AN ABSTRACT OF THE THESIS OF

Larry H. Nielsen for the degree of Master of Science in Nuclear Engineering presented on October 24, 1980

Title: An Assessment of Describing Function Methods Applied to Reactor Fuel Performance Analysis

Abstract Approved: Redacted for Privacy
K.L. Peddicord

Large digital computer reactor fuel performance codes perform repeated calculations of temperature profiles during transients. The thermal response is required to calculate the mechanical behavior. This mechanical response in turn feeds back to the thermal calculations through mechanisms such as fuel swelling or shrinking, causing changes in the fuel-clad gap conductance. This calculation process can become very costly in terms of computer time.

Describing functions or input-dependent transfer functions have been shown to exhibit substantial savings in computing time when compared to traditional finite difference or finite element methods in solving the non-linear partial differential equations for heat transfer in a fuel rod. This thesis is a study of the feasibility of using a describing function technique to replace the

finite element based heat transfer section of a fuel performance code. The hybrid program gave results which compared favorably with the original code but showed longer running times because of the interaction with the deformation solution at each transient time step. The steady state solution steps, however, gave comparable results in one-tenth of the computing time.

Also developed were new polynomial expressions to replace the original expressions for uranium dioxide conductivity and integral conductivity. These new functions gave more accurate results and were computationally faster.

AN ASSESSMENT OF DESCRIBING FUNCTION METHODS APPLIED TO REACTOR FUEL PERFORMANCE ANALYSIS

by

Larry H. Nielsen

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Commencement June 1981

APPROVED:

Redacted for Privacy

Associate Professor of Nuclear Engineering
in Charge of Major

Redacted for Privacy

Head of Department of Nuclear Engineering

Redacted for Privacy

Dean of Graduate School

Date thesis is presented October 24, 1980

Text processed by CYBER 70/73 for Larry H. Nielsen

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

AN ASSESSMENT OF DESCRIBING FUNCTION METHODS APPLIED TO REACTOR FUEL PERFORMANCE ANALYSIS

I. INTRODUCTION

In the work, "Transient Fuel Pin Temperature Calculations Using Describing Functions", Guidotti developed a method utilizing describing functions or input-dependent transfer functions to determine a solution for the time dependent, non-linear partial differential equations for heat flow in a reactor fuel pin. The method was successfully demonstrated by the computer program, DFTEMP [1]. Since this program essentially generates a solution over the complete time domain of interest, the temperature conditions are available at any desired spatial location and at any time during a transient with no corresponding penalty in execution time. Conventional finite-difference or finite-element codes must integrate over each successive time step to obtain the response at some remote time. In comparisons with a non-linear heat conduction finite-element code, COYOTE [2], a factor of 40 decrease in running time was obtained with DFTEMP for a particular calculation of the temperature response of a fuel pin following a scram.

A disadvantage of this approach is that representations of the various parameters influencing the

heat transfer, most notably the fuel-clad gap conductance, must be known in advance for the entire transient. Large fuel performance codes calculate the gap conductance at each time step based on the physical properties of the pin such as the fuel swelling and gas pressures which are functions of the changing temperatures. To incorporate DFTEMP as a heat transfer package into a large fuel behavior code, the describing function operation must be modified to allow the transient temperatures to be calculated interactively with the mechanical behavior routines. This modification severely handicaps the major advantage of the describing function method.

This work is an extention of Guidotti's development. An assessment will be made of the suitability of using a modified version of DFTEMP as the fuel pin heat transfer module in a new fuel performance code, FREY (Fuel Rod Evaluation System) [3,4], developed by Anatech International Corporation for the Electric Power Research Institute of Palo Alto, California. A description of FREY is included as are the modifications made to DFTEMP necessary to implement it into the heat transfer overlay of FREY. As this is the first attempt to use describing functions in this manner, a simple transient was modeled - that of the drop in temperature following a reactor scram. Further investigations would be necessary to fully explore

the use of describing function methods to analyze fuel performance.

II. THE FUEL ROD EVALUATION SYSTEM CODE - FREY

The purpose of the computer code FREY is to provide a system capable of analyzing fuel rod transient behavior in current BWR and PWR designs. The code was written for the CDC 7600 computer system which precluded its use on Oregon State University's CDC 6600 system. All computer runs used in this assessment were made on the Lawrence Berkeley Laboratory CDC 7600.

The code employs the finite element method of analysis using the same grid for both the thermal and deformation solutions. The geometry is two-dimensional with a choice of either axisymmetric (r-z) or plane (r-θ) problem types. Major components of the fuel pins such as the pellets, cladding and end plugs are modeled using eight node quadrilateral elements. Special gap elements are used for the fuel-clad gap and pellet-pellet contacts which possess thermal conductivity, friction-slip and contact-release properties.

Since FREY is still evolving at the time of this writing, it is difficult at best to describe the operation of the code in detail. For the purpose of this work, the initial release version, FREY01, was used to replace the finite element heat transfer section with a describing function algorithm.

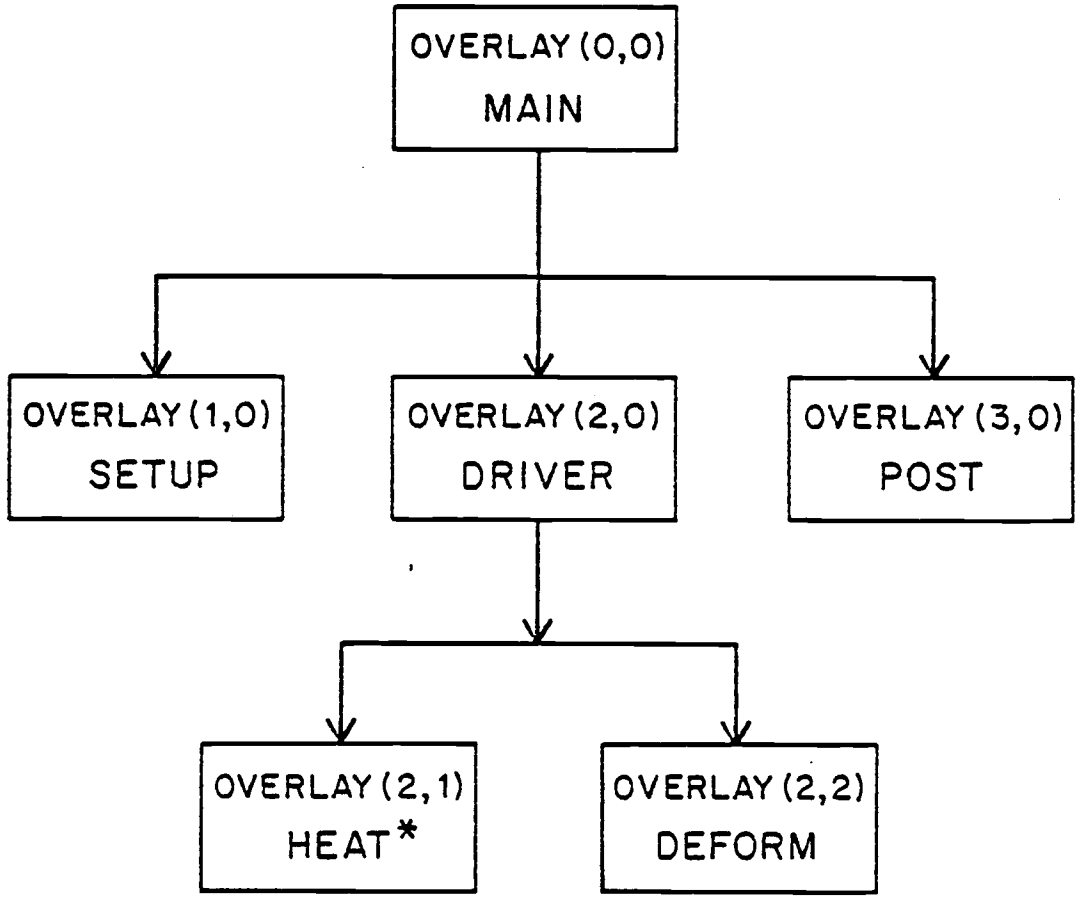
FREY01 is divided into three primary overlays; SETUP, DRIVER, and POST (see figure 1). The SETUP overlay is entered once for each problem. Relevant data is read from input and is used to generate the nodal point coordinates for the mesh. Other functions include the generation of element connectivity data; identification of fuel, clad, gap, and structural elements such as plenum springs; definition of the boundary conditions and problem constraints such as core pressure and spacer contact forces; and the setting of any initial conditions on stress or temperature. This information is stored in Large Core Memory (LCM) common for access by the other primary overlays.

The second primary overlay, DRIVER, is the heart of the program (see figure 2). It calls two secondary overlays, HEAT and DEFORM, which perform the step-by-step solution in succession. DRIVER reads in time-dependent history such as surface temperature, film coefficient, bulk flow and prior power which are used to calculate various parameters like the internal gas pressure and exposure history for the current fuel load cycle needed to compute gap conductivity.

The HEAT overlay was developed from the non-linear heat conduction code COYOTE, written by D. K. Gartling of Sandia Laboratories. Its structure was modified to model

