

JAERI - M
85-046

水平環状多層断熱層の熱伝達特性・第1報
— 数値計算コード —

1985年4月

日野竜太郎・下村 寛昭

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。
入手の間合わせは、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、お申しこしてください。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.

©Japan Atomic Energy Research Institute, 1985

編集兼発行 日本原子力研究所
印刷 柳高野高速印刷

水平環状多層断熱層の熱伝達特性・第1報
—数値計算コード—

日本原子力研究所東海研究所高温工学部
日野竜太郎・下村 寛昭

(1985年2月16日受理)

多目的高温ガス炉の水平高温配管において、環状断熱層内に局所的な間隙あるいは充てん密度の変化などが存在する場合の熱伝達特性を把握するため、断熱層内での自然対流の発生を考慮した数値計算コードを開発した。本コードでは、上記の局所的状態を多層モデルを用いて近似した。

本報では、解析モデルと数値計算手法及びプログラム構成について述べ、計算結果の一例を紹介する。

Heat Transfer Characteristics of Horizontally
Oriented Multi-Layered Annular Insulation (1)
- Computer Code for Numerical Calculation -

Ryutaro HINO and Hiroaki SIMOMURA

Department of High Temperature Engineering
Tokai Research Establishment, JAERI

(Received February 16, 1985)

A computer code has been developed to analyze the natural convection heat transfer in a horizontal annular insulation layer of a hot gas duct when local gaps and inhomogeneity of filling density of insulation materials exist. This computer code simulates local gaps and inhomogeneity of filling density by a multi-layer model.

This report describes an analytical model, a numerical method, an outline of program and some calculation results.

Keywords: Computer Code, Natural Convection, Heat Transfer, Annular Insulation Layer, Hot Gas Duct, Multi-Layer Model

目 次

1. まえがき	1
2. 理論解析	1
2.1 解析モデル	1
2.2 基礎式	2
2.3 実効熱伝導率	4
3. 数値解法	7
3.1 運動方程式の差分表現	7
3.2 エネルギー方程式の差分表現	9
3.3 数値計算	13
4. プログラム構成	18
4.1 計算コードのプログラム構成	18
4.2 計算コードの入力データ	18
4.3 図形処理プログラムの構成	19
4.4 図形処理プログラムの入力データ	20
5. 計算コード	23
6. 図形処理プログラム	49
7. 計算結果	67
参考文献	73

Contents

1. Introduction	1
2. Theoretical analysis	1
2.1 An analytical model	1
2.2 Basic equations	2
2.3 Effective thermal conductivities	4
3. Numerical method	7
3.1 Finite difference formulation of a momentum equation	7
3.2 Finite difference formulation of an energy equation	9
3.3 Numerical procedure	13
4. Outline of computer code	18
4.1 Program for computation	18
4.2 Input data of computation	18
4.3 Program for graphic display	19
4.4 Input data of graphic display	20
5. Listing of the code	23
6. Listing of the graphic routines	49
7. Numerical results	67

1. ま え が き

高温ガス炉の一次冷却系及び二次冷却系の配管は、高温高圧のガスを安全に、しかも放熱熱量を少なく輸送するために、一般に内部断熱高温配管が用いられる。この高温配管は、高温のガスに直接触れる金属管の周囲に断熱構造物（環状断熱層）を施行し、更にその外側の比較的低温部を耐圧管で覆う方式である。従って、環状断熱層は耐圧管の強度維持及び放熱損失の防止のために、十分な断熱性能を要求される。

環状断熱層の断熱性能、すなわち伝熱特性については、下村ら⁽¹⁾、越後ら⁽²⁾、増岡ら⁽³⁾、Caltagirone⁽⁴⁾などにより理論的解析あるいは実験的検討が行われ、水平環状断熱層内で発生する自然対流の影響を明らかにしている。しかし、これらの研究は基本的な伝熱特性の把握を目的としており、断熱層構造材の熱膨張等により発生する局所的すき間あるいは充てん密度の変化など実際的な状況における伝熱特性については、今後の課題として残されていた。

本解析は、上記の局所的状態を多層モデルを用いて巨視的にとらえ、水平環状断熱層内の伝熱及び流動状態について検討を行うものである。本報では、解析モデルと数値計算手法及びプログラム構成について述べ、計算結果の一例を紹介する。

2. 理 論 解 析

2.1 解析モデル

水平環状断熱層において、局所的な間隙を生じた状態あるいは断熱材の充てん密度が局所的に不均一な状態を単純化して、充てん密度の異なる断熱層が多数積層した状態を考える。図 2.1 に解析モデルを示す。断熱材は、半径 r_i 、温度 T_i （一定）の流路管を覆っており、外側は半径 r_o 、温度 T_o （一定）の耐圧管に接している。また、断熱材は半径方向に n 層に分割され、それぞれの層の熱伝導率、透過率、及び境界半径を λ_{mk} 、 K_k 、 r_k ($k = 1, 2, \dots, n$) とおく。但し、熱伝導率 λ_m は流体静止時の値である。

このような多層環状断熱層の伝熱特性を、断熱層内で発生する自然対流に着目して解析する。座標系は図 2.1 に示すように円柱座標系を用いる。また、定常状態を考え、更に次の仮定をおく。

- (1) 断熱層を透過する気体の運動は、次の Darcy 則に従うものとする。

$$\frac{\mu}{K_k} \vec{U} = -\text{grad} p + \rho \vec{g} \quad (2.1)$$

- (2) 物性値は、浮力に関する密度について

$$\rho = \rho_o \{ 1 - \beta (T_k - T_o) \} \quad (2.2)$$

β : 気体の熱膨張率

と温度変化を与え、その他は一定とする。

1. ま え が き

高温ガス炉の一次冷却系及び二次冷却系の配管は、高温高圧のガスを安全に、しかも放熱熱量を少なく輸送するために、一般に内部断熱高温配管が用いられる。この高温配管は、高温のガスに直接触れる金属管の周囲に断熱構造物（環状断熱層）を施行し、更にその外側の比較的低温部を耐圧管で覆う方式である。従って、環状断熱層は耐圧管の強度維持及び放熱損失の防止のために、十分な断熱性能を要求される。

環状断熱層の断熱性能、すなわち伝熱特性については、下村ら⁽¹⁾、越後ら⁽²⁾、増岡ら⁽³⁾、Caltagirone⁽⁴⁾などにより理論的解析あるいは実験的検討が行われ、水平環状断熱層内で発生する自然対流の影響を明らかにしている。しかし、これらの研究は基本的な伝熱特性の把握を目的としており、断熱層構造材の熱膨張等により発生する局所的すき間あるいは充てん密度の変化など実際的な状況における伝熱特性については、今後の課題として残されていた。

本解析は、上記の局所的状態を多層モデルを用いて巨視的にとらえ、水平環状断熱層内の伝熱及び流動状態について検討を行うものである。本報では、解析モデルと数値計算手法及びプログラム構成について述べ、計算結果の一例を紹介する。

2. 理 論 解 析

2.1 解析モデル

水平環状断熱層において、局所的な間隙を生じた状態あるいは断熱材の充てん密度が局所的に不均一な状態を単純化して、充てん密度の異なる断熱層が多数積層した状態を考える。図 2.1 に解析モデルを示す。断熱材は、半径 r_i 、温度 T_i （一定）の流路管を覆っており、外側は半径 r_o 、温度 T_o （一定）の耐圧管に接している。また、断熱材は半径方向に n 層に分割され、それぞれの層の熱伝導率、透過率、及び境界半径を λ_{mk} 、 K_k 、 r_k ($k=1, 2, \dots, n$) とおく。但し、熱伝導率 λ_m は流体静止時の値である。

このような多層環状断熱層の伝熱特性を、断熱層内で発生する自然対流に着目して解析する。座標系は図 2.1 に示すように円柱座標系を用いる。また、定常状態を考え、更に次の仮定をおく。

- (1) 断熱層を透過する気体の運動は、次の Darcy 則に従うものとする。

$$\frac{\mu}{K_k} \vec{U} = -\text{grad} p + \rho \vec{g} \quad (2.1)$$

- (2) 物性値は、浮力に関する密度について

$$\rho = \rho_o \{ 1 - \beta (T_k - T_o) \} \quad (2.2)$$

β : 気体の熱膨張率

と温度変化を与え、その他は一定とする。

- (3) エネルギー輸送において、粘性による散逸は無視し、熱放射の効果は考慮しない。
 (4) 気体の流動状態又は層内の温度分布は、それぞれ左右対称とする。

2.2 基礎式

2.1 に記した仮定から、それぞれの断熱層における基礎式は次のように与えられる。

$$\text{連続式} : \frac{\partial (r u_k)}{\partial r} + \frac{\partial v_k}{\partial \phi} = 0 \quad (2.3)$$

運動方程式：

$$-\frac{\partial p_k}{\partial r} + \rho_0 g \{ 1 - \beta (T_k - T_0) \} \cos \phi = \frac{\mu}{K_k} \mu_k \quad (2.4)$$

$$-\frac{1}{r} \frac{\partial p_k}{\partial \phi} - \rho_0 g \{ 1 - \beta (T_k - T_0) \} \sin \phi = \frac{\mu}{K_k} v_k \quad (2.5)$$

エネルギー式：

$$u_k \frac{\partial T_k}{\partial r} + \frac{v_k}{r} \frac{\partial T_k}{\partial \phi} = a_{mk} \left(\frac{\partial^2 T_k}{\partial r^2} + \frac{1}{r} \frac{\partial T_k}{\partial r} + \frac{1}{r^2} \frac{\partial^2 T_k}{\partial \phi^2} \right) \quad (2.6)$$

$$a_{mk} = \lambda_{mk} / \rho c_p \quad (2.7)$$

ここで、 a_{mk} は温度伝導率、 C_p は気体の比熱、 μ は粘性係数である。

境界条件を次のように与える。まず、断熱層は固体壁で覆われており、また壁面温度は一定であることより、

$$r = r_i ; u_i = 0, T_i = T_i \quad (2.8)$$

$$r = r_o ; u_n = 0, T_n = T_o \quad (2.9)$$

温度場及び流れ場の対称性より、

$$\phi = 0, \pi ; \frac{\partial T}{\partial \phi} = 0, v = 0 \quad (r_i \leq r \leq r_o) \quad (2.10)$$

さらに、各層の境界 ($r = r_k$) では、温度と速度が連続であることより、

$$r = r_k ; T_{k+1} = T_k \quad (2.11)$$

$$u_{k+1} = u_k \quad (2.12)$$

$$v_{k+1} = v_k \quad (2.13)$$

半径方向 r ，温度 T を次のように無次元化する。

$$R = \frac{r}{(r_o - r_i)} \quad (2.14)$$

$$\theta_k = \frac{T_k - T_o}{T_i - T_o} \quad (2.15)$$

更に，半径方向速度 u ，周方向速度 v を無次元流れ関数 ψ を導入して，

$$u_k = \frac{a_{m1}}{(r_o - r_i)R} \frac{\partial \psi_k}{\partial \phi} \quad (2.16)$$

$$v_k = -\frac{a_{m1}}{(r_o - r_i)} \frac{\partial \psi_k}{\partial R} \quad (2.17)$$

と無次元化し，レイリー数 Ra 及びダルシー数 Da を次のように定義する。

$$Ra_k = \frac{g\beta(T_i - T_o)(r_o - r_i)^3}{\nu a_{mk}} \quad (2.18)$$

$$Da_k = \frac{K_k}{(r_o - r_i)^2} \quad (2.19)$$

ここで， ν は気体の動粘性係数である。上記の無次元数を用いて式(2.3)～(2.13)を無次元化する。

運動方程式：

$$\begin{aligned} -Ra_k Da_k \left(\frac{a_{mk}}{a_{m1}} \right) \left(\frac{\partial \theta_k}{\partial R} \sin \phi + \frac{1}{R} \frac{\partial \theta_k}{\partial \phi} \cos \phi \right) \\ = \frac{\partial^2 \psi_k}{\partial R^2} + \frac{1}{R} \frac{\partial \psi_k}{\partial R} + \frac{1}{R^2} \frac{\partial^2 \psi_k}{\partial \phi^2} \end{aligned} \quad (2.20)$$

エネルギー式：

$$\frac{\partial \psi_k}{\partial \phi} \frac{\partial \theta_k}{\partial R} - \frac{\partial \psi_k}{\partial R} \frac{\partial \theta_k}{\partial \phi} = \left(\frac{a_{mk}}{a_{m1}} \right) R \left(\frac{\partial^2 \theta_k}{\partial R^2} + \frac{1}{R} \frac{\partial \theta_k}{\partial R} + \frac{1}{R^2} \frac{\partial^2 \theta_k}{\partial \phi^2} \right) \quad (2.21)$$

境界条件：

$$R = \frac{r_i}{r_o - r_i} \quad ; \quad \frac{\partial \psi_i}{\partial \phi} = 0, \quad \theta_i = 1 \quad (2.22)$$

$$R = \frac{r_o}{r_o - r_i} \quad ; \quad \frac{\partial \psi_n}{\partial \phi} = 0, \quad \theta_n = 0 \quad (2.23)$$

$$\phi = 0, \pi \quad ; \quad \frac{\partial \theta}{\partial \phi} = 0, \quad \frac{\partial \psi}{\partial R} = 0 \quad (2.24)$$

$$\left(\frac{r_i}{r_o - r_i} \leq R \leq \frac{r_o}{r_o - r_i} \right)$$

$$R = \frac{r_k}{r_o - r_i} \quad ; \quad \theta_{k+1} = \theta_k \quad (2.25)$$

$$\frac{\partial \psi_{k+1}}{\partial \phi} = \frac{\partial \psi_k}{\partial \phi} \quad (2.26)$$

$$\frac{\partial \psi_{k+1}}{\partial R} = \frac{\partial \psi_k}{\partial R} \quad (2.27)$$

2.3 実効熱伝導率

環状断熱層の実効熱伝導率 λ_e を次のように定義する。

$$q_r = -2\pi \lambda_e \frac{T_i - T_o}{\ell_n (r_o / r_i)} \quad (2.28)$$

ここで、熱量 q_r は内壁及び外壁近傍の温度勾配より、

$$q_r = -2\pi r_i \lambda_{m1} \left(\frac{\partial T}{\partial r} \right)_{r=r_i} \quad (2.29)$$

$$q_r = -2\pi r_o \lambda_{mn} \left(\frac{\partial T}{\partial r} \right)_{r=r_o} \quad (2.30)$$

で与えられる。

式 (2.29) 及び (2.30) を式 (2.28) に代入して整理すると、

$$\frac{\lambda_e}{\lambda_{m1}} = - \frac{r_i \ln(r_o/r_i)}{T_i - T_o} \left(\frac{\partial T}{\partial r} \right)_{r=r_i} \quad (2.31)$$

$$\frac{\lambda_e}{\lambda_{mn}} = - \frac{r_o \ln(r_o/r_i)}{T_i - T_o} \left(\frac{\partial T}{\partial r} \right)_{r=r_o} \quad (2.32)$$

更に、上式を無次元化すると、

$$\frac{\lambda_e}{\lambda_{m1}} = - \frac{r_i \ln(r_o/r_i)}{r_o - r_i} \left(\frac{\partial \theta}{\partial R} \right)_{R=1} = \frac{r_i}{r_o - r_i} \quad (2.33)$$

$$\frac{\lambda_e}{\lambda_{mn}} = - \frac{r_o \ln(r_o/r_i)}{r_o - r_i} \left(\frac{\partial \theta}{\partial R} \right)_{R=r_o/r_i} = \frac{r_o}{r_o - r_i} \quad (2.34)$$

となる。式 (2.33) 及び (2.34) は、流体静止時の断熱層の熱伝導率 λ_{m1} 及び λ_{mn} に対する実効熱伝導率 λ_e の相対的大きさを表している。

式 (2.33) 及び (2.34) より、円周方向に平均した $(\lambda_e/\lambda_{m1})_m$ 及び $(\lambda_e/\lambda_{mn})_m$ は次式で与えられる。

$$\left(\frac{\lambda_e}{\lambda_{m1}} \right)_m = - \frac{r_i \ln(r_o/r_i)}{\pi (r_o - r_i)} \int_0^\pi \left(\frac{\partial \theta}{\partial R} \right)_{R=1} d\phi = \frac{r_i}{r_o - r_i} \quad (2.35)$$

$$\left(\frac{\lambda_e}{\lambda_{mn}} \right)_m = - \frac{r_o \ln(r_o/r_i)}{\pi (r_o - r_i)} \int_0^\pi \left(\frac{\partial \theta}{\partial R} \right)_{R=r_o/r_i} d\phi = \frac{r_o}{r_o - r_i} \quad (2.36)$$

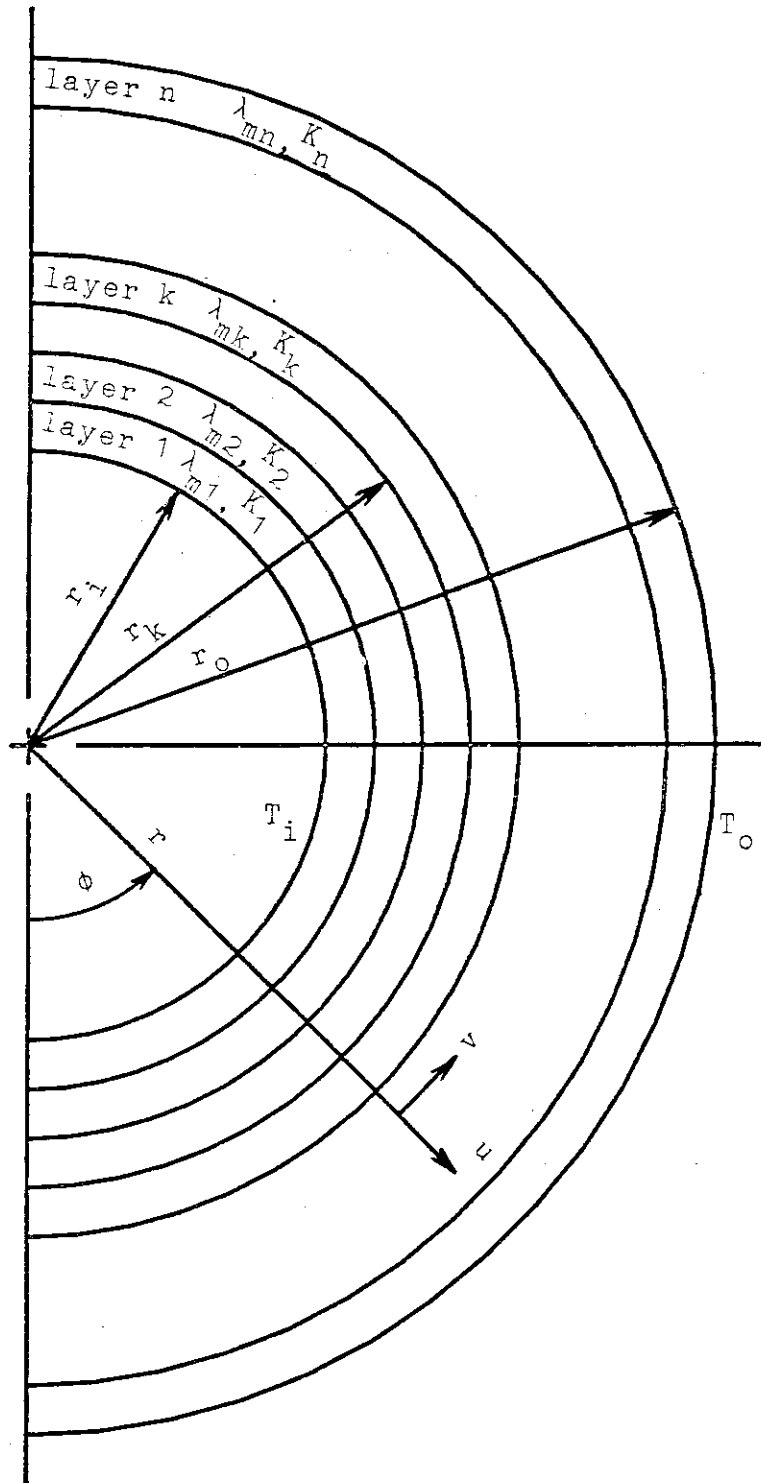


図 2.1 解析モデル

3. 数 値 解 法

運動方程式 (2.20) 及びエネルギー式 (2.21) を境界条件 (2.22) ~ (2.27) を用いて数值的に解き、無次元流れ関数 ψ 及び無次元温度 θ を求める。数値計算は差分法を用いて行なう。計算領域を図 3.1 に示す。計算領域は ϕ 方向に i_{\max} , R 方向に j_{\max} に mesh 分割され、各 mesh 中央にて $\psi_{i,j}$ 及び $\theta_{i,j}$ を定義する。また、 $i=0$, $i_{\max}+1$ は $\phi=0$, π の境界、 $j=0$, $j_{\max}+1$ は内側及び外側境界とする。なお、 ϕ 方向の mesh 分割は均等に行い、R 方向の mesh 分割は各断熱層で任意に行う。

