JAERI-M
90-122

IMPLEMENTATION OF AN IMPLICIT METHOD INTO HEAT CONDUCTION CALCULATION OF TRAC-PF1/MOD2 CODE

August 1990

Hajime AKIMOTO, Yutaka ABE, Akira OHNUKI
and Yoshio MURAO

日 本 原 子 力 研 究 所
Japan Atomic Energy Research Institute

JAERI-M レポートは，日本原子力研究所が不定期に公刊している研究報告書です。

入手の問合わせは，日本原子力研究所技術情報部情報資料課（〒319－11 茨城県那珂郡東海村）あて，お申しこしください。なお，このほかに財団法人原子力弘済会資料センター（〒319-11 茨城県那珂郡東海村日本原子力研究所内）で複写による実費頒布をおこなっております。

JAERI-M reports are issued irregularly.

Inquiries about availability of the reports should be addressed to Information Division, Department of Technical Information, Japan Atomic Energy Research Institute, Tokai-mura, Naka-gun, Ibaraki-ken 319-11, Japan.

編集兼発行　日本原子力研究所
印　　刷　山田軽印刷所

Implementation of an implicit method into heat conduction calculation of TRAC-PF1/MOD2 code

Hajime AKIMOTO, Yutaka ABE, Akira OHNUKI and Yoshio MURAO

Department of Reactor Engineering
Tokai Research Establishment
Japan Atomic Energy Research Institute
Tokai-mura, Naka-gun, Ibaraki-ken

(Received July 7, 1990)

A two-dimensional unsteady heat conduction equation is solved in the TRAC-PF1/MOD2 code to calculate temperature transients in fuel rod. A large CPU time is often required to get stable solution of temperature transients in the TRAC calculation with a small axial node size (less than 1.0mm), because the heat conduction equation is discretized explicitly. To eliminate the restriction of the maximum time step size by the heat conduction calculation, an implicit method for solving the heat condition equation was developed and implemented into the TRAC code. Several assessment calculations were performed with the original and modified TRAC codes. It is confirmed that the implicit method is reliable and is successfully implemented into the TRAC code through comparison with theoretical solutions and assessment calculation results. It is demonstrated that the implicit method makes the heat conduction calculation practical even for the analyses of temperature transients with the axial node size less than 0.1mm.

Keywords: Reactor Safety, PWR, CCTF, SCTF, Reflood, Implicit Method, Heat Conduction, TRAC-PF1/MOD2

陰解法による熱伝導方程式解析ルーチンの TRAC－PF1/MOD2コードへの組み込み

日本原子力研究所東海研究所原子炉工学部
秋本　肇・阿部　豊・大貫　晃・村尾　良夫

（1990年7月7日受理）

TRAC－PF1/MOD2コードでは，燃料棒内の温度分布計算のために，2次元非定常熱伝導方程式が差分法により解かれる。差分法として一部に陽解法が用いられているため，小さなノード分割を用いる計算では，安定な解を得るために小さなタイムステップサイズを用いる必要があり，多大な計算時間を必要としていた。大きなタイムステップサイズでも安定に計算できるようにするために，二次元非定常熱伝導方程式の陰解法ルーチンを，TRAC－PF1/MOD2コードに組み込んだ。陰解法ルーチンを組み込んだ修正版とオリジナル版を用い評価計算を行った。解析解並びにオリジナルのTRAC－PF1/MOD2コードの結果との比較により，今回整備した陰解法ルーチンが信頼できるものであり，TRAC－PF1/MOD2コードに正しく組み込まれていることを確認した。陰解法ルーチンの組み込みにより，ノード分割が0.1 mm以下にしても効率よく燃料棒内の温度分布を計算できるようになった。

東海研究所：〒319-11　茨城県那珂郡東海村白方字白根2－4

Contents

1. INTRODUCTION

The TRAC-PF1 code is a best-estimate code being developed at Los Alamos National Laboratory (LANL) to provide advanced best-estimate predictions of postulated accidents in light-water reactors and of thermal-hydraulic behaviors in many test facilities[1]. The code features either one or three dimensional treatment based on a two-fluid non-equilibrium hydrodynamic model with flow-regime-dependent constitutive treatment. The stability-enhancing two-step (SETS) numerical algorithm is used in the code and permits to violate the mathematical Courant condition. The code can simulate various important thermal-hydraulic phenomena in light water reactors.

In the TRAC-PF1 code, the temperature transients of fuel rods and heat structures are calculated solving a two dimensional unsteady heat conduction equation. The heat conduction equation is solved by a finite difference method with dynamic-mesh rezoning in axial direction. The heat conduction equation is discretized explicitly in axial direction while it is implicitly in radial direction. Since the solution method of the two-dimensional heat conduction equation in the present TRAC-PF1/MOD2 is explicit in the axial direction, the time step size Δt is restricted to be less than $(\Delta Z)^2/2D$ in order to get a stable solution with the explicit method, where ΔZ and D are noding size and thermal diffusivity, respectively.

The solution method mentioned above may restrict Δt very small if small ΔZ is required. Actually, Dr. P. Coddington and Mr. R. O'Mahoney pointed out that fine noding size less than 1.0 mm must be allowed to obtain a converged axial-conduction solution when reflood phenomena is analyzed with TRAC-PF1/MOD2 moving-mesh axial-conduction model, because the axial temperature profile in the vicinity of the quench front is very steep[2]. The very fine axial noding to get accurate solution of the axial temperature profile requires time-step size less than 1 msec and results in large CPU time. If the heat conduction equation is solved fully implicitly, the restriction of time-step size can be eliminated and CPU time can be reduced.

Japan Atomic Energy Research Institute (JAERI) conducted the modification of the TRAC-PF1/MOD2 to install the fully implicit solution method of heat conduction equation developed as one of contributions of JAERI in the code improvement program organized by the United State Nuclear Regulatory Commission. The installation work of the implicit heat conduction model into TRAC-PF1/MOD2 code was conducted at LANL by JAERI from December 5 to 9 in 1988.

This report describes the details of the implicit solution method of two dimensional heat conduction equation, modifications of TRAC-PF1/MOD2 code to install the implicit solution method, and the results of assessment calculation for the implicit solution method, which was done at JAERI and at LANL.

2. FINITE DIFFERENTIAL EQUATION

With a cell noding shown in Fig. 1, the differential equation of two-dimensional unsteady heat-conduction can be discretized as shown below if an implicit form is applied in axial direction as well as in radial direction:

$$
\begin{aligned}
& (\rho Cp)_{i,j} \; ((T_{i,j}^{n+1} - T_{i,j}^{n})/\Delta t) \; V_{i,j} \\
= \; & q_{i,j}{'''} V_{i,j} \\
& + k_{i+1/2,j} \; ((T_{i+1,j}^{n+1} - T_{i,j}^{n+1})/ \; \Delta r_i \;) \; A_{i+1/2} \\
& + k_{i-1/2,j} \; ((T_{i-1,j}^{n+1} - T_{i,j}^{n+1})/\Delta r_{i-1}) \; A_{i-1/2} \\
& + k_{i,j+1/2} \; ((T_{i,j+1}^{n+1} - T_{i,j}^{n+1})/ \; \Delta Z_j \;) \; A_i^{*} \\
& + k_{i,j-1/2} \; ((T_{i,j-1}^{n+1} - T_{i,j}^{n+1})/\Delta Z_{j-1}) \; A_i^{*} \qquad (1)
\end{aligned}
$$

where

ρ	: density	(kg/m^3),
Cp	: specific heat	(J/kgK),
T	: temperature	(K),
V	: cell volume	(m^3),
q'''	: heat generation rate per volume	(W/m^3),
k	: thermal conductivity	(W/mK),
A	: area in radial direction	(m^2),
A*	: area in axial direction	(m^2),
Δt	: time-step size	(s),
Δr	: cell length in radial direction	(m),
ΔZ	: cell length in axial direction	(m),

code improvement program organized by the United State Nuclear Regulatory Commission. The installation work of the implicit heat conduction model into TRAC-PF1/MOD2 code was conducted at LANL by JAERI from December 5 to 9 in 1988.

This report describes the details of the implicit solution method of two dimensional heat conduction equation, modifications of TRAC-PF1/MOD2 code to install the implicit solution method, and the results of assessment calculation for the implicit solution method, which was done at JAERI and at LANL.

2. FINITE DIFFERENTIAL EQUATION

With a cell noding shown in Fig. 1, the differential equation of two-dimensional unsteady heat-conduction can be discretized as shown below if an implicit form is applied in axial direction as well as in radial direction:

$$
\begin{aligned}
&(\rho Cp)_{i,j}\ ((T_{i,j}^{n+1}-T_{i,j}^{n})/\Delta t)\ V_{i,j} \\
= \ & q_{i,j}'''V_{i,j} \\
&+ k_{i+1/2,j}\ ((T_{i+1,j}^{n+1}-T_{i,j}^{n+1})/\ \Delta r_i\ )\ A_{i+1/2} \\
&+ k_{i-1/2,j}\ ((T_{i-1,j}^{n+1}-T_{i,j}^{n+1})/\Delta r_{i-1})\ A_{i-1/2} \\
&+ k_{i,j+1/2}\ ((T_{i,j+1}^{n+1}-T_{i,j}^{n+1})/\ \Delta Z_j\ )\ A_i^* \\
&+ k_{i,j-1/2}\ ((T_{i,j-1}^{n+1}-T_{i,j}^{n+1})/\Delta Z_{j-1})\ A_i^* \qquad (1)
\end{aligned}
$$

where

ρ	: density	(kg/m^3),
Cp	: specific heat	(J/kgK),
T	: temperature	(K),
V	: cell volume	(m^3),
q'''	: heat generation rate per volume	(W/m^3),
k	: thermal conductivity	(W/mK),
A	: area in radial direction	(m^2),
A*	: area in axial direction	(m^2),
Δt	: time-step size	(s),
Δr	: cell length in radial direction	(m),
ΔZ	: cell length in axial direction	(m),

i : cell index in radial direction,

j : cell index in axial direction,

n : index of old time-step,

n+1 : index of new time-step.

Equation (1) can be described as follows;

$$a_{1,i,j}T_{i-1,j}^{n+1} + a_{2,i,j}T_{i,j}^{n+1} + a_{3,i,j}T_{i+1,j}^{n+1} + a_{4,i,j}T_{i,j-1}^{n+1} + a_{5,i,j}T_{i,j+1}^{n+1} = b_{i,j} \quad (2)$$

where

$$a_{1,i,j} = -k_{i-1/2,j}\, A_{i-1/2}/\Delta r_{i-1}$$

$$a_{2,i,j} = (\rho Cp)_{i,j}\, V_{i,j}/\Delta t + k_{i+1/2,j}A_{i+1/2}/\Delta r_i + k_{i-1/2,j}A_{i-1/2}/\Delta r_{i-1} + k_{i,j+1/2}A_i^*/\Delta Z_j + k_{i,j-1/2}A_i^*/\Delta Z_{j-1}$$

$$a_{3,i,j} = -k_{i+1/2,j}\, A_{i+1/2}/\Delta r_i$$

$$a_{4,i,j} = -k_{i,j-1/2}\, A_i^*/\Delta Z_{j-1}$$

$$a_{5,i,j} = -k_{i,j+1/2}\, A_i^*/\Delta Z_j$$

$$b_{i,j} = \{q_{i,j}''' + (\rho Cp)_{i,j}T_{i,j}^n/\Delta t\}\, V_{i,j} \quad (3)$$