Figure 1

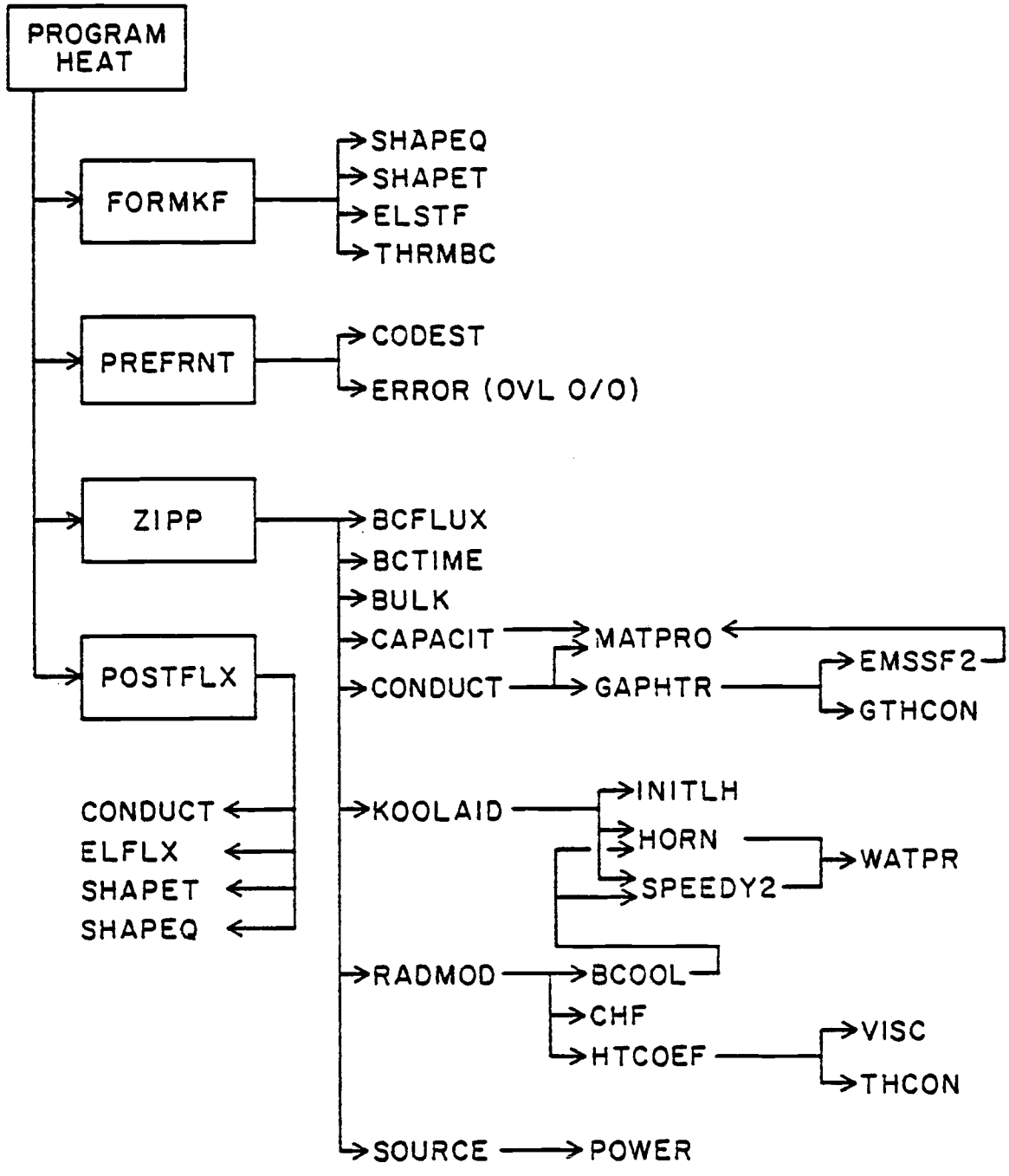
FREY Overlay Structure



*Replaced With Describing Function Temperature Method

Figure 2

HEAT Subroutine Flow Diagram



fuel pin geometries exclusively which precludes the use of FREY for analyzing generic problems. The development of the finite element equations for transient conduction heat transfer problems is well documented and is only briefly outlined here for the purpose of showing the functions of the various subroutines comprising HEAT. The region to be analyzed is divided into a number of smaller simply shaped elements. A set of nodal points is established for each element at which the dependent variable (temperature) is to be evaluated. For each of the elements, the temperature field can be approximated by:

$$T(x_i,t) = \sum_{n=1}^{N} \Phi_n(x_i)\ \Theta_n(t) \qquad \text{II.1}$$

or in matrix notation:

$$T(x_i,t) = \underset{\sim}{\Phi}_n^T(x_i)\ \underset{\sim}{\Theta}_n(t) \qquad \text{II.2}$$

where:

$\underset{\sim}{\Phi}_n$ is an N dimensional vector of interpolating or shape functions

$\underset{\sim}{\Theta}_n$ is a vector of nodal point unknowns

N is the number of nodes in the element

This approximation is substituted into the differential equation for heat transfer:

$$\rho\ c_p \frac{\partial T}{\partial t} - \frac{\partial}{\partial x_i}\left(k_{ij} \frac{\partial T}{\partial x_j} \right) - Q = 0 \qquad \text{II.3}$$

where: ρ = density

c_p = heat capacity

k_{ij} = conductivity tensor

Q = volumetric heat source

t = time

x_i = spatial coordinate

T = temperature

Since the assumed temperature field is only an approximation to the true solution, a set of residual equations results from the substitution. Galerkin [5] showed that the residual for each element, R, is orthogonal to the space spanned by the shape function for that element, or

$$\int_{\Omega_e} \underset{\sim}{\Phi}\, R \, d\Omega = 0 \qquad \text{II.4}$$

where Ω_e is the region enclosed by the element.

Making the above substitution and including the external boundary conditions for the element leads to a matrix equation for each element of the following form once the form of the shape function $\underset{\sim}{\Phi}$ is known.

$$\underset{\approx}{M}\dot{\underset{\sim}{\Theta}} + \underset{\approx}{K}\underset{\sim}{\Theta} = \underset{\sim}{F}_Q + \underset{\sim}{F} \qquad \text{II.5}$$

where $\underset{\approx}{M} = \int_{\Omega_e} \rho\, c_p\, \underset{\sim}{\Phi}\, \underset{\sim}{\Phi}^T \, d\Omega$ $\qquad \underset{\sim}{F}_Q = \int_{\Omega_e} \underset{\sim}{\Phi}\, Q \, d\Omega$

$$\underset{\approx}{K} = \int_{\Omega_e} \frac{\partial \Phi}{\partial x_i} k_{ij} \frac{\partial \Phi^T}{\partial x_j} \, d\Omega \qquad \underset{\sim}{F} = \int_{\Gamma_e} \underset{\sim}{\Phi}\, (q_n + q_c + q_r)\, d\Gamma$$

q_n = applied heat flux normal to boundary

q_c = heat flux due to convection

q_r = heat flux due to radiation

By imposing continuity of temperature constraints on adjacent elements, a set of matrix equations can be assembled for the entire area of interest. The only boundary conditions of interest are those on external surfaces since effects from internal boundaries are cancelled by the adjoining elements. This part of the solution process is performed by subroutine FORMKF which calls other subroutines to assemble the geometry, evaluate the shape functions for the type of element used, compute the element matrices and set up the thermal boundary conditions.

The next step is to prepare the element matrices for solution with a pre-front operation which determines the connectivity of the various elements and establishes a "destination vector" for the matrix equations solutions. The equations are solved for one element at a time with the equations for adjacent elements called in as needed. Once the dependent variable has been determined at all the nodes in an element, the solution moves to the next element. Subroutine ZIPP performs the actual solution for the temperature field after determining the coolant conditions, material properties, coolant dependent heat transfer

boundary conditions and the volumetric heating in the fuel. The local coolant conditions are available from either the initial conditions or the previous iteration through a common block. The temperature dependent material properties, the heat capacity and thermal conductivity, are calculated for each element through calls to the MATPRO [6] subroutine package from the routines CAPACIT and CONDUCT. To model the fuel-clad gap, two node gap elements are used. These elements have a thermal conductivity value calculated by GAPHTR using other MATPRO routines.

Essential to the proper modeling of actual in-core conditions are the thermodynamic correlations used to characterize the heat transfer at the clad surface. The governing boundary condition is given by

$$q = h\,(T_w - T_{ref}) \qquad \text{II.6}$$

where h and T_{ref} are to be determined.

To find the reference temperature, FREY01 employs a coolant enthalpy model similar to that in FRAP-T4 which applies the principles of conservation of mass and energy in a series of vertically stacked one-dimensional control volumes corresponding to the exterior clad node points. The model assumes that the upper plenum conditions lag those of the lower plenum by one time step and that the

mass flow rate varies linearly with elevation and is high enough so that the heat can be transferred to the coolant without significantly affecting the coolant conditions.

The correlations used for the determination of the heat transfer coefficient, h, were taken from the RETRAN code. There are seven possible modes for heat transfer:

1) Forced Convection to Liquid (Dittus-Boelter)
2) Fully Developed Surface Boiling (Thom)
3) Forced Convection Vaporization (Schrock-Grossman)
4) Forced Convecton Transition Boiling (McDonough, Milich, King)
5) Pool Stable Flux Boiling (Berenson)
6) Pool Transition Boiling (Berenson)
7) Forced Convection to Superheated Vapor (Dittus-Boelter)

To determine the proper mode, it is first necessary to determine the value of critical heat flux, again using the methods from RETRAN (see figure 3). Once the CHF is known, a wall temperature corresponding to this value of heat flux can be found from the Thom correlation,

$$T_{w_{CHF}} = T_{sat} + .072 \exp\left(-\frac{P}{1260}\right) Q_{CHF}^{.5} \qquad \text{II.7}$$

then a series of tests can be made (see figure 4) to select the proper heat transfer mode.

Finally, the volumetric heat term is calculated for each element in the fuel using the subroutine POWER from FRAP-T4 which completes the information needed for assembly of the global matrix equations.