3.1 運動方程式の差分表現

式 (2.20) は、中心差分近似を用いて次のように表せる。

$$\begin{aligned}
 -Ra_k Da_k \left(\frac{a_{mk}}{a_{m1}} \right) & \left\{ \frac{\theta_{i,j+1} - \theta_{i,j-1}}{1/2(\Delta R_{j+1} + 2\Delta R_j + \Delta R_{j-1})} \sin \phi_i \right. \\
 & \left. + \frac{\theta_{i+1,j} - \theta_{i-1,j}}{2\Delta \phi} \cos \phi_i \right\} \\
 = & \frac{(\Delta R_j + \Delta R_{j+1}) \psi_{i,j+1} - (\Delta R_{j+1} + 2\Delta R_j + \Delta R_{j-1}) \psi_{i,j} + (\Delta R_{j+1} + \Delta R_j) \psi_{i,j-1}}{1/2 (\Delta R_{j+1} + \Delta R_j) \Delta R_j (\Delta R_j + \Delta R_{j+1})} \\
 & + \frac{1}{R_j} \frac{\psi_{i,j+1} - \psi_{i,j-1}}{1/2 (\Delta R_{j+1} + 2\Delta R_j + \Delta R_{j-1})} + \frac{1}{R_j^2} \frac{\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}}{\Delta \phi^2} \quad (3.1)
 \end{aligned}$$

上式を $\psi_{i,j}$ について整理すると、

$$a_{ij} \psi_{i-1,j} + b_{ij} \psi_{i,j} + c_{ij} \psi_{i+1,j} + d_{ij} \psi_{i,j-1} + e_{ij} \psi_{i,j+1} = f_{ij} \quad (3.2)$$

ここに、

$$a_{ij} = \frac{1}{R_j^2 \Delta \phi^2} \quad (3.3)$$

$$b_{ij} = - \frac{2(\Delta R_{j+1} + 2\Delta R_j + \Delta R_{j-1})}{(\Delta R_{j+1} + \Delta R_j) \Delta R_j (\Delta R_j + \Delta R_{j+1})} - \frac{2}{R_j^2 \Delta \phi^2} \quad (3.4)$$

$$c_{ij} = \frac{1}{R_j^2 \Delta \phi^2} \quad (3.5)$$

$$d_{ij} = \frac{2}{\Delta R_j (\Delta R_j + \Delta R_{j-1})} - \frac{2}{R_j (\Delta R_{j+1} + 2 \Delta R_j + \Delta R_{j-1})} \quad (3.6)$$

$$e_{ij} = \frac{2}{(\Delta R_{j+1} + \Delta R_j) \Delta R_j} + \frac{2}{R_j (\Delta R_{j+1} + 2 \Delta R_j + \Delta R_{j-1})} \quad (3.7)$$

$$f_{ij} = -Ra_k Da_k \left(\frac{a_{mk}}{a_{m1}} \right) \left\{ \frac{2(\theta_{i,j+1} - \theta_{i,j-1})}{\Delta R_{j+1} + 2\Delta R_j + \Delta R_{j-1}} \sin \phi_j \right. \\ \left. + \frac{\theta_{i+1,j} - \theta_{i-1,j}}{2\Delta \phi} \cos \phi_i \right\} \quad (3.8)$$

但し、 ΔR_j 及び $\Delta \phi$ の境界条件は次のとおりである。

$$j = 1 \quad ; \Delta R_{j-1} = 0 \quad (3.9)$$

$$j = j_{\max} \quad ; \Delta R_{j+1} = 0 \quad (3.10)$$

$$i = 1, i_{\max} \quad ; \Delta \phi = \frac{3}{2} \Delta \phi \quad (3.11)$$

$$\Delta \phi^2 = \left(\frac{3}{4} \Delta \phi \right)^2 \quad (3.12)$$

また、 $\psi_{i,j}$ は境界条件 (2.22) ~ (2.24) より、

$$(1) R = \frac{r_i}{r_0 - r_i} \quad \text{のとき}$$

$$\frac{\partial \psi}{\partial \phi} = 0 \quad \text{であるから}$$

$$\psi_{0,0} = \psi_{1,0} = \psi_{2,0} = \dots = \psi_{i_{\max},0} = \psi_{i_{\max}+1,0} \quad (3.13)$$

$$(2) R = \frac{r_0}{r_0 - r_i} \quad \text{のとき}$$

$$\frac{\partial \psi}{\partial \phi} = 0 \quad \text{であるから}$$

$$\psi_{0, j_{\max}+1} = \psi_{1, j_{\max}+1} = \dots = \psi_{i_{\max}+1, j_{\max}+1} \quad (3.14)$$

(3) $\phi = 0, \pi$ のとき

$$\psi_{0,0} = \psi_{0,1} = \dots = \psi_{0, j_{\max}+1} \quad (3.15)$$

$$\psi_{i_{\max}+1,0} = \psi_{i_{\max}+1,1} = \dots = \psi_{i_{\max}+1, j_{\max}+1} \quad (3.16)$$

となる。

3.2 エネルギー方程式の差分表現

式(2.21)の右辺は中心差分近似を用い、左辺については温度微分項に風上差分近似を用いる。これは、左辺対流項の非線形性によって生じる解の不安定性を避けるためである。

式(2.21)右辺は次のように差分近似される。

$$\frac{a_{mk}}{a_{m1}} R_j \left\{ \frac{(\Delta R_j + \Delta R_{j-1}) \theta_{i,j+1} - (\Delta R_{j+1} + 2\Delta R_j + \Delta R_{j-1}) \theta_{i,j} + (\Delta R_{j+1} + \Delta R_j) \theta_{i,j-1}}{\Delta \phi^2} + \frac{1}{R_j} \cdot \frac{\theta_{i,j+1} - \theta_{i,j-1}}{1/2 (\Delta R_{j+1} + 2\Delta R_j + \Delta R_{j-1})} + \frac{1}{R_j^2} \frac{\theta_{i+1,j} - 2\theta_{i,j} + \theta_{i-1,j}}{\Delta \phi^2} \right\} \quad (3.17)$$

更に、 $\theta_{i,j}$ について整理すると、

$$a_{ij} \theta_{i-1,j} + b_{ij} \theta_{i,j} + c_{ij} \theta_{i+1,j} + d_{ij} \theta_{i,j-1} + e_{ij} \theta_{i,j+1} \quad (3.18)$$

ここに、

$$a_{ij} = \frac{a_{mk}}{a_{m1}} \frac{1}{R_j \cdot \Delta \phi^2} \quad (3.19)$$

$$b_{ij} = -\frac{a_{mk}}{a_{m1}} R_j \left\{ \frac{2(\Delta R_{j+1} + 2\Delta R_j + \Delta R_{j-1})}{(\Delta R_{j+1} + \Delta R_j) \Delta R_j (\Delta R_j + \Delta R_{j-1})} - \frac{2}{R_j^2 \Delta \phi^2} \right\} \quad (3.20)$$

$$c_{ij} = \frac{a_{mk}}{a_{m1}} \frac{1}{R_j \Delta \phi^2} \quad (3.21)$$

$$d_{ij} = \frac{a_{mk}}{a_{m1}} R_j \left\{ \frac{2}{\Delta R_j (\Delta R_j + \Delta R_{j-1})} - \frac{2}{R_j (\Delta R_{j+1} + 2\Delta R_j + \Delta R_{j-1})} \right\} \quad (3.22)$$

$$e_{ij} = \frac{a_{mk}}{a_{m1}} R_j \left\{ \frac{2}{\Delta R_j (\Delta R_j + \Delta R_{j-1})} + \frac{2}{R_j (\Delta R_{j+1} + 2\Delta R_j + \Delta R_{j-1})} \right\} \quad (3.23)$$

となる。

一方、式(2.21)の左辺は温度微分項に風上差分を適用するため、

$$(1) \quad \frac{\partial \psi}{\partial \phi} > 0, \quad -\frac{\partial \psi}{\partial R} > 0 \quad \text{のとき}$$

$$\left. \frac{\partial \psi}{\partial \phi} \right|_{i,j} = \frac{\theta_{i,j} - \theta_{i,j-1}}{1/2 (\Delta R_j + \Delta R_{j-1})} - \left. \frac{\partial \psi}{\partial R} \right|_{i,j} = \frac{\theta_{i,j} - \theta_{i-1,j}}{\Delta \phi} \quad (3.24)$$

$$(2) \quad \frac{\partial \psi}{\partial \phi} < 0, \quad -\frac{\partial \psi}{\partial R} > 0 \quad \text{のとき}$$

$$\left. \frac{\partial \psi}{\partial \phi} \right|_{i,j} = \frac{\theta_{i,j+1} - \theta_{i,j}}{1/2 (\Delta R_{j+1} + \Delta R_j)} - \left. \frac{\partial \psi}{\partial R} \right|_{i,j} = \frac{\theta_{i,j} - \theta_{i-1,j}}{\Delta \phi} \quad (3.25)$$

$$(3) \quad \frac{\partial \psi}{\partial \phi} > 0, \quad -\frac{\partial \psi}{\partial R} > 0 \quad \text{のとき}$$

$$\left. \frac{\partial \psi}{\partial \phi} \right|_{i,j} = \frac{\theta_{i,j} - \theta_{i,j-1}}{1/2 (\Delta R_j + \Delta R_{j-1})} - \left. \frac{\partial \psi}{\partial R} \right|_{i,j} = \frac{\theta_{i+1,j} - \theta_{i,j}}{\Delta \phi} \quad (3.26)$$

$$(4) \quad \frac{\partial \psi}{\partial \phi} < 0, \quad -\frac{\partial \psi}{\partial R} < 0 \quad \text{のとき}$$

$$\left. \frac{\partial \psi}{\partial \phi} \right|_{i,j} = \frac{\theta_{i,j+1} - \theta_{i,j}}{1/2 (\Delta R_j + \Delta R_{j-1})} - \left. \frac{\partial \psi}{\partial R} \right|_{i,j} = \frac{\theta_{i+1,j} - \theta_{i,j}}{\Delta \phi} \quad (3.27)$$

となる。

従って、式(3.18)～(3.27)より、式(2.21)は次のように表すことができる。

$$a'_{ij} \theta_{i-1,j} + b'_{ij} \theta_{i,j} + c'_{ij} \theta_{i+1,j} + d'_{ij} \theta_{i,j-1} + e'_{ij} \theta_{i,j+1} = 0 \quad (3.28)$$

ここに、

$$(1) \quad \frac{\partial \psi}{\partial \phi} > 0, \quad -\frac{\partial \psi}{\partial R} > 0 \quad \text{のとき}$$

$$a'_{ij} = a_{ij} - \left. \frac{\partial \psi}{\partial R} \right|_{i,j} \frac{1}{\Delta \phi} \quad (3.29)$$

$$b'_{ij} = b_{ij} - \left. \frac{\partial \psi}{\partial \phi} \right|_{i,j} \frac{2}{\Delta R_j + \Delta R_{j+1}} + \left. \frac{\partial \psi}{\partial R} \right|_{i,j} \frac{1}{\Delta \phi} \quad (3.30)$$

$$c'_{ij} = c_{ij} \quad (3.31)$$

$$d'_{ij} = d_{ij} + \left. \frac{\partial \psi}{\partial \phi} \right|_{i,j} \frac{2}{\Delta R_j + \Delta R_{j-1}} \quad (3.32)$$

$$e'_{ij} = e_{ij} \quad (3.33)$$

(2) $\frac{\partial \psi}{\partial \phi} < 0$, $-\frac{\partial \psi}{\partial R} > 0$ のとき

$$a'_{ij} = a_{ij} - \left. \frac{\partial \psi}{\partial R} \right|_{i,j} \frac{1}{\Delta \phi} \quad (3.34)$$

$$b'_{ij} = b_{ij} + \left. \frac{\partial \psi}{\partial \phi} \right|_{i,j} \frac{2}{\Delta R_{j+1} + \Delta R_j} + \left. \frac{\partial \psi}{\partial R} \right|_{i,j} \frac{1}{\Delta \phi} \quad (3.35)$$

$$c'_{ij} = c_{ij} \quad (3.36)$$

$$d'_{ij} = d_{ij} \quad (3.37)$$

$$e'_{ij} = e_{ij} - \left. \frac{\partial \psi}{\partial \phi} \right|_{i,j} \frac{2}{\Delta R_{j+1} + \Delta R_j} \quad (3.38)$$

(3) $\frac{\partial \psi}{\partial \phi} > 0$, $-\frac{\partial \psi}{\partial R} < 0$ のとき

$$a'_{ij} = a_{ij} \quad (3.39)$$

$$b'_{ij} = b_{ij} - \left. \frac{\partial \psi}{\partial \phi} \right|_{i,j} \frac{2}{\Delta R_j + \Delta R_{j-1}} - \left. \frac{\partial \psi}{\partial R} \right|_{i,j} \frac{1}{\Delta \phi} \quad (3.40)$$

$$c'_{ij} = c_{ij} + \left. \frac{\partial \psi}{\partial R} \right|_{i,j} \frac{1}{\Delta \phi} \quad (3.41)$$

$$d'_{ij} = d_{ij} + \left. \frac{\partial \psi}{\partial \phi} \right|_{i,j} \frac{2}{\Delta R_j + \Delta R_{j-1}} \quad (3.42)$$

$$e'_{ij} = e_{ij} \quad (3.43)$$

$$(4) \quad \frac{\partial \psi}{\partial \phi} < 0, \quad -\frac{\partial \psi}{\partial R} < 0 \quad \text{のとき}$$

$$a'_{ij} = a_{ij} \quad (3.44)$$

$$b'_{ij} = b_{ij} + \frac{\partial \psi}{\partial \phi} \Big|_{i,j} \frac{2}{\Delta R_{j+1} + \Delta R_j} - \frac{\partial \psi}{\partial R} \Big|_{i,j} \frac{1}{\Delta \phi} \quad (3.45)$$

$$c'_{ij} = c_{ij} + \frac{\partial \psi}{\partial R} \Big|_{i,j} \frac{1}{\Delta \phi} \quad (3.46)$$

$$d'_{ij} = d_{ij} \quad (3.47)$$

$$e'_{ij} = e_{ij} - \frac{\partial \psi}{\partial \phi} \Big|_{i,j} \frac{2}{\Delta R_{j+1} + \Delta R_j} \quad (3.48)$$

但し,

$$\frac{\partial \psi}{\partial \phi} \Big|_{i,j} = \frac{\psi_{i+1,j} - \psi_{i-1,j}}{2 \Delta \phi} \quad (3.49)$$

$$\frac{\partial \psi}{\partial R} \Big|_{i,j} = \frac{\psi_{i,j+1} - \psi_{i,j-1}}{1/2 (\Delta R_{j+1} + 2 \Delta R_j + \Delta R_{j-1})} \quad (3.50)$$

ここで、 ΔR_j 及び $\Delta \phi$ の境界条件は次のとおりである。

$$j = 1 \quad ; \Delta R_{j-1} = 0 \quad (3.51)$$

$$j = j_{\max} \quad ; \Delta R_{j+1} = 0 \quad (3.52)$$

$$i = 1, i_{\max} \quad ; \Delta \phi = \frac{3}{2} \Delta \phi \quad (3.53)$$

$$\Delta \phi^2 = \left(\frac{3}{4} \Delta \phi \right)^2 \quad (3.54)$$

また、 $\theta_{i,j}$ は境界条件 (2.22) ~ (2.24) より、

$$(1) \quad R = \frac{r_i}{r_0 - r_i} \quad \text{のとき}$$

$\theta_1 = 1$ であるから

$$\theta_{0,0} = \theta_{1,0} = \theta_{2,0} = \dots = \theta_{j_{\max},0} = 1 \quad (3.55)$$

$$(2) \quad R = \frac{r_0}{r_0 - r_i} \quad \text{のとき}$$

ディスクに出力された計算結果は、図形処理プログラムの入力データとして読み込まれ、図形処理演算を施される。結果は、流れ関数線図、等温度線図、速度ベクトル図、実効熱伝導率分布図等に整理され、それぞれの図をレーザープリンターで出力する。