Equation (2) is rewritten with matrix as follows;

$$A \cdot T = B \quad (4)$$

or

$$\begin{bmatrix} D^{(1)} & E^{(1)} & & & & & 0 \\ C^{(2)} & D^{(2)} & E^{(2)} & & & & \\ & C^{(3)} & D^{(3)} & E^{(3)} & & & \\ & & \cdot & & & & \\ & & & \cdot & & & \\ & & & & \cdot & & \\ 0 & & & & & \cdot & \\ & & & & & \cdot & \\ & & & C^{(NZ-1)} & D^{(NZ-1)} & E^{(NZ-1)} \\ & & & & C^{(NZ)} & D^{(NZ)} \end{bmatrix} \begin{bmatrix} T^{(1)} \\ T^{(2)} \\ T^{(3)} \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ T^{(NZ-1)} \\ T^{(NZ)} \end{bmatrix} = \begin{bmatrix} B^{(1)} \\ B^{(2)} \\ B^{(3)} \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ B^{(NZ-1)} \\ B^{(NZ)} \end{bmatrix} \quad (5)$$

where

$$T^{(j)} = \begin{bmatrix} T_{1,j} \\ T_{2,j} \\ . \\ . \\ T_{NR,j} \end{bmatrix} \qquad B^{(j)} = \begin{bmatrix} b_{1,j} \\ b_{2,j} \\ . \\ . \\ b_{NR,j} \end{bmatrix} \qquad (j = 1,\ NZ) \tag{6}$$

$$C^{(j)} = \begin{bmatrix} a_{4,1,j} & & & 0 \\ & a_{4,2,j} & & \\ & & \cdot & \\ & 0 & & \cdot \\ & & & a_{4,NR,j} \end{bmatrix} \qquad (j=2,NZ) \tag{7}$$

$$D^{(j)} = \begin{bmatrix} a_{2,1,j} & a_{3,1,j} & & & 0 \\ a_{1,2,j} & a_{2,2,j} & a_{3,2,j} & & \\ & a_{1,3,j} & a_{2,3,j} & a_{3,3,j} & \\ & & \cdot & & \\ & 0 & & \cdot & \\ & & & a_{1,NR,j} & a_{2,NR,j} \end{bmatrix} \qquad (j=1,NZ) \tag{8}$$

$$E^{(j)} = \begin{bmatrix} a_{5,1,j} & & & 0 \\ & a_{5,2,j} & & \\ & & \cdot & \\ & 0 & & \cdot \\ & & & a_{5,NR,j} \end{bmatrix} \qquad (j=1,NZ-1) \tag{9}$$

Matrix A is a symmetrical band matrix.

The treatment at gap is also modified as described below. The continuity relationship of heat flux between gap inner surface and gap outer surface is

$$A_{gap}^{-} q_{gap}''^{-} = A_{gap}^{+} q_{gap}''^{+}, \tag{10}$$

where A_{gap}^{-}, A_{gap}^{+}, $q_{gap}''^{-}$ and $q_{gap}''^{+}$ are inner surface area of the gap, outer surface area of the gap, heat flux at inner surface of the gap and heat flux at outer surface of the gap,respectively. When gap conductance is taken as $h_{gap,j}$, Eq.(10) should be equal to

$$2\pi\cdot(r_{gap}^{-}+r_{gap}^{+})/2\cdot(\Delta Z_j+\Delta Z_{j-1})/2\cdot h_{gap,j}\cdot(T_{gap,j}^{-n+1}-T_{gap,j}^{+n+1}). \quad (11)$$

These changes are corresponding to the following replacement of second and third terms of right-hand side of Eq.(1).

$$k_{i+1/2,j}\ (T_{i+1,j}^{n+1} - T_{i,j}^{n+1})\ A_{i+1/2}/\Delta r_i$$
$$\rightarrow\ h_{gap,j}\ (\ T_{gap,j}^{-n+1} - T_{gap,j}^{+n+1}\ )\ A_{i+1/2}, \quad (12)$$

and

$$k_{i-1/2,j}\ (T_{i-1,j}^{n+1} - T_{i,j}^{n+1})\ A_{i-1/2}/\Delta r_{i-1}$$
$$\rightarrow\ h_{gap,j}\ (\ T_{gap,j}^{-n+1} - T_{gap,j}^{+n+1}\ )\ A_{i-1/2}. \quad (13)$$

In the code, the coefficients $a_{3,i,j}$ and $a_{1,i,j}$ in Eq.(3) are changed as follows, based on the above replacement,

$$a_{3,i,j}=h_{gap,j}\cdot A_{i+1/2}\ /\ \Delta r_i \qquad \text{at gap inner surface,} \quad (14)$$

and

$$a_{1,i,j}=h_{gap,j}\cdot A_{i-1/2}\ /\ \Delta r_{i-1} \qquad \text{at gap outer surface.} \quad (15)$$

Above modification makes the matrix D in Eq.(5) symmetric and accordingly the matrix A in Eq.(5) becomes symmetric. Although the coefficient $a_{2,i,j}$ is also changed, the change does not have any effect on the symmetry of the matrix D in Eq.(5) because $a_{2,i,j}$ is a diagonal part of the matrix D in Eq.(5). In case that the matrix A is symmetric, fast numerical solution procedure such as Cholesky resolution method can be adopted to reduce the number of the numerical operations.

3. SOLUTION METHOD

To get solution of Eq.(4), it is required to get an inversion of matrix A. Since the matrix A is a symmetrical matrix, modified Cholesky method is applicable to get inversion of matrix A. Once the matrix A is inverted, the solution T is obtained by

$$T = A^{-1}\ B. \quad (16)$$

tance is taken as $h_{gap,j}$, Eq.(10) should be equal to

$$2\pi \cdot (r_{gap}^{-}+r_{gap}^{+})/2 \cdot (\Delta Z_j+\Delta Z_{j-1})/2 \cdot h_{gap,j} \cdot (T_{gap,j}^{-n+1}-T_{gap,j}^{+n+1}). \qquad (11)$$

These changes are corresponding to the following replacement of second and third terms of right-hand side of Eq.(1).

$$k_{i+1/2,j}\ (T_{i+1,j}^{n+1} - T_{i,j}^{n+1})\ A_{i+1/2}/\Delta r_i$$
$$\rightarrow\ h_{gap,j}\ (\ T_{gap,j}^{-n+1} - T_{gap,j}^{+n+1}\ )\ A_{i+1/2}, \qquad (12)$$

and

$$k_{i-1/2,j}\ (T_{i-1,j}^{n+1} - T_{i,j}^{n+1})\ A_{i-1/2}/\Delta r_{i-1}$$
$$\rightarrow\ h_{gap,j}\ (\ T_{gap,j}^{-n+1} - T_{gap,j}^{+n+1}\ )\ A_{i-1/2}. \qquad (13)$$

In the code, the coefficients $a_{3,i,j}$ and $a_{1,i,j}$ in Eq.(3) are changed as follows, based on the above replacement,

$$a_{3,i,j}=h_{gap,j} \cdot A_{i+1/2}\ /\ \Delta r_i \qquad \text{at gap inner surface,} \qquad (14)$$

and

$$a_{1,i,j}=h_{gap,j} \cdot A_{i-1/2}\ /\ \Delta r_{i-1} \qquad \text{at gap outer surface.} \qquad (15)$$

Above modification makes the matrix D in Eq.(5) symmetric and accordingly the matrix A in Eq.(5) becomes symmetric. Although the coefficient $a_{2,i,j}$ is also changed, the change does not have any effect on the symmetry of the matrix D in Eq.(5) because $a_{2,i,j}$ is a diagonal part of the matrix D in Eq.(5). In case that the matrix A is symmetric, fast numerical solution procedure such as Cholesky resolution method can be adopted to reduce the number of the numerical operations.

3. SOLUTION METHOD

To get solution of Eq.(4), it is required to get an inversion of matrix A. Since the matrix A is a symmetrical matrix, modified Cholesky method is applicable to get inversion of matrix A. Once the matrix A is inverted, the solution T is obtained by

$$T = A^{-1} B. \qquad (16)$$

4. CODING CHANGE

In the followings, the modified parts in the TRAC-PF1/MOD2 code for changing the solution method of axial heat conduction from explicit to implicit are described. Table 1 shows the tree of the related subroutines. In Table 1, the subroutine INPUT is a control routine in the input overlay, and subroutines CORE3 and HTSTR3 are control routines for core component and heat structure component in postpass overlay, respectively.

4.1 Added common blocks

The following common blocks are added to specify the option parameter to switch the solution method of heat conduction between the explicit method and the implicit method by user input, and to keep the working area for the implicit heat conduction calculation.

```
      PARAMETER ( JRZMAX= 100 , JRRMAX=20 )
      COMMON / CNRSLV / NRSLV
      COMMON / CNRSLV / AR(JRRMAX+1,JRZMAX*JRRMAX) ,
     *                  BB(JRRMAX,JRZMAX) ,
     *                  WW(JRRMAX,JRZMAX)
```

The meaning of each variables in the added commons is listed below:

JRZMAX = maximum number of axial nodes,
JRRMAX = maximum number of radial nodes,
AR = array to store the matrix elements of heat conduction equation,
BB = array to store the constant vector of heat conduction equation,
WW = array to solve the symmetrical band matrix.

Parameters JRZMAX and JRRMAX define the dimension of the arrays AR, BB and WW. If a user inputs the number of nodes more than JRZMAX or JRRMAX, the job will be stopped with warning message. If a user needs more nodes than the default values of JRZMAX and JRRMAX, it is necessary to change the values in the PARAMETER statement and compile again.

NRSLV is an option flag that allows users to select the solution method for heat conduction calculation; If NRSLV=0, then explicit method will be selected. If NRSLV=1, then implicit method will be selected. NRSLV is set in the namelist INOPTS. If a user sets non-zero or non-unity value, the job is executed with a default value, i.e. NRSLV=0.

4.2 New subroutine

A new subroutine BANSOL is added to the TRAC-PF1/MOD2 code.

(1) SUBROUTINE:BANSOL

```
SUBROUTINE BANSOL ( MDIM , N , M1 , AR , B , W , KEY , ERR )
DIMENSION AR(MDIM,1) , B(1) , W(1)
LOGICAL ERR
```

This subroutine is devoted to solve the following equation;

A x = B,

where A is the symmetrical matrix of Nth dimension with half band length of M.

The meaning of the argument of this subroutine is listed below;

MDIM = Length of array AR.

N = Dimension of the equation.

M1 = M+1 (half band length + 1).
Necessary length of array AR is M1*N.

AR = Coefficient matrix which stores the results of Cholesky resolution.
The lower part of the symmetrical matrix is stored. Actual relationship between coefficient matrix A and AR is as follows;
AR(L,K) = A(K,K+L-M-1).

B = Constant vector or solution vector.

KEY = Option flag to solution procedure;
1 = get solution,
2 = only forward elimination,
3 = only back substitution.

ERR = Error flag. When ERR is true, matrix is singular.

This subroutine is called by subroutine RODHT.

4.3 Modified subroutines

Modified subroutines and modifications are described below;

(1) SUBROUTINE; BLKDAT

This subroutine sets the initial values with DATA statements. The common block CNRSLV and the following DATA statement are added to set the default value of NRSLV;

DATA NRSLV / 0 /

(2) SUBROUTINE; COREC1

In order to compare the computational results with the analytical solution, this subroutine was modified to force a heat transfer coefficient a constant value in particular assessment calculation. This modification was deleted in final modified version after completion of assessment calculations.

(3) SUBROUTINE; COREC3

This subroutine is the interface routine to solve the heat conduction equation for fuel rods of 1-D CORE component.

The followings are modified:

i) Add common CNRSLV,

ii) Add statements to check the size of work area and

iii) change arguments to call subroutine FROD.

(4) SUBROUTINE; CORE3

This subroutine is the interface routine to solve the heat conduction equation for heat structure components.

The modified parts are

i) Add arguments AR,BB and WW and dimension statements and

ii) Add arguments AR,BB and WW to call subroutine FROD.

(5) SUBROUTINE; FROD

This subroutine calculates heat generation rate by reaction between zircaloy and water and calculates of gap conductance, and calls subroutine RODHT in which the heat conduction equation is solved.

The modification parts are

i)Addition of arguments AR,BB and WW in this subroutine and addition of dimension statements of these variables and

ii)Addition of arguments AR,BB and WW to call subroutine RODHT.

(6) SUBROUTINE; HTSTR3

This subroutine is the control routine to solve the heat conduction equation for heat structure component.

The modified parts are

i) Addition of common CNRSLV,

ii) Checking the size of work area and

iii) Addition of arguments AR,BB and WW to call subroutine CORE3.

(7) SUBROUTINE; INPUT

This subroutine is the control routine for reading input data.

The modified parts are

i) Add common CNRSLV,

ii) Add the variable NRSLV into NAMELIST INOPTS and

iii) Read NRSLV in NAMELIST.