Figure 3

Critical Heat Flux Correlations

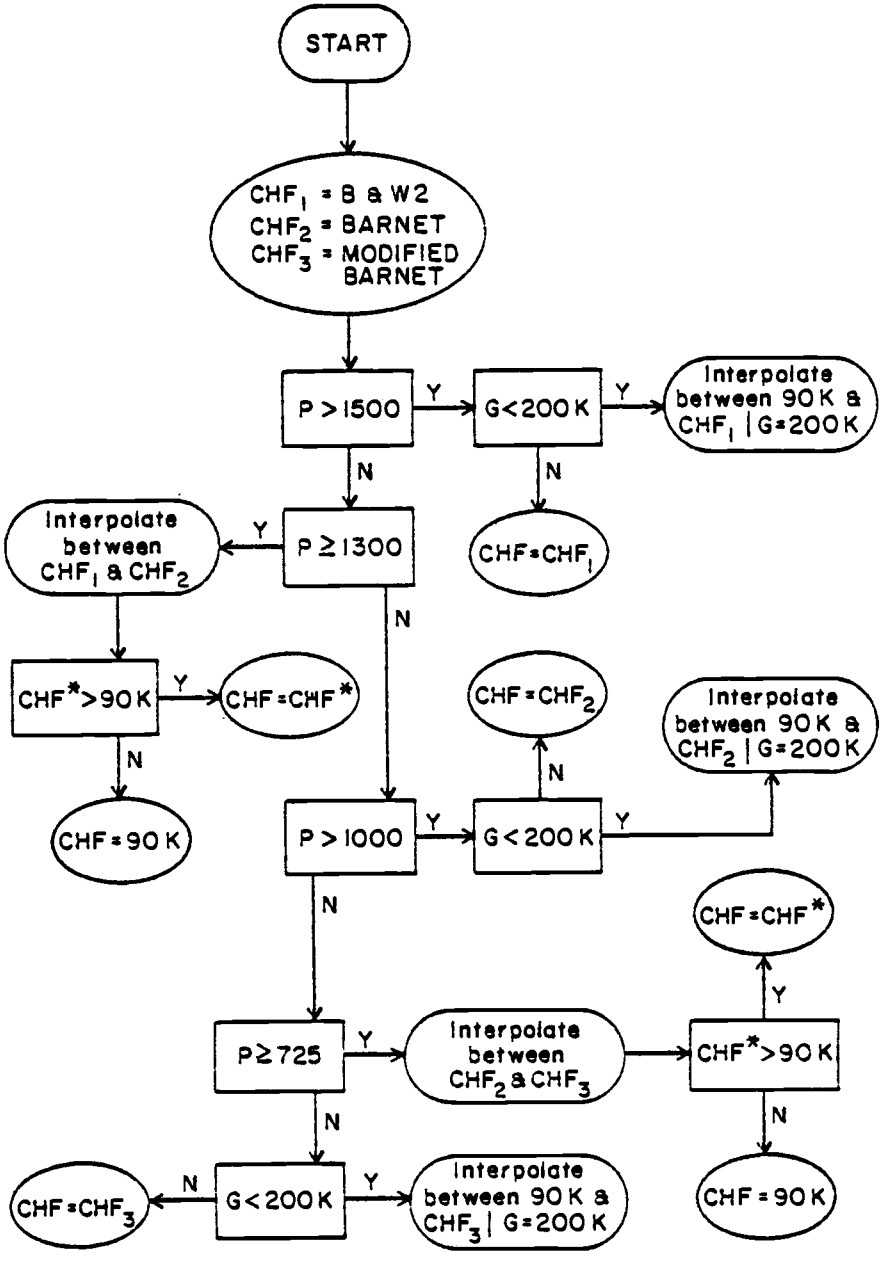
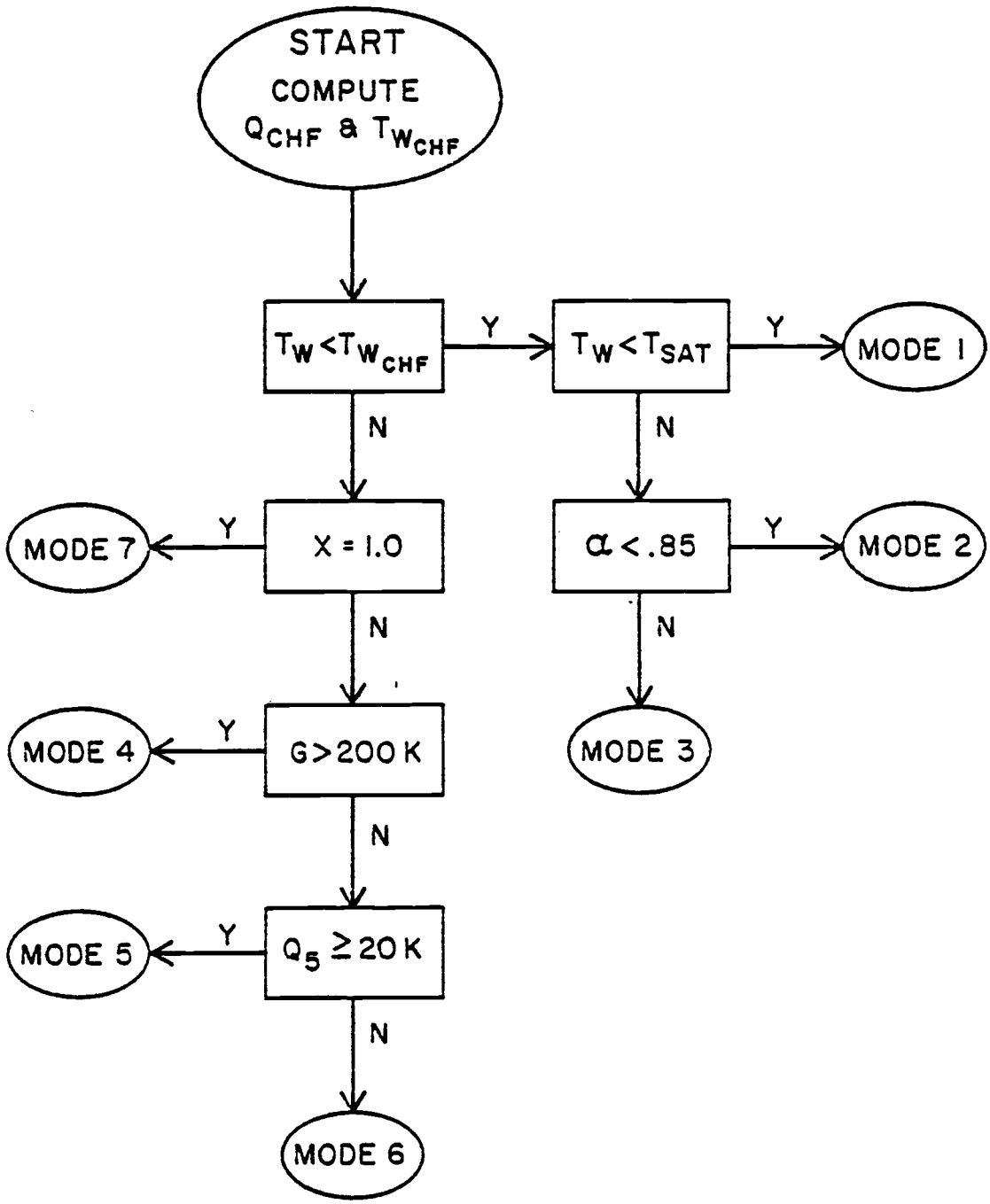


Figure 4

Heat Transfer Modes



The solution process for the assembled matrix equations proceeds along one path for steady state problems and another for transient analysis. The steady state case results in a reduced matrix equation of the form:

$$\underset{\approx}{K}(\underset{\sim}{\Theta}) \cdot \underset{\sim}{\Theta} = \underset{\sim}{F}(\underset{\sim}{\Theta}) \qquad \text{II.8}$$

Since both $\underset{\approx}{K}$ and $\underset{\sim}{F}$ are functions of $\underset{\sim}{\Theta}$, an iterative approach is used to converge on the solution. ZIPP uses the Picard iteration method or successive substitution which is given by:

$$\underset{\approx}{K}(\underset{\sim}{\Theta}^{n}) \cdot \underset{\sim}{\Theta}^{n+1} = \underset{\sim}{F}(\underset{\sim}{\Theta}^{n}) \qquad \text{II.9}$$

where n and n+1 refer to the iteration number. Here $\underset{\sim}{\Theta}^{n+1}$ is determined from the linear equations (II.9) knowing $\underset{\sim}{\Theta}^{n}$.

An implicit averaged Crank-Nicholson scheme is used to perform the time integration for a transient analysis which results in a high degree of numerical stability allowing the use of large time increments. This central difference method is similar to the standard Crank-Nicholson procedure but the solution is not extrapolated to the end of the time interval. The integration of the matrix equations takes the form:

$$\left[\frac{2}{\Delta t}\,\underset{\approx}{M}(\underset{\sim}{\Theta}^{a}) + \underset{\approx}{K}(\underset{\sim}{\Theta}^{a})\right]\underset{\sim}{\Theta}^{a} = \underset{\sim}{F}(\underset{\sim}{\Theta}^{a}) + \frac{2}{\Delta t}\,\underset{\approx}{M}(\underset{\sim}{\Theta}^{a})\,\underset{\sim}{\Theta}^{n} \qquad \text{II.10}$$

where $\underset{\sim}{\Theta}^{a} = \dfrac{\underset{\sim}{\Theta}^{n+1} + \underset{\sim}{\Theta}^{n}}{2}$

n = time iteration

Δt = time step

To evaluate the temperature dependent terms, $\underset{\sim}{\Theta}^a$ is approximated by $\underset{\sim}{\Theta}^n$ which is a good assumption for problems where no step discontinuities exist. This form reduces to the steady state equation if $\Delta t \to \infty$.

The last function of the HEAT overlay is to calculate the heat flux field at every element boundary using the newly computed temperature profile. This option provides for a detailed look at the heat transfer conditions for any time step but is not an integral part of the solution since only the heat flux at the surface of the clad is needed for analysis at subsequent time steps.

Once the temperature profile is stored in common memory, control is shifted to the DEFORM secondary overlay. It performs the deformation solution in much the same way as HEAT performs the temperature solution. The element stiffness and force matrices are calculated using the proper shape functions and the material properties from calls to MATPRO with the latest temperature solution. The gap element equations are assembled with provisions for contact and stick or slip conditions. The matrix equation solution is also of the frontal solver type using Gaussian elimination on an element-to-element basis without assembling the global matrix equations. The

back-substitution then produces the incremental displacements for each node.

Subroutine STRESS then determines if the solution is complete for the current time step or if iteration is necessary. Since the deformation depends on the temperature which, in turn, is affected by the swelling or shrinking of the fuel resulting in changing gap conductance, at least one iteration is needed. When the solution converges, the incremental displacements are summed and the stresses and strains for each element are calculated. Element history data is updated and output for strains, stresses, plasticity index, damage index, failure index and gap information is printed if desired. Control is then passed back to DRIVER for the next time step or another iteration.

If the solution for the entire transient is complete, the last primary overlay, POST, is called to process the output data and when completed, to construct graphics of the element grid (mainly for debugging), contours of stress and temperature, and time-history plots of deformation and temperature. Future plans include possible inclusion of a plotting package that will prepare movie films of isotherms or strains during transient analyses.

III. THE DESCRIBING FUNCTION METHOD

The use of describing functions to calculate transient fuel pin temperatures was demonstrated in the previous work by Guidotti. Three basic steps were used to solve the non-linear partial differential equation which governs the heat transfer process. First, Kirchoff transformations were used to recast the equation into a linear partial differential equation. These transformations, the integral conductivity and the integral heat capacity or enthalpy, then allow the use of linear operators - the Laplace and Hankel integral transforms, which convert the partial differential equation in time and space into an algebraic expression in the doubly transformed domain. The third step is to relate the new variables to each other with describing functions, yielding a single algebraic equation in one variable which can be solved to obtain the desired temperature at any value of time and position.

Starting with the general heat conduction equation,

$$\nabla \cdot k \nabla T + q''' = \rho c_p \frac{\partial T}{\partial t} \qquad \text{III.1}$$

simplifying assumptions can be made to yield the specific form used to analyze a reactor fuel pin. First the axial conduction along a typical 3.8 meter length of the rod can be neglected due to the small temperature gradient in this

direction and because of the large thermal resistance caused by the inter-pellet gaps every 1.4 cm. along the fuel stack. Azimuthal variations in the temperature can also be neglected since the pin is geometrically axisymmetric with only small random variations due to cracking or offset stacking within the confines of the cladding. This reduces the problem for the two regions; in the fuel with its highly non-linear thermal properties:

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\, k_f(T)\frac{\partial T_f}{\partial r}\right) + q'''(r,t) = \rho_f\, c_{p_f}(T)\frac{\partial T_f}{\partial t} \qquad \text{III.2}$$

and in the clad, assuming constant properties and no heat generation, Fourier's Equation:

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial T_c}{\partial r}\right) = \frac{1}{\alpha_c}\frac{\partial T_c}{\partial t} \qquad \text{III.3}$$

The four boundary conditions needed for solution are:

1) zero temperature gradient at the fuel centerline,

$$\frac{\partial T_f}{\partial r}(0,t) = 0 \qquad \text{III.4}$$

2) continuity of heat flux at the fuel-clad interface,

$$-R\, k_f(T)\frac{\partial T_f}{\partial r}(R,t) = -R_i\, k_c\frac{\partial T_c}{\partial r}(R_i,t) \qquad \text{III.5}$$

3) a contact resistance at the fuel-clad interface,

$$-k_f(T)\frac{\partial T_f}{\partial r}(R,t) = h_g(t)\left[T_f(R,t) - T_c(R_i,t)\right] \qquad \text{III.6}$$

4) convection heat transfer at the clad surface,

$$-k_c \frac{\partial T_c}{\partial r}(R_o,t) = h\left[T_c(R_o,t) - T_\infty(t)\right] \qquad \text{III.7}$$

It should be noted that the third boundary condition makes allowances for a gap conductance term that is time dependent. This is caused by changes in the gap size as a result of fuel shrinking and swelling. The behavior of the gap conductance term must be known in advance to solve the transient problem explicitly. The two necessary initial conditions are found by solving the steady state problem, that is, the above equations with the time derivatives set to zero.

