAMED-IN-PLACE, RIGID	6,20	2320555
PUREF	7301	POLYURETHANE FOAM, FLEXIBLE	6,20,38	2020555
PURER	7311	POLYURETHANE RUBBER L-100	6,38	20555
PVINA	7321	POLYVINYL ALCOHOL	6	5555
PVINB	7331	POLYVINYL BUTYRAL	6,38	555
PVINC	7341	POLYVINYL CARBAZOLE	20	5555
PVCAF	7356	POLYVINYL CHLORIDE ACYTATE, FLEXIBLE	38	2225555
PVCAR	7357	POLYVINYL CHLORIDE ACYTATE, RIGID	38	2225555
PVCFL	7351	POLYVINYL CHLORIDE, FLEXIBLE	6,38,20	2222555
PVCRD	7352	POLYVINYL CHLORIDE, RIGID	6,38,20	2222555
PVTPX	7371	POLYVINYL TPX-R	5,38	5555
PVDCM	7362	POLYVINYLIDENE CHLORIDE	6,38	22555
PVDCF	7361	POLYVINYLIDENE CHLORIDE FILM	6,38	555
PVDCK	7366	POLYVINYLIDENE FLUORIDE (KYNAR)	5.38	5555
PORAL	5281	PORCELAIN, HIGH ALUMINA	20	3005500
PORZR	5282	PORCELAIN, HIGH ZIRCON	20	305535
PORMT	5283	PORCELAIN, MAGNESIUM TITANATE	20,48	305530
PORCE	5286	PORCELAIN, ORDINARY	1,20	3325535
PORBE	5280	PORCELAIN 4811 (BEO)	20	305530
PORC5	5284	PORCELAIN 576	20	325535
K	1226	POTASSIUM	1,27,28,42,41,43,51	0
KLIQ	1227	POTASSIUM (LIQUID)	14,27,28,43,47,51	22
KALUM	4972	POTASSIUM ALUM (K2SO4.AL2(SO4)3.24H2O)	1,27	550
ORTHC	4328	POTASSIUM ALUMINUM SILICATE (ORTHOCLAS)	27,51	325555
KBR	4828	POTASSIUM BROMIDE (KBR) (PRSD 8 KB)	1,27,47	210
KBRX	4830	POTASSIUM BROMIDE (KBR) (SINGLE XTAL)	1,27,47,51	210
KBR50	4833	POTASSIUM BROMIDE (KBR 50, KCL 50)	1,27,47,51	3303330
KBR75	4832	POTASSIUM BROMIDE (KBR 75, KCL 25)	1,27,47,51	3303330
KBR90	4831	POTASSIUM BROMIDE (KBR 90, KCL 10)	1,27,47,51	3303330
KCL	4836	POTASSIUM CHLORIDE (KCL) (PRSD 8 KB)	1,27,47,51	10
KCLSY	4837	POTASSIUM CHLORIDE (KCL) (SYLVITE XTAL)	1,27,47,51	10
KBR50	4833	POTASSIUM CHLORIDE (KCL 50, KBR 50)	1,27,47,51	3303330
KCLSO	4838	POTASSIUM CHLORIDE (KCL 50, NACL 50)	1,27,47,51	303330
KCL75	4834	POTASSIUM CHLORIDE (KCL 75, KBR 25)	1,27,47,51	3303330
KCL90	4835	POTASSIUM CHLORIDE (KCL 90, KBR 10)	1,27,47,51	3303330
KCROM	4233	POTASSIUM CHROMATE (K2O.2CRO3) (M AXIS)	1,27	300050