(8) SUBROUTINE; RODHT

Heat conduction equation is solved in this subroutine. Original routine for the explicit method was rewrite using standard FORTRAN77. In this routine, the statements for the implicit method were added. The code structure and differential method were not changed except for the treatment of differential scheme at gap in fuel rods as described in Section 2.

Since the time step control is not necessary in the implicit method, the variable DIFMIN is set to 1.E8 if the implicit method is used.

5. ASSESSMENT CALCULATIONS AT JAERI

To check the reliability of the code modification, four cases of assessment calculations were performed with the implicit method in JAERI: that is,

Case 1 : One dimensional heat conduction problem in radial direction at steady state,

Case 2 : One dimensional heat conduction problem in axial direction at steady state,

Case 3 : Two dimensional heat conduction problem at steady state,

Case 4 : Two dimensional heat conduction problem at transient case.

Assessment calculations of cases 1, 2 and 3 were conducted with modified TRAC-PF1/MOD2 in which the implicit heat conduction calculation method is installed. In cases 1, 2 and 3, numerical results will be compared with theoretical solutions to check whether coding is performed accurately or not.

Assessment calculation of case 4 was conducted with modified TRAC-PF1/MOD1 in which the same implicit heat conduction calculation method is installed, because conversion of 1-D VESSEL component in TRAC-PF1/MOD2 CRAY-version into FACOM-version had not finished yet in JAERI when this assessment calculations were conducted in JAERI. Assessment calculation of case 4, however, provides useful information to discuss on CPU time which indicates a benefit of the implicit method through comparisons with results from the explicit method.

These assessment calculations were performed with a FACOM M-780 computer at JAERI.

5.1 One dimensional heat conduction problem in radial direction at steady state

The differential equation for one dimensional heat conduction problem in radial direction is given by

$$1/r \, d/dr(rk(dT/dr)) = -q''' \tag{17}$$

Figure 2 shows a schematic where an analytical solution is obtained. The boundary conditions are listed below.

$$dT/dr = 0 \quad \text{at } r = 0,$$

$$-k(dT/dr) = h_3 (T_3 - T_4) \quad \text{at } r = r_3,$$

$$q''' = \begin{cases} q_0 & (0 \leqq r \leqq r_1), \\ 0 & (r_1 < r), \end{cases} \tag{18}$$

$$h = h_{gap} \quad (\text{at gap}).$$

The solution is given by

$$T_3 = Q/(2\pi r_3 l h_3) + T_4, \tag{19}$$

$$T_2 = T_3 - q_1 r_1 \ln(r_2/r_3)/k_2, \tag{20}$$

$$T_1 = T_2 + q_1/h_{gap}, \tag{21}$$

$$T_0 = T_1 + q_1 r_1/2k_1, \tag{22}$$

$$q_1 = Q/(2\pi r_1 l), \tag{23}$$

$$T = \begin{cases} T_3 - q_1 r_1 \ln(r/r_3)/k_2 & (r_2 < r < r_3) \\ T_1 + q_1 (r_1{}^2 - r^2)/(2k_1 r_1) & (r < r_1) \end{cases} \tag{24}$$

where

- h_3 : heat transfer coefficient at rod surface, (W/m^2K)
- T_4 : fluid temperature, (K)
- Q : total power, (W)
- l : length of rod, (m)
- k : thermal conductivity. (W/(mK))

A numerical calculation was performed under conditions summarized below:

r_1 = 6.35 (mm), r_2 = 6.426 (mm), r_3 = 7.239 (mm),

k_1 = 2 (W/(mK)), k_2 = 13.8 (W/(mK)),

T_4 = 300 (K), h_3 = 2836 ($W/(m^2K)$),

Q = 1000 (W), l = 0.1 (m), h_{gap} = 1000 ($W/(m^2K)$).

Table 2 and Fig. 3 show comparisons with analytical solution for one dimensional heat conduction problem in radial direction at steady state. Excellent agreement is observed in Table 2 and Fig. 3. These results confirm that the coding related to the heat conduction in radial direction is reliable in the modified code.

5.2 One dimensional heat conduction problem in axial direction at steady state

The differential equation for one dimensional heat conduction in axial direction at steady state is given by

$$d/dZ(k(dT/dZ)) = -q'''. \qquad (25)$$

Figure 4 illustrates boundary conditions that an analytical solution of Eq. (25) is derived. The boundary conditions is described by

$$\begin{aligned} dT/dZ &= 0 && \text{at } Z = 0, \\ T &= T_1 && \text{at } Z = L, \\ q''' &= q_0(1-Z/L), && \\ k &= \text{constant}. && \end{aligned} \qquad (26)$$

The solution is given by

$$T = T_1 - q_0(Z^2-L^2)/(2k) + q_0(Z^3-L^3)/(6kL). \qquad (27)$$

A numerical calculation was performed with following conditions:

T_1 = 416 (K), k = 2.0 (W/(mK)),

L = 0.1 (m), r_1 = 6.35 (mm),

$Q = \pi r_1^2 L q_0/2 = 5$ (W).

Table 3 and Fig. 5 show comparisons with analytical solution for one dimensional heat conduction problem at steady state. The numerical results by the modified code show excellent agreement with the analytical solution. These results confirm that the coding related to the heat conduction in axial direction is implemented into the TRAC code accurately.

5.3 Two dimensional heat conduction problem at steady state

The differential equation for two dimensional heat conduction problem at steady state is given by

$$k(T_{rr} + (1/r)T_r + T_{zz}) + q''' = 0. \tag{28}$$

In Eq. (28), subscripts r and z indicate partial differentiation with respect to r and z, respectively.

Figure 6 shows boundary conditions where an analytical solution of Eq. (28) is solved. The boundary conditions are described as follows;

$$\begin{aligned} T_z &= 0 && \text{at } Z = \pm L/2, \\ -kT_r &= h(T - T_\infty) && \text{at } r = a, \\ T_r &= 0 && \text{at } r = 0, \\ h &= \text{constant}, \\ q''' &= \text{constant}, \\ T_\infty &= \begin{cases} T_1 & (-L/2 \leqq Z < 0), \\ T_2 & (0 \leqq Z \leqq L/2). \end{cases} \end{aligned} \tag{29}$$

The analytical solution is given by

$$T(r,Z) = \sum_{n=0}^{\infty} T_n(r)\phi_n(Z), \tag{30}$$

where

$$T_n(r) = \begin{cases} q_0(a^2-r^2)/(4k) + q_0 a/(2h) + t_0, & (n=0) \\ (L/n\pi)^2 q_n/k + \dfrac{\sigma\ \{t_n-(L/n\pi)2q_n\}\ I_0(n\pi r/L)}{\sigma I_0(n\pi a/L) + (n\pi/L)I_1(n\pi a/L)} & (n\neq 0) \end{cases} \tag{31}$$

$$\sigma = h/k \tag{32}$$

$$q_n = \begin{cases} \sqrt{L}q & (n=0) \\ 0 & (n\neq 0) \end{cases} \tag{33}$$

$$t_n = \begin{cases} \sqrt{L}(T_1+T_2)/2 & (n=0) \\ \sqrt{2L}(T_1-T_2)/(n\pi)\ \sin(n\pi/2) & (n\neq 0) \end{cases} \tag{34}$$

$$\phi_n(Z) = \begin{cases} \sqrt{(1/L)} & (n=0) \\ \sqrt{(2/L)}\cos(n\pi Z/L) & (n\neq 0) \end{cases} \quad (35)$$

I_0 and I_1 indicate Bessel functions.

A numerical calculation was performed with following conditions:

$a = 5$ (mm), $L = 0.2$ (m),

$T_1 = 300$ (K), $T_2 = 500$ (K),

$h = 1000$ (W/(m^{2}K)), $k = 2$ (W/(mK)),

$Q = \pi a^2 Lq = 1000$ (W).

Table 4 and Fig. 7 show comparisons with analytical solution along center line of rod. The analytical solution was calculated taking a summation upto n=500. The calculated results by the modified code show excellent agreement with the analytical solution. Table 5 and Fig. 8 show comparisons with analytical solution along a radius at midplane of rod. The calculated results by the modified code show excellent agreement with the analytical solution. These results also confirm that the modified code is reasonably accurate.

5.4 Two dimensional heat conduction problem at transient case

Figure 9 shows input schematics used for assessment calculation on two dimensional heat conduction problem at transient case. The boundary conditions were specified with measured results from CCTF flat radial power test. The heater rod was modeled by six radial nodes. The fine mesh logic was used for axial noding of heater rod.

Table 6 summarizes calculation conditions in the sensitivity study to assess the effect of the solution method on calculation statistics. A parameter study for minimum allowable axial-node-size and solution method was performed.

Figures 10 and 11 show effect of minimum axial node size DZNHT with implicit or explicit method on clad temperature at elevation of 1.83 m. The case for explicit method with DZNHT=0.1 mm was calculated only by 135.7 s because of CPU time limit. The coarse node shows slightly higher clad temperature in both explicit and implicit methods. However, the clad temperature is almost identical if DZNHT is less than 0.010 mm. Figures 12 and 13 shows effect of minimum axial node size DZNHT on quench front propagation in explicit or implicit methods. The coarser node resulted in slightly slower quench front propagation if DZNHT is greater than 0.010 mm in this calculation. These results suggest that a minimum axial node size less than 0.01 mm is required to get a solution independent of minimum axial node size.

Figure 14 shows a comparison of clad temperature at elevation of 1.83 m between implicit and explicit methods. Although the minimum axial node size DZNHT less than 0.010 mm is expected to eliminate effect of node size, the comparison was made for the case with DZNHT = 1.0 mm because the CPU time by the explicit method with DZNHT=0.01 mm is estimated to be more than 10^6 s. It should be noted that the result shown in Fig. 14 may include the effect of node side as well as the effect of the solution method. In Fig. 14, the result from the implicit method shows good agreement with that from the explicit method except the period right before the quench. The reason why the result from the implicit method showed slightly slow quench propagation compared to that from the explicit method is uncertain at present.

Figures 15 and 16 show comparisons of CPU time between explicit and implicit methods with DZNHT = 1.0 mm and 0.1 mm. Figures 15 and 16 show that the implementation of the implicit method resulted in the bigger time step size as expected and resulted in the faster calculation.

Table 7 summarizes CPU statistics. Because calculation for solving the heat conduction equation is controlled by subroutine FROD, the total CPU time at subroutine FROD and subsequent routine is considered to represent the pure CPU cost for solving the heat conduction equation. The CPU time at subroutine FROD is about 25 % in these calculations.

Table 7 show that the implementation of the implicit method resulted in the increase of CPU time at subroutine FROD by about 60 %. The increase causes the CPU time per a time step to be increase by about 20 %. If the implicit method is used under conditions where time-step size is not controlled by the heat conduction calculation, the use of the implicit method can cause a penalty in the CPU time upto 60 % compared to the case when the explicit method is used.

For the average time step size, the implicit method has no limitation from the heat conduction on the time step size, while the explicit method restricts the time step size severely in these calculations. The average time step size with the implicit method is about 1.5 times that of the explicit method in case of DZNHT = 1.0 mm. It is about 10 in case of DZNHT = 0.1 mm based on the average between 0 and 135.7 s.

In this calculation, the fine mesh logic was turned on at 122.19 s with time step number of 361. The average time step size between 122.19 and 135.7 is only 0.001 in the calculation with the explicit method with DZNHT = 0.1 mm. To complete calculation by 600 s by the explicit method with DZNHT = 0.1 mm, the CPU time is estimated to be about 40000 s. The CPU time is much bigger than that by the implicit method with DZNHT = 0.1 mm. These results confirm that the implicit method is useful if a calculation is performed with a fine axial noding.

6. ASSESSMENT CALCULATIONS AT LANL

6.1 Installation procedure to LANL CRAY computer

In LANL, assessment calculations are conducted to check whether the JAERI provided implicit heat conduction calculation method is correctly installed into TRAC-PF1/MOD2 LANL version or not. The detail of these assessment calculations is described below.