To remove the non-linear terms from equation III.2, two new variables are defined, the integral conductivity, I, and the integral heat capacity, U. These are given by:

$$I = \int_{T_{ref}}^{T} k(T)\, dT \qquad U = \rho \int_{T_{ref}}^{T} c_p(T)\, dT \qquad \text{III.8,9}$$

The use of the integral conductivity function is well documented. Expressions for both integral conductivity and the inverse function are presented in MATPRO. The integral heat capacity can easily be calculated by analytical integration of an expression for the heat capacity also given in MATPRO. When these are substituted in equation

III.2 the result is a linear partial differential equation in the new variables,

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial I}{\partial r}\right) + q'''(r,t) = \frac{\partial U}{\partial t} \qquad \text{III.10}$$

For his analysis, Guidotti used a quadratic heat source variation of the form:

$$q'''(r,0) = q_o\left[1 + a\left(\frac{r}{R}\right)^2\right] \qquad \text{III.11}$$

where q_o and a are constants that can be determined from the linear power rating and peak-to-average power ratio. This form closely approximates the actual heat source distribution with its central depression. Once the form of the heat source is known, the steady state solution to equation III.10 can be found, giving the initial temperature profile for the transient analysis.

The second step in the process is to reduce equation III.10 to an algebraic equation using integral transforms. Because of the cylindrical geometry which results in solutions involving Bessel functions, the finite Hankel transform was chosen to reduce the spatial term. The finite Hankel transform of a radially dependent function, g(r), is defined as:

$$H_o\{g(r)\} = \bar{g}(\xi_\ell) = \int_0^R r\, J_o(\xi_\ell r)\, g(r)\, dr \qquad \text{III.12}$$

where J_o is the Bessel function of order zero and the transform variable, ξ_ℓ ,is evaluated from the roots of:

$$J_o(\xi_\ell R) = 0 \qquad \text{III.13}$$

The use of this transform will result in a zero value of the transformed function at the fuel surface (r = R). This makes it necessary to redefine the terms of equation III.10 relative to their fuel pellet surface values, for example,

$$I(r,t) = I_s(r,t) + I(R,t) \qquad \text{III.14}$$

where $$I_s(R,t) = 0 \qquad \text{III.15}$$

The time derivative of equation III.10 is reduced to an algebraic expression through the use of the Laplace transform, defined as:

$$\mathcal{L}\{f(t)\} = \bar{f}(s) = \int_0^\infty e^{-st} f(t)\, dt \qquad \text{III.16}$$

In contrast to the Hankel transform, the Laplace transform has an infinite upper limit of integration which results in a continuous function in the transformed domain while the Hankel transform variable, ξ_ℓ, takes on discrete values determined by the root equation III.13.

The result of applying these two transforms to equation III.10 gives the algebraic expression,

$$-\xi_\ell^2 \; \bar{\bar{I}}_s(\xi_\ell,s) + \bar{\bar{q}}'''(\xi_\ell,s) = s\,\bar{\bar{U}}_s(\xi_\ell,s) - \bar{\bar{U}}_s(\xi_\ell,0)$$
$$+ \frac{R\,J_1(\xi_\ell R)}{\xi_\ell}\left[s\,\bar{U}(R,s) - U(R,0) - \bar{q}''(R,s) \right] \qquad \text{III.17}$$

This expression contains three unknowns, $\bar{\bar{I}}_s(\xi_\ell,s)$, $\bar{\bar{U}}(\xi_\ell,s)$, and $\bar{U}(R,s)$. The remaining terms can be calculated by performing the indicated integrations on the heat source and the initial integral heat capacity available from the steady state solution. Two more equations are needed for closure of the set. One comes from the boundary conditions, involving the solution in the clad, and the other from the application of the describing function method.

The describing function extends the linear theory of transfer functions to non-linear systems. For a linear system, the transfer function relates the influence of a certain change in the input to the output of the system. Transfer functions are widely used in the study of mechanical and electrical systems. If a transfer function were to be used to analyze a non-linear system, it would first be necessary to choose a nominal value for the input about which the system could be linearized. As long as the true input did not vary far from the nominal value, the

calculated solution would be a good representation of the actual system response. This type of analysis is common in mechanics problems, for example in the study of the motion of a mass-spring system where the initial perturbation is small, allowing the use of a linear spring force law. When the range of inputs is large enough to invalidate the use of true linearization, another concept must be used, that of quasi-linearization which leads to the use of the describing function.

The describing function uses properties of the input to generate an operator which can be used to solve the non-linear system. Using the fact that since both the integral heat capacity, U, and integral conductivity, I, are functions of temperature, and therefore are related to one another, Guidotti used statistical methods to find a polynomial relationship for U as a function of I:

$$U = u_1 I + u_2 I^2 \qquad \text{III.18}$$

Then, since the expected form of the temperature transient following a scram is an exponentially decreasing function, he assumed the following form for the fuel surface value of the integral conductivity during the transient:

$$I_s(r,t) = A_\ell J_o(\xi_\ell r) e^{-b\ell t} \qquad \text{III.19}$$

where A_ℓ and b_ℓ are known constants.

Using this functional form for I_s gave values from equation III.18 for U_s and allowed the calculation of the material describing function, D. Like the more common transfer function, the describing function is defined as the ratio of the integral transform of the output to the same transform of the input, or

$$D = \frac{\bar{\bar{U}}_s}{\bar{\bar{I}}_s} \qquad \text{III.20}$$

with the difference being that the transfer function is valid only over a restricted range of inputs while the describing function is useful over the entire range since it uses the properties of the assumed input form, the magnitude and time constant of the exponential decrease, in its calculations.

To apply the describing function, a temperature profile is assumed and values for I_s and U_s are calculated. These are then used in equation III.17 to calculate a new value for the integral conductivity. The solution is then repeated until the convergence criterion is met, usually in two or three iterations. It is important to note that these iterations involve the entire transient, not just a single time step as with conventional finite difference or finite element schemes. This is the significant advantage of the method, that the solution is available for any

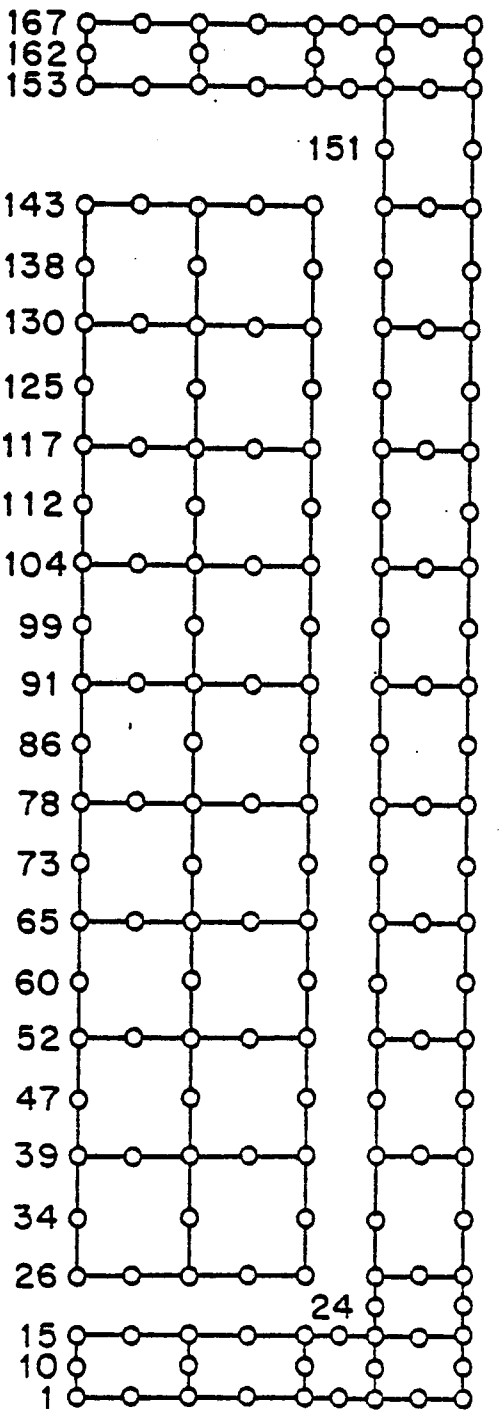
desired value of time or radial position. The calculation of a transient response of 100 seconds duration takes no more time than that for 10 seconds duration. For situations that can be modeled appropriately, very significant gains in computational efficiency can be realized. For this reason, it is desirable to implement the method of describing functions into a fuel performance code, such as FREY, which requires repeated temperature calcuations to compute the stresses and resulting deformations in a fuel pin during a transient.

IV. IMPLEMENTATION OF THE METHOD

As mentioned previously, the computer code DFTEMP showed a dramatic reduction in computing time over the conventional finite element code, COYOTE. DFTEMP was designed to calculate a single radial temperature profile at some arbitrary vertical location along a fuel pin. In order to model the entire pin, repeated calculations have to be made, one at each axial location at which the temperature is desired. This was easily implemented due to the regular pattern established by the generation of the finite element mesh. Since it was desired to change only the heat transfer portion of FREY, the mesh generation and the deformation solution routines continue to use the finite element formulation. Figure 5 shows the element and noding scheme used in this study. Elements 1 through 18 represent the fuel area, 23 through 33 the clad, and the remainder of the eight node elements at the top and bottom are for the end caps. Two node elements connect the nodes along the periphery of the fuel section with the neighboring nodes in the clad and end caps, modeling the gas gap and plenum. Although each of the 271 pellets in the fuel stack could be represented with a unique element, this greatly increases the computing time and does not lead to a significant gain in the information obtained.

Figure 5

Noding Diagram



To couple the DFTEMP solution to the element nodal array, a calculation of the temperature profile was made at the intersection of each of the fuel stack elements and at the top and bottom of the stack. The temperatures at nodal locations between these computed locations were obtained by averaging the temperatures of the nodes immediately above and below. This is justified because of the smooth gradient in the axial direction throughout the fuel. If axial conduction can be neglected, this vertical stacking of one-dimensional heat transfer problems gives a good representation of the two-dimensional case and is used in many finite difference type codes. It does, however, cause some difficulty in calculating temperatures in the end caps of the fuel rod. For the purposes of this study, the end cap temperatures were equated to the temperature of the clad elements at the top and bottom extremes of the fuel stack. This introduces very little error since the linear power of the rod is lowest at the ends and only small variations in temperature are seen in this region. The derivation of a describing function method to calculate two-dimensional heat transfer would be an extremely difficult task. This approximation has even more validity when an examination of the FREY01 output showed that the nodes in the row along the midplane of each end cap had a temperature lower than that of either vertically adjacent

node. Physically, since the end caps cannot be a heat sink, the lowest temperature must be on the outside surface.