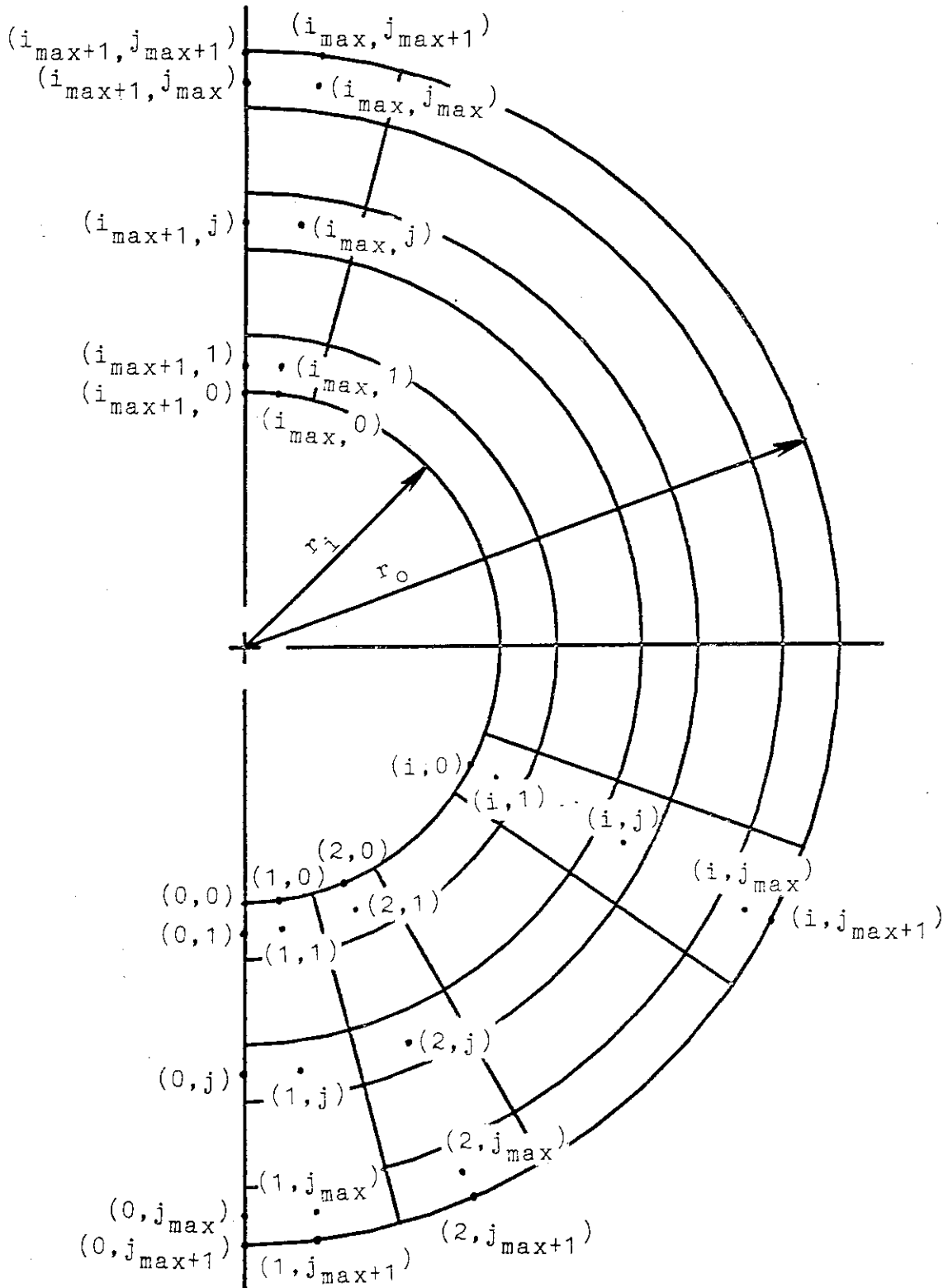


図 3.1 計算領域

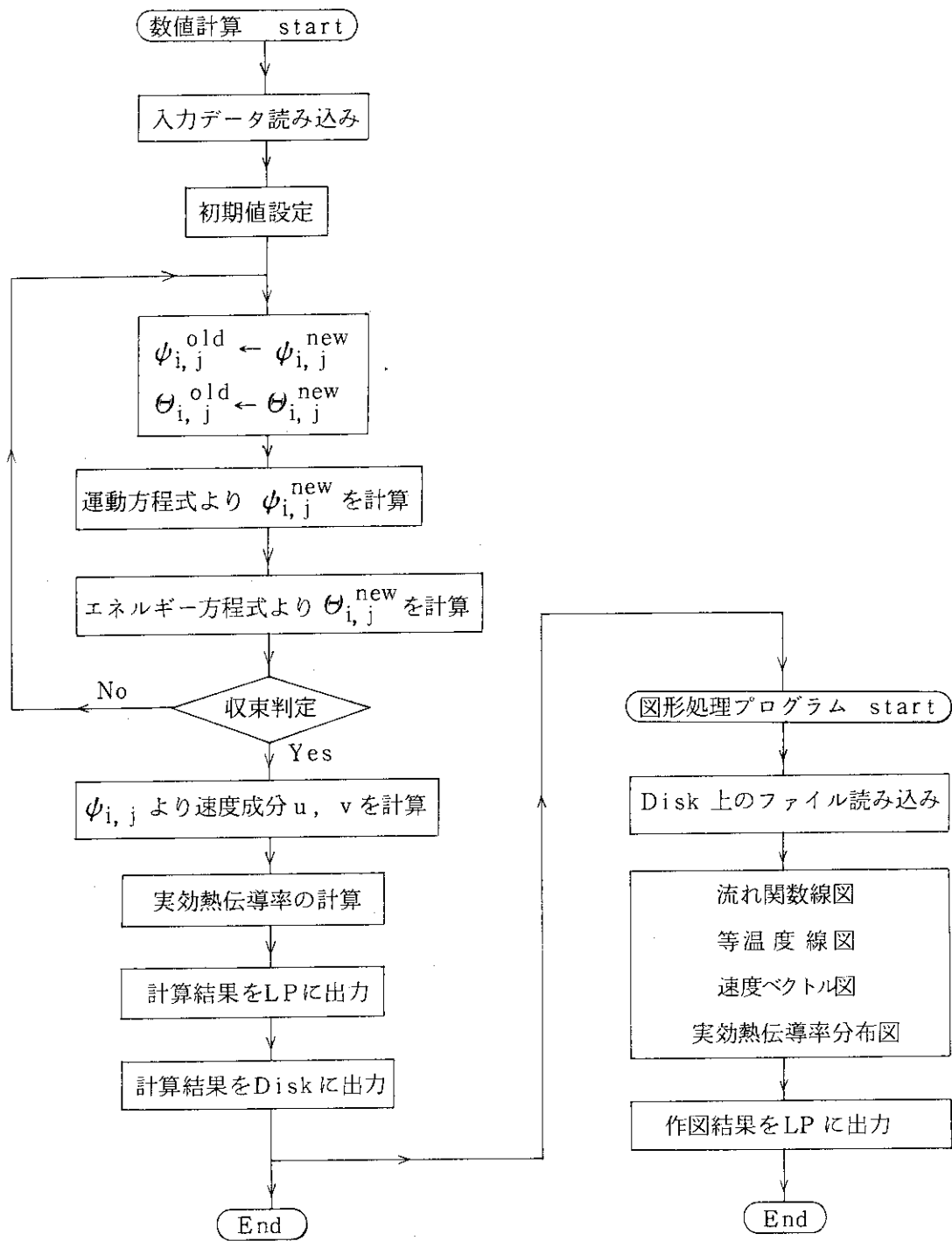


図 3.2 数値計算の概要 (フローチャート)

4. プログラム構成

4.1 計算コードのプログラム構成

計算コードのプログラム構成を図 4.1 に示す。数値計算を実行・制御するメインプログラム (MAIN) を中心にして、次のサブプログラム群が連結されている。

- 1) DATAIN : 入力データカードを読み込む。
- 2) INISET : $\psi_{i,j}$ 及び $\Theta_{i,j}$ の初期値設定。
- 3) ERRMSG : 入力データカードの読み込みエラーや不足エラーのとき、あるいはメッシュ分割総数の制限エラーなどのときにエラーメッセージ出力。
- 4) GLPOUT : 断熱層数, メッシュ分割総数, 半径方向及び円周方向のメッシュ幅を出力。
- 5) MOTION : 差分表現した運動方程式を n 次の連立方程式に変換し, サブプログラム MATRIX を用いて $\psi_{i,j}$ を求める。
- 6) ENERGY : 差分表現したエネルギー式を n 次の連立方程式に変換し, MATRIX を用いて $\Theta_{i,j}$ を求める。
- 7) MATRIX : n 次の連立方程式を 3 重対角行列に変換し, ガウス消却法により解を算出する。
- 8) VELCTY : 流れ関数 $\psi_{i,j}$ の計算結果から, R 方向及び ϕ 方向の速度成分 u, v を求める。
- 9) THRMAL : 実効熱伝導率の計算。
- 10) OUTPUT : 流れ関数 $\psi_{i,j}$, 無次元温度 $\Theta_{i,j}$, 速度成分 u, v 及び実効熱伝導率を LP に出力する。
- 11) RESULT 1 : 流れ関数 $\psi_{i,j}$ 及び無次元温度 $\Theta_{i,j}$ の LP 出力形式の設定。
- 12) RESULT 2 : 速度成分 u, v の LP 出力形式の設定。
- 13) FIGURE : 計算結果の値をディスクに順編成ファイルとして書き出す。

4.2 計算コード入力データ

本プログラムの入力データは、カード入力データである。以下にその様式を記す。なお、記号はプログラムの変数名を表している。

- (1) 制御データカード (1 枚)

カラム	記号	記述形式	説明
1~5	I OPT 1	I 5	計算結果のファイル書込み指定 = 0 : 書き込まない ≠ 0 : 書き込む

(2) 形状データカード (1枚)

カラム	記号	記述形式	説明
1~5	KMAX	I 5	断熱層数
6~10	I MAX	I 5	ϕ 方向分割数 (偶数)
11~20	R MAX	F 10.0	r_o/r_i 断熱層の外壁までの半径

(3) 層パラメータカード (KMAX枚)

カラム	記号	記述形式	説明
1~5	JMAX	I 5	第K層R方向分割数
6~15	R	F 10.0	r_k/r_{in} 第K層半径
16~25	RA	F 10.0	レーリー数
26~35	DA	F 10.0	ダルシー数
36~45	AM	F 10.0	a_{mk}/a_{m1} 温度伝導率比

(4) 収束計算パラメータカード (1枚)

カラム	記号	記述形式	説明
1~5	ITMAX	I 5	反復回数の制限値
6~15	ESP 1	F 10.0	$\psi_{i,j}$ の収束判定値 (ϵ_1)
16~25	ESP 2	F 10.0	$\theta_{i,j}$ の収束判定値 (ϵ_2)

4.3 図形処理プログラムの構成

本プログラムは、計算コードの出力した順編成ファイルを読み込んで、流れ関数 ψ 及び無次元温度 θ の等高線図及び鳥かん図、速度ベクトル図、実効熱伝導率分布、半径方向及び角度方向の温度分布図を出力するためのものである。等高線図、鳥かん図及び速度ベクトル図の作図には ARGUS を用い、実効熱伝導率分布、半径方向及び角度方向の温度分布図の作図には PXCNTL を用いている。ARGUS 及び PXCNTL は、いずれも原研で所有管理されている汎用図形処理コードである。

図 4.2 にプログラム構成を示す。図形処理を実行・制御するメインプログラム (MAIN) を中心に次のサブプログラム群が連結されている。以下に主要なサブプログラム群の内容を記す。

- 1) DATAIN : 計算コードで出力した順編成ファイルを読み込む。
- 2) CARDIN : 入力データカードを読み込む。
- 3) ERRMSG : 入力カードのデータ読み込みエラーのときにエラーメッセージを出力する。
- 4) DSPARG : 等高線図、鳥かん図及び速度ベクトル図を作図する。
- 5) DSPGRP : 実効熱伝導率分布図、半径比 $r/r_{in} = \text{const.}$ 及び角度 $\phi = \text{const.}$ の温度分布を作図する。
- 6) VALFIX : 入力データの半径比 r/r_{in} 及び角度 ϕ に最も近い半径及び角度の $\theta_{i,j}$ の配列要素を抽出する。

4.4 図形処理プログラムの入力データ

図形処理プログラムのカード入力データの様式を以下に記す。但し、それぞれの作図出力の指定には各1枚のカードが必要である。

1) 流れ関数の等高線図及び鳥かん図：

カード1カラム目に A を指定する。

2) 無次元温度の等高線図及び鳥かん図：

カード1カラム目に B を指定する。

3) 速度ベクトル図：

カード1カラム目に C を指定する。

4) 内外壁実効熱伝導率の ϕ 方向分布：

カード1カラム目に D を指定する。

5) ϕ 方向温度分布 ($r/r_{in} = \text{const.}$)

カード1カラム目に E ,

4, 5カラム目に ϕ 方向分布出力個数 (≤ 10), 6~10, 11~15, 16~20, ……カラムに ϕ 方向分布を出力する r/r_{in} の値を出力個数だけ指定する。

6) R方向温度分布：

カード1カラム目に F ,

4, 5カラム目にR方向分布出力個数 (≤ 10), 6~10, 11~15, 16~20, ……カラムにR方向分布を出力する ϕ 値を出力個数だけ指定する。

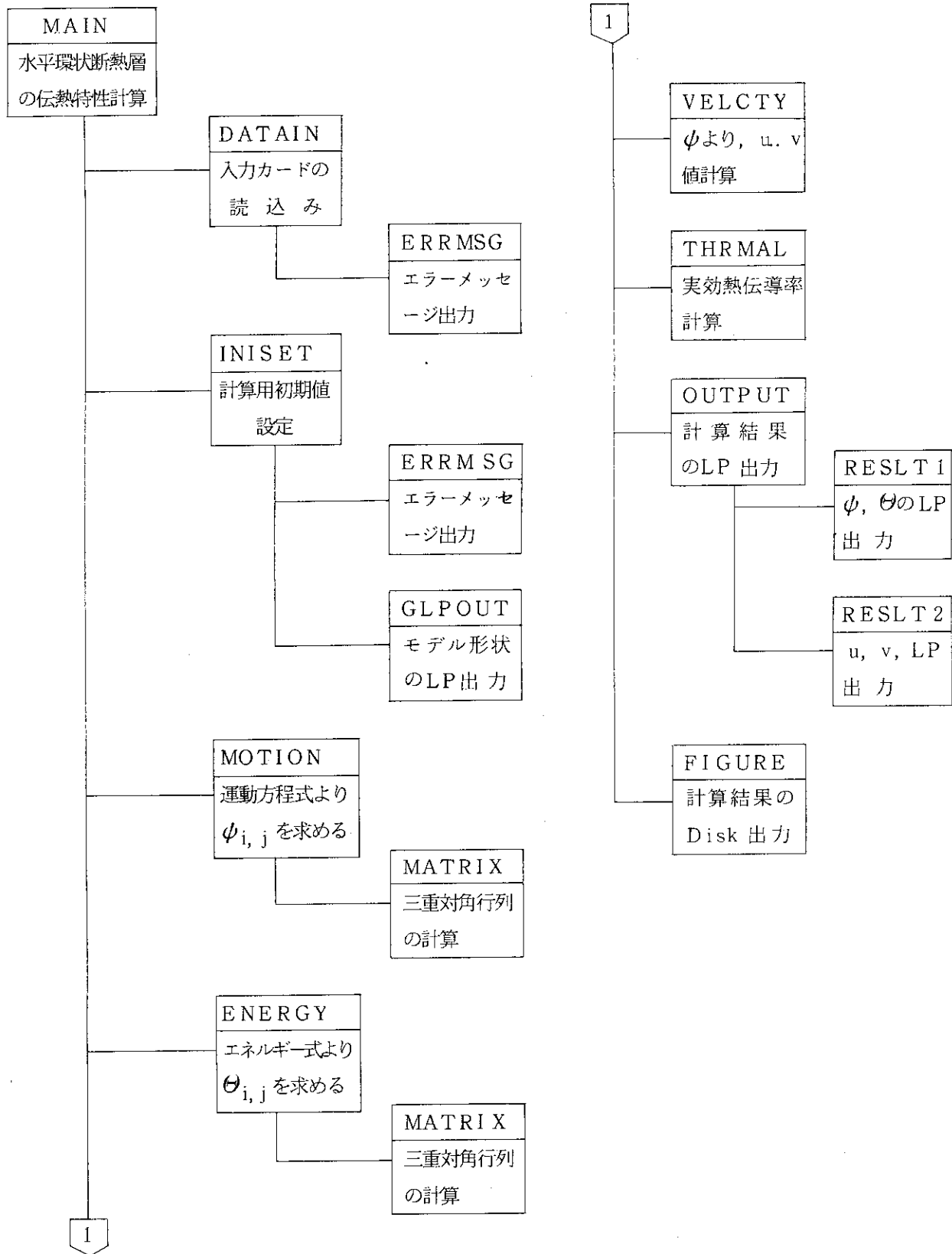


図 4.1 計算コードのプログラム構成

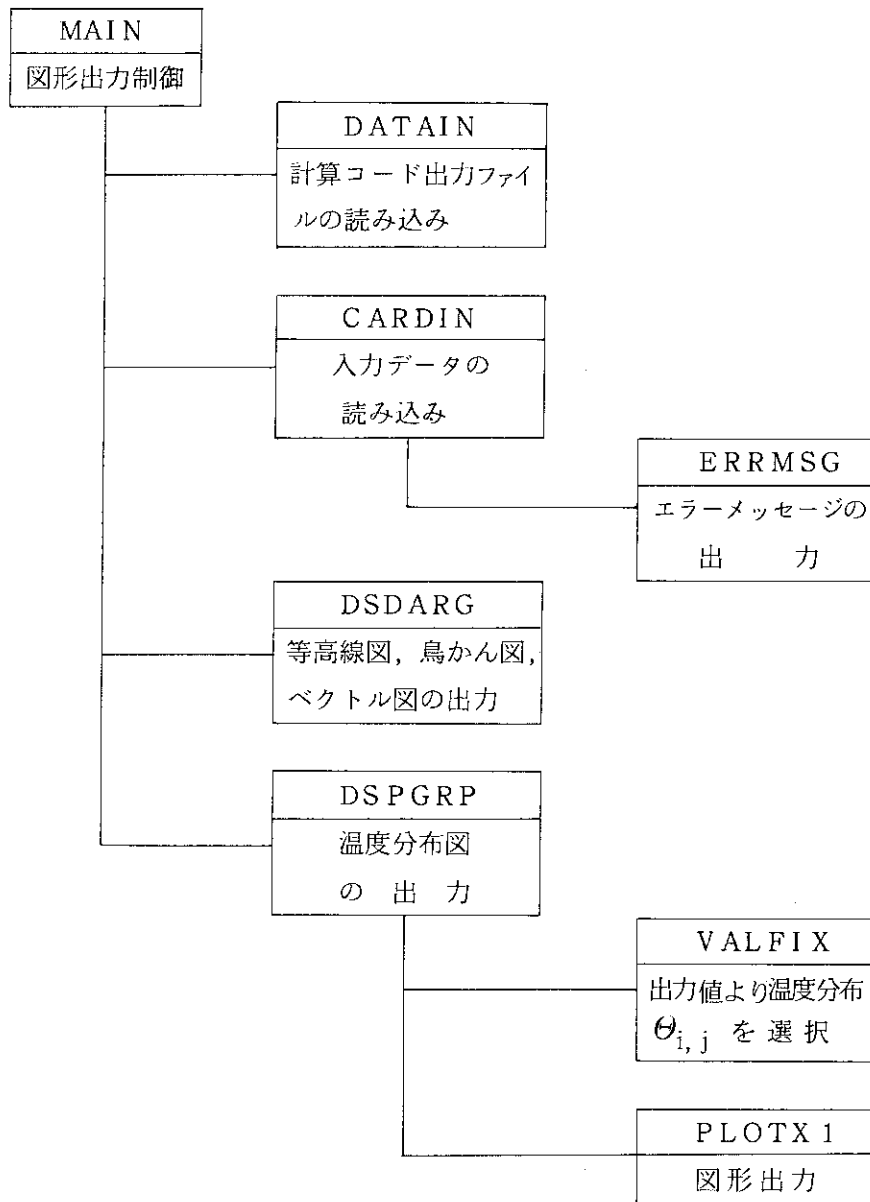


図 4.2 図形処理プログラムの構成

5. 計算コード

計算コードのCOMMONブロックとパラメータセットの内容は次のとおりである。

COMMON/CARD 1/				
No.	Variable Name	Type	Content.	Unit
1	IOPT1	I * 1	計算結果 disk 出力 Option = 0 : 出力せず ≠ 0 : 出力する	—

COMMON/CARD 2/				
No.	Variable Name	Type	Content.	Unit
1	KMAX	I * 4	断熱層数 ($1 \leq, \leq LMAX$)	—
2	IMAX	I * 4	円周方向分割数 ($1 \leq, \leq JMAXX$)	—
3	RMAX	R * 8	r_o/r_i 断熱材の外側半径の無次元化	—

COMMON/CARD 3/				
No.	Variable Name	Type	Content.	Unit
1	JMAX (LMAXX)	I * 4	第K層半径方向分割数 ($1 \leq, \leq KMAXX$)	—
2	R (LMAXX)	R * 8	r_k/r_i 第K層無次元化半径	—
3	RA (LMAXX)	R * 8	Rak (レーリー数)	—
4	DA (LMAXX)	R * 8	Dak (ダルシー数)	—
5	AM (LMAXX)	R * 8	a_{mk}/a_{m1} 第K層温度伝導率比	—

COMMON/CARD 4/				
No.	Variable Name	Type	Content.	Unit
1	ITMAX	I * 4	収束計算反復回数 ($1 \leq, \leq MAXCVG$)	—
2	ESP1	R * 8	$\psi_{i,j}$ 収束判定値	—
3	ESP2	R * 8	$\theta_{i,j}$ 収束判定値	—

COMMON / GMTRY /				
No.	Variable Name	Type	Content.	Unit
1	IXMAX	I * 4	円周方向総分割数 ($1 \leq, \leq \text{IMAXX}$)	—
2	JXMAX	I * 4	半径方向総分割数 ($1 \leq, \leq \text{JMAXX}$)	—
3	DR (LMAXX)	R * 8	第K層 R 方向 Mesh 幅	—
4	DPHI	R * 8	ϕ 方向 Mesh 幅	rad.
5	LPOINT (LMAXX)	R * 8	第K層を分割した最も内側の層アドレス	—
6	RX (MAXX)	R * 8	Mesh 中心 R 座標	—
7	PHIX (MAXX)	R * 8	Mesh 中心 ϕ 座標	rad.