(1) Calculation with original TRAC-PF1/MOD2

At first, assessment calculations are conducted with original TRAC-PF1/MOD2 to establish data base before modification by using the following input data which are provided by LANL as TRAC-PF1/MOD2 sample input.

A) SLAB1,

B) POWER2,

C) W4LOOP,

D) CCTF.

(2) Calculation with explicit method in modified TRAC-PF1/MOD2

After the modification was completed, the calculations with explicit method in the modified TRAC-PF1/MOD2 are conducted using the same input data that were used in the assessment calculation with original TRAC-restricts the time step size severely in these calculations. The average time step size with the implicit method is about 1.5 times that of the explicit method in case of DZNHT = 1.0 mm. It is about 10 in case of DZNHT = 0.1 mm based on the average between 0 and 135.7 s.

In this calculation, the fine mesh logic was turned on at 122.19 s with time step number of 361. The average time step size between 122.19 and 135.7 is only 0.001 in the calculation with the explicit method with DZNHT = 0.1 mm. To complete calculation by 600 s by the explicit method with DZNHT = 0.1 mm, the CPU time is estimated to be about 40000 s. The CPU time is much bigger than that by the implicit method with DZNHT = 0.1 mm. These results confirm that the implicit method is useful if a calculation is performed with a fine axial noding.

6. ASSESSMENT CALCULATIONS AT LANL

6.1 Installation procedure to LANL CRAY computer

In LANL, assessment calculations are conducted to check whether the JAERI provided implicit heat conduction calculation method is correctly installed into TRAC-PF1/MOD2 LANL version or not. The detail of these assessment calculations is described below.

(1) Calculation with original TRAC-PF1/MOD2

At first, assessment calculations are conducted with original TRAC-PF1/MOD2 to establish data base before modification by using the following input data which are provided by LANL as TRAC-PF1/MOD2 sample input.

A) SLAB1,

B) POWER2,

C) W4LOOP,

D) CCTF.

(2) Calculation with explicit method in modified TRAC-PF1/MOD2

After the modification was completed, the calculations with explicit method in the modified TRAC-PF1/MOD2 are conducted using the same input data that were used in the assessment calculation with original TRAC-

PF1/MOD2 to check whether the modification has no influence on the other parts of the code, or not.

(3) Calculation with implicit method in modified TRAC-PF1/MOD2

The next, the calculations with implicit method in modified TRAC-PF1/MOD2 are conducted using the same input data that were used above assessment calculations to get the information on CPU time and effect of solution methods.

(4) Calculation of two dimensional heat conduction at steady state

This calculation is one of assessment calculations conducted at JAERI with modified TRAC-PF1/MOD2 to check the code accuracy. Since the modified TRAC-PF1/MOD2 shows excellent agreement with the analytical solution in JAERI, it is possible to check whether the modified code works well on LANL computer as well as on JAERI computer by comparing the result at LANL with the analytical solution.

6.2 Results of assessment calculations in LANL

Assessment calculations were performed with modified code which was made in LANL by JAERI. The calculations in terms of original version, modified version with explicit method and modified version with implicit method are agreed exactly with each other for each sample case except for the case which temperature field calculations within rods with gap between cladding and inside material were included. As described previously, differential scheme at gap was changed to make the matrix symmetric. The difference, however, is negligibly small. Actually, implicit calculation results of SLAB1 are exactly the same as original SLAB1 calculation results. Although there are small difference in POWER2, W4LOOP and CCTF calculations between original calculation and implicit calculation, the differences are very small. For example, the surface temperatures of average rod are shown in below.

(POWER2) Original: 603.90 K
Implicit: 605.00 K at 60.76 sec
(W4LOOP) Original: 603.77 K

Implicit: 605.12 K (steady state calculation)

(CCTF) Original: 693.98 K

Implicit: 693.98 K at 50.12 sec

The calculation statics are summarized in Table 8 for each sample case. Since the explicit results in this table were obtained using the modified code, the effect of the solution method, that is, explicit or implicit is directly investigated from this table.

In case of SLAB1 and POWER2, total time step number and time step size were the same for both methods, respectively. This result implies that the time step size is restricted not by heat conduction calculation but by the other calculations such as hydrodynamic calculation. CPU time per a time step in implicit calculation was larger than that in explicit calculation, because implicit method needs more numerical operations than explicit method.

In case of W4LOOP, there was no difference on each items between two solution methods. This result suggests that the fraction of CPU time of subroutine FROD and the subsequent routines are negligibly small against total CPU time in the W4LOOP calculation.

The benefit of the implicit method for heat conduction calculation can be observed in case of CCTF. In this case, the time step size in implicit calculation was larger than that in explicit calculation. But total CPU time in the implicit calculation is larger than that in explicit calculation, because the fine mesh calculation started at 62.8 s and the calculation was stopped at 100 s. That is, the duration time of the fine mesh calculation is so short that the large time step size cannot affect total CPU time significantly. Furthermore, the benefit of implicit calculation is not appeared since minimum axial node size DZNHT is set a large value, 5 mm. One can expect that total CPU time in implicit calculation is smaller than in explicit calculation, if the calculation is continued further than 100 s and/or the minimum axial node size DZNHT is set a smaller value.

The implicit calculation at LANL for two-dimensional heat conduction equation with modified code showed the identical results to the TRAC calculation results performed in JAERI. This fact confirms that the implicit calculation method of the two dimensional heat conduction equation is successfully installed into the TRAC-PF1/MOD2 code on LANL CRY computer as well as on JAERI FACOM computer.

7. SUMMARY AND RECOMMENDATION

(1) An implicit method for solving a two-dimensional unsteady heat conduction equation was developed and implemented into the TRAC-PF1/MOD2 code as one of contributions of Japan Atomic Energy Research Institute (JAERI) to the code improvement program organized by the United State Nuclear Regulatory Commission.

(2) The modified code with the implicit method, mentioned above, showed excellent agreement with the analytical solution of the heat conduction equation as the original code with the explicit method.

(3) The modified code with the implicit method predicted the same results under transient condition as the original code with the explicit method.

(4) The modified code was installed into the CRAY computer at Los Alamos National Laboratory (LANL). As a result of assessment calculations at JAERI and LANL, it was confirmed that the modified code gives the same answers on the LANL CRAY computer as it does on the JAERI FACOM computer.

(5) The implicit method, mentioned above, required more CPU time by about 60 % per a time step to solve the implicit heat conduction equation than the explicit method in the original code. However, the total CPU time can be decrease because the implicit method allows a big time step size if the fine mesh algorithm is used.

Acknowledgment

They appreciate Drs. J. Spore and P. Giguere of Los Alamos National Laboratory (LANL) for their assistance to using computer system in LANL and understanding of the TRAC-PF1/MOD2 code. And they also appreciate Dr. H. Kamo of Japan Information Service and Mr. M. Enomoto of ISL for their assistance during this work in Japan and LANL.

References

(1) Los Alamos National Laboratory: NUREG/CR-3858,LA-10157-MS.R4,July 1986

(2) O'Mahoney: AEEW-M2590, July 1989.

7. SUMMARY AND RECOMMENDATION

(1) An implicit method for solving a two-dimensional unsteady heat conduction equation was developed and implemented into the TRAC-PF1/MOD2 code as one of contributions of Japan Atomic Energy Research Institute (JAERI) to the code improvement program organized by the United State Nuclear Regulatory Commission.

(2) The modified code with the implicit method, mentioned above, showed excellent agreement with the analytical solution of the heat conduction equation as the original code with the explicit method.

(3) The modified code with the implicit method predicted the same results under transient condition as the original code with the explicit method.

(4) The modified code was installed into the CRAY computer at Los Alamos National Laboratory (LANL). As a result of assessment calculations at JAERI and LANL, it was confirmed that the modified code gives the same answers on the LANL CRAY computer as it does on the JAERI FACOM computer.

(5) The implicit method, mentioned above, required more CPU time by about 60 % per a time step to solve the implicit heat conduction equation than the explicit method in the original code. However, the total CPU time can be decrease because the implicit method allows a big time step size if the fine mesh algorithm is used.

Acknowledgment

They appreciate Drs. J. Spore and P. Giguere of Los Alamos National Laboratory (LANL) for their assistance to using computer system in LANL and understanding of the TRAC-PF1/MOD2 code. And they also appreciate Dr. H. Kamo of Japan Information Service and Mr. M. Enomoto of ISL for their assistance during this work in Japan and LANL.

References

(1) Los Alamos National Laboratory: NUREG/CR-3858,LA-10157-MS.R4,July 1986

(2) O'Mahoney: AEEW-M2590, July 1989.

7. SUMMARY AND RECOMMENDATION

(1) An implicit method for solving a two-dimensional unsteady heat conduction equation was developed and implemented into the TRAC-PF1/MOD2 code as one of contributions of Japan Atomic Energy Research Institute (JAERI) to the code improvement program organized by the United State Nuclear Regulatory Commission.

(2) The modified code with the implicit method, mentioned above, showed excellent agreement with the analytical solution of the heat conduction equation as the original code with the explicit method.

(3) The modified code with the implicit method predicted the same results under transient condition as the original code with the explicit method.

(4) The modified code was installed into the CRAY computer at Los Alamos National Laboratory (LANL). As a result of assessment calculations at JAERI and LANL, it was confirmed that the modified code gives the same answers on the LANL CRAY computer as it does on the JAERI FACOM computer.

(5) The implicit method, mentioned above, required more CPU time by about 60 % per a time step to solve the implicit heat conduction equation than the explicit method in the original code. However, the total CPU time can be decrease because the implicit method allows a big time step size if the fine mesh algorithm is used.

Acknowledgment

They appreciate Drs. J. Spore and P. Giguere of Los Alamos National Laboratory (LANL) for their assistance to using computer system in LANL and understanding of the TRAC-PF1/MOD2 code. And they also appreciate Dr. H. Kamo of Japan Information Service and Mr. M. Enomoto of ISL for their assistance during this work in Japan and LANL.

References

(1) Los Alamos National Laboratory: NUREG/CR-3858,LA-10157-MS.R4,July 1986

(2) O'Mahoney: AEEW-M2590, July 1989.

Table 1 List of changed subroutines

Subroutine name	Function
INPUT	Controls data input
BLKDAT	Set initial value of variables
COREC3	Controls temperature calculation in fuel rods in 1-D core component
HTSTR3	Controls temperature calculation in heat structures
CORE3	Calculates temperature profile in heat structures
FROD	Calculates temperature profile in fuel rods
RODHT	Calculates fuel rod temperature field
BANSOL	Solves linear system of the form AX = B where A is symmetric

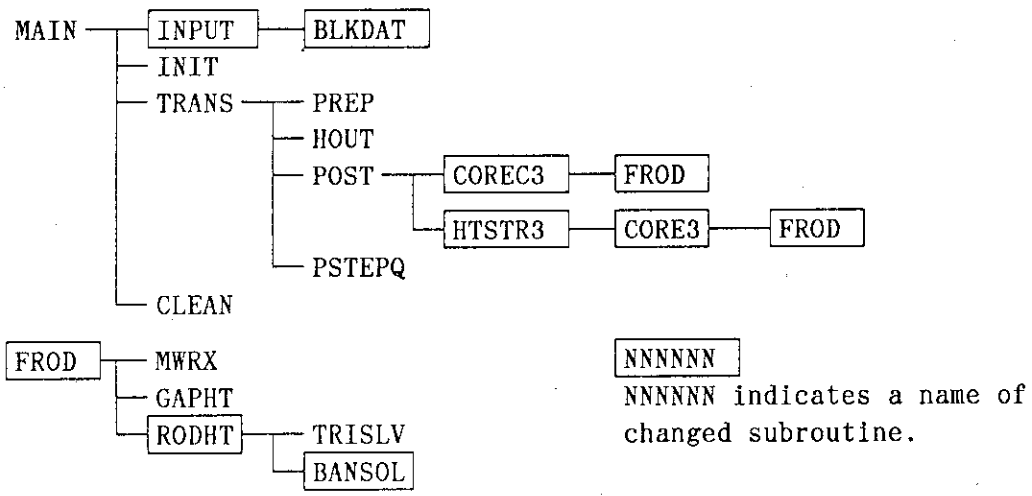


Table 2 Comparison with analytical solution for one dimensional heat conduction problem in radial direction at steady state

Node location (mm)	Analytical solution	TRAC calculation		
		4 nodes	8 nodes	18 nodes
0.000	1039.8	1042.4(+2.6)	1042.4(+2.6)	1042.4(+2.6)
1.830	1006.7	-	-	1009.4(+2.7)
2.590	973.6	-	-	976.2(+2.6)
3.175	940.3	-	942.9(+2.6)	942.9(+2.6)
3.670	906.9	-	-	909.5(+2.6)
4.100	873.9	-	-	876.5(+2.6)
4.490	840.9	-	843.5(+2.6)	843.5(+2.6)
4.850	807.7	-	-	810.3(+2.6)
5.185	774.5	-	-	777.1(+2.6)
5.500	741.3	-	743.9(+2.6)	743.9(+2.6)
5.800	707.8	-	-	710.4(+2.6)
6.080	675.0	-	-	677.6(+2.6)
6.350	641.9	644.5(+2.6)	644.5(+2.6)	644.5(+2.6)
6.426	391.3	392.4(+1.0)	392.4(+1.0)	392.4(+1.0)
6.670	387.0	-	-	388.0(+1.0)
6.840	384.1	-	385.1(+1.0)	385.1(+1.0)
7.040	380.7	-	-	381.7(+1.0)
7.239	377.5	378.5(+1.0)	378.5(+1.0)	378.5(+1.0)

Note)
Gap inner surface : 6.35 (mm)
Gap outer surface : 6.426 (mm)
Rod outer surface : 7.239 (mm)

Number in parenthesis indicates difference between analytical solution and calculated result.