The transient enthalpy rise model contained in FREY01 to compute the coolant conditions was also discarded in favor of a simple heat balance. Since at this time, the describing function method is useful only for transients involving an exponential decrease in fuel temperature typical of a scram or a dropped control rod, the only mode of heat transfer of concern is single phase, subcooled liquid, forced convection. For this reason, the coolant temperatures can be obtained by a straight forward heat balance - that all of the heat produced in the fuel is transferred to the coolant. This is given by:

$$T_c^{i+1} = T_c^i + \frac{\chi_i \, \Delta z_i}{\dot{m}_c \, c_p} \qquad \text{IV.1}$$

where:

T_c^{i+1} = coolant temperature at i^{th} axial location

χ_i = linear power in i^{th} segment

$\Delta z_i = z_{i+1} - z_i$, z= axial height

$\dot{m}_c$ = mass flow rate of coolant

c_p = heat capacity of coolant

The sophistication of the enthalpy rise model was seen as

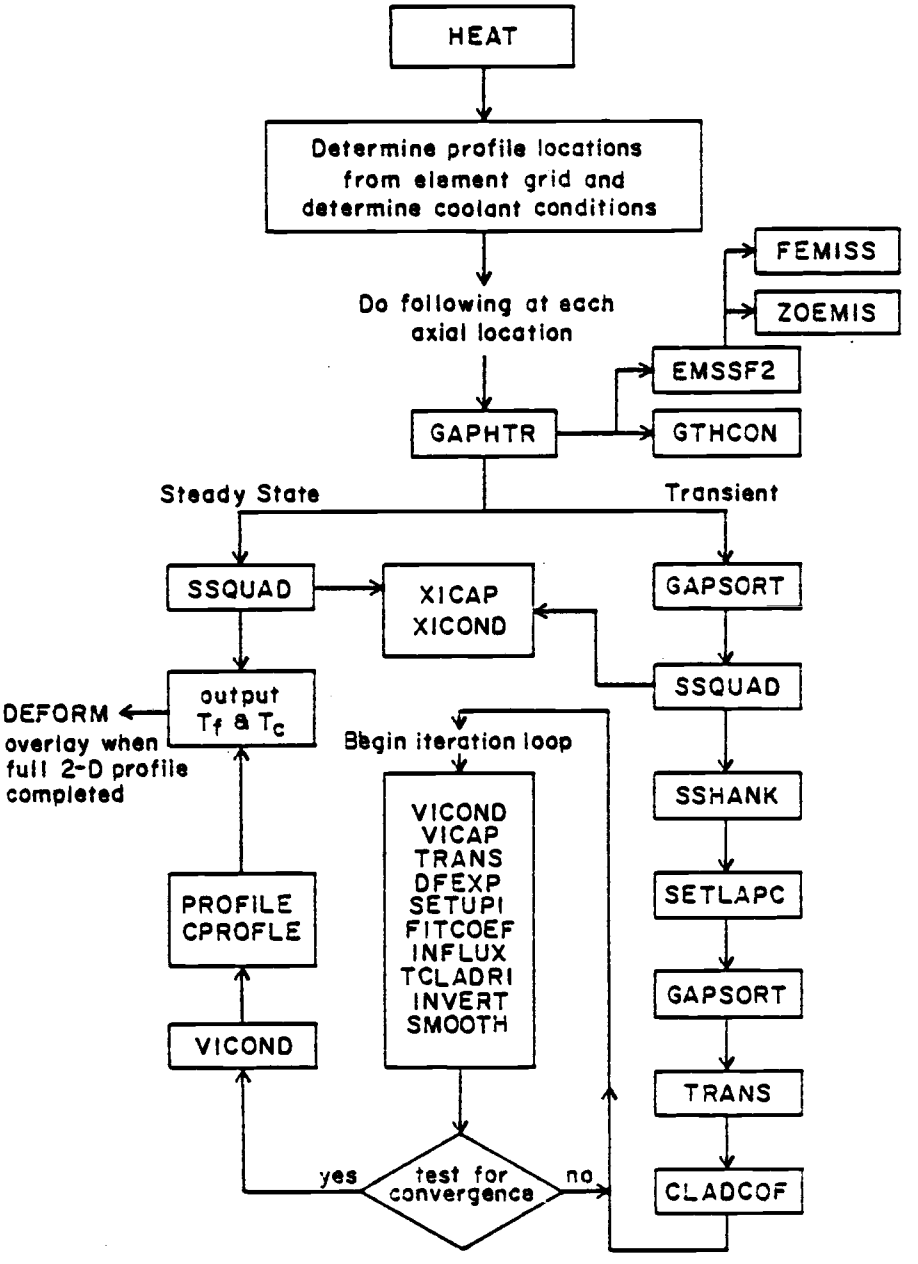
computationally too expensive and unnecessary considering the limitations on the types of transients that could be analyzed. The Dittus-Boelter correlation was used to determine the film heat transfer coefficient at the clad surface which gave a value at each axial profile position remaining nearly constant (within one percent) over the duration of the transient. These constant values were then applied to the corresponding locations along with the bulk temperature for that position obtained from the heat balance, thus providing the information necessary to evaluate the clad surface boundary condition.

The operation of FREY01 is such that the initial conditions for the transient to be analyzed are obtained by several steady state calculations while the power is ramped up to the desired pre-transient level. For this case, 10 one second time steps are used to reach the initial conditions. Power is increased linearly for the first six steps and held constant for the next four to allow all temperatures, heat fluxes and other parameters to stabilize.

Figure 6 shows the flow chart for the algorithm used in the modified version of FREY01, called FREYDF. Subroutine GAPHTR is used in the same way as in FREY01 to compute the gap conductance at each axial location. For the pre-transient steady state calculations, SSQUAD (for

Figure 6

FREYDF Subroutines and Operation



Steady State QUADratic heat source profile) is called by FREYDF to compute the steady state temperatures. This routine is computationally very fast since with the use of the integral conductivity transform, the solution to the linearized steady state equations for heat transfer can be programmed directly.

The expressions given in MATPRO - Version 10 for the integral conductivity function for UO_2 fuel and for the inverse function, contain an inherent inaccuracy. When a temperature value is converted to $\int_{0°C}^{T} k\,dt$ and that result used in the inverse function to give the temperature back again, an error of up to 25 degrees K can be introduced. In addition, the MATPRO expressions have a discontinuity at 1650 °C using different expressions below and above this temperature. This adds another complexity to their use.

In an attempt to reduce the errors involved, the expression given in MATPRO for the thermal conductivity of UO_2 was used to generate data pairs of temperature and the corresponding conductivity value. A polynomial regression was performed on the data and a model was obtained for the conductivity as a function of temperature. This is given as:

$$K = .0797465 - .890857\times10^{-4}\,T + .264249\times10^{-7}\,T^2 + .305447\times10^{-10}\,T^3 - .242610\times10^{-13}\,T^4 + .482966\times10^{-17}\,T^5 \qquad \text{IV.2}$$

where K is the thermal conductivity (W/cm°C) and T is the temperature (°C). This fifth order expression gives essentially the same conductivity values as MATPRO but has the advantage of no discontinuities. The expression is valid from 0°C to the UO_2 melting point (2840°C).

To obtain the integral conductivity function, the polynomial expression was simply integrated analytically. For the inverse function, statistical methods were again used. Data pairs of temperature and integral conductivity were generated and regressed to find an expression for the temperature as a function of integral conductivity. The resulting expressions are:

$$I = .0797465\ T - 4.454285\times10^{-5}\ T^2 + 8.8083\times10^{-9}\ T^3 + 7.636175\times10^{-12}\ T^4 - 4.8522\times10^{-15}\ T^5 + 8.049433\times10^{-19}\ T^6 \qquad \text{IV.3}$$

$$T = -37.5779 + 23.7335\ I - .985593\ I^2 + .0452817\ I^3 - .822123\times10^{-3}\ I^4 + .624382\times10^{-5}\ I^5 + .64841\ I^6 - .224516\times10^{-9}\ I^7 \qquad \text{IV.4}$$

where I is the integral conductivity (W/cm) and T is the temperature (°C).

The following table shows a comparison between the standard MATPRO expressions and these new polynomial correlations. The temperature in the first column was converted to $\int_{0°C}^{T} k\ dt$ and then back to temperature using both methods.

T(K)	Using MATPRO	Using Polynomials
500	492	500
700	692	698
900	888	903
1100	1083	1103
1300	1279	1298
1500	1478	1495
1700	1683	1697
1900	1898	1903
2100	2102	2108
2300	2301	2305
2500	2500	2494
2700	2699	2689
2900	2900	2911

Although the MATPRO expressions gave better results above the 1650 °C (1923 K) division, the polynomials were a better choice for lower temperatures and gave satisfactory results over the entire range. The polynomial functions result in simpler programming since one expression is valid over the full range of temperature and are computationally more efficient since evaluating a simple polynomial by looping is generally faster than evaluating exponentials as in MATPRO. These functions were employed in the subroutines XICOND for a single value of $\int_{0°C}^{T} k\,dt$ or the inverse and in VICOND for a vector of values to be transformed.

For the transient case, subroutine GAPSORT stores the values of h_{gap} calculated at each time step for later use in the algorithm. SSQUAD then calculates a steady state profile which is used to give an initial guess for the time

dependent fuel surface temperature. SSHANK gives the Hankel transform of the heat source. In his development, Guidotti assumed a heat source of the form:

$$q'''(r,t) = q'''(r) \cdot f(t) \qquad \text{IV.5}$$

The heat source maintains its initial quadratic shape throughout the transient but decreases in magnitude as the power decays. This allows the use of the same routine to calculate the Hankel transformed heat source regardless of the time.

Subroutine SETLAPC is used to adjust constants used in the numerical Laplace transform routines to allow for transforming at any desired time. The time shifting property of the Laplace transform is shown by:

$$\mathcal{L}\{f(ct)\} = \frac{1}{c}\,\bar{f}\left(\frac{s}{c}\right) \qquad \text{IV.6}$$

This property is used in FREYDF to set the maximum time for the transient at each step. The numerical Laplace transform technique results in a solution at 15 values of time corresponding to the 15 s-domain points used in the method. Each successive time step is added to the first step, giving the solution from the start of the transient out to the current time. With the 0.5 second steps used throughout the transient analysis, FREYDF calculates the response for the first half second at step one, from zero

to one second at step two, zero to 1.5 seconds at the third step, etc. This is possible because the algorithm does not require any more computing time as the transient time is lengthened; the s-domain points are simply spread out to cover the entire transient. The GAPSORT subroutine contains the values of h_{gap} calculated for each half second time step at the beginning of the step. These are incorporated into the fuel-clad interface boundary condition by assigning a value of h_{gap} to each of the 15 time points depending on within which previous time step the current time points fall.

The remainder of the algorithm used to perform the transient calculations is the same that Guidotti used in his single profile calculations. It is interesting to note that the entire process of calculating the transient temperature profiles is based on the fuel surface conditions as a function of time. Knowing the shape of the heat source allows one to calculate the fuel temperature at any location once the surface value of the integral conductivity is known. From the initial guess of the fuel surface temperature as a function of time calculated by SSQUAD, the integral conductivity, I(R,t) and the integral heat capacity, U(R,t) are calculated. The numerical Laplace transform is applied to give $\bar{U}(R,s)$ (subroutine TRANS). This is the first of the three unknowns needed to

solve equation III.17. Next, the material describing function is calculated (DFEXP) using the polynomial relationship between U and I and the assumed exponential form for the surface value of the integral conductivity. With the describing function relating $\bar{\bar{U}}$ and $\bar{\bar{I}}$, a value for $\bar{\bar{I}}_s(\xi_\ell,s)$ is obtained (SETUPI) which is used to generate a new estimate of the describing function input coefficients, A_ℓ and b_ℓ, from a least squares fit of

$$\bar{\bar{I}}_s(\xi_\ell,s) = \frac{A_\ell R^2 J_1^2(\xi_\ell R)}{2(s+b_\ell)} \qquad \text{IV.7}$$

in subroutine FITCOEF. The Laplace domain fuel surface value of the heat flux is calculated in INFLUX and used along with the Laplace transform of the coolant temperature in the linear clad solution in TCLADRI to find the inner clad surface temperature in the Laplace domain.