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
KCROX	4232	POTASSIUM CHROMATE (K2O.2CRO3) (S AXIS)	1,27	300050
KF	4840	POTASSIUM FLUORIDE (KF) (PRSD 8 KB)	1,27,47	10
KI	4842	POTASSIUM IODIDE (KI) (PRSD 8 KB)	1,27,47	10
KFECN	4974	POTASSIUM FERROCYANIDE (K4FE(CN)6.3H2O)	1,27	550
KNO3	4976	POTASSIUM NITRATE (KNO3) (PR 8000 KB)	1,27	0
PR	1361	PRASEODYMIUM	41,27,20,42	205
PRB6	4628	PRASEODYMIUM BORIDE (PRB6) (95 PC DENS)	48	305555
PM	1421	PROMETHIUM	20,27,42	3332035
PA	1476	PROTACTINIUM	20,27	30005
PYREX	5016	PYREX GLASS	2,20,27,51	500
PYRC6	5018	PYROCERAM 9606 CERAMIC GLASS	20	5500
PYRC8	5019	PYROCERAM 9608 CERAMIC GLASS	20	5500
PYROP	4990	PYROPHYLLITE (PARALLEL TO BEDDING)	27,51	2305530
PYRON	4991	PYROPHYLLITE (PERPEND. TO BEDDING)	27,51	2305530
SIO2A	4066	QUARTZ CRYSTAL, A AXIS (SIO2)	1,2,20,27,37,47,51	2200
SIO2C	4064	QUARTZ CRYSTAL, C AXIS (SIO2)	1,2,20,27,37,47,51	2200
SANQF	6032	QUARTZ FLOUR, FINE (DRY)	51	305535
SANQN	6033	QUARTZ FLOUR, FINE (H2O 21 PC)	51	305555
SIO2	4062	QUARTZ GLASS, FUSED (SIO2)	1,20,27,47,51	2200
SANQP	6039	QUARTZ POWDER, COARSE (H2O 24)	51	305555
SANDY	6034	QUARTZ SAND (DRY) (AV PROP) (SEE REF)	51	2325555
SANDQ	6036	QUARTZ SAND (WET) (H2O 4-23) (AV PROP)	51	2325555
RA	1479	RADIUM	27	2040025
RNGAS	1561	RADON GAS	27	40505
RATTN	8211	RATTAN	44	25555
RE	1251	RHENIUM	27,43,41,42	2
RH	1281	RHODIUM	20,27,43,42,41	0
ROCKS	6022	ROCK OR STONE (AVERAGE PROPERTIES)	1,46	3335555
STBRC	7431	RUBBER, BUNA, WITH CARBON BLACK	6,20,39	200555
BUTYR	7031	RUBBER, BUTYL	6,39	555
RUBBD	7382	RUBBER, DIELECTRIC MIX	6,20	550
RUBBK	7383	RUBBER, HIGH K	6,20	550
RUBBN	7381	RUBBER, NATURAL	6	555
RUBBF	7386	RUBBER, NATURAL, FOAM	6,20	320555
NEOPR	7136	RUBBER, NEOPRENE	6.38,39	555
NITRR	7141	RUBBER, NITRILE		5555
PURER	7311	RUBBER, POLYURETHANE ELASTOMER L-100	6,38	20555
RB	1291	RUBIDIUM	1,27,28,42,43,47	40024
RBLIQ	1292	RUBIDIUM (LIQUID)	1,14,27,28,43,47	20022
RBCL	4844	RUBIDIUM CHLORIDE (RBCL) (PRSD 8 KB)	1,27,47	10
RBI	4846	RUBIDIUM IODIDE (RBI) (PRSD 8 KB)	1,27,47	10
RU	1301	RUTHENIUM	20,42,1,27	202020
SM	1371	SAMARIUM	20,27,41,42,43	30205
SMB6	4632	SAMARIUM BORIDE (SMB6) (96.8 PC DENSE)	48	300555
SMS	4748	SAMARIUM SULFIDE (SMS)	20,48	300550
SANDY	6034	SAND (DRY) (D = 1.36 TO 1.84) (AV PROP)	51	2325555
SANDN	6038	SAND, NORTHWAY (H2O 4-10) (AV PROP)	51	2325555
SANDQ	6036	SAND, QUARTZ (WET) (H2O 4-23) (AV PROP)	2,51	2325555
SANDM	6026	SANDSTONE (AV PROP) (SEE REF 51)	2,27,46,51	25555
SANDS	6024	SANDSTONE (HIGH DENSITY)	2,27,46,51	25555
SANDL	6025	SANDSTONE (LOW DENSITY)	2,27,46,51	25555
ALOXD	4012	SAPPHIRE (AL2O3) (POLYXTAL) (100 PC D)	1,20,27,32,47,48,51	200
ALOXS	4016	SAPPHIRE (AL2O3) (SINGLE XTAL) (AV DIR)	1,20,27,32,47,48,51	202
SC	1341	SCANDIUM	20,27,41,43	30005
SE	1326	SELENIUM (GREY)	27,43,42,1	20005
SHALE	6028	SHALE (AV PROP) (SEE REF 51)	51	2225555
BRISI	5294	SILICA BRICK, HARD FIRED (SIO2 94-95)	20	330
SIO2	4062	SILICA GLASS, FUSED OR VITROUS	1,20,27,47,51	2200
BRISL	5296	SILICEOUS BRICK (SIO2 89, AL2O3 9)(25P)	47	335
SI	1256	SILICON	20,1,27,41,43,42,48	0
SIB4	4614	SILICON BORIDE (SIB4)	27,48	300530
SICAL	4539	SILICON CARBIDE (SIC) (BRICK AL2O3 1.7)	20,27,32,47	2500
SICCF	4535	SILICON CARBIDE (SIC) (CARBOFRAX BRICK)	20,27,32,47	2500

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
SICFV	4542	SILICON CARBIDE (SIC) (FOAM, IN VACUUM)	20,32,47	2500
SICFB	4536	SILICON CARBIDE (SIC) (FRIT BND BRICK)	20,27,32,47	2500
SICKT	4534	SILICON CARBIDE (SIC) (KT GRADE)	20,27,32,47,48	2500
SICNB	4541	SILICON CARBIDE (SIC) (NITRIDE BONDED)	20,32,47,48	2503
SICPA	4543	SILICON CARBIDE (SIC) (POWDER, IN AIR)	20,32,47	2500
SICPH	4544	SILICON CARBIDE (SIC) (POWDER, IN HE)	20,32,47	2500
SICRD	4540	SILICON CARBIDE (SIC) (REFRACTORY D-30)	20,27,32,47	2500
SICRP	4538	SILICON CARBIDE (SIC) (REXTAL, 65-70D)	1,20,27,32,47	2500
SICRX	4537	SILICON CARBIDE (SIC) (REXTAL, 80-100D)	20,27,32,47,48	2500
SICX1	4531	SILICON CARBIDE (SIC) (SINGLE XTAL)	20,27,32,47.	22500
SICX2	4532	SILICON CARBIDE (SIC) (SINGLE XTAL)	20,32,47,48	22505
SICSB	4533	SILICON CARBIDE (SIC) (SLF BND, HE ATM)	20,27,32,47	2500
SICSI	5424	SILICON CARBIDE + SI (SIC 76, SI 24)	20	3305550
SI3NL	4444	SILICON NITRIDE (SI3N4) (70 PC DENSE)	20,27,47,48	20502
SI3N4	4442	SILICON NITRIDE (SI3N4) (85 PC DENSE)	20,27,47,48	20502
SIO2F	4068	SILICON OXIDE (SIO2) (FOAM, 1 ATM AIR)	1,20,27,47	2200
SIO2	4062	SILICON OXIDE (SIO2) (FUSED QUARTZ)	1,20,27,47,51	2200
SIO2A	4066	SILICON OXIDE (SIO2) (QUARTZ, A AXIS)	1,2,20,27,37,47,51	2200
SIO2C	4064	SILICON OXIDE (SIO2) (QUARTZ, C AXIS)	1,2,20,27,37,47,51	2200
SITE	4936	SILICON TELLURIDE (SITE)	20	305555
SILFF	7401	SILICONE FOAM, FLEXIBLE (LRL)	6,20	2022555
SILFR	7402	SILICONE FOAM, RIGID, VARIOUS	6,20	2022555
SILRH	7423	SILICONE RUBBER, HIGH K (SEE REF 5)	5,6,41,44	220555
SILRL	7421	SILICONE RUBBER, LOW K (SEE REF 5)	5,6,41,44	220555
SILRM	7422	SILICONE RUBBER, MEDIUM K (SEE REF 5)	5,6,41,44	220555
SILRl	7424	SILICONE RUBBER, RTV 521 AND 093-009	5,6,41,44	220555
SILMD	7411	SILICONE, MOLDED, VARIOUS FILLERS	6,20,38	2222555
BRISK	5298	SILLIMANITE BRICK (22 PC POROSITY)	20	300530
AG	1126	SILVER	20,6,1,27,2,32,42,47,	0
AG	1126	SILVER	+14,43	0
AGLIQ	1127	SILVER (LIQUID)	27,32,42,43	30005
SILVA	3751	SILVER ALLOYS, STERLING AND COIN	1	325
SNTE2	4940	SILVER ANTIMONY TELLURIDE (SNTE 25 PC)	20	3305555
AGCL	4848	SILVER CHLORIDE (AGCL) (SINGLE CRYSTAL)	27,47,51	15
AG2SE	4942	SILVER SELENIDE (AG2SE)	20,27,47	20502
SLATE	6030	SLATE (AV PROP) (SEE REF 51)	51	2225555
GLASC	5032	SODA-LIME GLASS	20,27	3325532
NA	1221	SODIUM	2,1,27,28,41,42,43,51	0
NALIQ	1222	SODIUM (LIQUID)	1,14,27,28,42,43,47,	1
NALIQ	1222	SODIUM (LIQUID)	+51	1
ANALC	4329	SODIUM ALUMINUM SILICATE HYDRATE (XTAL)	27,51	305555
NABR	4850	SODIUM BROMIDE (NABR) (PRSD 8 KB)	1,27,47	10
NACLO	4978	SODIUM CHLORATE (NACLO3) (CRYSTAL)	1,27,51	505
NACL	4852	SODIUM CHLORIDE (NACL) (CLEAR CRYSTAL)	1,27,47,51,52	10
NACLR	4854	SODIUM CHLORIDE (NACL) (OPAQUE, IMPURE)	1,27,47,51,52	10
KCL50	4838	SODIUM CHLORIDE (NACL 50, KCL 50)	1,27,47,51	303330
NAF	4856	SODIUM FLUORIDE (NAF) (PRSD 8 KB)	1,27,47	10
SOILS	6046	SOIL (AV PROPS) (SEE REFS)	2,46,51	2225555
SOIL3	6044	SOIL, CLAY (WET)	51	2225555
SANQF	6032	SOIL, FINE QUARTZ FLOUR (DRY)	51	305535
SANQW	6033	SOIL, FINE QUARTZ FLOUR (H2O 21 PC)	51	305555
SOILl	6042	SOIL, LOAM (DRY) (AV PROP) (SEE REFS)	2,51	2225555
SOIL2	6043	SOIL, LOAM (H2O 4-27 PC) (AV, SEE REFS)	2,51	2225555
SOILM	6048	SOIL, MARS SURFACE (SEE UCRL-50309)		5555
SOLID	6050	SOIL, SANDY DRY	46	5555
SOILW	6052	SOIL, SANDY (H2O 8)	46	5555
ART7	9007	SOLID, REGULATING (D=C=X=TM=1,HM=10**9)		0
ART6	9006	SOLID, UNIT LATENT HEAT (D=C=X=TM=HM=1		0
ART3	9003	SOLID, UNIT PROPERTY (D=C=X=1)		0
SNOWF	8231	SNOW, FRESH	2,27	55
SNOWP	8232	SNOW, PACKED	27,46	55
SOLD1	3611	SOLDER (PB37.5, SN37.5, IN25)	36	3355
SOLD2	3612	SOLDER (PB39.2, SN60.8)	34,41	5