Table-3 Comparison with analytical solution for one dimensional heat conduction problem in axial direction at steady state

Node location (mm)	Analytical solution	TRAC calculation			
		3 nodes	6 nodes	11 nodes	21 nodes
0	1732	1894(+162)	1756(+24)	1737(+5)	1732(0)
5	1727	-	-	-	1727(0)
10	1713	-	-	1717(+4)	1712(-1)
15	1689	-	-	-	1689(0)
20	1658	-	1677(+19)	1662(+4)	1658(0)
25	1619	-	-	-	1618(-1)
30	1572	-	-	1575(+3)	1571(-1)
35	1518	-	-	-	1518(0)
40	1458	-	1472(+14)	1460(+2)	1457(-1)
45	1392	-	-	-	1391(-1)
50	1321	1401(+80)	-	1322(+1)	1320(-1)
55	1244	-	-	-	1243(-1)
60	1163	-	1172(+9)	1164(+1)	1162(-1)
65	1079	-	-	-	1077(-2)
70	990	-	-	991(+1)	989(-1)
75	899	-	-	-	898(-1)
80	805	-	809(+4)	805(0)	804(-1)
85	710	-	-	-	708(-2)
90	613	-	-	612(+1)	611(-2)
95	515	-	-	-	513(-2)
100	416	416(0)	416(0)	416(0)	416(0)

Note)
Number in parenthesis indicates difference between analytical solution and calculated result.

Table 4 Comparison with analytical solution along a center line of rod in a two dimensional heat conduction problem at steady state

Node location (mm)	Analytical solution	TRAC calculation			
		3 nodes	21 nodes	41 nodes	Fine mesh logic
0	658.1	658.3(0.2)	658.1(0.0)	658.1(0.0)	658.1(0.0)
10	658.1	-	658.1(0.0)	658.1(0.0)	658.1(0.0)
20	658.1	-	658.1(0.0)	658.1(0.0)	658.1(0.0)
30	658.1	-	658.1(0.0)	658.1(0.0)	658.1(0.0)
40	658.1	-	658.1(0.0)	658.1(0.0)	658.1(0.0)
50	658.1	-	658.1(0.0)	658.1(0.0)	658.1(0.0)
60	658.1	-	658.1(0.0)	658.1(0.0)	658.1(0.0)
70	658.1	-	658.2(0.1)	658.1(0.0)	658.1(0.0)
80	658.2	-	658.8(0.6)	658.4(0.2)	658.3(0.1)
90	662.6	-	667.7(5.1)	664.3(1.7)	661.3(-1.3)
100	758.1	758.1(0.0)	758.1(0.0)	758.1(0.0)	758.1(0.0)
110	853.9	-	848.5(-5.0)	851.9(-1.6)	854.9(1.4)
120	857.5	-	857.4(-0.5)	857.8(-0.1)	857.9(0.0)
130	858.1	-	858.0(-0.1)	858.1(0.0)	858.1(0.0)
140	858.1	-	858.1(0.0)	858.1(0.0)	858.1(0.0)
150	858.1	-	858.1(0.0)	858.1(0.0)	858.1(0.0)
160	858.1	-	858.1(0.0)	858.1(0.0)	858.1(0.0)
170	858.1	-	858.1(0.0)	858.1(0.0)	858.1(0.0)
180	858.1	-	858.1(0.0)	858.1(0.0)	858.1(0.0)
190	858.1	-	858.1(0.0)	858.1(0.0)	858.1(0.0)
200	858.1	857.9(-0.2)	858.1(0.0)	858.1(0.0)	858.1(0.0)

Note) 4 nodes in radial direction

Number in parenthesis indicates difference between analytical solution and calculated result.

Table 5 Comparison with analytical solution along a radius at midplane of rod in a two dimensional heat conduction problem at steady state

Node location (mm)	Analytical solution	TRAC calculation		
		2 nodes	4 nodes	11 nodes
0.0	758.1	758.1(0.0)	758.1(0.0)	758.1(0.0)
0.5	756.1	-	-	756.1(0.0)
1.0	750.1	-	-	750.1(0.0)
1.5	740.2	-	-	740.2(0.0)
2.0	726.3	-	-	726.3(0.0)
2.5	708.4	-	-	708.4(0.0)
3.0	686.5	-	686.5(0.0)	686.5(0.0)
3.5	660.6	-	-	660.6(0.0)
4.0	630.8	-	630.8(0.0)	630.8(0.0)
4.5	596.9	-	-	597.0(0.1)
5.0	559.1	559.2(0.1)	559.2(0.1)	559.2(0.1)

Note) 21 nodes in axial direction

Number in parenthesis indicates difference between analytical solution and calculated result.

Table 6 Summary of calculation condition

Case number	Solution method	Parameters for fine mesh calculation		
		DTXHT (K)	DZNHT (mm)	NZMAX (-)
1	Implicit	5	1.000	200
2	Implicit	5	0.100	200
3	Implicit	5	0.010	200
4	Implicit	5	0.001	200
5	Explicit	5	1.000	200
6	Explicit	5	0.100	200

Note) DTXHT : Maximum ΔT (K) above which rows of nodes are inserted
DZNHT : Minimum ΔZ (m) below which no additional rows of nodes are inserted
NZMAX : Maximum number of rows of nodes

Table 7 Effect of solution method on CPU time

	Solution method					
	Implicit method DZNHT(mm)				Explicit method DZNHT(mm)	
	1.0	0.1	0.01	0.001	1.0	0.1
Total CPU time (s)	607.59 (1.0)	665.87 (1.10)	678.03 (1.12)	679.47 (1.12)	781.55 (1.29)	1199.21 (-)
Total time step number	5873	5898	5866	5866	8958	13907
Transient time (s)	600.2	600.1	600.1	600.1	600.1	135.7
CPU time of each part (s)						
PREP	249.00	295.01	309.18	308.61	374.30	697.12
OUTER	163.73	165.77	164.32	165.36	209.98	182.46
POST	189.20	199.23	198.81	199.69	190.88	315.86
FROD	165.74	175.31	175.31	175.76	154.92	263.19
others	5.66	5.86	5.72	5.81	6.39	3.77
Total number FROD called	5904	5930	5904	5904	8986	13933
CPU time per a call of FROD	0.02807 (1.0)	0.02956 (1.05)	0.02969 (1.06)	0.02977 (1.06)	0.01724 (0.61)	0.01889 (0.67)
CPU time per a time step	0.10249 (1.0)	0.11190 (1.09)	0.11461 (1.12)	0.11484 (1.12)	0.08653 (0.84)	0.08596 (0.84)
Average time step size	0.10220 (1.0)	0.10175 (1.00)	0.10230 (1.00)	0.10230 (1.00)	0.06699 (0.66)	0.00976 (0.10)

Note) CPU time of POST includes CPU time of FROD.
Fine mesh calculation logic was turned on at 122.19 s with time step number of 361.
Number in parenthesis indicates a ratio to the corresponding value of implicit method with DZNHT=1.0 mm.

Table 8 Effect of solution method on CPU time (assessed in LANL)

Sample data	W4LOOP		CCTF		SLAB1	
Solution method	Implicit	Explicit	Implicit	Explicit	Implicit	Explicit
Total CPU time (S)	67.535 (1.0)	67.639 (1.0)	184.893 (1.0)	126.928 (0.69)	2.131 (1.0)	1.880 (0.88)
Total time step number	326 (1.0)	326 (1.0)	1196 (1.0)	1245 (1.04)	20 (1.0)	20 (1.0)
Transient time (s)	89.468 (1.0)	89.468 (1.0)	100.053 (1.0)	100.014 (1.0)	0.3013 (1.0)	0.3013 (1.0)
CPU time per a time step	0.2076 (1.0)	0.20748 (1.0)	0.1546 (1.0)	0.1020 (0.66)	0.10655 (1.0)	0.0940 (0.88)
Average time step size	0.27444 (1.0)	0.27444 (1.0)	0.0837 (1.0)	0.0803 (0.96)	0.01507 (1.0)	0.01507 (1.0)

Sample data	POWER2	
Solution method	Implicit	Explicit
Total CPU time (S)	7.972 (1.0)	6.612 (0.83)
Total time step number	355 (1.0)	355 (1.0)
Transient time (s)	60.7645 (1.0)	60.7645 (1.0)
CPU time per a time step	0.022456 (1.0)	0.018651 (0.83)
Average time step size	0.17117 (1.0)	0.17117 (1.0)

Note) Number in parenthesis indicates a ratio to the corresponding value with implicit method.

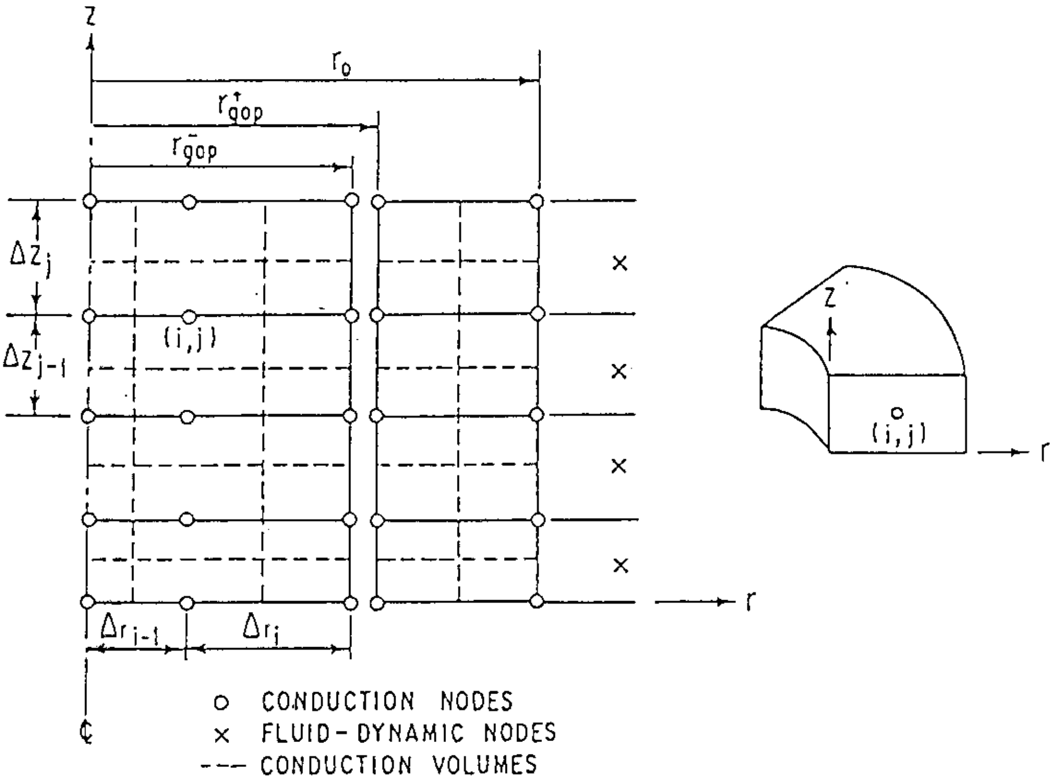
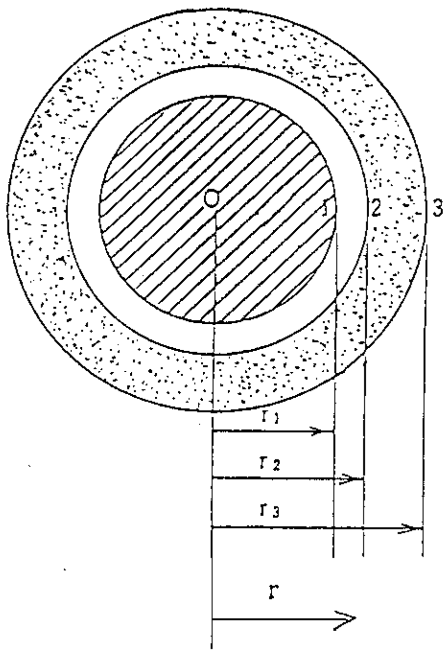