COMMON / FLUID /				
No.	Variable Name	Type	Content.	Unit
1	THTN (0 :IMAXX 1,)	R * 8	Mesh 中心温度の New 値	—
2	PSIN (0 :IMAXX 1,)	R * 8	Mesh 中心流れ関数値の New 値	—
3	THTO (0 :IMAXX 1,)	R * 8	Mesh 中心温度の Old 値	—
4	PSIO (0 :IMAXX 1,)	R * 8	Mesh 中心流れ関数値の Old 値	—

COMMON / VECTR /				
No.	Variable Name	Type	Content.	Unit
1	U (IMAXX, JMAXX)	R * 8	R 方向速度成分	1/m
2	V (IMAXX, JMAXX)	R * 8	ϕ 方向速度成分	1/m

COMMON / LAMDA /				
No.	Variable Name	Type	Content.	Unit
1	ALAMI (0 :IMAXX 1)	R * 8	λ_e / λ_{m1} 内壁の実効熱伝導率	—
2	ALAMO (0 :IMAXX 1)	R * 8	λ_e / λ_{mn} 外壁の実効熱伝導率	—
3	ALAMIM	R * 8	内壁平均実効熱伝導率	—
4	ALAMOM	R * 8	外壁 "	—
5	RADAAV	R * 8	断熱層平均 $Ra * Da$	—
6	AMAV	R * 8	" 温度伝導率比	—

PARAMETER			
Variable Name	type	Value	Comment
LMAXX	I * 4	100	断熱層の最大配列個数
IMAXX	I * 4	100	ϕ 方向 Mesh の最大配列個数
IMAXX1	I * 4	MAXX + 1	IMAXX + 1
JMAXX	I * 4	100	R方向 Mesh の最大配列個数
JMAXX1	I * 4	JMAXX + 1	JMAXX + 1
KMAXX	I * 4	100	第K層のR方向 Mesh の最大配列個数

計算コードは次のとおりである。


```

C
C *****
C * MAIN PROGRAM *
C *****
C
C*****  DECLARATION  *****
C
000001  IMPLICIT  REAL*8  (A-H,O-Z)
*INCLUDE PARAM, FIXED, INSOURCE
000002  PARAMETER  ( LMAXX=100 ,KMAXX = 100 )
000003  PARAMETER  ( IMAXX=100 ,IMAXX1 = IMAXX+1 ,IMAXX2 = IMAXX+2 )
000004  PARAMETER  ( JMAXX=100 ,JMAXX1 = JMAXX+1 ,JMAXX2 = JMAXX+2 )
*INCLUDE CARD, FIXED, INSOURCE
000005  COMMON  /CARD1/  IOPT01
000006  COMMON  /CARD2/  KMAX ,IMAX ,RMAX
000007  COMMON  /CARD3/  JMAX(LMAXX) ,R(LMAXX) ,RA(LMAXX) ,DA(LMAXX) ,
AM(LMAXX)
000008  COMMON  /CARD4/  ESP1 ,ESP2 ,ITMAX
*INCLUDE COMGF, FIXED, INSOURCE
000009  COMMON  /GMTRY/  IXMAX, JXMAX, DR(LMAXX), DPHI,
LPOINT(LMAXX), RX(JMAXX), PHIX(IMAXX)
000010  COMMON  /FLUID/  THTN(O:IMAXX1,O:JMAXX1),PSIN(O:IMAXX1,O:JMAXX1)
THTO(O:IMAXX1,O:JMAXX1),PSIO(O:IMAXX1,O:JMAXX1)
*INCLUDE COMUV, FIXED, INSOURCE
000011  COMMON  /VECTR/  U(IMAXX,JMAXX),V(IMAXX,JMAXX)
C
C*****  PROCEDURE  *****
C
C*** INPUT OF CARD DATA ***
C
000012  CALL  DATAIN(IRTN)
C
000013  IF( IRTN .NE. 0 ) GO TO 3000
C
C*** INITIAL VALUE SETTING ***
C
000014  CALL  INISET(IRTN)
C
000015  IF( IRTN .NE. 0 ) GO TO 3000
C
C*** CONVERGENCE DISPOSITION ***
C
000016  ITR = 0
C
000017  SGMTN = 0.0
000018  SGMPN = 0.0
C
000019  DO 4000  J = 1,JXMAX
000020  DO 4100  I = 1,IXMAX
000021  SGMTN = SGMTN + DABS(THTN(I,J))
000022  SGMPN = SGMPN + DABS(PSIN(I,J))
000023  4100  CONTINUE
000024  4000 CONTINUE
C
000025  3100 CONTINUE
C
000026  ITR = ITR + 1
C
000027  DO 4200  J = 0,JXMAX+1
000028  DO 4300  I = 0,IXMAX+1
000029  THTO(I,J) = THTN(I,J)
000030  PSIO(I,J) = PSIN(I,J)
000031  4300  CONTINUE
000032  4200 CONTINUE
C
C*** THETA CALCULATION BY EQUATION OF MOTION ***
C
000033  CALL  MOTION(IRTN)
C
000034  IF( IRTN .NE. 0 ) GO TO 3000
C
C*** PSI CALCULATION BY EQUATION OF ENERGY ***
C
000035  CALL  ENERGY(IRTN)
C
000036  IF( IRTN .NE. 0 ) GO TO 3000
C

```

```

C*** CONVERGENCE DESITION ***
C
000037      SGMT0 = SGMTN
000038      SGMP0 = SGMPN
000039      SGMTN = 0
000040      SGMPN = 0
C
000041      DO 4400 J = 1,JXMAX
000042          DO 4500 I = 1,IXMAX
000043              SGMTN = SGMTN + DABS(THTN(I,J))
000044              SGMPN = SGMPN + DABS(PSIN(I,J))
000045      4500      CONTINUE
000046      4400 CONTINUE
C
000047      ESPT = DABS( (SGMTN-SGMT0)/SGMTN )
000048      ESPP = DABS( (SGMPN-SGMP0)/SGMPN )
C
000049      IF( MOD(ITR,30).EQ.1 ) WRITE(6,6015)ITR,ESPT,ESPP
000050      IF( ( ESPT .LT. ESP1 ) .AND. ( ESPP .LT. ESP2 ) ) THEN
000051          WRITE(6,6020) ITR,ESPT,ESPP
000052          GO TO 3200
000053      ENDIF
C
000054      IF( ITR .GE. ITMAX ) THEN
000055          WRITE(6,6030) ITR,ESPT,ESPP
000056          GO TO 3200
000057      ENDIF
C
000058                                          GO TO 3100
C
000059      3000 CONTINUE
000060          WRITE(6,6000)
000061                                          GO TO 9999
C
000062      3200 CONTINUE
C
C
C*** VELOCITY(U & V) ***
C
000063      CALL      VELCTY
C
C*** EFFECTIVE THERMAL CONDUCTIVITY ***
C
000064      CALL      THRMAL
C
C*** PRINT OUT THE RESULT ***
C
000065      CALL      OUTPUT
C
C*** FIGURE DATA SET ***
C
000066      IF( IOPT01.NE.0 ) CALL      FIGURE
C
C
000067      9999 CONTINUE
C
C
000068      STOP
C
000069      6000 FORMAT(1H0,20X,'ERROR STOP')
000070      6015 FORMAT(5X,'ITERATION COUNT = ',15,'      ESPT = ',1PD12.4,
.          '      ESPP = ',1PD12.4)
000071      6020 FORMAT(////,5X,'CALCULATION CONVERGED',/,
.          '      10X,'ITERATION COUNT = ',15,
.          '      ESPT = ',1PD12.4,'      ESPP = ',1PD12.4)
C
000072      6030 FORMAT(////,5X,'CALCULATION NOT CONVERGED',/,
.          '      10X,'ITERATION COUNT = ',15,
.          '      ESPT = ',1PD12.4,'      ESPP = ',1PD12.4)
000073      END

```

```

00006600
00006700
00006800
00006900
00007000
00007100
00007200
00007300
00007400
00007500
00007600
00007700
00007800
00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00009400
00009500
00009600
00009700
00009800
00009900
00010000
00010100
00010200
00010300
00010400
00010500
00010600
00010700
00010800
00010900
00011000
00011100
00011200
00011300
00011400
00011500
00011600
00011700
00011800
00011900
00012000
00012100
00012200
00012300
00012400
00012500
00012600
00012700
00012800
00012900
00013000
00013100
00013200
00013300
00013400
00013500

```

```

000001      SUBROUTINE  DATAIN(IRTN)                                00000100
C                                                    00000200
C*****                                                    00000300
C*                                                    *00000400
C*      FUNCTION ;                                                    *00000500
C*                                                    *00000600
C*      OUTPUT OF INPUT DATA CARD IMAGE LIST AND                *00000700
C*                                                    *00000800
C*      SET-UP OF INPUT DATA VALUE.                            *00000900
C*                                                    *00001000
C*      DATE      ; 84/02/15                                    *00001100
C*                                                    *00001200
C*      ARGUMENT ;                                           (TYPE,I/O,UNIT) *00001300
C*                                                    *00001400
C*      (1) IRTN  *** RETURN CODE                               (I*4,0,-) *00001500
C*                                                    *00001600
C*                                                    *00001700
C*                                                    *00001800
C*****                                                    *00001900
C                                                    00002000
C*****  DECLARATION  ****                                     00002100
C                                                    00002200
000002      IMPLICIT  REAL*8  (A-H,O-Z)                            00002300
*INCLUDE PARAM, FIXED, INSOURCE                                00002400
000003      PARAMETER  ( LMAXX=100 ,KMAXX = 100 )                  00000010
000004      PARAMETER  ( IMAXX=100 ,IMAXX1 = IMAXX+1 ,IMAXX2 = IMAXX+2 ) 00000020
000005      PARAMETER  ( JMAXX=100 ,JMAXX1 = JMAXX+1 ,JMAXX2 = JMAXX+2 ) 00000030
*INCLUDE CARD, FIXED, INSOURCE                                00002500
000006      COMMON  /CARD1/  IOPT01                                00000011
000007      COMMON  /CARD2/  KMAX      ,IMAX      ,RMAX            00000012
000008      COMMON  /CARD3/  JMAX(LMAXX) ,R(LMAXX) ,RA(LMAXX) ,DA(LMAXX),
AM(LMAXX)
000009      COMMON  /CARD4/  ESP1      ,ESP2      ,ITMAX          00000040
000010      CHARACTER*80  CARD                                    00002600
000011      CHARACTER*21  LABEL                                  00002700
C                                                    00002800
C*****  PROCEDURE.  ****                                     00002900
C                                                    00003000
000012      IRTN = 0                                           00003100
000013      IERR = 0                                           00003200
000014      ICD = 0                                           00003300
000015      KMAX = 0                                           00003400
000016      MAXCVG = 1000                                       00003500
000017      3000 CONTINUE                                       00003600
C                                                    00003700
C***  CARD READING ***                                       00003800
C                                                    00003900
000018      READ(5,5000,END=3333) CARD                          00004000
C                                                    00004100
000019      ICD = ICD + 1                                       00004200
C                                                    00004300
000020      IF( MOD(ICD,50) .EQ. 1 ) THEN                       00004400
000021          WRITE(6,6000)                                    00004500
000022          WRITE(6,6100) (NO,NO=1,8)                      00004600
000023      ENDIF                                              00004700
C                                                    00004800
C***  CARD IMAGE WRITING ***                                   00004900
C                                                    00005000
000024      WRITE(6,6200) ICD,CARD                              00005100
C                                                    00005200
000025      IF( MOD(ICD,50) .EQ. 0 ) THEN                       00005300
000026          WRITE(6,6100) (NO,NO=1,8)                      00005400
000027      ENDIF                                              00005500
000028      GO TO 3000                                          00005600
000029      3333 CONTINUE                                       00005700
C                                                    00005800
C***  INPUT DATA REWIND ***                                  00005900
C                                                    00006000
000030      REWIND 5                                           00006100
C                                                    00006200
C                                                    00006300
C****  SET-UP INPUT DATA VALUE & ****                       00006400
C****  INPUT DATA SUMMARY WRITING ****                       00006500
C ,                                                            00006600
000031      WRITE(6,6300)                                       00006700

```

```

C*** FORM DATUM SET ***
C
000032      INCD = 0
000033      READ(5,7000) IOPT01
000034      READ(5,7000) KMAX,IMAX,RMAX
C
000035      INCD = INCD + 1
000036      LABEL = 'FORM DATA'
C
000037      ICHECK = 0
000038      IF( ( KMAX.LT.1 ) .OR. ( KMAX.GT.LMAXX ) ) ICHECK = 1
000039      IF( ( IMAX.LT.1 ) .OR. ( IMAX.GT.IMAXX ) ) ICHECK = ICHECK + 10
000040      IF( RMAX.LE.1 ) ICHECK = ICHECK + 100
C
000041      IF( ICHECK.NE.0 ) THEN
000042          IRTN = 1
000043          CALL ERRMSG( IRTN )
000044          WRITE(6,6800) INCD,LABEL,ICHECK
000045
000046      ENDIF
C
000047      WRITE(6,6400) KMAX,IMAX,RMAX
C
C*** EACH LAYER PARAMETER SET ***
C
000048      WRITE(6,6500)
000049      DO 4000 K = 1,KMAX
000050          READ(5,7100) JMAX(K),R(K),RA(K),DA(K),AM(K)
C
000051          INCD = INCD + 1
000052          LABEL = 'EACH LAYER PARAM DATA'
C
000053          ICHECK = 0
000054          IF( ( JMAX(K).LT.1 ) .OR. ( JMAX(K).GT.KMAXX ) )
C
000055              IF( ( R(K).LT.1 ) .OR. ( R(K).GT.RMAX ) )
C
000056              IF( RA(K).LE.0
C
000057              IF( DA(K).LE.0
C
000058              IF( AM(K).LE.0
C
000059          IF( ICHECK.NE.0 ) THEN
000060              IRTN = 1
000061              CALL ERRMSG( IRTN )
000062              WRITE(6,6800) INCD,LABEL,ICHECK
000063
000064          ENDIF
C
000065          WRITE(6,6600) JMAX(K),R(K),RA(K),DA(K),AM(K)
C
000066      4000 CONTINUE
000067      WRITE(6,6650)
C
C*** CONVERGE COMPUTE PARAMETER SET ***
C
000068      READ(5,7200) ITMAX,ESP1,ESP2
C
000069      INCD = INCD + 1
000070      LABEL = 'CONV. PARAM DATA'
C
000071      ICHECK = 0
000072      IF( ( ITMAX.LT.1 ) .OR. ( ITMAX.GT.MAXCVG ) ) ICHECK = 1
000073      IF( ESP1.LE.0 ) ICHECK = ICHECK + 10
000074      IF( ESP2.LE.0 ) ICHECK = ICHECK + 100
C
000075      IF( ICHECK.NE.0 ) THEN
000076          IRTN = 1
000077          CALL ERRMSG( IRTN )
000078          WRITE(6,6800) INCD,LABEL,ICHECK
000079      ENDIF
C
000080      WRITE(6,6700) ITMAX,ESP1,ESP2
C
C
C
000081      9000 CONTINUE
C
C
000082      RETURN
C

```

```

00006800
00006900
00007000
00007100
00007200
00007300
00007400
00007500
00007600
00007700
00007800
00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00009400
00009500
00009600
00009700
00009800
00009900
00010000
00010100
00010200
00010300
00010400
00010500
00010600
00010700
00010800
00010900
00011000
00011100
00011200
00011300
00011400
00011500
00011600
00011700
00011800
00011900
00012000
00012100
00012200
00012300
00012400
00012500
00012600
00012700
00012800
00012900
00013000
00013100
00013200
00013300
00013400
00013500
00013600
00013700
00013800
00013900
00014000
00014100
00014200
00014300
00014400
00014500
00014600
00014700
00014800
00014900
00015000

```



```

000001      SUBROUTINE      ERRMSG(IRTN)                                00000100
C                                                    00000200
C*****                                                    00000300
C*                                                    *00000400
C*      FUNCTION ;                                                    *00000500
C*                                                    *00000600
C*      OUTPUT OF ERROR MESSEGE.                                       *00000700
C*                                                    *00000800
C*      DATE      ; 84/02/15                                           *00000900
C*                                                    *00001000
C*      ARGUMENT ;                                                    (TYPE,I/O,UNIT) *00001100
C*                                                    *00001200
C*      (1) IRTN  ---  RETURN CODE                                     (I*4,I,-)* 00001300
C*                                                    *00001400
C*      =1 ; INPUT DATA FORMAT ERROR                                 *00001500
C*      =2 ; BEYOND THE CARD'S LIMITS                               *00001600
C*      =3 ; BEYOND THE LIMITS (JXMAX)                             *00001700
C*      =4 ; ALLAY ELEMENT NUMBER ERROR (RX)                       *00001800
C*                                                    *00001900
C*****                                                    00002000
C                                                    00002100
C*****      PROCEDURE      ***** .                                00002200
C                                                    00002300
C                                                    00002400
C*** INPUT DATA FORMAT ERROR ***                                    00002500
C                                                    00002600
000002      IF( IRTN .EQ. 1 ) THEN                                       00002700
000003      WRITE(6,6000)                                                 00002800
C                                                    00002900
C*** BEYOND THE CARD'S LIMITS ***                                    00003000
C                                                    00003100
000004      ELSEIF( IRTN .EQ. 2 ) THEN                                    00003200
000005      WRITE(6,6100)                                                 00003300
C                                                    00003400
C*** BEYOND THE LIMITS (JXMAX) ***                                    00003500
C                                                    00003600
000006      ELSEIF( IRTN .EQ. 3 ) THEN                                    00003700
000007      WRITE(6,6200)                                                 00003800
C                                                    00003900
C*** ALLAY ELEMENT NUMBER ERROR (RX) ***                               00004000
C                                                    00004100
000008      ELSEIF( IRTN .EQ. 4 ) THEN                                    00004200
000009      WRITE(6,6300)                                                 00004300
C                                                    00004400
000010      ENDIF                                                         00004500
C                                                    00004600
000011      RETURN                                                         00004700
C                                                    00004800
C                                                    00004900
000012      6000 FORMAT(1H ,20X,'CODE-1 INPUT DATA FORMAT ERROR')     00005000
000013      6100 FORMAT(1H ,20X,'CODE-2 BEYOND THE CARD'S LIMITS')     00005100
000014      6200 FORMAT(1H ,20X,'CODE-3 BEYOND THE LIMITS (JXMAX)')     00005200
000015      6300 FORMAT(1H ,20X,'CODE-4 ALLAY ELEMENT NUMBER ERROR (RX)') 00005300
C                                                    00005400
000016      END                                                           00005500