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
SOLD4	3613	SOLDER (PB50, SN50)	1,41	50
SOLD3	3614	SOLDER (PB60, SN40)	1,41	55
SPINX	4224	SPINEL (MGO.AL203) (SINGLE CRYSTAL)	20,27,47,48,51	500
SPINL	4225	SPINEL (MGO.AL203) (100 PC DENSE POLYX)	20,27,47,48,51	500
H2OGS	4113	STEAM (H2O) (GAS) (SATD)	1,2,14,27,46	205550
H2OGL	4112	STEAM (H2O) (GAS) (1 ATM)	1,2,14,27,46	200050
STEAT	4326	STEATITE (MGO.SIO2) (COMMERCIAL GRADES)	1,20,27,51	505
STALL	3101	STEEL, ALLOY AND MILD (4130, 4340)	6,41,46	3302
STALC	3106	STEEL, ALLOY, CAST	6	3355
STCAR	3116	STEEL, CARBON, TYPE 1020 (0.2 - 0.6 C)	1,20,37	300
STCFC	3111	STEEL, FREE CUTTING, EUTECTOID	2,4,6,20,41	300
STHSP	3121	STEEL, HIGH SPEED (M1, M10, M-2, T1)	20	3303355
COMTL	2636	STEEL, MULTIMET (N-155) (LOW C)	1,14,20,41	330
COMTM	2634	STEEL, MULTIMET (N-155) (WROUGHT)	1,14,20,41	330
STST4	3141	STEEL, STAINLESS (CR 12-13, NI 0-3)	1,2,4,14,20	300
STST3	3131	STEEL, STAINLESS (CR 16-26, NI 8-36)	1,2,4,6,20,37	2023333
SSC08	3208	STEEL, STAINLESS CA15, CA40 (CAST)	1	355
SSC09	3209	STEEL, STAINLESS CB30, CC50 (CAST)	1	355
SSC13	3213	STEEL, STAINLESS CK, CH, HI (CAST)	1	355
SSC12	3212	STEEL, STAINLESS CF (CAST)	1	355
SSC04	3204	STEEL, STAINLESS CN-7M (CAST)	1	355
SSC06	3206	STEEL, STAINLESS HA (CAST)	1	355
SSC07	3207	STEEL, STAINLESS HC, HD (CAST)	1	355
SSC11	3211	STEEL, STAINLESS HE (CAST)	1	355
SSC05	3205	STEEL, STAINLESS HF (CAST)	1	355
SSC10	3210	STEEL, STAINLESS HH, HL, HX (CAST)	1	355
SSC03	3203	STEEL, STAINLESS HT (CAST)	1	355
SSC02	3202	STEEL, STAINLESS HU (CAST)	1	355
SSC01	3201	STEEL, STAINLESS HW (CAST)	1	355
STST3	3131	STEEL, STAINLESS SERIES 300	1.2,4,6,20,37	2023333
STST4	3141	STEEL, STAINLESS SERIES 400	1,2,4,20	300
STPH4	3171	STEEL, STAINLESS 17-4PH	20,14	23302
STPH7	3176	STEEL, STAINLESS 17-7PH	20,14	2003300
ST199	3186	STEEL, STAINLESS 19-9DL	41	2300
STST2	3126	STEEL, STAINLESS 201 AND 202	1	3355
ST304	3133	STEEL, STAINLESS 304	1,4,37	300
ST347	3135	STEEL, STAINLESS 321 AND 347	1,2,4,20,37,14	300
STS4A	3146	STEEL, STAINLESS 430, 430F, AND 431	1,2,4,6,20,14	302
STS4B	3151	STEEL, STAINLESS 446	1,14,20	302
STST5	3161	STEEL, STAINLESS 501 AND 502	2,4	2303330
WCC02	5453	STEEL, TOOL, TUNGSTEN CARBIDE CA2	20	3300530
WCC01	5452	STEEL, TOOL, TUNGSTEN CARBIDE CA4	20	3300530
STUHS	3181	STEEL, ULTRA HIGH STRENGTH TYPE 300-M	1	355
SR	1296	STRONTIUM	20,28,42,27	240024
SRSO4	4980	STRONTIUM SULFATE (SRSO4) (CRYSTAL)	1,27,51	550
SRTIT	4234	STRONTIUM TITANATE (SRO.TIO2) (100 D)	20,48	500
SRTIP	4235	STRONTIUM TITANATE (SRO.TIO2) (80 D)	20,48	500
STCO1	5426	STRONTIUM TITANATE + CO (CO 10)	20	3300535
STCO2	5427	STRONTIUM TITANATE + CO (CO 20)	20	3300535
STCO3	5428	STRONTIUM TITANATE + CO (CO 30)	20	3300535
STCO4	5429	STRONTIUM TITANATE + CO (CO 40)	20	3300535
STBRC	7431	STYRENE-BUTADIENE RUBBER + CARBON BLK	6,20,39	200555
S	1331	SULFUR	27,1,42,30,2,41	10001
SO2	4071	SULFUR DIOXIDE (SO2) (GAS)	1	5500
TALC	4994	TALC	20,27,51	5505
TA	1161	TANTALUM	20,1,27,32,37,43	20002
TANBV	3784	TANTALUM ALLOY (TA62, MB30, V7.5)	20	3300
TAWHF	3785	TANTALUM ALLOY (TA89, W9, HF2)	20	300
TA10W	3781	TANTALUM ALLOY (TA90, W10)	20,36	500
TACUZ	3783	TANTALUM ALLOY (TA98, CUO.7, ZRO.7)	20	3300
TANBL	3782	TANTALUM ALLOY (TA99.5, NBO.5)	20	3303330
TASB	4944	TANTALUM ANTIMONIDE (TASB)	20	3305552
TABED	4718	TANTALUM BERYLLIDE (TABE12) (HP)	20,48	