Fig. 1 Cell noding for heat conduction calculation



Boundary conditions

$$\frac{dT}{dr} = 0 \qquad (r = 0)$$

$$-k\frac{dT}{dr} = h_3(T_3 - T_4) \qquad (r = r_3)$$

$$q''' = \begin{cases} q_8 & (0 \leqq r \leqq r_1) \\ 0 & (r_1 \leqq r) \end{cases}$$

Fig. 2 Boundary conditions of one dimensional heat conduction problem in radial direction

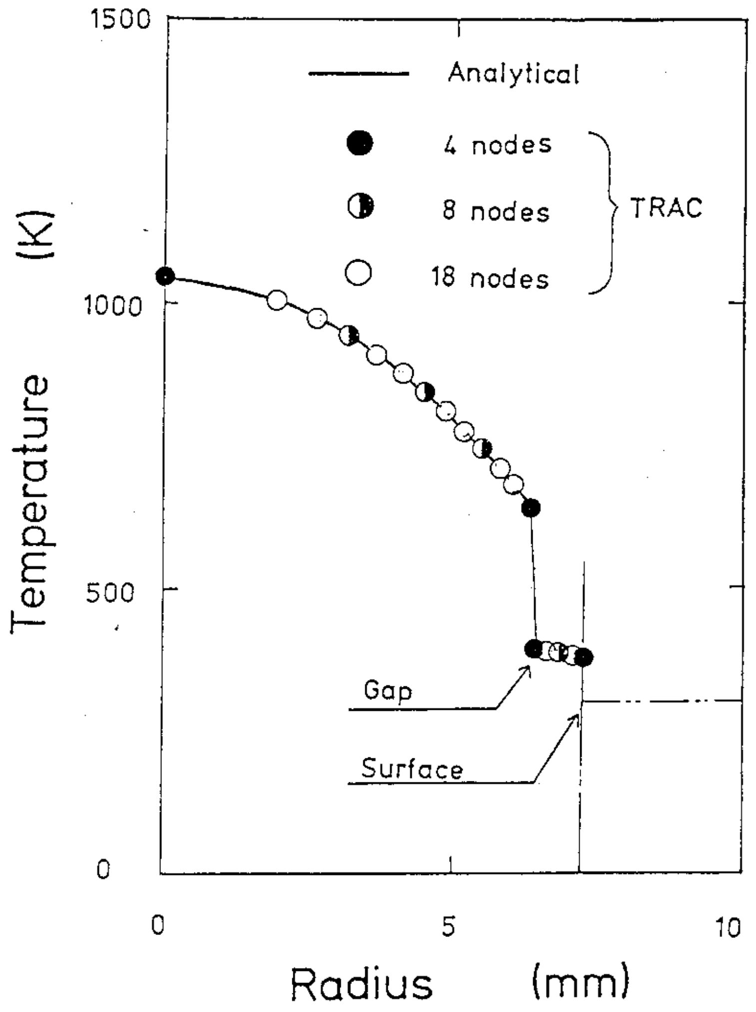


Fig. 3 Comparison with analytical solution for one dimensional heat conduction problem in radial direction

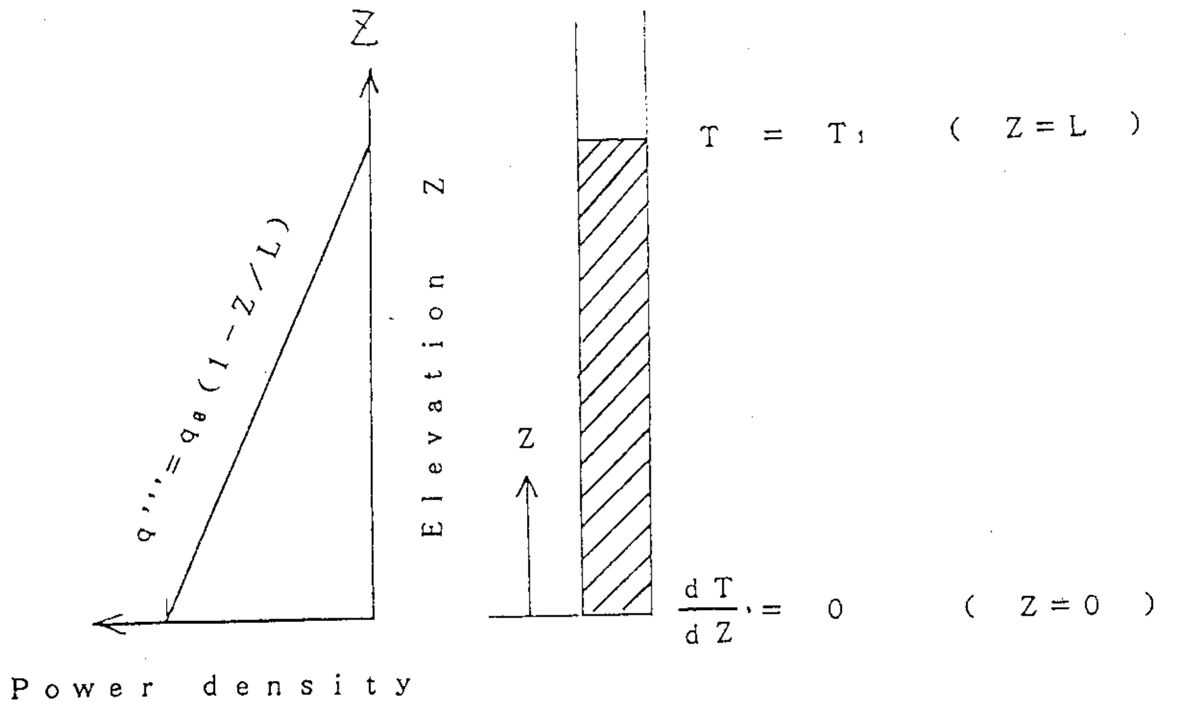


Fig. 4 Boundary conditions of one dimensional heat conduction problem in axial direction

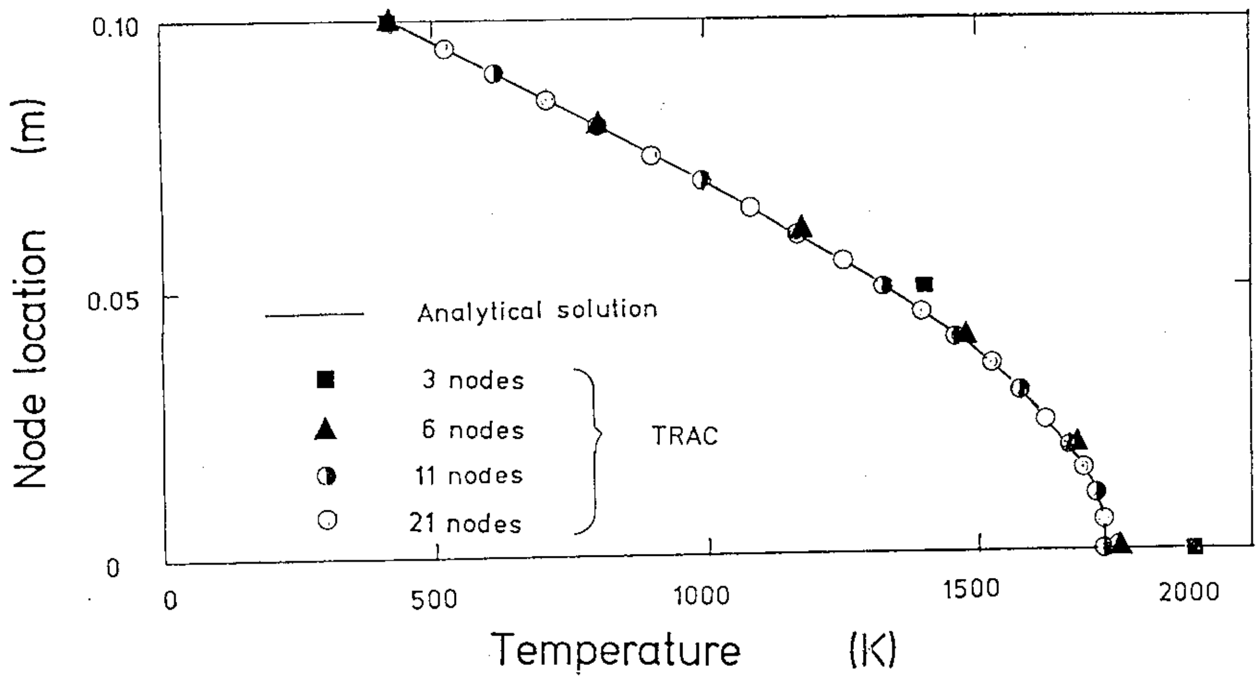
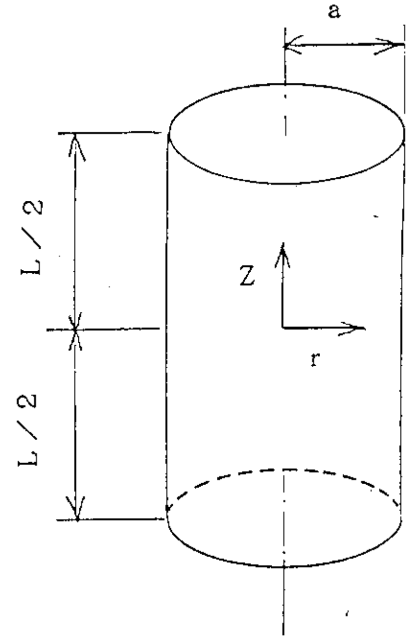


Fig. 5 Comparison with analytical solution for one dimensional heat conduction problem in axial direction



Boundary conditions

$$\frac{\partial T}{\partial Z} = 0 \qquad (Z = \pm L/2)$$

$$\frac{\partial T}{\partial r} = 0 \qquad (r = 0)$$

$$-k\frac{\partial T}{\partial r} = h\,(T - T_\infty) \qquad (r = a)$$

$$T_\infty = \begin{cases} T_1 & (-L/2 \leqq Z < 0) \\ T_2 & (0 \leqq Z \leqq L/2) \end{cases}$$

$$q''' = q$$

Fig. 6 Boundary conditions of two-dimensional heat conduction problem at steady state