```

```

000001      SUBROUTINE      INISET(IRTN)                                00000100
C                                                    00000200
C*****                                                    00000300
C                                                    *00000400
C      FUNCTION ;                                                    *00000500
C                                                    *00000600
C      SET-UP OF INITIAL VALUE IN THE USE OF CONVERGE COMPUTE.      *00000700
C                                                    *00000800
C      DATE      ; 84/02/15                                          *00000900
C                                                    *00001000
C      ARGUMENT ;                                                    (TYPE,I/O,UNIT) *00001100
C                                                    *00001200
C      (1) IRTN --- RETURN CODE                                     (I*4,0,-) *00001300
C                                                    =0 ; NORMAL END *00001400
C                                                    =1 ; ABNORMAL END *00001500
C                                                    *00001600
C*****                                                    *00001700
C                                                    00001800
C***** DECLARATION *****                                       00001900
C                                                    00002000
000002      IMPLICIT      REAL*8      (A-H,O-Z)                    00002100
*INCLUDE PARAM, FIXED, INSOURCE                                00002200
000003      PARAMETER      ( LMAXX=100 ,KMAXX = 100      )          00000010
000004      PARAMETER      ( IMAXX=100 ,IMAXX1 = IMAXX+1 ,IMAXX2 = IMAXX+2 ) 00000020
000005      PARAMETER      ( JMAXX=100 ,JMAXX1 = JMAXX+1 ,JMAXX2 = JMAXX+2 ) 00000030
*INCLUDE CARD, FIXED, INSOURCE                                00002300
000006      COMMON /CARD1/ IOPT01                                  00000011
000007      COMMON /CARD2/ KMAX      ,IMAX      ,RMAX                                  00000012
000008      COMMON /CARD3/ JMAX(LMAXX) ,R(LMAXX) ,RA(LMAXX) ,DA(LMAXX) ,
. AM(LMAXX)                                                    00000030
000009      COMMON /CARD4/ ESP1      ,ESP2      ,ITMAX                                  00000040
*INCLUDE COMGF, FIXED, INSOURCE                                00002400
000010      COMMON /GMTRY/ IXMAX, JXMAX, DR(LMAXX), DPHI,
. LPOINT(LMAXX), RX(JMAXX), PHIX(IMAXX)                        00000020
000011      COMMON /FLUID/ THTN(O:IMAXX1,O:JMAXX1),PSIN(O:IMAXX1,O:JMAXX1) 00000030
. THTD(O:IMAXX1,O:JMAXX1),PSID(O:IMAXX1,O:JMAXX1) 00000040
. PSINRM(O:IMAXX1,O:JMAXX1)                                    00000050
000012      DATA PAI / 3.141592654 /                                00002500
C                                                    00002600
C***** PROCEDURE *****                                       00002700
C                                                    00002800
000013      IRTN      = 0                                          00002900
C                                                    00003000
C*** JXMAX CALCULATION ***                                       00003100
C                                                    00003200
000014      JXMAX      = 0                                          00003300
000015      DO 4050 J = 1,KMAX                                       00003400
000016      JXMAX = JXMAX + JMAX(J)                                    00003500
000017      4050 CONTINUE                                           00003600
C                                                    00003700
000018      IF( JXMAX .GT. JMAXX ) THEN                                00003800
000019      IRTN = 3                                                  00003900
000020      CALL      ERRMSG( IRTN )                                    00004000
000021                                                    GO TO 8000 00004100
000022      ENDIF                                                    00004200
C                                                    00004300
C*** IXMAX SET ***                                               00004400
C                                                    00004500
000023      IXMAX      = IMAX                                         00004600
C                                                    00004700
C*** DDR CALCULATION ***                                         00004800
C                                                    00004900
000024      IF( KMAX .NE.1 ) THEN                                       00005000
000025      DO 4100 I = 1,KMAX-1                                       00005100
000026      DR(I) = (R(I+1) - R(I))/JMAX(I)                            00005200
000027      4100 CONTINUE                                           00005300
000028      ENDIF                                                    00005400
C                                                    00005500
000029      DR(KMAX) = (RMAX - R(KMAX))/JMAX(KMAX)                   00005600
C                                                    00005700
C*** DPHIX CALCULATION ***                                         00005800
C                                                    00005900
000030      DPHI      = PAI/IXMAX                                       00006000
C                                                    00006100
C*** LPOINT CALCULATION ***                                         00006200
C                                                    00006300
000031      LPOINT(1) = 1                                             00006400
000032      IF( KMAX .NE. 1 ) THEN                                       00006500
000033      DO 4300 I = 1,KMAX-1                                       00006600
000034      LPOINT(I+1) = LPOINT(I) + JMAX(I)                         00006700
000035      4300 CONTINUE                                           00006800
000036      ENDIF                                                    00006900

```

```

C
C*** RX CALCULATION ***
C
000037      J1 = 0
000038      DO 4450 JK = 1,KMAX
000039          RR = R(JK) + 0.5*DR(JK)
000040          JINI = LPOINT(JK)
000041          IF( JK.LT.KMAX ) JFIN = LPOINT(JK+1) - 1
000042          IF( JK.EQ.KMAX ) JFIN = JMAXX
000043          DO 4440 J = JINI,JFIN
000044              RX(J) = RR
000045              RR = RR + DR(JK)
000046      4440      CONTINUE
000047      4450 CONTINUE
C
C*** PHIX CALCULATION ***
C
000048      PPHI = 0.5*DPHI
000049      DO 4500 I = 1,IXMAX
000050          PHIX(I) = PPHI
000051          PPHI = PPHI + DPHI
000052      4500 CONTINUE
C
C*** GEOMERT DATA LPOUT ***
C
000053      CALL      GLPOUT
C
C*** THETA INITIALIZATION ***
C
000054      DO 4550 J = 0,JXMAX+1
000055          IF( J.EQ.0 ) THEN
000056              THITA = 1.0
000057          ELSEIF( J.EQ.JXMAX+1 ) THEN
000058              THITA = 0.0
000059          ELSE
000060              THITA = (RMAX - RX(J))/(RMAX - 1.0)
000061          ENDIF
000062          DO 4540 I = 0,IXMAX+1
000063              THTN(I,J) = THITA
000064      4540      CONTINUE
000065      4550 CONTINUE
C
C*** PSI INITIALIZATION ***
C
000066      DO 4700 J = 0,JXMAX+1
000067          DO 4710 I = 0,IXMAX+1
000068              PSIN(I,J) = 0.0
000069      4710      CONTINUE
000070      4700 CONTINUE
C
000071      8000 CONTINUE
C
C
000072      RETURN
000073      END
00007000
00007100
00007200
00007300
00007400
00007500
00007600
00007700
00007800
00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00009400
00009500
00009600
00009700
00009800
00009900
00010000
00010100
00010200
00010300
00010400
00010500
00010600
00010700
00010800
00010900
00011000
00011100
00011200
00011300
00011400
00011500
00011600
00011700
00011800
00011900
00012000
00012100
00012200
00012300
00012400

```



```

000001      SUBROUTINE      GLPOUT      00000100
C
000002      IMPLICIT      REAL*8      (A-H,O-Z)      00000200
*INCLUDE PARAM, FIXED, INSOURCE      00000300
000003      PARAMETER      ( LMAXX=100 ,KMAXX = 100      )      00000010
000004      PARAMETER      ( IMAXX=100 ,IMAXX1 = IMAXX+1 ,IMAXX2 = IMAXX+2 )      00000020
000005      PARAMETER      ( JMAXX=100 ,JMAXX1 = JMAXX+1 ,JMAXX2 = JMAXX+2 )      00000030
*INCLUDE CARD, FIXED, INSOURCE      00000500
000006      COMMON /CARD1/      IOPT01      00000011
000007      COMMON /CARD2/      KMAX      ,IMAX      ,RMAX      00000012
000008      COMMON /CARD3/      JMAX(LMAXX) ,R(LMAXX) ,RA(LMAXX) ,DA(LMAXX),      00000020
AM(LMAXX)      00000030
000009      COMMON /CARD4/      ESP1      ,ESP2      ,ITMAX      00000040
*INCLUDE COMGF, FIXED, INSOURCE      00000600
000010      COMMON /GMTRY/      IXMAX, JXMAX, DR(LMAXX), DPHI,      00000010
LPOINT(LMAXX), RX(JMAXX), PHIX(IMAXX)      00000020
000011      COMMON /FLUID/      THTN(O:IMAXX1,O:JMAXX1),PSIN(O:IMAXX1,O:JMAXX1)      00000030
, THTO(O:IMAXX1,O:JMAXX1),PSIO(O:IMAXX1,O:JMAXX1)      00000040
, PSINRM(O:IMAXX1,O:JMAXX1)      00000050
000012      DATA FFFF /57.29578/      00000700
C
C*****      GEOMETRY DATA WRITING.      *****      00000800
C
000013      WRITE(6,6010)      00001100
000014      WRITE(6,6020) KMAX, IXMAX, JXMAX      00001200
000015      WRITE(6,6030) DPHI*FFFF      00001300
000016      WRITE(6,6040) (K,DR(K),K=1,KMAX)      00001400
000017      WRITE(6,6050)      00001500
000018      WRITE(6,6060)      00001600
C
C
000019      LL1 = JXMAX/7      00001700
000020      LL2 = MOD(JXMAX,7)      00001800
000021      DO 4150 I = 1,LL1      00001900
000022      I1 = (I - 1)*7      00002000
000023      L1 = I1 + 1      00002100
000024      L2 = I1 + 7      00002200
000025      WRITE(6,6110) (II,II=L1,L2)      00002300
000026      WRITE(6,6120) (RX(II),II=L1,L2)      00002400
000027      4150 CONTINUE      00002500
000028      IF( LL2.NE.0 ) THEN      00002600
000029      I1 = LL1*7      00002700
000030      L1 = I1 + 1      00002800
000031      L2 = I1 + LL2      00002900
000032      WRITE(6,6110) (II,II=L1,L2)      00003000
000033      WRITE(6,6120) (RX(II),II=L1,L2)      00003100
000034      ENDIF      00003200
C
C
000035      WRITE(6,6080)      00003300
000036      LL1 = IXMAX/7      00003400
000037      LL2 = MOD(IXMAX,7)      00003500
000038      DO 4250 I = 1,LL1      00003600
000039      I1 = (I - 1)*7      00003700
000040      L1 = I1 + 1      00003800
000041      L2 = I1 + 7      00003900
000042      WRITE(6,6110) (II,II=L1,L2)      00004000
000043      WRITE(6,6120) (PHIX(II)*FFFF,II=L1,L2)      00004100
000044      4250 CONTINUE      00004200
000045      IF( LL2.NE.0 ) THEN      00004300
000046      I1 = LL1*7      00004400
000047      L1 = I1 + 1      00004500
000048      L2 = I1 + LL2      00004600
000049      WRITE(6,6110) (II,II=L1,L2)      00004700
000050      WRITE(6,6120) (PHIX(II)*FFFF,II=L1,L2)      00004800
000051      ENDIF      00004900
000052      WRITE(6,6150)      00005000
C
C
C
000053      RETURN      00005100
C
000054      6010 FORMAT(1H1,///,36X,      00005200
,      00005300
'*****      GEOMETRY DATA SUMMARY LIST      *****'///)      00005400
000055      6020 FORMAT(///,15X,'KMAX = ',I5,/,      00005500
,      00005600
15X,'IXMAX = ',I5,/,      00005700
,      00005800
15X,'JXMAX = ',I5,/,      00005900
,      00006000
25X,'KMAX : ADIABATIC LAYER COUNT',/,      00006100
,      00006200
25X,'IXMAX : TOTAL CIRCUMFERENCE DEVISION COUNT',/,      00006300
,      00006400
25X,'JXMAX : TOTAL RADIAL DEVISION COUNT')      00006500
000056      6030 FORMAT(///,15X,'DPHI = ',F10.4)      00006600
000057      6040 FORMAT( 15X,'DR(',I2,') = ', F10.4)      00006700

```

000058	6050	FORMAT(//	00007000
	.	25X,'DPHI : MESH INTERVAL FOR PHI (DEG) ',//	00007100
	.	25X,'DR(K) : MESH INTERVAL OF K-TH LAYER FOR R')	00007200
		C	00007300
		C	00007400
000059	6060	FORMAT(1H1,////,	00007500
	.	15X,'<< MESH CENTER COORDINATE >>',//	00007600
	.	17X,'< R - COORDINATE >',//)	00007700
000060	6080	FORMAT(//, 17X,'< PHI - COORDINATE (DEG) >',//)	00007800
000061	6110	FORMAT(17X,7(:3X,'<',14,'>',3X))	00007900
000062	6120	FORMAT(17X,7(1X,F10.4,1X),//)	00008000
000063	6150	FORMAT(1H1)	00008100
		C	00008200
		C	00008300
000064		END	00008400

```

000001      SUBROUTINE      MOTION(IRTNO)                                00000100
C                                                    00000200
C*****                                                    00000300
C*                                                    *00000400
C*      FUNCTION ;                                                    *00000500
C*                                                    *00000600
C*      THIS SUBROUTINE IS THAT SOLVE THE EQUATION OF MOTION AND      *00000700
C*                                                    *00000800
C*      GET THE STREAM-FUNCTION.                                        *00000900
C*                                                    *00001000
C*      DATE      ; 84/02/15                                          *00001100
C*                                                    *00001200
C*      ARGUMENT ;                                                    (TYPE,I/O,UNIT) *00001300
C*                                                    *00001400
C*      (1) IRTN  --- RETURN CODE                                     (I*4,O,-) *00001500
C*                                                    =0 ; NORMAL END *00001600
C*                                                    =1 ; ABNORMAL END *00001700
C*                                                    *00001800
C*****                                                    00001900
C                                                    00002000
C*****      DECLARATION      *****                                00002100
C                                                    00002200
000002      IMPLICIT      REAL*8      (A-H,O-Z)                        00002300
*INCLUDE PARAM,FIXED,INSOURCE                                00002400
000003      PARAMETER      ( LMAXX=100 ,KMAXX = 100 )                00000010
000004      PARAMETER      ( IMAXX=100 ,IMAXX1 = IMAXX+1 ,IMAXX2 = IMAXX+2 ) 00000020
000005      PARAMETER      ( JMAXX=100 ,JMAXX1 = JMAXX+1 ,JMAXX2 = JMAXX+2 ) 00000030
*INCLUDE CARD,FIXED,INSOURCE                                00002500
000006      COMMON      /CARD1/      IDPT01                            00000011
000007      COMMON      /CARD2/      KMAX      ,IMAX      ,RMAX        00000012
000008      COMMON      /CARD3/      JMAX(LMAXX) ,R(LMAXX) ,RA(LMAXX) ,DA(LMAXX), 00000020
AM(LMAXX)                                00000030
000009      COMMON      /CARD4/      ESP1      ,ESP2      ,ITMAX      00000040
*INCLUDE COMGF,FIXED,INSOURCE                                00002600
000010      COMMON      /GMTRY/      IXMAX, JXMAX, DR(LMAXX), DPHI,    00000010
LPOINT(LMAXX), RX(JMAXX), PHIX(IMAXX)    00000020
000011      COMMON      /FLUID/      THTN(O:IMAXX1,O:JMAXX1),PSIN(O:IMAXX1,O:JMAXX1) 00000030
,THTO(O:IMAXX1,O:JMAXX1),PSIO(O:IMAXX1,O:JMAXX1) 00000040
,PSINRM(O:IMAXX1,O:JMAXX1)                00000050
C                                                    00002700
000012      DIMENSION A(IMAXX), B(IMAXX), C(IMAXX), S(IMAXX)        00002800
C                                                    00002900
000013      IRTNO = 0                                                00003000
000014      K      = 0                                                00003100
000015      DPHI2 = DPHI**2                                          00003200
C                                                    00003300
000016      DO 4800 J = 1,JXMAX                                       00003400
C                                                    00003500
000017      ISET      = 0                                             00003600
000018      IF( K.LT.KMAX .AND. J.EQ.LPOINT(K+1) ) THEN            00003700
000019      K      = K + 1                                             00003800
000020      K1      = LPOINT(K)                                        00003900
000021      IF( K.LT.KMAX ) K2 = LPOINT(K+1) - 1                    00004000
000022      IF( K.EQ.KMAX ) K2 = IXMAX                                00004100
000023      RADAK      = RA(K)*DA(K)                                    00004200
000024      AMK      = AM(K)                                           00004300
000025      ENDFIF                                                    00004400
C                                                    00004500
000026      IF( J.EQ.K1 ) THEN                                         00004600
000027      IF( K.EQ.1 ) DRO1 = 0.0                                    00004700
000028      IF( K.GT.1 ) DRO1 = DR(K-1)                                00004800
000029      DRO2      = DR(K)                                           00004900
000030      DRO3      = DR(K)                                           00005000
000031      ISET      = 1                                             00005100
000032      ENDFIF                                                    00005200
C                                                    00005300
000033      IF( J.EQ.K1+1 ) THEN                                       00005400
000034      DRO1      = DR(K)                                           00005500
000035      ISET      = 1                                             00005600
000036      ENDFIF                                                    00005700
000037      IF( J.EQ.K2 ) THEN                                         00005800
000038      IF( K.LT.KMAX ) DRO3 = DR(K+1)                            00005900
000039      IF( K.EQ.KMAX ) DRO3 = 0.0                                00006000
000040      ISET      = 1                                             00006100
000041      ENDFIF                                                    00006200
C                                                    00006300
000042      IF( ISET.NE.0 ) THEN                                       00006400
000043      DR1      = DRO3 + DRO2                                       00006500
000044      DR2      = DRO2                                               00006600
000045      DR3      = DRO2 + DRO1                                       00006700
000046      DR4      = DRO3 + 2.0*DRO2 + DRO1                           00006800
000047      DR5      = DR1*DR2                                           00006900

```

```

000048      DR6  = DR2*DR3
000049      DR7  = DR1*DR2*DR3
000050      ENDIF
C
C
000051      A1    = 1.0/(RX(J)**2*DPHI2)
000052      B1    = - 2.0*DR4/DR7 - 2.0*A1
000053      C1    = A1
000054      TEMP  = 2.0/(RX(J)*DR4)
000055      D1    = 2.0/DR6 - TEMP
000056      E1    = 2.0/DR5 + TEMP
C
C
000057      DO 4500 I = 1, IXMAX
C
C      DTDR  = 2.0*(THTN(I,J+1) - THTN(I,J-1))/DR4
000058      DPP   = 2.0*DPHI
000059      IF( I.EQ.1 .OR. I.EQ.IXMAX ) DPP = 1.5*DPHI
000060      DTDP  = (THTN(I+1,J) - THTN(I-1,J))/DPP
C
000062      A(I)  = A1
000063      B(I)  = B1
000064      C(I)  = C1
000065      D     = D1
000066      E     = E1
000067      F     = - RADAK*AMK*( DTDR*SIN(PHIX(I))
*          + DTDP*COS(PHIX(I))/RX(J) )
000068      S(I)  = F - D*PSIN(I,J-1) - E*PSIO(I,J+1)
C
000069      IF( I.EQ.1 ) THEN
000070          A(I) = 2.0*A(I)
000071          B(I) = B(I) - A1
000072          S(I) = S(I) - A(I)*PSIN(I-1,J)
000073      ENDIF
000074      IF( I.EQ.IXMAX ) THEN
000075          B(I) = B(I) - A1
000076          C(I) = 2.0*C(I)
000077          S(I) = S(I) - C(I)*PSIN(I+1,J)
000078      ENDIF
C
000079      4500 CONTINUE
C
C
000080      CALL MATRIX( IXMAX, A, B, C, S, PSIN(1,J), IRTNO )
000081      4800 CONTINUE
C
C
C
000082      RETURN
000083      END
00007000
00007100
00007200
00007300
00007400
00007500
00007600
00007700
00007800
00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00009400
00009500
00009600
00009700
00009800
00009900
00010000
00010100
00010200
00010300
00010400
00010500
00010600
00010700
00010800
00010900
00011000
00011100
00011200
00011300
00011400
00011500
00011600
00011700
00011800
00011900

```

```

000001      SUBROUTINE      MATRIX(IDIM,A,B,C,S,X,IRTN)      00000100
C                                                    00000200
C*****00000300
C*                                                    *00000400
C*      FUNCTION ;      *00000500
C*                                                    *00000600
C*      THIS SUBROUTINE IS THAT COMPUTE OF TRIPLE DIAGONAL. *00000700
C*                                                    *00000800
C*      DATE      ; 84/02/15      *00000900
C*                                                    *00001000
C*      ARGUMENTS ;      (TYPE,I/O,UNIT) *00001100
C*                                                    *00001200
C*      (1) IDIM --- DEGREE OF MATRIX      (I*4,I,-) *00001300
C*                                                    *00001400
C*      (2) A --- 1ST ELEMENT      (R*8,I,-) *00001500
C*      (IDIM) *00001600
C*                                                    *00001700
C*      (3) B --- 2ND ELEMENT      (R*8,I,-) *00001800
C*      (IDIM) *00001900
C*                                                    *00002000
C*      (4) C --- 3RD ELEMENT      (R*8,I,-) *00002100
C*      (IDIM) *00002200
C*                                                    *00002300
C*      (5) S --- CONSTANT MATRIX      (R*8,I,-) *00002400
C*      (IDIM) *00002500
C*                                                    *00002600
C*      (6) X --- UNKNOWN MATRIX      (R*8,O,-) *00002700
C*      (IDIM) *00002800
C*                                                    *00002900
C*      (7) IRTN --- RETURN CODE      (I*4,O,-) *00003000
C*                                                    *00003100
C*                                                    =0 ; NORMAL END
C*                                                    *00003200
C*                                                    =1 ; UN-COMPUTE
C*                                                    *00003300
C*****00003400
C                                                    00003500
C*****      DECLARATION      *****      00003600
C                                                    00003700
000002      IMPLICIT      REAL*8      (A-H,O-Z)      00003800
C                                                    00003900
000003      REAL*8      A(IDIM),B(IDIM),C(IDIM),S(IDIM),X(IDIM)      00004000
C                                                    00004100
C*****      PROCEDURE      *****      00004200
C                                                    00004300
000004      IRTN = 0      00004400
000005      IF( IDIM .LT. 2 )      GO TO 8000      00004500
000006      K = IDIM + 1      00004600
000007      DO 4000 I = 1, IDIM      00004700
000008      K = K - 1      00004800
000009      IF( B(1) .EQ. 0.0 )      GO TO 8000      00004900
000010      P = B(1)      00005000
000011      Q = -C(1)/P      00005100
000012      U = S(1)/P      00005200
000013      IF( K .LT. 2 )      GO TO 3000      00005300
C                                                    00005400
000014      DO 4100 KK = 2,K      00005500
000015      P = A(KK)*Q + B(KK)      00005600
000016      IF( P .EQ. 0.0 )      GO TO 8000      00005700
000017      Q = -C(KK)/P      00005800
000018      U = (S(KK) - A(KK)*U)/P      00005900
000019      4100 CONTINUE      00006000
C                                                    00006100
000020      IF( K .LT. IDIM )      GO TO 3000      00006200
000021      X(IDIM) = U      00006300
000022      GO TO 4000      00006400
000023      3000 CONTINUE      00006500
000024      X(K) = Q*X(K+1) + U      00006600
000025      4000 CONTINUE      00006700
000026      GO TO 9999      00006800
000027      8000 CONTINUE      00006900
000028      ITRN = 1      00007000
000029      9999 CONTINUE      00007100
000030      RETURN      00007200
000031      END      00007300