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
TABE8	4720	TANTALUM BERYLLIDE (TA2BE17)	20,48	500
TAB	4642	TANTALUM BORIDE (TAB) (PSD, SNTR, 85D)	20,27,47,48	500
TAB2	4644	TANTALUM BORIDE (TAB2)	20,27,47,48	2020502
TAC	4562	TANTALUM CARBIDE (TAC)	20,27,48,54	2020502
WCCO5	5456	TANTALUM CARBIDE + WC CERMET K601	49	300530
TAN	4446	TANTALUM NITRIDE (TAN)	20,27,48	2220525
TA2N	4448	TANTALUM NITRIDE (TA2N)	20,27,48	2220522
TASI2	4784	TANTALUM SILICIDE (TASI2)	20,27	300535
TC	1164	TECHNICIUM	27,42	40025
FCTFE	7111	TEFLON	1,6,20,38	2500
FCTFR	7112	TEFLON, REINFORCED	20	2500
TE	1488	TELLURIUM	27,20,42,41,1	20002
TB	1386	TERBIUM	20,27,41,42,43	30205
TBB6	4638	TERBIUM BORIDE (TBB6) (94.3 PC DENSE)	48	305555
TETRL	7512	TETRYL H.E.	8	300555
TL	1336	THALLIUM	41,27,1,42,47	0
TLLIQ	1337	THALLIUM (LIQUID)	1,14,27,42,43,47	55
TH	1166	THORIUM	20,32,34,27,37,43,44,	2
TH	1166	THORIUM	+41	2
THB4	4646	THORIUM BORIDE (THB4)	20,27,48	2002502
THB6	4648	THORIUM BORIDE (THB6)	20,27,48	2000535
THC	4564	THORIUM CARBIDE (THC) (80 PC DENSE)	20,27.48,54	300530
THC2	4565	THORIUM CARBIDE (THC2) (69 PC DENSE)	20,27,48,54	300530
THO2	4072	THORIUM OXIDE (THO2) (96-100 PC DENSE)	1,20,27,47,48,51	2002300
TM	1406	THULIUM	20,27,41,42,43	30005
SN	1171	TIN	2,1,27,32,42,41,47	0
SNLIQ	1172	TIN (LIQUID)	1,14,27,32,41,42,43	0
SOLD1	3611	TIN ALLOY (SN37.5, PB37.5, IN25)	36	3355
SOLD3	3614	TIN ALLOY (SN40, PB60) (SOLDER)	1,41	55
SOLD4	3613	TIN ALLOY (SN50, PB50) (SOLDER)	1,41	50
SOLD2	3612	TIN ALLOY (SN60.8, PB39.2) (SOLDER)	34,41	5
SNO2	4074	TIN OXIDE (SNO2) (93-95 PC DENSE)	1,20,27,47,48	2300
SNTE2	4940	TIN TELLURIDE (SNTE 25, AGSBTE2 75)	20	3305555
SNTE6	4939	TIN TELLURIDE (SNTE 60, AGSBTE2 40)	20	3305555
SNTE8	4938	TIN TELLURIDE (SNTE 80, AGSBTE2 20)	20	3305555
TI	1176	TITANIUM	20,27,32,37,42,43,41,	20002
TI	1176	TITANIUM	+1	20002
TIAL1	2712	TITANIUM ALLOY (TI BAL, AL2, MN2)	20	3303330
TIAL3	2716	TITANIUM ALLOY (TI BAL, AL4, CU2, ZR2)	20	3303330
TIAL7	2730	TITANIUM ALLOY (TI BAL, AL4, CU4, SN2)	20	3303330
TIA11	2732	TITANIUM ALLOY (TI BAL, AL4, MO3, V1)	20	3233323
TIAL5	2720	TITANIUM ALLOY (TI BAL, AL4, V1, MOO.6)	20	3303330
TIAL2	2714	TITANIUM ALLOY (TI BAL, AL4, V2, MO1)	20	3303330
TIAL4	2718	TITANIUM ALLOY (TI BAL, AL4, V3, MO1.5)	20	3303330
TIAL6	2728	TITANIUM ALLOY (TI BAL, AL4, ZR3.5)	20	3303330
TIA12	2736	TITANIUM ALLOY (TI BAL, AL5, SN2.5)	41,45	303330
T1555	2738	TITANIUM ALLOY (TI BAL, AL5, SN5, ZR5)	45	303330
TIA13	2740	TITANIUM ALLOY (TI BAL, AL6, V4)	20,37,45	300
TIA15	2742	TITANIUM ALLOY (TI BAL, ALB, MO1, V1)	45	3330
TIA14	2746	TITANIUM ALLOY (TI BAL, CR3.4, MO2.1)	20	3303330
TIAL8	2752	TITANIUM ALLOY (TI BAL, SN4.8, AL4.5)	20	3303330
TIAL9	2754	TITANIUM ALLOY (TI BAL, SN5.5, AL2)	20	3303330
TIV14	2768	TITANIUM ALLOY (TI BAL, V14, CR10, AL4)	20	3003300
TIV15	2766	TITANIUM ALLOY (TI BAL, V15, AL2.8)	20	3003300
TIA10	2770	TITANIUM ALLOY (TI BAL, ZR3, AL2)	20	3303330
TI100	2750	TITANIUM ALLOY C100M (RC130A) (MN7.9)	20	203320
TIH40	2722	TITANIUM ALLOY HYLITE 40 C130AM,RC130B	1,20,41	303330
TIH50	2724	TITANIUM ALLOY HYLITE 50 (IMI550)	20,45	303330
TIH51	2726	TITANIUM ALLOY HYLITE 51 (IM1551)	45	303330
TIH55	2760	TITANIUM ALLOY HYLITE 55 (IMI)	20,45	303330
TIH60	2758	TITANIUM ALLOY HYLITE 60 (IMI)	20,45	303330
TIH65	2756	TITANIUM ALLOY HYLITE 65 (IMI)	45	303330
TI679	2764	TITANIUM ALLOY IMI 679 (SN11, ZR5, AL2)	45	303330