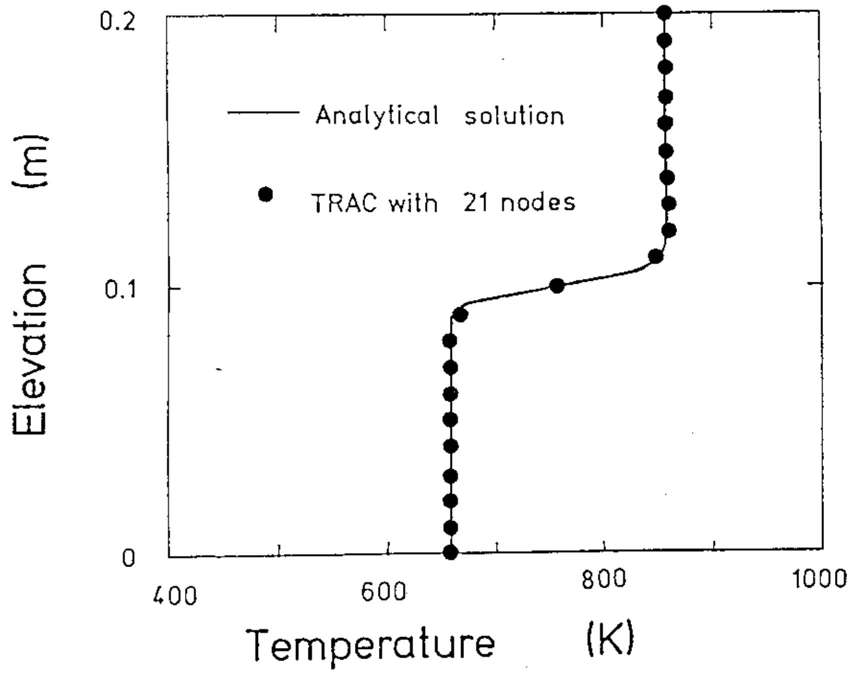


Fig. 7 Comparison with analytical solution along a rod center line in two dimensional heat conduction problem

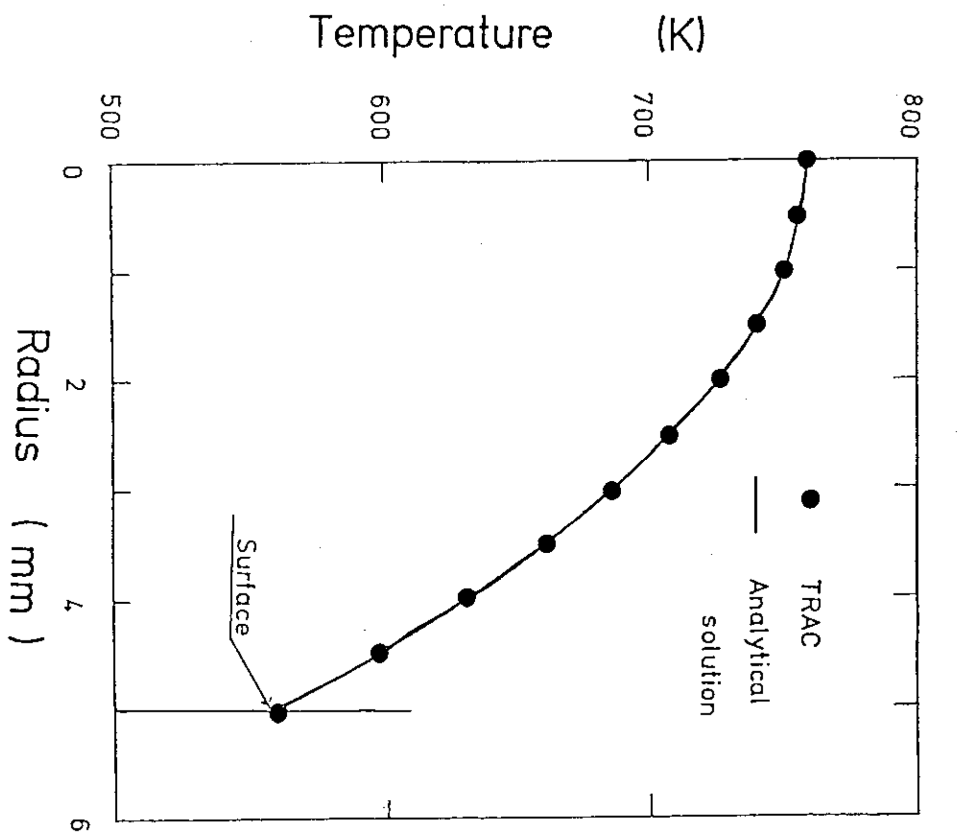


Fig. 8 Comparison with analytical solution along a radius at midplane of rod in two dimensional heat conduction problem

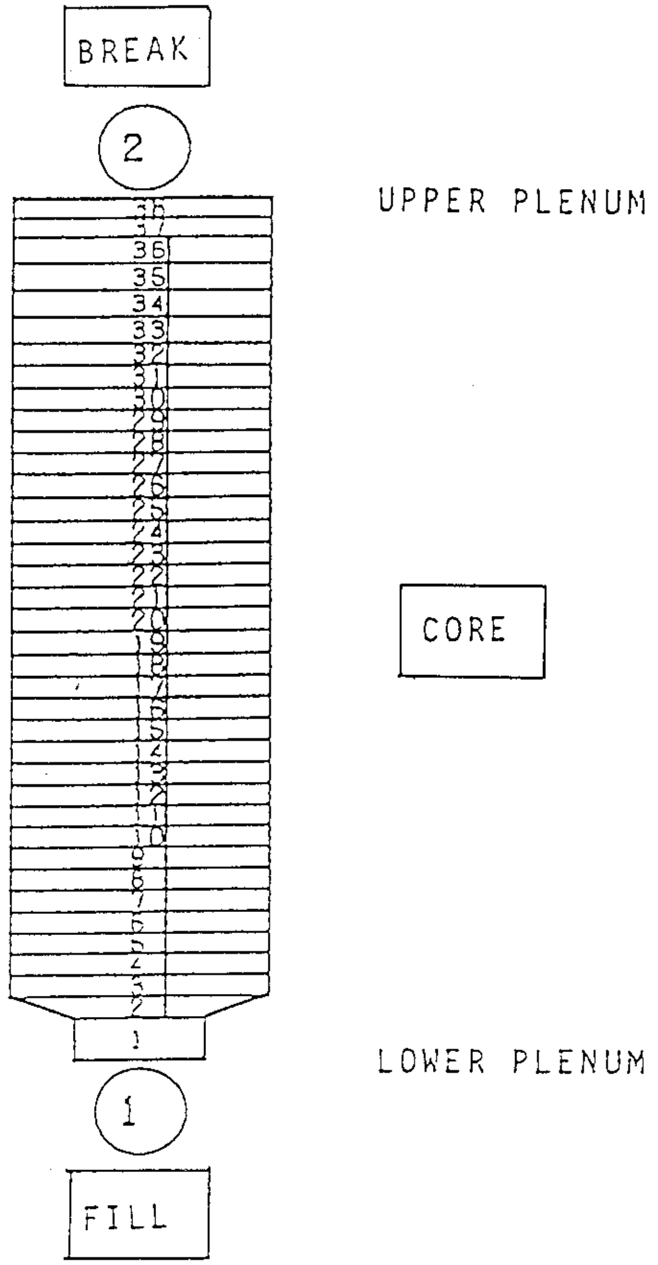


Fig. 9 TRAC noding for a CCTF test analysis

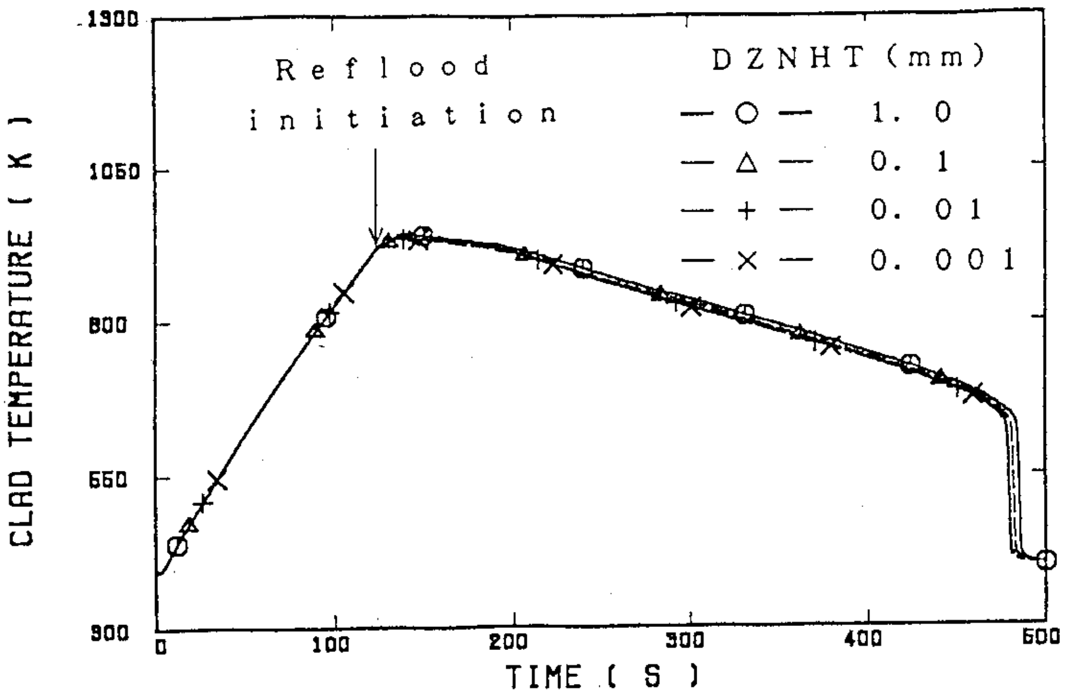


Fig. 10 Effect of minimum allowable axial node size DZNHT on clad temperature at elevation of 1.83 m (Implicit method)

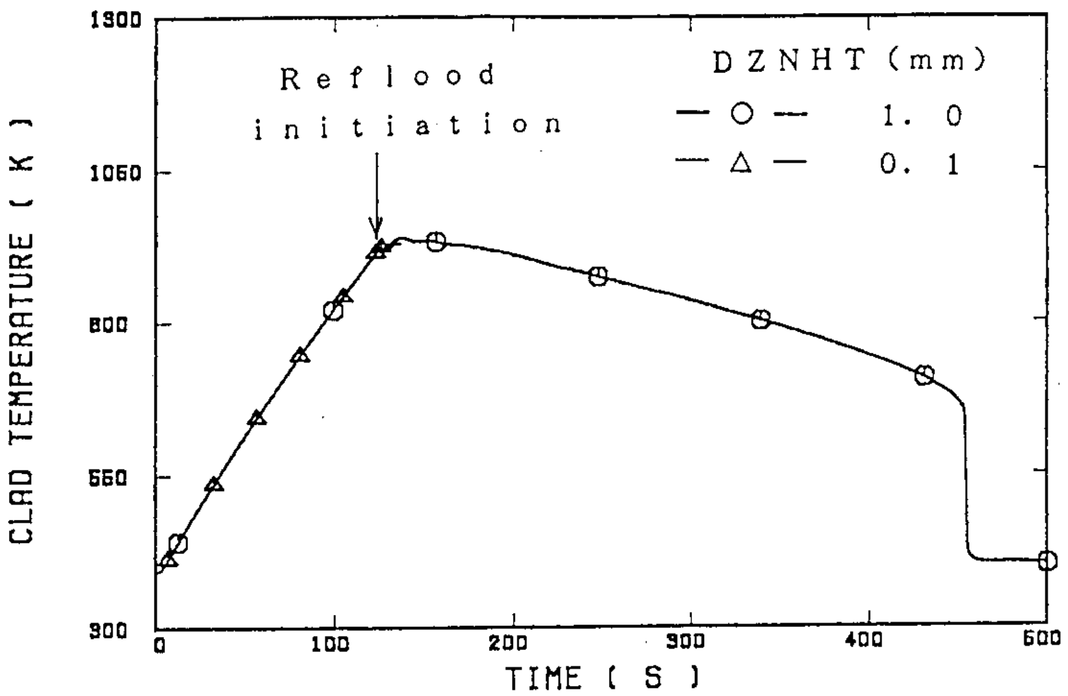


Fig. 11 Effect of minimum allowable axial node size DZNHT on clad temperature at elevation of 1.83 m (Explicit method)