```

```

000001      SUBROUTINE      ENERGY(IRTNO)                                00000100
C                                                    00000200
C*****00000300
C*                                                    *00000400
C*      FUNCTION ;                                                    *00000500
C*                                                    *00000600
C*      THIS SUBROUTINE IS THAT SOLVE THE EQUATION OF ENERGY AND      *00000700
C*      GET THE TEMPERATURE FIELD.                                       *00000800
C*                                                    *00000900
C*      DATE      ; 84/02/15                                             *00001000
C*                                                    *00001100
C*      ARGUMENT ;                                                    (TYPE,I/O,UNIT) *00001300
C*                                                    *00001400
C*      (1) IRTN --- RETURN CODE                                       (I*4,O,-) *00001500
C*                                                    *00001600
C*                                                    =0 ; NORMAL END
C*                                                    *00001700
C*                                                    =1 ; ABNORMAL END
C*                                                    *00001800
C*****00001900
C                                                    00002000
C*****      DECLARATION      ***** 00002100
C                                                    00002200
000002      IMPLICIT      REAL*8      (A-H,O-Z) 00002300
*INCLUDE PARAM, FIXED, INSOURCE 00002400
000003      PARAMETER      ( LMAXX=100 ,KMAXX = 100      ) 00000010
000004      PARAMETER      ( IMAXX=100 ,IMAXX1 = IMAXX+1 ,IMAXX2 = IMAXX+2 ) 00000020
000005      PARAMETER      ( JMAXX=100 ,JMAXX1 = JMAXX+1 ,JMAXX2 = JMAXX+2 ) 00000030
*INCLUDE CARD, FIXED, INSOURCE 00002500
000006      COMMON /CARD1/ IOPT01 00000011
000007      COMMON /CARD2/ KMAX ,IMAX ,RMAX 00000012
000008      COMMON /CARD3/ JMAX(LMAXX) ,R(LMAXX) ,RA(LMAXX) ,DA(LMAXX), 00000020
AM(LMAXX) 00000030
000009      COMMON /CARD4/ ESP1 ,ESP2 ,ITMAX 00000040
*INCLUDE COMGF, FIXED, INSOURCE 00002600
000010      COMMON /GMTRY/ IXMAX, JXMAX, DR(LMAXX), DPHI, 00000010
LPOINT(LMAXX), RX(JMAXX), PHIX(IMAXX) 00000020
000011      COMMON /FLUID/ THTN(O:IMAXX1,O:JMAXX1),PSIN(O:IMAXX1,O:JMAXX1) 00000030
, THTO(O:IMAXX1,O:JMAXX1),PSIO(O:IMAXX1,O:JMAXX1) 00000040
, PSINRM(O:IMAXX1,O:JMAXX1) 00000050
C                                                    00002700
000012      DIMENSION A(IMAXX), B(IMAXX), C(IMAXX), S(IMAXX) 00002800
C                                                    00002900
000013      IRTNO = 0 00003000
000014      K = 0 00003100
000015      DPHI2 = DPHI**2 00003200
C                                                    00003300
000016      DO 4800 J = 1,JXMAX 00003400
C                                                    00003500
000017      ISET = 0 00003600
000018      IF( K.LT.KMAX .AND. J.EQ.LPOINT(K+1) ) THEN 00003700
000019      K = K + 1 00003800
000020      K1 = LPOINT(K) 00003900
000021      IF( K.LT.KMAX ) K2 = LPOINT(K+1) - 1 00004000
000022      IF( K.EQ.KMAX ) K2 = IXMAX 00004100
000023      RADAK = RA(K)*DA(K) 00004200
000024      AMK = AM(K) 00004300
000025      ENDIF 00004400
C                                                    00004500
000026      IF( J.EQ.K1 ) THEN 00004600
000027      IF( K.EQ.1 ) DRO1 = 0.0 00004700
000028      IF( K.GT.1 ) DRO1 = DR(K-1) 00004800
000029      DRO2 = DR(K) 00004900
000030      DRO3 = DR(K) 00005000
000031      ISET = 1 00005100
000032      ENDIF 00005200
C                                                    00005300
000033      IF( J.EQ.K1+1 ) THEN 00005400
000034      DRO1 = DR(K) 00005500
000035      ISET = 1 00005600
000036      ENDIF 00005700
000037      IF( J.EQ.K2 ) THEN 00005800
000038      IF( K.LT.KMAX ) DRO3 = DR(K+1) 00005900
000039      IF( K.EQ.KMAX ) DRO3 = 0.0 00006000
000040      ISET = 1 00006100
000041      ENDIF 00006200
C                                                    00006300
000042      IF( ISET.NE.0 ) THEN 00006400
000043      DR1 = DRO3 + DRO2 00006500
000044      DR2 = DRO2 00006600
000045      DR3 = DRO2 + DRO1 00006700
000046      DR4 = DRO3 + 2.0*DRO2 + DRO1 00006800
000047      DR5 = DR1*DR2 00006900

```

```

000048      DR6  = DR2*DR3                      00007000
000049      DR7  = DR1*DR2*DR3                 00007100
000050      ENDIF                                00007200
C                                                    00007300

C
C
000051      A1   = AMK/(RX(J)*DPHI2)             00007400
000052      B1   = -AMK*RX(J)*(2.0*DR4/DR7 + 2.0/(RX(J)**2*DPHI2)) 00007500
000053      C1   = A1                           00007600
000054      TEMP = 2.0/(RX(J)*DR4)              00007700
000055      D1   = AMK*RX(J)*(2.0/DR6 - TEMP)    00007800
000056      E1   = AMK*RX(J)*(2.0/DR5 + TEMP)  00007900
C                                                    00008000
C                                                    00008100
C                                                    00008200
000057      DO 4500 I = 1,IXMAX                 00008300
C                                                    00008400
C                                                    00008500
000058          DPDR = 2.0*(PSIN(I,J+1) - PSIN(I,J-1))/DR4 00008600
000059          DPP  = 2.0*DPHI                  00008700
000060          IF( I.EQ.1 .OR. I.EQ.IXMAX ) DPP = 1.5*DPHI 00008800
000061          DPDP = (PSIN(I+1,J) - PSIN(I-1,J))/ DPP    00008900
C                                                    00009000
C                                                    00009100
000062          A(I) = A1                        00009200
000063          B(I) = B1                        00009300
000064          C(I) = C1                       00009400
000065          D   = D1                        00009500
000066          E   = E1                       00009600
C                                                    00009700
C                                                    00009800
000067          IF( I.EQ.1 ) THEN               00009900
000068              A(I) = 2.0*A(I)             00010000
000069              B(I) = B(I) - A1           00010100
000070          ENDIF                            00010200
000071          IF( I.EQ.IXMAX ) THEN           00010300
000072              B(I) = B(I) - A1           00010400
000073              C(I) = 2.0*C(I)           00010500
000074          ENDIF                            00010600
C                                                    00010700
C                                                    00010800
000075          DPP  = DPHI                     00010900
000076          IF( DPDP.GE.0.0 .AND. DPDR.LE.0.0 ) THEN 00011000
000077              IF( I.EQ.1 ) DPP = 0.5*DPHI 00011100
000078              A(I) = A(I) - DPDR/DPP      00011200
000079              B(I) = B(I) - DPDP*2.0/DR3 + DPDR/DPP 00011300
000080              D   = D + DPDP*2.0/DR3      00011400
000081          ELSE IF ( DPDP.LT.0.0 .AND. DPDR.LE.0.0 ) THEN 00011500
000082              IF( I.EQ.1 ) DPP = 0.5*DPHI 00011600
000083              A(I) = A(I) - DPDR/DPP      00011700
000084              B(I) = B(I) + DPDP*2.0/DR1 + DPDR/DPP 00011800
000085              E   = E - DPDP*2.0/DR1      00011900
000086          ELSE IF ( DPDP.GE.0.0 .AND. DPDR.GT.0.0 ) THEN 00012000
000087              IF( I.EQ.IXMAX ) DPP = 0.5*DPHI 00012100
000088              B(I) = B(I) - DPDP*2.0/DR3 - DPDR/DPP 00012200
000089              C(I) = C(I) + DPDR/DPP      00012300
000090              D   = D + DPDP*2.0/DR3      00012400
000091          ELSE                              00012500
000092              IF( I.EQ.IXMAX ) DPP = 0.5*DPHI 00012600
000093              B(I) = B(I) + DPDP*2.0/DR1 - DPDR/DPP 00012700
000094              C(I) = C(I) + DPDR/DPP      00012800
000095              E   = E - DPDP*2.0/DR1      00012900
000096          ENDIF                            00013000
C                                                    00013100
C                                                    00013200
000097          IF( I.EQ.1 ) B(I) = B(I) + A(I) 00013300
000098          IF( I.EQ.IXMAX ) B(I) = B(I) + C(I) 00013400
000099          S(I) = - D*THTN(I,J-1) - E*THTN(I,J+1) 00013500
C                                                    00013600
C                                                    00013700
000100      4500 CONTINUE                       00013800
C                                                    00013900
C                                                    00014000
C                                                    00014100
000101      CALL MATRIX( IXMAX, A, B, C, S, THTN(1,J), IRTNO ) 00014200
C                                                    00014300
C                                                    00014400
000102          THTN(0,J) = THTN(1,J)
000103          THTN(IXMAX+1,J) = THTN(IXMAX,J)
C
000104      4800 CONTINUE
C
C
000105      RETURN
000106      END

```

```

000001      SUBROUTINE  VELCTY                                00000100
C                                                    00000200
C*****00000300
C*                                                    *00000400
C*      FUNCTION ;                                         *00000500
C*                                                    *00000600
C*      THIS SUBROUTINE GETS THE VELOCITY OF RADIAL(U) AND *00000700
C*                                                    *00000800
C*      CIRCUMFERENCE(V) DIRECTION.                    *00000900
C*                                                    *00001000
C*                                                    *00001100
C*****00001200
C                                                    00001300
C*****  DECLARATION  ***** 00001400
C                                                    00001500
000002      IMPLICIT    REAL*8    (A-H,O-Z)              00001600
*INCLUDE PARAM,  FIXED,  INSOURCE 00001700
000003      PARAMETER  ( LMAXX=100 ,KMAXX  = 100      ) 00000010
000004      PARAMETER  ( IMAXX=100 ,IMAXX1 = IMAXX+1 ,IMAXX2 = IMAXX+2 ) 00000020
000005      PARAMETER  ( JMAXX=100 ,JMAXX1 = JMAXX+1 ,JMAXX2 = JMAXX+2 ) 00000030
*INCLUDE CARD,   FIXED,  INSOURCE 00001800
000006      COMMON  /CARD1/  IOPT01,                    00000011
000007      COMMON  /CARD2/  KMAX      ,IMAX      ,RMAX    00000012
000008      COMMON  /CARD3/  JMAX(LMAXX) ,R(LMAXX) ,RA(LMAXX) ,DA(LMAXX), 00000020
.          AM(LMAXX) 00000030
000009      COMMON  /CARD4/  ESP1      ,ESP2      ,ITMAX    00000040
*INCLUDE COMGF,  FIXED,  INSOURCE 00001900
000010      COMMON  /GMTRY/  IXMAX, JXMAX, DR(LMAXX), DPHI, 00000010
.          LPOINT(LMAXX), RX(JMAXX), PHIX(IMAXX) 00000020
000011      COMMON  /FLUID/  THTN(O:IMAXX1,O:JMAXX1),PSIN(O:IMAXX1,O:JMAXX1) 00000030
.          ,THTO(O:IMAXX1,O:JMAXX1),PSIO(O:IMAXX1,O:JMAXX1) 00000040
.          ,PSINRM(O:IMAXX1,O:JMAXX1) 00000050
*INCLUDE COMUV,  FIXED,  INSOURCE 00002000
000012      COMMON  /VECTR/  U(IMAXX,JMAXX),V(IMAXX,JMAXX) 00000010
C                                                    00002100
C                                                    00002200
000013      K      = 0 00002300
C                                                    00002400
000014      DO 4800 J = 1,JXMAX 00002500
C                                                    00002600
000015      ISET   = 0 00002700
000016      IF( K.LT.KMAX .AND. J.EQ.LPOINT(K+1) ) THEN 00002800
000017      K      = K + 1 00002900
000018      K1     = LPOINT(K) 00003000
000019      IF( K.LT.KMAX ) K2 = LPOINT(K+1) - 1 00003100
000020      IF( K.EQ.KMAX ) K2 = IXMAX 00003200
000021      ENDIF 00003300
C                                                    00003400
000022      IF( J.EQ.K1 ) THEN 00003500
000023      IF( K.EQ.1 ) DRO1 = 0.0 00003600
000024      IF( K.GT.1 ) DRO1 = DR(K-1) 00003700
000025      DRO2 = DR(K) 00003800
000026      DRO3 = DR(K) 00003900
000027      ISET = 1 00004000
000028      ENDIF 00004100
C                                                    00004200
000029      IF( J.EQ.K1+1 ) THEN 00004300
000030      DRO1 = DR(K) 00004400
000031      ISET = 1 00004500
000032      ENDIF 00004600
000033      IF( J.EQ.K2 ) THEN 00004700
000034      IF( K.LT.KMAX ) DRO3 = DR(K+1) 00004800
000035      IF( K.EQ.KMAX ) DRO3 = 0.0 00004900
000036      ISET = 1 00005000
000037      ENDIF 00005100
C                                                    00005200
000038      IF( ISET.NE.0 ) DRR = 0.5*(DRO3 + 2.0*DRO2 + DRO1) 00005300
C                                                    00005400
C                                                    00005500
000039      DO 4500 I = 1,IXMAX 00005600
000040      DPP = 2.0*DPHI 00005700
000041      IF( I.EQ.1 .OR. I.EQ.IXMAX ) DPP = 1.5*DPHI 00005800
000042      U(I,J) = ( PSIN(I+1,J) - PSIN(I-1,J) )/(RX(J)*DPP) 00005900
000043      V(I,J) = -( PSIN(I,J+1) - PSIN(I,J-1) )/DRR 00006000
000044      4500 CONTINUE 00006100
C                                                    00006200
C                                                    00006300
000045      4800 CONTINUE 00006400
C                                                    00006500
C                                                    00006600
000046      RETURN 00006700
000047      END 00006800

```



```

000001      SUBROUTINE      THRMAL                      00000100
C                                                    00000200
C*****                                                    00000300
C*                                                    *00000400
C*      FUNCTION ;                                       *00000500
C*                                                    *00000600
C*      THIS SUBROUTINE IS THAT CALCULATES             *00000700
C*                THE EFFECTIVE THERMAL CONDUCTIVITY. *00000800
C*                                                    *00000900
C*      DATE      ; 84/03/30                          *00001000
C*                                                    *00001100
C*****                                                    00001200
C                                                    00001300
C*****      DECLARATION      *****                    00001400
C                                                    00001500
000002      IMPLICIT      REAL*8      (A-H,O-Z)         00001600
*INCLUDE PARAM, FIXED, INSOURCE                       00001700
000003      PARAMETER      ( LMAXX=100 ,KMAXX = 100      ) 00000010
000004      PARAMETER      ( IMAXX=100 ,IMAXX1 = IMAXX+1 ,IMAXX2 = IMAXX+2 ) 00000020
000005      PARAMETER      ( JMAXX=100 ,JMAXX1 = JMAXX+1 ,JMAXX2 = JMAXX+2 ) 00000030
*INCLUDE CARD, FIXED, INSOURCE                       00001800
000006      COMMON /CARD1/  IOPT01                    00000011
000007      COMMON /CARD2/  KMAX      ,IMAX      ,RMAX    00000012
000008      COMMON /CARD3/  JMAX(LMAXX) ,R(LMAXX) ,RA(LMAXX) ,DA(LMAXX), 00000020
AM(LMAXX)                                           00000030
000009      COMMON /CARD4/  ESP1      ,ESP2      ,ITMAX   00000040
*INCLUDE COMGF, FIXED, INSOURCE                       00001900
000010      COMMON /GMTRY/  IXMAX, JXMAX, DR(LMAXX), DPHI, 00000010
LPOINT(LMAXX), RX(JMAXX), PHIX(IMAXX)             00000020
000011      COMMON /FLUID/  THTN(O:IMAXX1,O:JMAXX1),PSIN(O:IMAXX1,O:JMAXX1) 00000030
, THTO(O:IMAXX1,O:JMAXX1),PSIO(O:IMAXX1,O:JMAXX1) 00000040
, PSINRM(O:IMAXX1,O:JMAXX1)                       00000050
*INCLUDE COMLM, FIXED, INSOURCE                       00002000
000012      COMMON /LAMDA/  ALAMI(O:IMAXX1), ALAMO(O:IMAXX1), 00000010
*      ALAMIM,      ALAMOM,      00000011
*      RADAAV,      AMAV      00000020
C                                                    00002100
C*****      PROCEDURE      *****                    00002200
C                                                    00002300
000013      PI = 3.141592654                          00002400
C                                                    00002500
000014      SUM1      = 0.0                            00002600
000015      SUM2      = 0.0                            00002700
000016      DO 4010 K = 1,KMAX                          00002800
000017      RR1      = R(K)**2                          00002900
000018      IF( K.NE.KMAX ) RR2 = R(K+1)**2            00003000
000019      IF( K.EQ.KMAX ) RR2 = RMAX**2              00003100
000020      TEMPO     = RR2 - RR1                        00003200
000021      TEMP1    = TEMPO*RA(K)*DA(K)                00003300
000022      TEMP2    = TEMPO*AM(K)                     00003400
000023      SUM1     = SUM1 + TEMP1                     00003500
000024      SUM2     = SUM2 + TEMP2                     00003600
000025      4010 CONTINUE                              00003700
C                                                    00003800
000026      TEMP3    = RMAX**2 - 1.0                   00003900
000027      RADAAV   = SUM1/TEMP3                       00004000
000028      AMAV     = SUM2/TEMP3                       00004100
C                                                    00004200
C                                                    00004300
000029      DLOGR    = DLOG(RMAX)                      00004400
000030      DO 4100 I = 0,IXMAX+1                      00004500
000031      ALAMI(I) = -2.0/(RMAX-1)*DLOGR*(THTN(I,1)-THTN(I,0))/DR(1) 00004600
000032      ALAMO(I) = -2.0/(1-1/RMAX)*DLOGR*          00004700
(THTN(I,JXMAX+1)-THTN(I,JXMAX))/DR(KMAX)          00004800
000033      4100 CONTINUE                              00004900
C                                                    00005000
C                                                    00005100
000034      SUM1     = 0.0                              00005200
000035      SUM2     = 0.0                              00005300
000036      JXMAX2   = JXMAX/2.0                       00005400
C                                                    00005500
000037      DO 4200 I = 1,JXMAX2                       00005600
000038      SUM1     = SUM1 +                           00005700
.      (ALAMI(2*I-2)+4.0*ALAMI(2*I-1)+ALAMI(2*I))*DPHI/3.0 00005800
000039      SUM2     = SUM2 +                           00005900
.      (ALAMO(2*I-2)+4.0*ALAMO(2*I-1)+ALAMO(2*I))*DPHI/3.0 00006000
000040      4200 CONTINUE                              00006100
C                                                    00006200
000041      ALAMIM =SUM1/PI                             00006300
000042      ALAMOM =SUM2/PI                             00006400
C                                                    00006500
C                                                    00006600