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
TI680	2762	TITANIUM ALLOY IMI 680 (SN11, MO4, AL2)	45	303330
TI140	2748	TITANIUM ALLOY TI140A (FE2, CR2, MO2)	20	3303330
TI150	2744	TITANIUM ALLOY TI150A (CR2.7, FE1.4)	20	3003300
TI155	2734	TITANIUM ALLOY TI155A (AL5,FE2,CR1,MO1)	1,20	3303330
TIBED	4722	TITANIUM BERYLLIDE (TIBE12) (HP, 95D)	20,48	2500
TIB2	4650	TITANIUM BORIDE (TIB2) (HP, 95 PC DENS)	20,27,47,48	500
TIC93	4567	TITANIUM CARBIDE (TIC) (93 PC DENSE)	20,27,47,48,54	22502
TIC96	4566	TITANIUM CARBIDE (TIC) (96 PC DENSE)	20,27,47,48,54	22502
TICC2	5431	TITANIUM CARBIDE + CO (CO18, NBC,TAC15)	20	3305535
TICC1	5430	TITANIUM CARBIDE + CO (TIC 80, CO 20)	20	305535
TICC3	5432	TITANIUM CARBIDE + NBC + NI	20	3305535
TICCR	5442	TITANIUM CARBIDE + NI OR CO (AVG PROP)	48	2220555
TICC5	5434	TITANIUM CARBIDE CERMET K138A	49	305530
TICC6	5435	TITANIUM CARBIDE CERMET K151A	49	305530
TIC10	5439	TITANIUM CARBIDE CERMET K161B	20	3005500
TICC4	5433	TITANIUM CARBIDE CERMET K162B	49	305530
TICC7	5436	TITANIUM CARBIDE CERMET K163Bl	49	305530
TICC8	5437	TITANIUM CARBIDE CERMET K164B	49	305530
TICC9	5438	TITANIUM CARBIDE CERMET K165	49	305530
TINIT	4452	TITANIUM NITRIDE (TIN) (HP, 70-90 DENS)	20,27,47,48	2020502
TIO2A	4084	TITANIUM OXIDE (TIO2) (RUTILE, A AXIS)	1,20,27,32,47,48,51	20202
TIO2C	4082	TITANIUM OXIDE (TIO2) (RUTILE, C AXIS)	1,20,27,32,47,48,51	20202
TIO2	4086	TITANIUM OXIDE (TIO2) (RUTILE, 100 D)	1,20,27,32,47,48,51	200
TNT	7513	TNT H.S. (2,4,6-TRINITROTOLUENE) (CAST)	8	555
TOPAZ	4301	TOPAZ (2ALFO.SIO2) (A-AXIS)	27,51	305550
TOPAC	4302	TOPAZ (2ALFO.SIO2) (C-AXIS)	27,51	305551
TOURM	4996	TOURMALINE	27,51	305550
T-GAS	1566	TRITIUM GAS		3335533
W	1181	TUNGSTEN	20,1,32,34,27,37,42,	0
W	1181	TUNGSTEN	+41,43	0
W25RE	3801	TUNGSTEN ALLOY (W75, RE25)		3303330
WNICU	3802	TUNGSTEN ALLOY (W90, NI6, CU2-4)	20	303330
WB	4652	TUNGSTEN BORIDE (WB)	20,27,47,48	2500
WC	4568	TUNGSTEN CARBIDE (WC)	20,27,47,48,54	2500
WCC12	5463	TUNGSTEN CARBIDE CERMET CARBOLOY 883	50	300530
WCC11	5462	TUNGSTEN CARBIDE CERMET CARBOLOY 905	50	300530
WCC10	5461	TUNGSTEN CARBIDE CERMET CARBOLOY 999	50	300530
WCC02	5453	TUNGSTEN CARBIDE CERMET CA2 TOOL STEEL	20	3300530
WCC0l	5452	TUNGSTEN CARBIDE CERMET CA4 TOOL STEEL	20	3300530
WCC03	5454	TUNGSTEN CARBIDE CERMET K6 AND K96	20,49	300530
WCCOS	5456	TUNGSTEN CARBIDE CERMET K601	49	300530
WCC08	5459	TUNGSTEN CARBIDE CERMET K701	49	300530
WCC09	5460	TUNGSTEN CARBIDE CERMET K801 (WC + NI)	49	300530
WCC07	5458	TUNGSTEN CARBIDE CERMET K92	49	300530
WCC06	5457	TUNGSTEN CARBIDE CERMET K94 AND K1	49	300530
WO3	4092	TUNGSTEN OXIDE (WO3) (POLYXTAL. POROUS)	20,27,47,48	20305
WSE2	4946	TUNGSTEN SELENIDE (WSE2)	20,27	3303550
WSI2	4786	TUNGSTEN SILICIDE (WSI2) (HP, 95 DENSE)	20,27,48	2002500
WTE2	4947	TUNGSTEN TELLURIDE (WTE2)	20,27	3303555
WTIC5	5470	TUNGSTEN TITANIUM CARBIDE KM	49	300530
WCC04	5465	TUNGSTEN TITANIUM CARBIDE K2S	20,49	300530
WTIC7	5472	TUNGSTEN TITANIUM CARBIDE K21	49	300530
WTIC6	5471	TUNGSTEN TITANIUM CARBIDE K4H	49	300530
WTIC4	5469	TUNGSTEN TITANIUM CARBIDE K5H	49	300530
WTIC3	5468	TUNGSTEN TITANIUM CARBIDE K81	49	300530
WTIC2	5467	TUNGSTEN TITANIUM CARBIDE K84	49	300530
WTIC1	5466	TUNGSTEN TITANIUM CARBIDE K86, K7H, K3H	49	300530
U	1186	URANIUM	20,31,32,36,27,37,42,	0
U	1186	URANIUM	+41,43,44,34	0
MULBY	3831	URANIUM ALLOY MULBERRY	34,36,37	0
U18ZR	3833	URANIUM ALLOY (U82, ZR18)	20,34	303330
U90FS	3837	URANIUM ALLOY (U90, FS10)	20	303330
U10MO	3832	URANIUM ALLOY (U90, MO10)	20,35,36	300