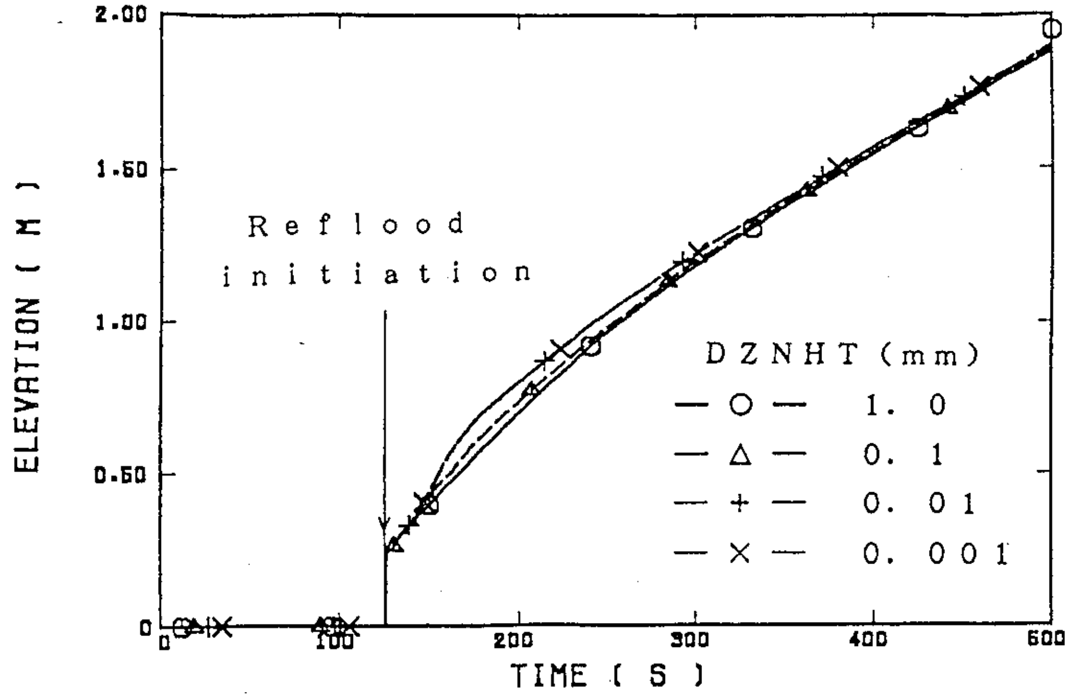


Fig. 12 Effect of minimum allowable axial node size DZNHT on quench propagation (Implicit method)

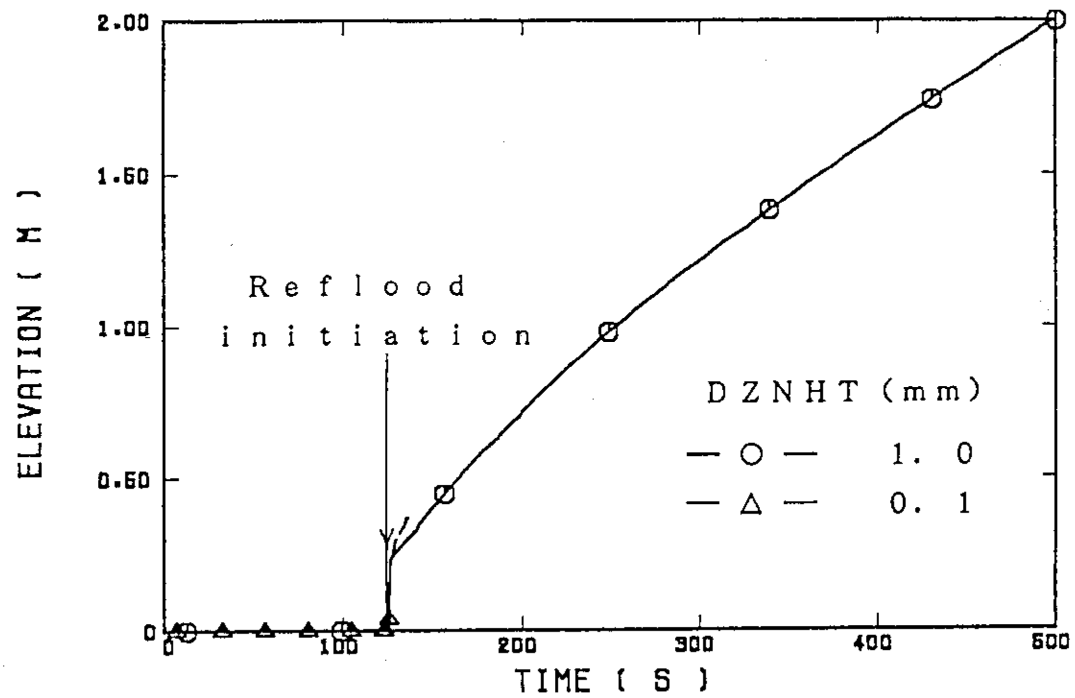


Fig. 13 Effect of minimum allowable axial node size DZNHT on quench propagation (Explicit method)

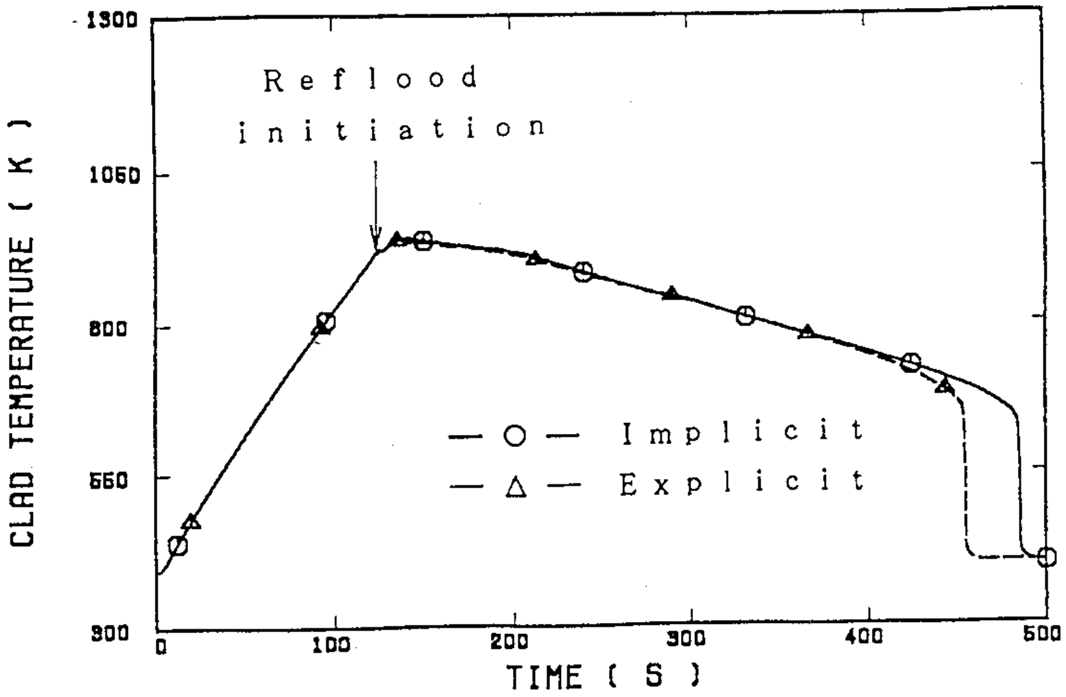


Fig. 14 Effect of solution method on clad temperature at elevation of 1.83 m (DZNHT = 1.0 mm)

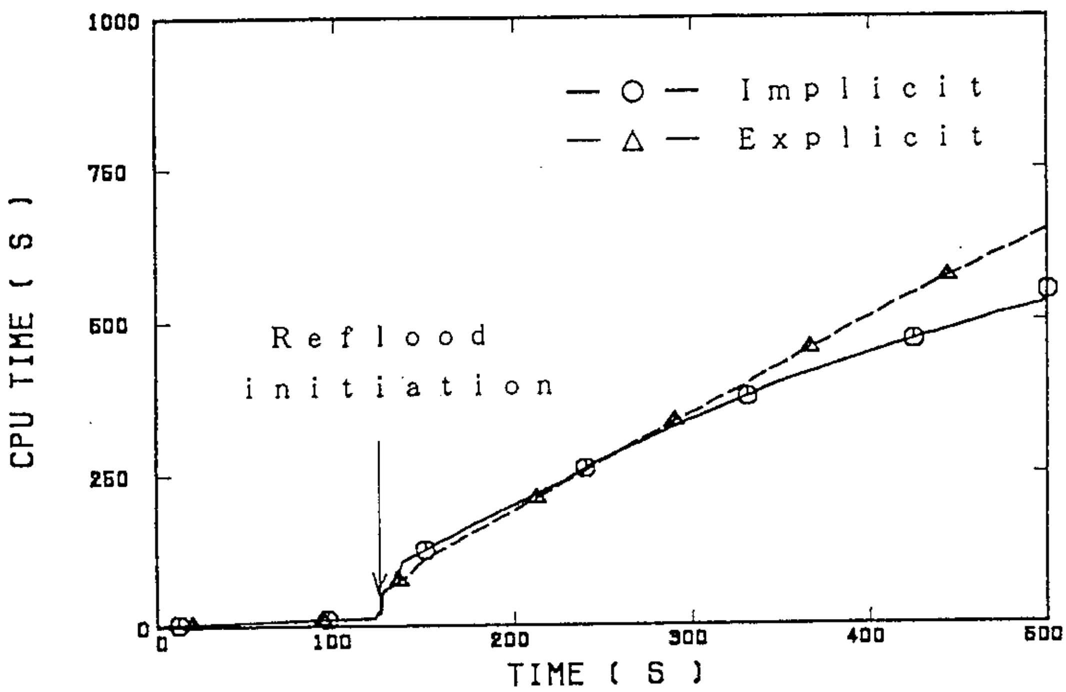


Fig. 15 Effect of solution method on CPU time (DZNHT = 1.0 mm)

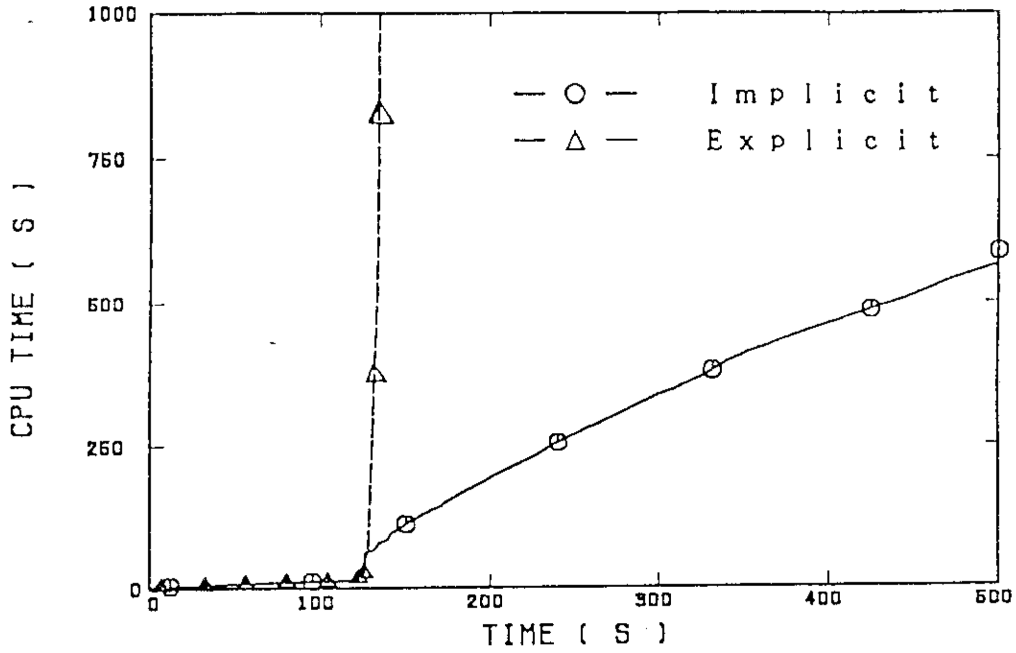


Fig. 16 Effect of solution method on CPU time (DZNHT = 0.1 mm)