Subroutines INVERT and SMOOTH give the inverse Laplace transform of the inner clad temperature and the fuel surface heat flux which are used in the fuel-clad interface boundary condition to provide a new estimate of the fuel surface temperature,

$$T_f(R,t) = \frac{q''_f(t)}{h_{gap}(t)} + T_c(R_i,t) \qquad \text{IV.8}$$

A test is made for the convergence of $T_f(R,t)$ and if met, the fuel and clad temperatures are calculated from the

integral conductivity profile by PROFILE and CPROFLE for any desired radial position. If the convergence test fails, the iteration starts again with the new value of T (R,t) modified slightly to accelerate the convergence. This modification is calculated by

$$T_f(R,t)^{i+1} = (1 - ACCEL) \cdot T_f(R,t)^{i-1} + ACCEL \cdot T_f(R,t)^{i} \qquad IV.9$$

where i refers to the iteration number. This acceleration factor changes with the value of the size of the transient step that FREYDF is using. For the first few small steps, ACCEL is typically 0.2. The numerical Laplace transform and its inverse are performed by Bellman's method [7]. The vector of functional values to be transformed is multiplied by a matrix to give the transformed vector. The inverse of this matrix is then used to convert the s-domain vector back to the time domain. As the Laplace constants are shifted to smaller times, the matrix becomes ill-conditioned. This causes oscillations in the time dependent temperatures if a radically different fuel surface temperature is used for the next iteration with the Laplace constants shifted for small times. As the time steps become larger, the value of ACCEL increases to 0.5 where it remains. The changing acceleration factor, obtained from parametric studies of DFTEMP, is used to optimize the iteration process and converge in the fewest

number of passes, usually four or five.

The following section discusses the results obtained by using FREYDF to calculate the fuel behavior following a reactor scram.

V. RESULTS

To assess the applicability of using the describing function heat transfer overlay coupled with the remainder of FREY01, an input file was created that could be used unaltered with FREYDF or FREY01. The input consists of initial values of parameters such as coolant flows, pressures, temperatures, etc.; histories for power, pressure, inlet temperature, mass flow and time; axial power distribution; and the information necessary to generate the finite element mesh. Relevant input data is listed in table V-1. To generate a power history after the scram, the ANS Standard for Decay Heat Source [8] was used. These values are read by FREY01 and used to calculate the volumetric heating in each fuel element throughout the transient. FREYDF also uses the ANS Standard in its calculations of the values for $q'''(t)$ at each of the internal time points.

Figure 7 shows a comparison of the fuel centerline temperature at a location near the axial midplane of the rod (node 91) obtained from four sources. The solid line is from the FREYDF output, the triangles from FREY01, and the crosses and circles from the codes DFTEMP and COYOTE, respectively. These last two codes were employed as a verification method when an error in the FREY01 code

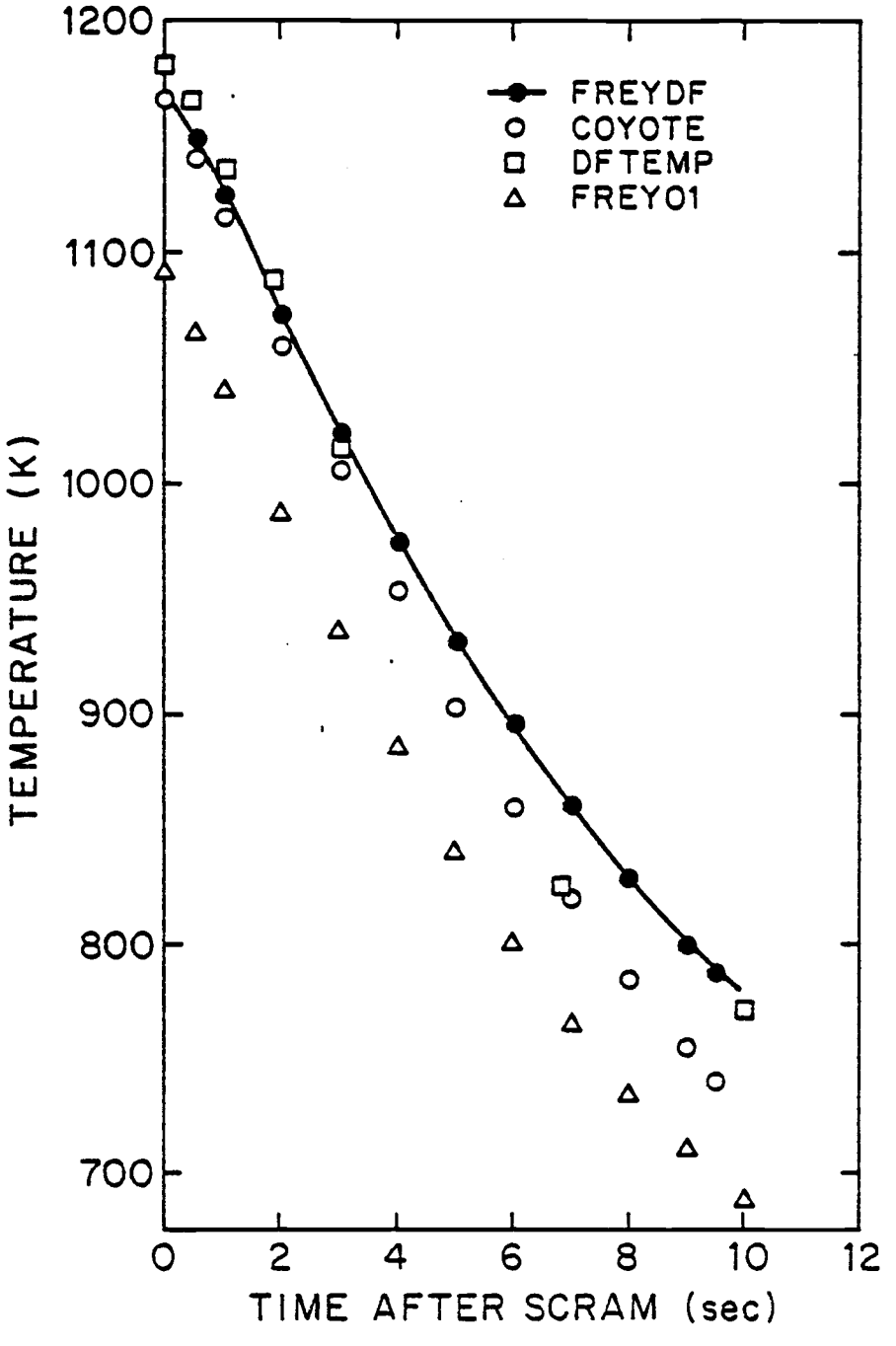
TABLE V-1

INPUT DATA

Fuel radius	.00601 m
Clad inner radius	.00605 m
Clad outer radius	.00699 m
Fuel stack height	3.80 m
Total rod height	4.213 m
Inlet temperature	543.5 K
Inlet pressure	15.21 MPa
Mass flux	3552.63 kg/m -sec
Average linear power	16.0 KW/m
Axial power profile	chopped cosine
Fuel	UO_2
Clad	Zircaloy - 4
Fill gas	He
Number of elements	66 total
fuel	18
clad	13
end caps	6
gap	24
plenum	5
Number of nodes	175

Figure 7

Fuel Centerline Profile
Node 91



initially caused it to give erroneous results. Upon correction, the proper results are as plotted. The problem modeled with COYOTE was a single radial row of two fuel elements, a gap element, and a clad element - a slice out of the full mesh model in FREY01 with the same dimensions. The nodal temperatures were initialized to those obtained by FREYDF just prior to the scram. Material properties were calculated using the appropriate MATPRO routines and the volumetric heating was initialized to the value at the selected element in FREYDF with the power decay again specified by the ANS Standard decay heat source. Although the model was the same dimensions as the segment of the FREY01 mesh, the upper and lower sides were at adiabatic conditions, thereby neglecting axial conduction. Differences between the plots for FREYDF, COYOTE, and DFTEMP are largely a result of the inability to model a changing coolant temperature or a radially varying heat source with COYOTE and using only a rough estimate of the time variation of the coolant temperature and gap conductance values in DFTEMP. In addition, the temperature convergence criterion for the FREYDF iterations was set at five degrees K which could be decreased at the expense of increased running time. This effect, coupled with the errors involved in transforming the temperatures to integral conductivity values and back again, account for

the discrepancies which are at most about six percent.

Figure 8 is a plot of the difference between the FREYDF and FREY01 centerline temperatures throughout the transient. The nearly constant separation between the two is due to the inclusion of axial conduction effects in FREY01 which causes all of its temperatures to be lower than the corresponding values in FREYDF. There is also a difference in the way the heat source is evaluated. FREYDF uses a value interpolated directly from the input axial profile at the desired height. FREY01 uses a value interpolated from the input profile based on the average height of the element. These effects are seen throughout all of the pre-transient time steps and result in an initial profile prior to the scram which is lower for FREY01 than for FREYDF. The important consideration is that both codes show the same shape for the transient profile, demonstrating the ability of the describing function method to accurately account for the non-linear effects.

A plot of the fuel surface temperature vs. time for the two codes is shown in figure 9. Again, the difference between the two plots is nearly constant but of a lower magnitude than at the centerline because of the non-linear nature of the thermal properties and the differences in the radial heat source profiles. Figure 10 shows the initial