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
UFSZR	3838	URANIUM ALLOY (U93, FS5, ZR2)	20	303330
UCREU	3835	URANIUM ALLOY (U94.4, CR5.6) (EUTECTIC)	20	3003300
UZR05	3840	URANIUM ALLOY (U95, ZR5)	20	30303330
UO4NB	3834	URANIUM ALLOY (U96, NB4)	20,34	303330
U97FS	3836	URANIUM ALLOY (U97, FS3)	20	303330
UZR02	3839	URANIUM ALLOY (U98.5, ZR1.5)	20	303330
UBE13	4724	URANIUM BERYLLIDE (UBE13) (SNTRD, 61D)	20,48	305550
UB4	4654	URANIUM BORIDE (UB4)	20,48	300555
UCAMC	4572	URANIUM CARBIDE (UC) (ARCM OR CAST,99D)	20,48,53,54	2502
UCAVG	4574	URANIUM CARBIDE (UC) (AVG, VAR. TYPES)	20,48,53,54	2502
UCSNT	4573	URANIUM CARBIDE (UC) (SINTERED, 90D)	20,48,53,54	2502
UC2	4576	URANIUM CARBIDE (UC2)	20,27,48,54	2000500
UN	4454	URANIUM NITRIDE (UN) (HP, 95-98 DENSE)	20,27,48,PWAC481-65	2500
UO2PX	4096	URANIUM OXIDE (UO2) (POLYXTAL, 97 DENS)	20,27,47,48,51	500
UO2	4094	URANIUM OXIDE (UO2) (SINGLE CRYSTAL)	20,27,47,48,51	500
U3O8L	4099	URANIUM OXIDE (U3O8) (PRSD AT 100 PSI)	1,20,27,47,48	2002500
U3O8H	4098	URANIUM OXIDE (U3O8) (PRSD AT 4200 PSI)	1,20,27,47,48	2002500
UO2CR	5482	URANIUM OXIDE + CR (UO2 80 VOL) (97 D)	20	4305550
UOMO3	5486	URANIUM OXIDE + MO (UO2 70 VOL) (92 D)	20	4305550
UOMO2	5485	URANIUM OXIDE + MO (UO2 80 VOL) (91 D)	20	4305550
UOMOl	5484	URANIUM OXIDE + MO (UO2 80 VOL) (94 D)	20	4305550
UO2NB	5488	URANIUM OXIDE + NB (UO2 80 VOL)	20	4305550
UO2Sl	5490	URANIUM OXIDE + ST STEEL (UO2 70 VOL)	20	4305550
UO2S2	5491	URANIUM OXIDE + ST STEEL (UO2 80 VOL)	20	4305550
UO2Zl	5492	URANIUM OXIDE + ZR (UO2 43,ZR 57)(59 P)	20	4305530
UO2Z2	5493	URANIUM OXIDE + ZR (UO2 80,ZR 20)	20	305535
UPUOX	4242	URANIUM PLUTONIUM OXIDE (4UO2.PUO2) SEE	53	303530
U3SI	4788	URANIUM SILICIDE (U3SI)	20,48	20525
URFMA	7442	UREA-FORMALDEHYDE, ALPHA CELLULOSE FLLR	38	25555
URFMN	7441	UREA-FORKALDEHYDE, MOLDED	38	25555
ART2	9002	VACUUM, CONDUCTING (D=C=0, K=10**9)		0
ART1	9001	VACUUM, INSULATING (D=C=K=0)		0
V	1286	VANADIUM	20,32,27,41,42,43	0
VBE12	4726	VANADIUM BERYLLIDE (VBE12) (85 PC DENS)	20,48	505
VB2	4656	VANADIUM BORIDE (VB2)	20,27,48	20555
VC	4578	VANADIUM CARBIDE (VC)	20,27,47,48,54	2022502
VN	4456	VANADIUM NITRIDE (VN) (PR, STRD, 100 D)	20,27,47,48	22505
VERM2	5292	VERMICULITE, EXPANDED (D = 0.19-0.25)	20	305530
VERM3	5293	VERMICULITE, EXPANDED (D = 0.3)	20	305530
VERMI	4986	VERMICULITE BRICK	20	305550
VERMP	4988	VERMICULITE INSULATING POWDER	20	305550
SIO2	4062	VITROUS SILICA GLASS	1,20,27,4.',51	2200
VYCOR	5036	VYCOR GLASS	20,27	500
H2OGS	4113	WATER (H2O) (GAS) (SATD)	1,2,14,27,46	205550
H2OGl	4112	WATER (H2O) (GAS) (1 ATM)	1,2,14,27,46	200050
WATER	4114	WATER (H2O) (LIQUID)	2,14,27,46	0
ICE	4116	WATER (H2O) (SOLID) (ICE)	27,46,51	0
D2O	4118	WATER, HEAVY (D2O) (LIQUID)	14,27	300
GLASW	5028	WINDOW GLASS, LINE	20	305530
BALS1	8516	WOOD, BALSA (ACROSS GRAIN)	1,2,27	305555
BALS2	8517	WOOD, BALSA (ACROSS GRAIN)	1,2,27	305555
CYPRS	8518	WOOD, CYPRESS (ACROSS GRAIN)	1,27	305555
MAHOG	8520	WOOD, MAHOGANY (ACROSS GRAIN)	1,27	305555
MAPLE	8522	WOOD, MAPLE (ACROSS GRAIN)	1,2,27	305555
PINEW	8526	WOOD, NORWAY PINE (ACROSS GRAIN)	1,2,27	305555
OAKLD	8525	WOOD, OAK, RED, BLACK (ACROSS GRAIN)	1,2,27	05555
OAKHD	8524	WOOD, OAK, WHITE, LIVE (ACROSS GRAIN)	1,2,27	05555
PINEL	8527	WOOD, OREGON PINE (ACROSS GRAIN)	1,2,27	305555
PINEW	8529	WOOD, PINE (WITH GRAIN)	1,2,27	305555
WOOD2	8513	WOOD, SPRUCE (ACROSS GRAIN)	46	5555
WOOD1	8512	WOOD, SPRUCE (WITH GRAIN)	46	5555
TEAK	8530	WOOD, TEAK (ACROSS GRAIN)	1,2	305555
PINEV	8528	WOOD, VIRGINIA PINE (ACROSS GRAIN)	1,2,27	305555