```

```

000043 C          RETURN                                00006700
000044          END                                  00006800
                                                00006900

000001          SUBROUTINE OUTPUT                                00000100
C
C
C
000002          IMPLICIT REAL*8 (A-H,O-Z)                    00000500
*INCLUDE PARAM, FIXED, INSOURCE                          00000600
000003          PARAMETER ( LMAXX=100 ,KMAXX = 100 )          00000710
000004          PARAMETER ( IMAXX=100 ,IMAXX1 = IMAXX+1 ,IMAXX2 = IMAXX+2 ) 00000820
000005          PARAMETER ( JMAXX=100 ,JMAXX1 = JMAXX+1 ,JMAXX2 = JMAXX+2 ) 00000930
*INCLUDE COMGF, FIXED, INSOURCE                          000010700
000006          COMMON /GMTRY/ IXMAX, JXMAX, DR(LMAXX), DPHI, 00001110
          LPOINT(LMAXX), RX(JMAXX), PHIX(IMAXX)          00001220
000007          COMMON /FLUID/ THTN(O:IMAXX1,O:JMAXX1),PSIN(O:IMAXX1,O:JMAXX1)00001330
          ,THTO(O:IMAXX1,O:JMAXX1),PSIO(O:IMAXX1,O:JMAXX1)00001440
          ,PSINRM(O:IMAXX1,O:JMAXX1)                    00001550
*INCLUDE COMUV, FIXED, INSOURCE                          000016800
000008          COMMON /VECTR/ U(IMAXX,JMAXX),V(IMAXX,JMAXX) 00001790
*INCLUDE COMLM, FIXED, INSOURCE                          00001900
000009          COMMON /LAMDA/ ALAMI(O:IMAXX1), ALAMO(O:IMAXX1), 00002010
          * ALAMIM, ALAMOM,                               00002111
          * RADAAM, AMAV                                  00002220
C
C
C
000010          PSIMIN = PSIN(O,O)                          00002330
000011          PSIMAX = PSIN(O,O)                          00002440
                                                00002550
C
000012          DO 4010 J = 0,JXMAX                          00002660
000013          DO 4005 I = 0,IXMAX                          00002770
000014          IF( PSIN(I,J).LT.PSIMIN ) PSIMIN = PSIN(I,J) 00002880
000015          IF( PSIN(I,J).GT.PSIMAX ) PSIMAX = PSIN(I,J) 00002990
000016          4005 CONTINUE                                00003100
000017          4010 CONTINUE                                00003210
C
000018          PSIINT = PSIMAX - PSIMIN                      00003320
000019          FACTOR = 1.0/PSIINT                          00003430
                                                00003540
C
000020          DO 4020 J = 0,JXMAX                          00003650
000021          DO 4015 I = 0,IXMAX                          00003760
000022          PSINRM(I,J) = 1.0 + FACTOR*PSIN(I,J)        00003870
000023          4015 CONTINUE                                00003980
000024          4020 CONTINUE                                00004090
C
C
C
000025          CALL RESLT1                                  00004200
          * ( PSIN, 0, IXMAX+1, 0, JXMAX+1, 0, IMAXX1, 0, JMAXX1, 1 ) 00004310
000026          CALL RESLT1                                  00004420
          * ( PSINRM, 0, IXMAX+1, 0, JXMAX+1, 0, IMAXX1, 0, JMAXX1, 2 ) 00004530
000027          WRITE(6,6210) PSIINT                          00004640
000028          CALL RESLT1                                  00004750
          * ( THTN, 0, IXMAX+1, 0, JXMAX+1, 0, IMAXX1, 0, JMAXX1, 3 ) 00004860
000029          CALL RESLT2                                  00004970
          * ( U, V, 1, IXMAX, 1, JXMAX, 1, IMAXX, 1, JMAXX ) 00005080
C
000030          WRITE(6,6010) 'EFFECTIVE THERMAL CONDUCTIVITY' 00005190
000031          WRITE(6,6020)                                00005300
C
000032          IXXXX = IXMAX + 2                              00005410
000033          LL1 = IXXXX/10                                00005520
000034          LL2 = MOD(IXXXX,10)                          00005630
000035          DO 4150 I = 1,LL1                              00005740
000036          I1 = (I - 1)*10                               00005850
000037          L1 = I1                                        00005960
000038          L2 = I1 + 9                                    00006070
000039          WRITE(6,6110) (I,I=L1,L2)                    00006180
000040          WRITE(6,6120) (ALAMI(I),I=L1,L2)            00006290
000041          4150 CONTINUE                                00006400
000042          IF (LL2.NE.0 ) THEN                            00006510
000043          I1 = LL1*10                                    00006620
000044          L1 = I1                                        00006730
000045          L2 = I1 + LL2 - 1                             00006840
000046          WRITE(6,6110) (I,I=L1,L2)                    00006950
000047          WRITE(6,6120) (ALAMI(I),I=L1,L2)            00007060
000048          ENDF                                           00007170
C
000049          WRITE(6,6130)ALAMIM                          00007280
C
C
C

```

```

000050      WRITE(6,6030)
C
000051      DO 4250 I = 1,LL1
000052          I1      = (I - 1)*10
000053          L1      = I1
000054          L2      = I1 + 9
000055          WRITE(6,6110) (II,II=L1,L2)
000056          WRITE(6,6120) (ALAMO(II),II=L1,L2)
000057      4250 CONTINUE
000058          IF (LL2.NE.0 ) THEN
000059              I1      = LL1*10
000060              L1      = I1
000061              L2      = I1 + LL2 - 1
000062              WRITE(6,6110) (II,II=L1,L2)
000063              WRITE(6,6120) (ALAMO(II),II=L1,L2)
000064          ENDIF
000065          WRITE(6,6130)ALAMOM
C
C
C
000066      RETURN
C
000067      6010 FORMAT(1H1,///,36X,
*      '***** ',A30,' *****')
000068      6020 FORMAT(////,' << INNER WALL >>',/)
000069      6030 FORMAT(////,' << OUTER WALL >>',/)
000070      6110 FORMAT(3X,' I ',2X,10(3X,:',<',14,')',3X))
000071      6120 FORMAT(1X,'LAMDA(I)',1P10D12.4,/)
000072      6130 FORMAT(//,1X,'AVERAGE LAMDA = ',1PD12.4)
000073      6210 FORMAT(////,1X,'PSIMAX - PSIMIN = ',F12.4)
C
C
000074      END

```

```

00006700
00006800
00006900
00007000
00007100
00007200
00007300
00007400
00007500
00007600
00007700
00007800
00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00009400
00009500
00009600
00009700
00009800
00009900

```



```

000001      SUBROUTINE  RESULT2
* ( XDATA, YDATA, IX1, IX2, JY1, JY2, IXX1, IXX2, JYY1, JYY2 )
C
000002      IMPLICIT REAL*8      (A-H,O-Z)
000003      CHARACTER*30 DTNAME
000004      DIMENSION XDATA(IXX1:IXX2,JYY1:JYY2), YDATA(IXX1:IXX2,JYY1:JYY2)
C
000005      DATA DTNAME /'          VELOCITY          '/
C
000006      LLO      = 5
000007      LL1      = IX2 - IX1 + 1
000008      LL2      = LL1/LL0 + 1
000009      LL3      = MOD(LL1,LL0)
C
000010      WRITE(6,6000) DTNAME
C
000011      DO 4800 L1 = 1,LL2
C
000012          IF( L1.EQ.LL2 .AND. LL3.EQ.0 )                GO TO 4800
000013          L10      = 5
000014          IF( L1.EQ.LL2 ) L10 = LL3
000015          LINI     = IX1 + (L1 - 1)*LLO
000016          LFIN    = LINI + L10 - 1
000017          WRITE(6,6020) ( L2, L2 = LINI,LFIN)
C
000018          DO 4100 JJ = JY1,JY2
000019              WRITE(6,6030) JJ,(XDATA(L2,JJ),YDATA(L2,JJ),L2=LINI,LFIN)
000020      4100      CONTINUE
000021              INT = JY2 - JY1
000022              IF ( L1.NE.LL2 ) WRITE(6,6010)
C
000023      4800 CONTINUE
C
000024      RETURN
000025      6000 FORMAT(1H1,///,30X
*          , '***** ',A30 ' *****')
000026      6010 FORMAT(1H1)
000027      6020 FORMAT(1H0,///,3X,'J / I',1X,5(9X,:', '<',14,'>',9X),/)
000028      6030 FORMAT(1H ,1X,'<',14,'>',3X,5(:(' ',1PE10.3,' ',1PE10.3,')'))
C
000029      END

```

```

000001      SUBROUTINE      FIGURE                                00000100
C                                                    00000200
C*****                                                    00000300
C*                                                    *00000400
C*      FUNCTION                                                    *00000500
C*                                                    *00000600
C*      OUTPUT OF CONSTRUCTION DATA                                *00000700
C*                                                    *00000800
C*****                                                    00000900
C                                                    00001000
C*****      DECLARATION      *****                                00001100
C                                                    00001200
000002      IMPLICIT      REAL*8      (A-H,O-Z)                    00001300
*INCLUDE PARAM, FIXED, INSOURCE                                00001400
000003      PARAMETER      ( LMAXX=100 ,KMAXX = 100      )        00000010
000004      PARAMETER      ( IMAXX=100 ,IMAXX1 = IMAXX+1 ,IMAXX2 = IMAXX+2 ) 00000020
000005      PARAMETER      ( JMAXX=100 ,JMAXX1 = JMAXX+1 ,JMAXX2 = JMAXX+2 ) 00000030
*INCLUDE CARD, FIXED, INSOURCE                                00001500
000006      COMMON /CARD1/ IOPT01                                00000011
000007      COMMON /CARD2/ KMAX      ,IMAX      ,RMAX                                00000012
000008      COMMON /CARD3/ JMAX(LMAXX) ,R(LMAXX) ,RA(LMAXX) ,DA(LMAXX), 00000020
AM(LMAXX)                                00000030
000009      COMMON /CARD4/ ESP1      ,ESP2      ,ITMAX                                00000040
*INCLUDE COMGF, FIXED, INSOURCE                                00001600
000010      COMMON /GMTRY/ IXMAX, JXMAX, DR(LMAXX), DPHI, 00000010
LPOINT(LMAXX), RX(JMAXX), PHIX(IMAXX) 00000020
000011      COMMON /FLUID/ THTN(O:IMAXX1,O:JMAXX1),PSIN(O:IMAXX1,O:JMAXX1) 00000030
, THTO(O:IMAXX1,O:JMAXX1),PSIO(O:IMAXX1,O:JMAXX1) 00000040
, PSINRM(O:IMAXX1,O:JMAXX1) 00000050
*INCLUDE COMUV, FIXED, INSOURCE                                00001700
000012      COMMON /VECTR/ U(IMAXX,JMAXX),V(IMAXX,JMAXX) 00000010
*INCLUDE COMLM, FIXED, INSOURCE                                00001800
000013      COMMON /LAMDA/ ALAMI(O:IMAXX1), ALAMO(O:IMAXX1), 00000010
* ALAMIM, ALAMOM, 00000011
* RADA AV, AMAV 00000020
000014      REAL*4      PHIO,PHI2N                                00001900
000015      REAL*8      VX(O:IMAXX1,O:JMAXX1),VY(O:IMAXX1,O:JMAXX1) 00002000
000016      DIMENSION RADA(LMAXX)                                00002100
C                                                    00002200
C*****      PROCEDURE      *****                                00002300
C                                                    00002400
000017      PI = 3.141592654                                    00002500
000018      PI2 = 0.5*PI                                        00002600
000019      DO 4005 K=1,KMAX                                    00002700
000020          RADA(K) =RA(K)*DA(K)                            00002800
000021      4005 CONTINUE                                      00002900
C                                                    00003000
000022      DO 4010 I = 1,IXMAX                                00003100
000023          PHIX(I) = PHIX(I) - PI2                        00003200
000024      4010 CONTINUE                                      00003300
C                                                    00003400
C                                                    00003500
000025      DO 4200 JY = 0,JXMAX+1                            00003600
000026          DO 4300 IX = 0,IXMAX+1                        00003700
000027              IF( (IX .EQ. 0) .OR. (IX .EQ. IXMAX+1) .OR. 00003800
(CJY .EQ. 0) .OR. (JY .EQ. JXMAX+1) ) THEN 00003900
000028                  PSINRM(IX,JY) = 1.00 00004000
000029                  VX(IX,JY) = .000 00004100
000030                  VY(IX,JY) = .000 00004200
000031              ELSE 00004300
000032                  COSX = DCOS(PHIX(IX)) 00004400
000033                  SINX = DSIN(PHIX(IX)) 00004500
000034                  VX(IX,JY) = U(IX,JY)*COSX - V(IX,JY)*SINX 00004600
000035                  VY(IX,JY) = U(IX,JY)*SINX + V(IX,JY)*COSX 00004700
000036              ENDIF 00004800
000037          4300 CONTINUE 00004900
000038      4200 CONTINUE 00005000
C                                                    00005100
C                                                    00005200
000039      PHIO = - PI2 00005300
000040      PHI2N = PI2 00005400
C                                                    00005500
000041      WRITE(30) IXMAX,JXMAX,KMAX 00005600
000042      WRITE(30) (SINGL(R(K)),K=1,KMAX),SINGL(RMAX), 00005700
* (SINGL(RADA(K)),K=1,KMAX), (SINGL(AM(K)),K=1,KMAX), 00005800
* SINGL(RADA AV),SINGL(AMAV) 00005900
000043      WRITE(30) SINGL(R(1)),(SINGL(RX(J)),J=1,JXMAX),SINGL(RMAX) 00006000
000044      WRITE(30) PHIO,(SINGL(PHIX(I)),I=1,IXMAX),PHI2N 00006100
000045      WRITE(30) ((SINGL(PSINRM(IX,JY)),IX=0,IXMAX+1),JY=0,JXMAX+1) 00006200
000046      WRITE(30) ((SINGL(THTN(IX,JY)),IX=0,IXMAX+1),JY=0,JXMAX+1) 00006300
000047      WRITE(30) ((SINGL(VX(IX,JY)),IX=0,IXMAX+1),JY=0,JXMAX+1) 00006400
000048      WRITE(30) ((SINGL(VY(IX,JY)),IX=0,IXMAX+1),JY=0,JXMAX+1) 00006500

```

000049	WRITE(30)	(SNGL(ALAMI(JY)),JY=0,JXMAX+1)	00006600
000050	WRITE(30)	(SNGL(ALAMO(JY)),JY=0,JXMAX+1)	00006700
	C		00006800
000051	RETURN		00006900
	C		00007000
000052	END		00007100

6. 図形処理プログラム

図形処理プログラムの COMMON ブロックとパラメータセットの内容は次のとおりである。

COMMON / DSPDT /				
No.	Variable Name	Type	Content.	Unit
1	RX (IMAXX 3)	R*4	Mesh 中心 R 座標 (円筒環の中心の座標を含む)	—
2	PHIX (JMAXX 2)	R*4	Mesh 中心 ϕ 座標	—
3	RR (LMAX 1)	R*4	r_k/r_i 及び r_o/r_i 無次元化した第 K 層の半径 及び断熱層の外側半径	—
4	THT (IMAXX 4, JMAXX 2)	R*4	温度の無次元化数	—
5	PSI (IMAXX 3, JMAXX 2)	R*4	流れ関数値	—
6	I MAX	I*4	R 方向総 Mesh 数 ($1 \leq, \leq \text{IMAXX}$)	—
7	J MAX	I*4	ϕ 方向総 Mesh 数 ($1 \leq, \leq \text{JMAXX}$)	—
8	K MAX	I*4	断熱層数 ($1 \leq, \leq \text{LMAX}$)	—

COMMON / DSPAG /				
No.	Variable Name	Type	Content.	Unit
1	RXT (IMAXX 4)	R*4	Mesh 中心の R 座標 (円筒環の中心及び内壁より内側の座標を含む)	—
2	VX (I2J2)	R*4	速度の X 成分	1/m
3	VY (I2J2)	R*4	速度の Y 成分	1/m
4	XA (I2J2)	R*4	Mesh 中心の x 座標値	—
5	YA (I2J2)	R*4	Mesh 中心の y 座標値	—
6	RADA (LMAX)	R*4	第 K 層の (レーリー数) \times (ダルシー数)	—
7	AM (LMAX)	R*4	第 K 層の温度伝導率比	—
8	RADAAV	R*4	(レーリー数) \times (ダルシー数) の平均値	—
9	AMAV	R*4	温度伝導率比の平均値	—

COMMON / DSPGR /				
No.	Variable Name	Type	Content.	Unit
1	ALAM(IMAXX 2, 2)	R * 4	実効熱伝導率	—
2	CHRR (10)	R * 4	グラフ出力R方向温度分布の半径	—
3	CHRP (10)	R * 4	グラフ出力φ方向温度分布の角度	deg.
4	NR	I * 4	グラフ出力R方向温度分布の個数	—
5	NP	I * 4	グラフ出力φ方向温度分布の個数	—

COMMON / FIGPM /				
No.	Variable Name	Type	Content.	Unit
1	NDIV	I * 4	等高値パラメータ	—
2	DH (5)	R * 4	等高値	—

PARAMETER			
Variable Name	Type	Value	Comment
IMAXX	I * 4	100	R 方向総分割の最大配列個数
IMAXX 2	I * 4	IMAXX + 2	IMAXX + 2
IMAXX 3	I * 4	IMAXX + 3	IMAXX + 3
IMAXX 4	I * 4	IMAXX + 4	IMAXX + 4
JMAXX	I * 4	100	φ方向総分割の最大配列個数
JMAXX 2	I * 4	JMAXX + 2	JMAXX + 2
I 2 J 2	I * 4	IMAXX 2 × JMAXX 2	IMAXX 2 × JMAXX 2
LMAX	I * 4	100	第 K 層の R 方向分割の最大配列個数

図形処理プログラムを以下に記す。なお、サブルーチン PXCNTL の CALL 文
 CALL PXCNTL (IPFLAG, ITITLE, IXTITL, IYTITL, DXMIN, DXMAX,
 DYMIN, DYMAX, IFLAG, NP, XDATA, YDATA, LSLCT, LCMT)
 の各変数の内容は次のとおりである。