Figure 8

Centerline Temperature Difference

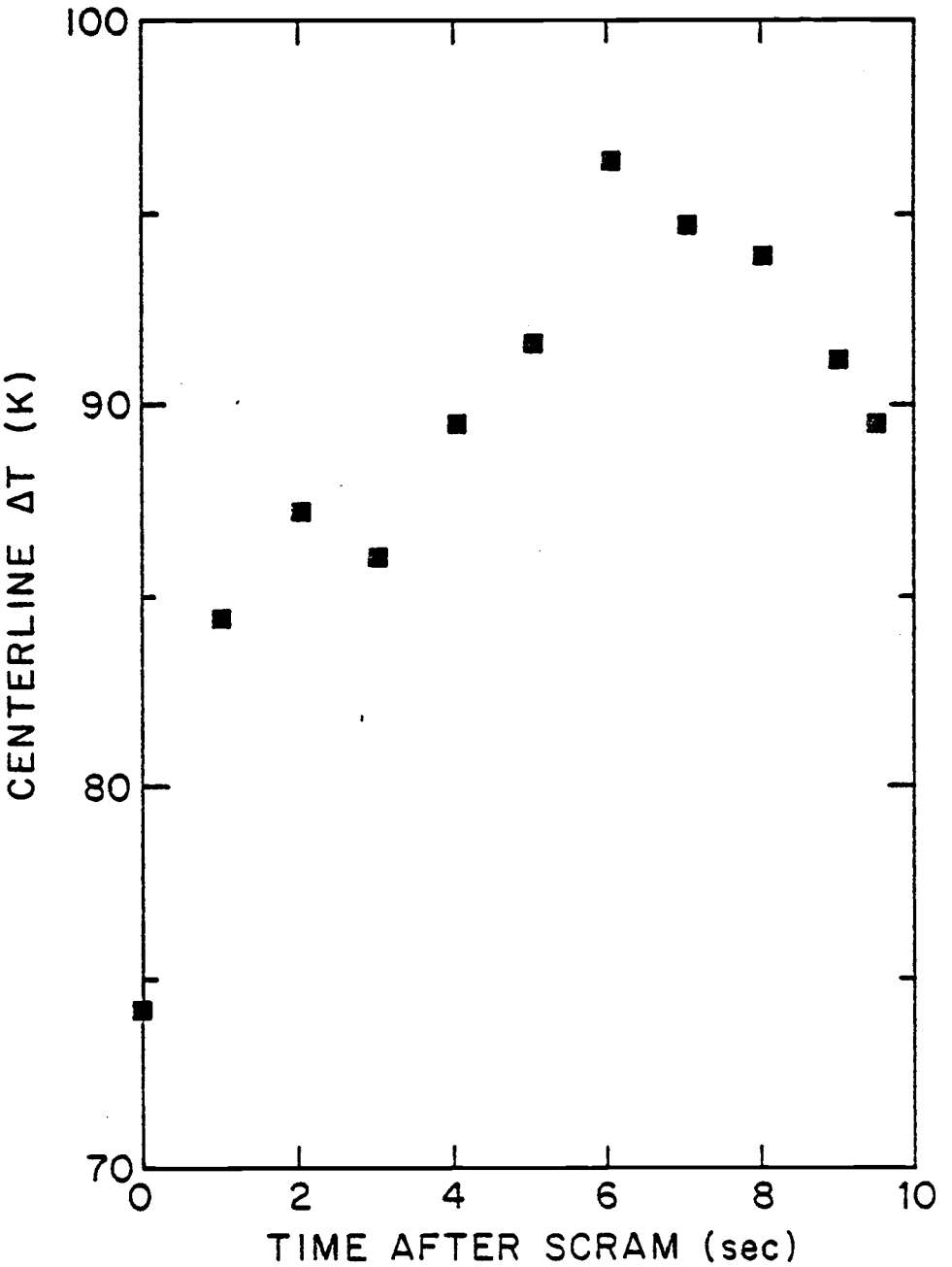


Figure 9

Fuel Surface Profile
Node 95

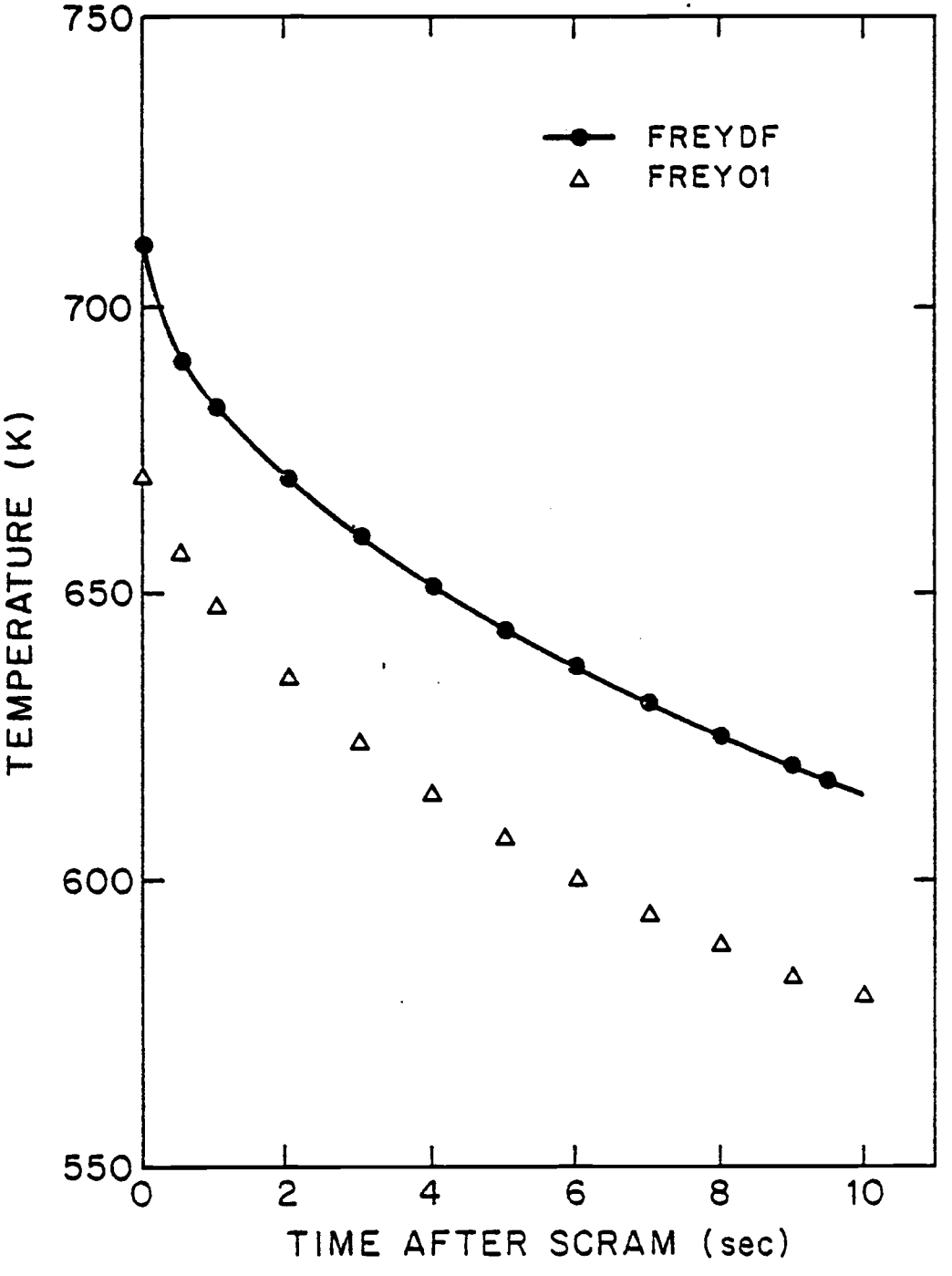
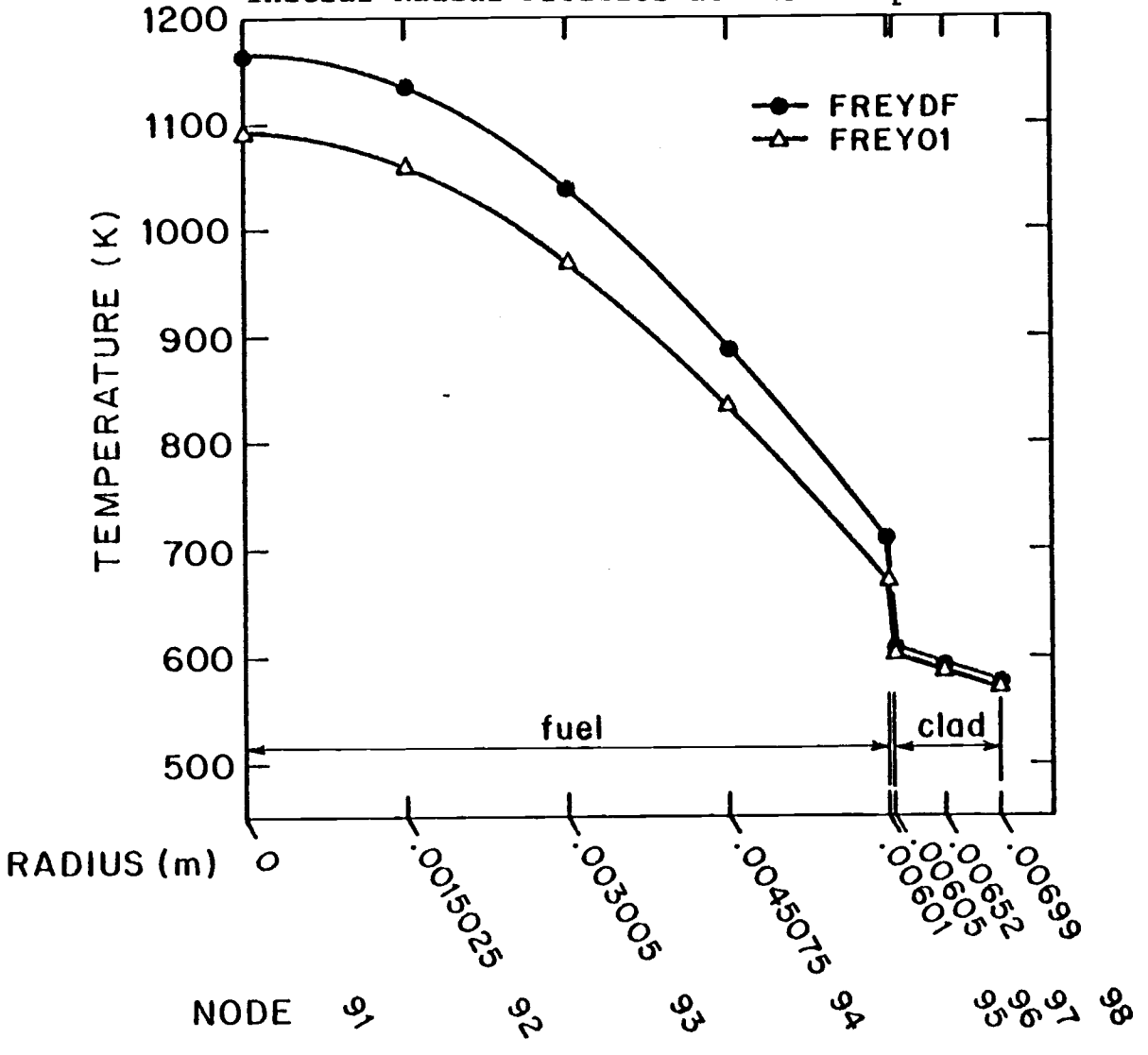


Figure 10
Initial Radial Profiles at Fuel Midplane



radial temperature distribution across the fuel rod. This clearly shows the decreasing difference between the results as the distance from the centerline increases.

Because of the averaging process for the volumetric heating value, FREY01 uses a higher heat source than FREYDF in the elements at the top and bottom of the fuel stack. Although the FREY01 elements are conducting heat to the end caps, this higher source term leads to higher temperatures in these locations as shown in figure 11. The axial profile shows the characteristic shape expected from a chopped cosine axial heat source. The peak temperature for both profiles is just above the midplane because the coolant temperature rises linearly along the rod length.

Figure 12 illustrates the differences in the clad midpoint temperatures vs. time for the two codes. FREYDF assumes temperature independent properties in the clad which leads to an overestimation of the temperature. Using temperature dependent properties, FREY01 shows a departure from the linear case but at 10 seconds is only approximately 15 degrees lower, a third of which is due to the initial difference in temperature.

The running times necessary to generate the solutions by the two methods is illustrated below. Times are averaged for both steady state and transient steps.