Table G.3. Material index (continued)

ABBR.	NO.	MATERIAL	REFERENCES	QUALITY INDEX DCKTHCK[a]
FIRWH	8532	WOOD, WHITE FIR (ACROSS GRAIN)	1,2	305555
PINEA	8534	WOOD, WHITS PINE (ACROSS GRAIN)	1,2,27	305555
XE	1541	XENON	27,42	30000
YB	1411	YTTERBIUM	20,27,41,42,43	30205
YBB6	4640	YTTERBIUM BORIDE (YBB6) (90.6 PC DENS)	48	305555
Y	1346	YTTRIUM	41,27,1,20,42,43	20005
YB6	4622	YTTRIUM BORIDE (YB6) (98.4 PC DENSE)	48	300555
Y2O3	4102	YTTRIUM OXIDE (Y2O3) (96-100 PC DENSE)	20,27,47,48	300
ZINC2	3861	ZILLOY 15	1,6,41	355
ZN	1191	ZINC	2,1,32,27,42,20,41,	0
ZN	1191	ZINC	+47,51	0
ZNLIQ	1192	ZINC (LIQUID)	14,20,27,32,43,47	0
ZINC1	3862	ZINC ALLOY ASTM B69	1,6,41,51	355
ZNSB	4948	ZINC ANTIMONIDE (ZNSB)	20,27,47	3303550
ZNFEO	4252	ZINC FERRITE (ZNO.FE2O3) (PR, FRD, VAC)	20,27	500
ZNO	4104	ZINC OXIDE (ZNO) (PRSD, FIRED, 100 D)	1,20,27,47,48,51	2000
ZNS	4750	ZINC SULFIDE (ZNS) (CUBIC CRYSTAL)	1,27,47,48,51	310
ZINC3	3863	ZINC-ALUMINUM ALLOY ASTM 23	1,6,41	355
ZINC4	3864	ZINC-ALUMINUM-COPPER ALLOY ASTM 25	1,6,41	355
ZIRCN	4330	ZIRCON (ZRO2.SIO2) (SINGLE CRYSTAL)	20,27,48,61	2505
PORZR	5282	ZIRCON PORCELAIN, HIGH	20	305535
ZR	1196	ZIRCONIUM	20,27,32,37,43,42,41	20002
ZR3ZI	3882	ZIRCONIUM ALLOY 3ZI (ZR97,AL1,SN1,MO1)	20	303330
ZIRCA	3881	ZIRCONIUM ALLOYS ZIRCALLOY 2 AND 3	1,6,14	200355
ZRBET	4728	ZIRCONIUM BERYLLIDE (ZRBE13)	20,48	500
ZRB2	4658	ZIRCONIUM BORIDE (ZRB2) (HP, 97 PC DEN)	20,27,47,48	20502
ZRB2B	5496	ZIRCONIUM BORIDE + BORON (BOROLITE 101)	47,48	505
ZRB2Z	5498	ZIRCONIUM BORIDE + MOSI2 (BORIDE Z)	47,48	2000500
ZRC	4580	ZIRCONIUM CARBIDE (ZRC) (HP OR SNT,94D)	20,27,48,54	22502
ZRH12	4892	ZIRCONIUM HYDRIDE (ZRH + ZRH2)	20	2225522
ZRNPF	4458	ZIRCONIUM NITRIDE (ZRN) (PR SR 88-90 D)	20,27,47,48	500
ZRNPS	4459	ZIRCONIUM NITRIDE (ZRN) (PR SR 93 D)	20,27,47,48	20502
ZROYZ	5495	ZIRCONIUM OXIDE (Y2O3 12, ZR 8) (97 D)	20	4305530
ZRO2	4106	ZIRCONIUM OXIDE (ZRO2) (MONOC., 100 D)	1,20,27,47,48,51	0
ZRO2C	4108	ZIRCONIUM OXIDE (ZRO2 96, CAO 6, 91 D)	1,20,27,47,48,51	0
ZROTI	5494	ZIRCONIUM OXIDE + TI CERMET ZT-15-M	20	5500
ZIRCN	4330	ZIRCONIUM SILICATE (ZRO2.SIO2) (XTAL)	20,27,48,61	2505
ZIRCP	4332	ZIRCONIUM SILICATE (ZRO2.SIO2) (100 D)	20,27,48,61	2500

[a]See text description of columns 73-79 for explanation of property abbreviation used in each column.

G.7 MATERIAL DATA LIST

The data for each material are contained on one to seven card-image records. A listing of the material data in the cgs-cal-°C units system may be found in ref. 1 or read from the appropriate file in the SCALE code package. The following format is used:

G.7.1 MATERIAL IDENTIFICATION CARD:

<u>Columns 1−5</u>

Alphanumeric identifier (same as in material index).

<u>Column 6</u>

Always left blank.

<u>Columns 7−10</u>

Material identification number (same as in material index).

<u>Columns 11−20</u>

Always left blank.

<u>Columns 21−25</u>

Number of data pairs listed in the table of specific heat vs temperature. This number does not exceed 12. If the number is not zero, it is always two or more.

<u>Columns 26−30</u>

Number of data pairs listed in the table of thermal conductivity vs temperature. This number does not exceed 12. If this number is not zero, it is always two or more.

<u>Columns 31−40</u>

Density of material in grams per cubic centimeter (g/cm^3). This is normally the value at or near room temperature or the lowest temperature for which specific heat or thermal conductivity is tabulated, whichever is highest. For use in codes that assume a fixed geometry, both the density and specified dimensions should correspond to the same conditions of temperature and pressure.

<u>Columns 41−50</u>

Specific heat of the material, in cal/g-°C, if constant or if the temperature dependence is unknown. When a table of specific heat vs temperature is listed, the value given here is normally the value at or near room temperature or at the lowest temperature tabulated, whichever is higher.

Columns 51−60

Thermal conductivity of the material, in cal/s-cm-°C, if constant or if the temperature dependence is unknown. When a table of thermal conductivity vs temperature is listed, the value given here is normally the value at or near room temperature or the lowest temperature listed in the table, whichever is higher.

Columns 61−70

Transition temperature of maximum temperature, in °C, at which the material may be used. If a latent heat is specified in columns 71-80, this is the transition temperature at which either a phase-change or solid-state transition occurs. If a material has more than one transition with a latent-heat effect, others are included in the table of specific heat vs temperature as a triangular peak superimposed over the actual specific heat curve. This triangular peak has a base width of 10°C and a height of 1/5 of the latent heat of transition, in cal/g, and is centered at the transition temperature.

Columns 71−80

Latent heat absorbed by the material, in cal/g, when the temperature is increased past the transition temperature specified in columns 61−70. Other latent-heat effects may be included in the specific heat table as described in the previous description (columns 61−70).

G.7.2 SPECIFIC HEAT TABLE CARDS

Only included if at least two data pairs are listed (no blank cards are needed).

Column 1

Always left blank.

Columns 2−10

Specific heat of the material, in cal/g-°C, at the temperature specified in columns 11−20.

Columns 11−20

Temperature, in °C, at which the material has the specific heat specified in columns 2−10.

Columns 21−30

Specific heat of the material, in cal/g-°C, at the temperature specified in columns 31−40.

Columns 31−40

Temperature, in °C, at which the material has the specific heat specified in columns

21—30. Always equal to or greater than the preceding temperature in the table.

The specific heat table card may list up to four data pairs using ten columns for each value of specific heat or temperature. As many as three cards with the same format can be used to list the number of data pairs specified in columns 21—25 of the material identification card.

G.7.3 THERMAL CONDUCTIVITY TABLE CARDS

Only included if at least two data pairs are listed (no blank cards are needed).

Column 1

Always left blank.

Columns 2—10

Thermal conductivity of the material, in cal/s-cm-°C, at the temperature specified in columns 11—20.

Columns 11—20

Temperature, in °C, at which the material has the thermal conductivity specified in columns 2—10.

Columns 21—30

Thermal conductivity of the material, in cal/s-cm-°C, at the temperature specified in columns 31—40.

Columns 31—40

Temperature, in °C, at which the material has the thermal conductivity specified in columns 21—30. Always equal to or greater than the preceding temperature in the table.

The thermal conductivity table card may list up to four data pairs using ten columns for each value of thermal conductivity. As many as three cards with the same format can be used to list the number of data pairs specified in columns 26—30 of the material identification card.

REFERENCES

1.   A. L. Edwards and P. T. Williams, "Thermal Material Properties Library," as described in Sect. M5 of *SCALE: A Modular Code System for Performing Standardized Computer Analyses for Licensing Evaluation*, NUREG/CR-0200 (Revision 4), Vol. 3, tentative publication date December 1991.

2.   A. L. Edwards, *A Compilation of Thermal Property Data for Computer Heat-Conduction Calculations*, UCRL-50589, University of California Lawrence Radiation Laboratory, February 24, 1969.

INTERNAL DISTRIBUTION

1. J. C. Anderson
2. J. P. Belk
3-22. K. W. Childs
23. J. C. C'₌veland
24. J. S. Crowell
25. R. D. Dabbs
26. M. R. Feldman
27. G. E. Giles
28. J. R. Kirkpatrick
29. G. H. Llewellyn
30. ?. E. Park
31. C. V. Parks
32. B. W. Patton
33. A. S. Quist

34. J. P. Sanders
35-37. C. H. Shappert
36. I. I. Siman-Tov
37. A. Sozer
38. M. J. Taylor
39. M. W. Wer.del
40. G. E. Whitesides/R. P. Leinius/ Applied Technology Library
41. K. E. Wilkes
42. P. T. Williams
43. Central Research Library
44. Document Records Section (Y-12)
45. Applied Technology Library
46. Laboratory Records
47. K-25 Site Records

EXTERNAL DISTRIBUTION

48-59. Office of Scientific and Technical Information, P.O. Box 62, Oak Ridge, TN 37831

60. Mr. Gail Allen, Westinghouse Hanford Company, P.O. Box 1970, MSIN HO-34, Richland, WA 99352

61. Mr. Martin K. Altenhofen, Environmental Health Physics Group, Battelle Pacific Northwest Laboratories, P.O. Box 999, Richland, WA 99352

62. Dr. Bryan Becker, University of Missouri, 600 W. Mechanic St., Independence, MO 64050-1799

63. Mr. George Cadelli, Building 730-A, Westinghouse Savannah River Company, Aiken SC 29808

END

DATE
FILMED

12/11/91