Figure 11

Initial Axial Profiles at Fuel Centerline

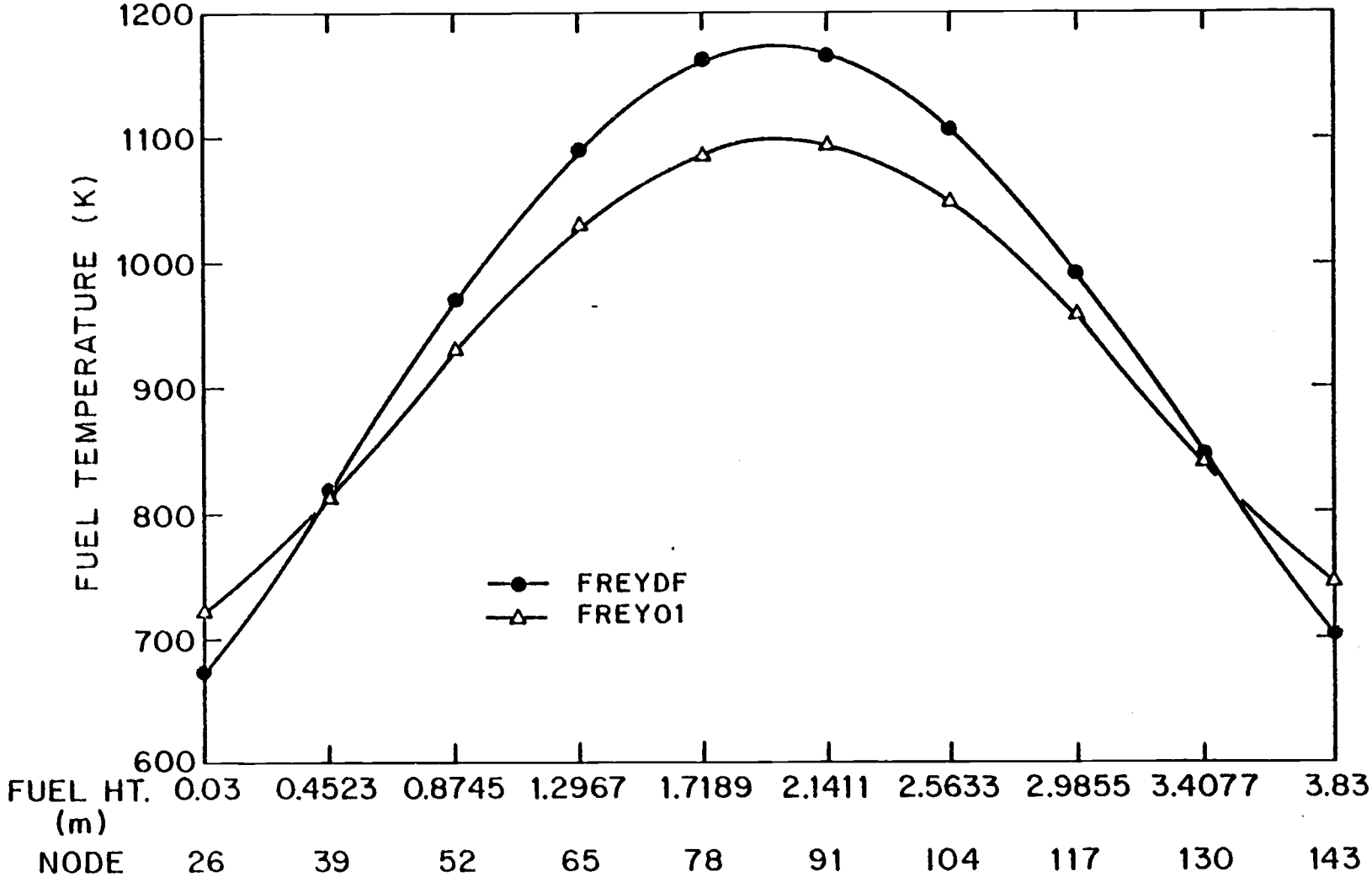
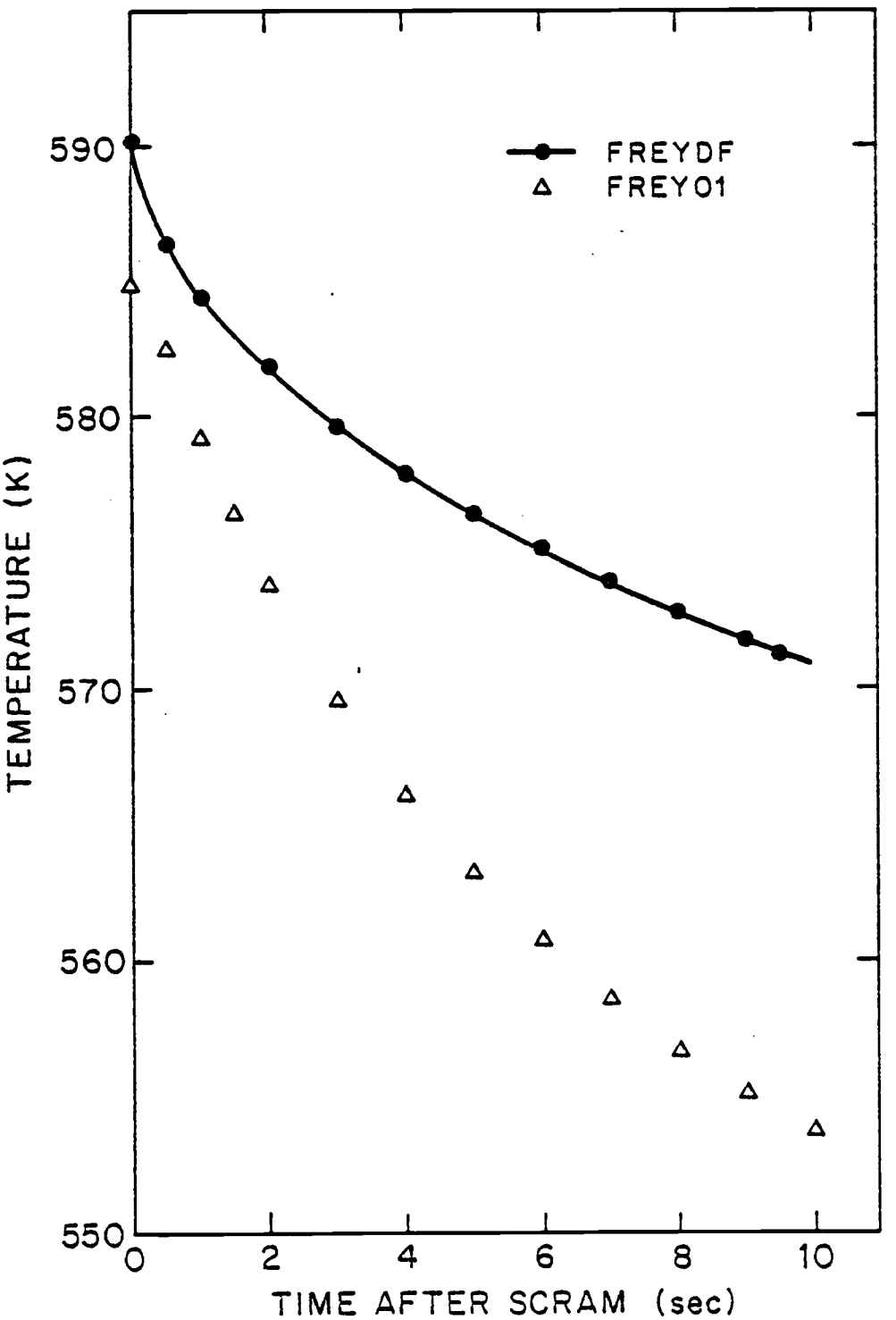


Figure 12

Clad Midpoint Profile
Node 97



	time per step FREY01	FREYDF
Steady State (10 steps)	0.25 sec	0.025 sec
Transient (20 steps)	0.27 sec	0.54 sec

This results in a total computing time of about 8 seconds for all the temperature calculations in FREY01 compared to about 11 seconds for FREYDF. The difference in these times is insignificant when compared to an average of 4.2 seconds per time step to calculate the mechanical response.

VI. CONCLUSION

The results of the previous section show that describing functions can be used to calculate the thermal response of a reactor fuel pin in a fuel performance code. The accuracy of the method is comparable to the finite element formulation but the running times are somewhat longer. The major advantage of the describing function technique is the development of an analytical solution to the time dependent, quasi-linearized, partial differential equations for heat transfer, making the conventional time integrations at each solution step unnecessary. This advantage was offset by the need to interact with the mechanical response routines at each half second interval. The full efficiency of this method might be realized by deriving describing functions for the mechanics formulations. Mathematical relationships are available for all of the physical features that would require consideration such as the swelling, strain rates, and stresses. An estimate of the expected response could be made about which the quasi-linearization process would lead to a describing function for each relationship, reducing those expressions to algebraic ones. These could then be solved simultaneously with the thermal response, giving the mechanical and thermal behavior in one solution step.

Although this process seems straightforward, there is a great deal of analytical work necessary before the method could be used in a computer program. This "front end" work is the trade-off for savings in computing time.

FREYDF is a very specialized code. It will only calculate the proper temperatures if the actual transient response is similar to the assumed form of the exponential input describing function. To be used as a multi-purpose heat transfer code, a generalized describing function would be necessary to adequately model the physical situation. In addition, there are limitations built into the code, because of the analytical work done to develop the algorithm, which make it suitable only for analyzing special cases. An example of this is the constants used in the exponential least squares fit of the coefficients and time constants for the input to the describing function. The constants are a result of integrating $rJ_0^3(\xi_\ell r)$ over the fuel region from 0 to R. Because of this, if the fuel pin size is changed, a new integration is needed. Although this could easily by done numerically within the code, it points out the need to generalize the algorithm.

Even with these limitations, the steady state routines in FREYDF could be very useful for the steady state solution routine in FREY01. The inclusion of the coolant enthalpy model and a method for properly treating the heat

transfer to the end caps would not significantly increase the running time and would produce a steady state solution much faster than the Picard iteration used now. It would also be desirable to investigate the optimization of the computational methods used in FREYDF. The routines used were not necessarily programmed for the greatest efficiency in time or storage requirements. If the describing function could be generalized, the transient portion of FREYDF would also be a valuable tool to be used for rough calculations where the approximate values of the gap conductance are known before hand. This would eliminate the need to interact with the mechanical routines at each time step. For example, for a transient such as a rapid power decrease where the fuel is known to shrink, thus opening the gap further, bounds could be set on the values for the gap conductance and FREYDF could be run with an extremely long time step. Test runs have been made using DFTEMP with step sizes of a few minutes with no loss of accuracy. Regardless of the step size, all of the thermal response information back to time zero is available in one step. In this way, very slow transients could easily be analyzed without the need for costly small time step sizes necessary with finite differencing methods.

VII. REFERENCES

1. Timothy Guidotti, "Transient Fuel Pin Temperature Calculations Using Describing Functions", OSU-NE-8003, 1980.

2. David K. Gartling, "COYOTE--A Finite Element Computer Program for Nonlinear Heat Conduction Problems", SAND77-1332, 1978.

3. Y.R. Rashid and M.N. Sharabi, "Analytical Methods for Transient Fuel Behavior", FREY Code Architecture, ANA79-1, 1979.

4. Randy J. James, Robert E. Nickell, and Robert S. Dunham, "Integration of the COYOTE Heat Transfer Program Into the FREY Fuel Pin Program", Final Report to EPRI, PT-U79-0403, 1979.

5. Bruce A. Finlayson, The Method of Weighted Residuals and Variational Principles, (New York: Academic Press, 1972).

6. Donald L. Hagrman and Gregory A Reymann, ed., "MATPRO - Version 10, A Handbook of Materials Properties for Use in the Analysis of Light Water Reactor Fuel Rod Behavior,", EG&G Idaho, Inc., TREE-NUREG-1180, 1978.

7. Richard Bellman, et: al., Numerical Inversion of the Laplace Transform (New York: American Elsevier Publishing Co., Inc., 1966).

8. ANS/ANSI 5.1, American Nuclear Society/American National Standards Institute standard ANS 5.1, "Decay Heat Power in Light Water Reactors", June 1978, ANS, 555 N. Kensington Ave., Lagrange Park, IL 60525.