Variable Name	Type	Content
IPFLAG	I*4	=1:新しい Graph を書く 0:古い Graph の上に重ねる 1:最後の Graph を書く
ITITLE	C*30	Graph Title (30文字以内の文字列)
IXTITL	C*30	x軸 Title (30文字以内の文字列)
IYTITL	C*30	y軸 Title (30文字以内の文字列)
DXMIN	R*4	x軸 data 最小値
DXMAX	R*4	x軸 data 最大値
DYMIN	R*4	y軸 data 最小値
DYMAX	R*4	y軸 data 最大値
IFLAG	I*4	Scoling =0:Linear-Linear =1:Linear-Log =2:Log-Linear =3:Log-Log
NP	I*4	data point 数
XDATA (NP)	R*4	x軸 data
YDATA (NP)	R*4	y軸 data
LSLCT	I*4	線種 $1 \leq LSLCT \leq 10$
LCMT	C*10	線種 Comment (10文字以内の文字列)

```

C*****00000100
C*00000200
C*   MAIN00000300
C*00000400
C*   FUNCTION ;00000500
C*00000600
C*   OUTPUT OF ISOTHERM (PLANE AND SOLID FIGURE)00000700
C*00000800
C*****00000900
C00001000
C***** DECLARATION *****00001100
*INCLUDE PARAMF, FIXED, INSOURCE00001200
000001 PARAMETER ( IMAXX = 100 , IMAXX2 = IMAXX + 2 )00000010
000002 PARAMETER ( IMAXX3 = IMAXX + 3 )00000020
000003 PARAMETER ( IMAXX4 = IMAXX + 4 )00000030
000004 PARAMETER ( JMAXX = 100 , JMAXX2 = JMAXX + 2 )00000040
000005 PARAMETER ( I2J2 = IMAXX2 * JMAXX2 )00000050
000006 PARAMETER ( LMAX = 100 , LMAX1 = LMAX + 1 )00000060
*INCLUDE COMFIG, FIXED, INSOURCE00001300
000007 COMMON /FIGPM/ NOIV, DH(5)00000010
*INCLUDE COMDSP, FIXED, INSOURCE00001400
000008 COMMON /DSPDT/ RX(IMAXX3) , PHIX(JMAXX2) , RR(LMAX1)00000010
      , THT(IMAXX4, JMAXX2) , PSI(IMAXX3, JMAXX2)00000020
      , IMAX, JMAX, KMAX00000030
*INCLUDE COMARG, FIXED, INSOURCE00001500
000009 COMMON /DSPAG/ RXT(IMAXX4)00000010
      , VX(I2J2) , VY(I2J2)00000020
      , XA(I2J2) , YA(I2J2)00000030
      , RADA(LMAX) , AM(LMAX)00000040
      , RADA AV , AM AV00000050
*INCLUDE COMGRP, FIXED, INSOURCE00001600
000010 COMMON /DSPGR/ ALAM(JMAXX2, 2) , CHRR(10) , CHR(10) ,00000010
      NR , NP00000020
000011 DIMENSION IFLAG(6)00001700
C00001800
C***** PROCEDURE00001900
C00002000
C***** READ DATA FROM DISK00002100
C00002200
000012 CALL DATAIN( IRET )00002200
C00002300
000013 IF( IRET.NE.0 ) GO TO 39000002400
C00002600
C***** INPUT OF CARD DATAS *****00002700
C00002800
000014 CALL CARDIN( N, IFLAG, IRET )00002900
C00003000
000015 IF( IRET.NE.0 ) GO TO 39000003100
C00003200
C***** OUTPUT FIGURE *****00003300
000016 DO 4500 I = 1, N00003400
C00003500
000017 IF( IFLAG(I).LE.3 ) THEN00003600
000018 CALL DSPARG( IFLAG(I) )00003700
000019 ELSE00003800
000020 CALL DSPGRP( IFLAG(I) )00003900
000021 ENDF00004000
C00004100
000022 4500 CONTINUE00004200
C00004300
C00004400
000023 GO TO 999900004500
C00004600
000024 3900 CONTINUE00004700
000025 WRITE(6, 6000)00004800
C00004900
000026 9999 CONTINUE00005000
C00005100
C00005200
000027 STOP00005300
C00005400
000028 6000 FORMAT(1H0, 20X, 'ERROR STOP')00005500
C00005600
000029 END00005700

```

```

000001      SUBROUTINE DATAIN( IRET )                                00000100
C*****                                                    *00000200
C*                                                    *00000300
C*      DATAIN                                                    *00000400
C*                                                    *00000500
C*      FUNCTION ;                                                    *00000600
C*                                                    *00000700
C*      READ DATA FILE                                                    *00000800
C*                                                    *00000900
C*****                                                    *00001000
C                                                    00001100
C*****      DECLARATION      *****                                00001200
C                                                    00001300
*INCLUDE PARAMF, FIXED, INSOURCE                                00001400
000002      PARAMETER      ( IMAXX = 100 , IMAXX2 = IMAXX + 2 )      00000010
000003      PARAMETER      (           IMAXX3 = IMAXX + 3 )          00000020
000004      PARAMETER      (           IMAXX4 = IMAXX + 4 )          00000030
000005      PARAMETER      ( JMAXX = 100 , JMAXX2 = JMAXX + 2 )      00000040
000006      PARAMETER      ( I2J2 = IMAXX2 * JMAXX2 )                00000050
000007      PARAMETER      ( LMAX = 100 , LMAX1 = LMAX + 1 )        00000060
*INCLUDE COMDSP, FIXED, INSOURCE                                00001500
000008      COMMON /DSPDT/  RX(IMAXX3) , PHIX(JMAXX2) , RR(LMAX1)    00000010
      .                    , THT(IMAXX4, JMAXX2) , PSI(IMAXX3, JMAXX2) 00000020
      .                    , IMAX, JMAX, KMAX                          00000030
*INCLUDE COMARG, FIXED, INSOURCE                                00001600
000009      COMMON /DSPAG/  RXT(IMAXX4)                               00000010
      .                    , VX(I2J2), VY(I2J2)                       00000020
      .                    , XA(I2J2), YA(I2J2)                       00000030
      .                    , RADA(LMAX), AM(LMAX)                     00000040
      .                    , RADA AV, AMAV                             00000050
*INCLUDE COMGRP, FIXED, INSOURCE                                00001700
000010      COMMON /DSPGR/  ALAM(JMAXX2, 2) , CHRR(10) , CHR(10) ,    00000010
      .                    NR, NP                                       00000020
000011      REAL*4          VXX(IMAXX2, JMAXX2), VYY(IMAXX2, JMAXX2) 00001800
C                                                    00001900
C*****      PROCEDURE      *****                                00002000
C                                                    00002100
000012      IRET = 0                                                00002200
C                                                    00002300
C                                                    00002400
C*****      INPUT OF FIGURE DATA      *****                    00002500
C                                                    00002600
000013      IREC = 1                                                00002700
000014      READ(30, END=3900)  JMAX, IMAX, KMAX                      00002800
C                                                    00002900
000015      IREC = 2                                                00003000
000016      READ(30, END=3900)  (RR(L), L=1, KMAX+1)                 00003100
      .                    , (RADA(L), L=1, KMAX) , (AM(L), L=1, KMAX) , RADA AV, AMAV 00003200
C                                                    00003300
000017      RX(1) = 0.0                                              00003400
000018      IREC = 3                                                00003500
000019      READ(30, END=3900)  (RX(I), I=2, IMAX+3)                 00003600
C                                                    00003700
000020      RXT(1) = RX(1)                                            00003800
000021      RXT(2) = RX(2) - (RX(IMAX+3) - RX(2))/50.                00003900
000022      DO 4600  IX = 3, IMAXX4                                   00004000
000023      RXT(IX) = RX(IX-1)                                        00004100
000024      4600 CONTINUE                                            00004200
C                                                    00004300
000025      IREC = 4                                                00004400
000026      READ(30, END=3900)  (PHIX(J), J=1, JMAX+2)              00004500
000027      IREC = 5                                                00004600
000028      READ(30, END=3900)  ((PSI(IX, JY), JY=1, JMAX+2), IX=2, IMAX+3) 00004700
000029      IREC = 6                                                00004800
000030      READ(30, END=3900)  ((THT(IX, JY), JY=1, JMAX+2), IX=3, IMAX+4) 00004900
C                                                    00005000
000031      DO 4100  JY = 1, JMAX+2                                   00005100
000032      THT(1, JY) = 1.0                                          00005200
000033      THT(2, JY) = 1.0                                          00005300
000034      PSI(1, JY) = 1.0                                          00005400
000035      4100 CONTINUE                                            00005500
C                                                    00005600
C      DO 4400  JY = 2, JMAX+1                                       00005700
C      DO 4500  IX = 3, IMAX+2                                       00005800
C%      PSI(IX, JY) = 1.0 - PSI(IX, JY)                                00005900
C      PSI(IX, JY) = ABS( PSI(IX, JY) )                               00006000
C4500 CONTINUE                                                       00006100
C4400 CONTINUE                                                       00006200
C                                                    00006300
000036      IREC = 7                                                00006400
000037      READ(30, END=3900)  ((VXX(IX, JY), JY=1, JMAX+2), IX=1, IMAX+2) 00006500
000038      IREC = 8                                                00006600

```

```

000039      READ(30,END=3900)((VYY(IX,JY),JY=1,JMAX+2),IX=1,IMAX+2)
C
000040      IDD = INT( IMAX/15 ) + 1
000041      JDD = INT( JMAX/15 ) + 1
000042      LXI = 0
000043      DO 4700 JY = 1,JMAX+2,JDD
000044          DO 4710 IX = 1,IMAX+2,IDD
000045              LXI = LXI + 1
000046              VX(LXI) = VXX(IX,JY)
000047              VY(LXI) = VYY(IX,JY)
000048              XA(LXI) = RX(IX+1)*COS(PHIX(JY))
000049              YA(LXI) = RX(IX+1)*SIN(PHIX(JY))
000050      4710      CONTINUE
000051      4700 CONTINUE
C
000052      IREC = 9
000053      READ(30,END=3900)(ALAM(IL,1),IL=1,JMAX+2)
000054      IREC = 10
000055      READ(30,END=9999)(ALAM(IL,2),IL=1,JMAX+2)
000056      GO TO 9999
C
C
000057      3900 CONTINUE
000058          WRITE(6,6910) IREC
000059      6910 FORMAT(10X,'FILE READ ERROR. RECORD NO. = 'I1)
000060          IRET = 1
C
000061      9999 CONTINUE
000062      RETURN
000063      END
C
000001      SUBROUTINE  DMIMAX(ADTUM,N,NDT,DMIN,DMAX)
C
C*****  DECLARATION  *****
C
000002      REAL*4      ADTUM(N)
C
C*****  PROCEDURE  *****
C
000003      DO 4000 L = 1,NDT
000004          IF( ADTUM(L) .LT. DMIN )  DMIN = ADTUM(L)
000005          IF( ADTUM(L) .GT. DMAX )  DMAX = ADTUM(L)
000006      4000 CONTINUE
C
C
000007      RETURN
C
000008      END

```

```

000001      SUBROUTINE      ERRMSG(IRTN)                                00000100
C                                                    00000200
C*****00000300
C*                                                    *00000400
C*      FUNCTION ;                                                    *00000500
C*                                                    *00000600
C*      OUTPUT OF ERROR MESSEGE.                                       *00000700
C*                                                    *00000800
C*      DATE      ; 84/03/30                                           *00000900
C*                                                    *00001000
C*      ARGUMENT ; (TYPE,I/O,UNIT) *00001100
C*                                                    *00001200
C*      (1) IRTN --- RETURN CODE (I*4,I,-) *00001300
C*                                                    *00001400
C*      =1 ; FIGURE SELECT FLAG ERROR *00001500
C*      =2 ; FIGURE PARAMETER ERROR *00001600
C*      =3 ; FIGURE PARAMETER NUMBER ERROR *00001700
C*                                                    *00001800
C*****00001900
C                                                    00002000
C***** PROCEDURE ***** 00002100
C                                                    00002200
C*** FIGURE SELECT FLAG ERROR *** 00002300
C                                                    00002400
000002      IF( IRTN .EQ. 1 ) THEN 00002500
000003          WRITE(6,6000) 00002600
C                                                    00002700
C*** FIGURE PARAMETER ERROR *** 00002800
C                                                    00002900
000004      ELSEIF( IRTN .EQ. 2 ) THEN 00003000
000005          WRITE(6,6100) 00003100
C                                                    00003200
C*** FIGURE PARAMETER NUMBER ERROR *** 00003300
C                                                    00003400
000006      ELSEIF( IRTN .EQ. 3 ) THEN 00003500
000007          WRITE(6,6200) 00003600
C                                                    00003700
000008      ENDIF 00003800
C                                                    00003900
000009      RETURN 00004000
C                                                    00004100
C                                                    00004200
000010      6000 FORMAT(1H0,20X,'CODE-1 FIGURE SELECT FLAG ERROR') 00004300
000011      6100 FORMAT(1H0,20X,'CODE-2 FIGURE PARAMETER ERROR') 00004400
000012      6200 FORMAT(1H0,20X,'CODE-3 FIGURE PARAMETER NUMBER ERROR') 00004500
C                                                    00004600
000013      END 00004700

```

```

000001      SUBROUTINE  CARDIN( N, IFLAG, IRET )
C
C*****
C*
C*      FUNCTION ;
C*
C*      OUTPUT OF INPUT DATA CARD IMAGE LIST AND
C*
C*      SET-UP OF INPUT DATA VALUE.
C*
C*      DATE      ; 84/03/30
C*
C*      ARGUMENT ;
C*
C*      N      : OUTPUT GRAPHIC COUNT
C*      IFLAG  : GRAPHIC FLAG
C*      IRET   : RETURN CORD
C*
C*****
C
C*****  DECLARATION  *****
C
*INCLUDE PARAMF, FIXED, INSOURCE
000002      PARAMETER ( IMAXX = 100 , IMAXX2 = IMAXX + 2 )
000003      PARAMETER (           IMAXX3 = IMAXX + 3 )
000004      PARAMETER (           IMAXX4 = IMAXX + 4 )
000005      PARAMETER ( JMAXX = 100 , JMAXX2 = JMAXX + 2 )
000006      PARAMETER ( I2J2 = IMAXX2 * JMAXX2 )
000007      PARAMETER ( LMAX = 100 , LMAX1 = LMAX + 1 )
*INCLUDE COMDSP, FIXED, INSOURCE
000008      COMMON /DSPDT/  RX(IMAXX3) , PHIX(JMAXX2) , RR(LMAX1)
      .                , THT(IMAXX4, JMAXX2) , PSI(IMAXX3, JMAXX2)
      .                , IMAX, JMAX, KMAX
*INCLUDE COMGRP, FIXED, INSOURCE
000009      COMMON /DSPGR/  ALAM(JMAXX2, 2) , CHRR(10) , CHR(10) ,
      .                , NR, NP
000010      DIMENSION      IFLAG(6)
C
000011      CHARACTER*80  CARD
000012      CHARACTER*5   CHRBLK(10)
000013      CHARACTER*1    IFLGXX
C
C*****  PROCEDURE  *****
C
C
000014      KMAX1 = KMAX + 1
000015      IRTN = 0
000016      ICDR = 0
000017      N = 0
C
C***  CARD READING ***
C
000018      WRITE(6, 6000)
000019      WRITE(6, 6100) (NO, NO=1, 8)
C
000020      3000 CONTINUE
C
000021      READ(5, 5000, END=3333) CARD
C
000022      ICD = ICD + 1
C
C***  CARD IMAGE WRITING ***
C
000023      WRITE(6, 6200) ICD, CARD
C
000024
000025      3333 CONTINUE
C
C***  INPUT DATA REWIND ***
C
000026      REWIND 5
C
C*****  SET-UP INPUT DATA VALUE & *****
C*****  INPUT DATA SUMMARY WRITING *****
C
C
C***  DROWING FIGURE SELECTION ***
C
000027      3050 CONTINUE

```

```

000028      READ(5,7000,END=9999)  IFLGXX,NUMXX,CHRBLK      00007000
C                                                    00007100
000029      IF ( N.GT.6 )      GO TO 9999      00007200
000030      IF ( IFLGXX.EQ.'A' )      GO TO 3100      00007300
000031      IF ( IFLGXX.EQ.'B' )      GO TO 3200      00007400
000032      IF ( IFLGXX.EQ.'C' )      GO TO 3300      00007500
000033      IF ( IFLGXX.EQ.'D' )      GO TO 3400      00007600
000034      IF ( IFLGXX.EQ.'E' )      GO TO 3500      00007700
000035      IF ( IFLGXX.EQ.'F' )      GO TO 3600      00007800
C                                                    00007900
C                                                    00008000
000036      GO TO 3910      00008100
C                                                    00008200
C                                                    00008300
000037      3100 CONTINUE      00008400
000038          N      = N + 1      00008500
000039          IFLAG(N) = 1      00008600
000040          GO TO 3050      00008700
C                                                    00008800
000041      3200 CONTINUE      00008900
000042          N      = N + 1      00009000
000043          IFLAG(N) = 2      00009100
000044          GO TO 3050      00009200
C                                                    00009300
000045      3300 CONTINUE      00009400
000046          N      = N + 1      00009500
000047          IFLAG(N) = 3      00009600
000048          GO TO 3050      00009700
C                                                    00009800
000049      3400 CONTINUE      00009900
000050          N      = N + 1      00010000
000051          IFLAG(N) = 4      00010100
000052          GO TO 3050      00010200
C                                                    00010300
000053      3500 CONTINUE      00010400
000054          IF( ( NUMXX.LT.1 ) .AND. ( NUMXX.GT.10 ) )      GO TO 3930      00010500
000055          N      = N + 1      00010600
000056          IFLAG(N) = 5      00010700
000057          NR      = NUMXX      00010800
000058          ICHECK = 0      00010900
C                                                    00011000
000059          DO 4510 II = 1,NR      00011100
000060              READ(CHRBLK(II),'(F5.0)')  CHR(II)      00011200
000061              IF( ( CHR(II).LT.RR(1) ) .OR.      00011300
                  ( CHR(II).GT.RR(KMAX1) ) )      00011400
                  ICHECK = ICHECK + 10**(II-1)      00011500
000062      4510 CONTINUE      00011600
C                                                    00011700
000063          IF( ICHECK .NE. 0 )      GO TO 3920      00011800
000064          GO TO 3050      00011900
C                                                    00012000
C                                                    00012100
000065      3600 CONTINUE      00012200
000066          IF( ( NUMXX.LT.1 ) .AND. ( NUMXX.GT.10 ) )      GO TO 3930      00012300
000067          N      = N + 1      00012400
000068          IFLAG(N) = 6      00012500
000069          NP      = NUMXX      00012600
000070          ICHECK = 0      00012700
C                                                    00012800
000071          DO 4610 II = 1,NP      00012900
000072              READ(CHRBLK(II),'(F5.0)')  CHR(II)      00013000
000073              IF( ( CHR(II).LT.O.O ) .OR.      00013100
                  ( CHR(II).GT.180.O ) )      00013200
                  ICHECK = ICHECK + 10**(II-1)      00013300
000074      4610 CONTINUE      00013400
C                                                    00013500
000075          IF( ICHECK .NE. 0 )      GO TO 3920      00013600
000076          GO TO 3050      00013700
C                                                    00013800
C                                                    00013900
C                                                    00014000
000077      3910 CONTINUE      00014100
000078          IRET      = 1      00014200
000079          CALL ERRMSG( IRET )      00014300
000080          GO TO 9999      00014400
C                                                    00014500
000081      3920 CONTINUE      00014600
000082          IRET      = 2      00014700
000083          CALL ERRMSG( IRET )      00014800
000084          GO TO 9999      00014900
C

```


000085	3930 CONTINUE	00015000
000086	IRET = 3	00015100
000087	CALL ERRMSG(IRET)	00015200
	C	00015300
000088	9999 CONTINUE	00015400
000089	RETURN	00015500
	C	00015600
000090	5000 FORMAT(A80)	00015700
000091	6000 FORMAT(1H1/1H ,36X, '*****' INPUT DATA CARD IMAGE LST '****')//)	00015800
000092	6100 FORMAT(1H0,20X,'NO.',2X,8('....*....'I1))	00015900
000093	6200 FORMAT(1H ,20X,12,3X,A80)	00016000
	C	00016100
000094	7000 FORMAT(A1,I4,10A5)	00016200
	C	00016300
	C	00016400
	C	00016500
000095	END	00016600

```

000001      SUBROUTINE      DSPARG( IFLAG )                                00000100
C                                                    00000200
C*****                                                    00000300
C*                                                    *00000400
C*      FUNCTION ;                                                    *00000500
C*                                                    *00000600
C*      OUTPUT OF STREAM-LINE (PLANE AND SOLID FIGURE) &          *00000700
C*                                                    *00000800
C*      DIMENSIONLESS THERM (PLANE AND SOLID FIGURE) &          *00000900
C*                                                    *00001000
C*      VELOCITY VECTOR.                                          *00001100
C*                                                    *00001200
C*      DATE      ; 84/03/30                                        *00001300
C*                                                    *00001400
C*      ARGUMENT ;                                              *00001500
C*                                                    *00001600
C*      (1) IFLAG = 1 : DRAW PSI DISTRIBUTION                      *00001700
C*      = 2 : DRAW THITA DISTRIBUTION                             *00001800
C*      = 3 : DRAW (U,V)                                         *00001900
C*                                                    *00002000
C*****                                                    00002100
C                                                    00002200
C*****      DECLARATION      *****                                00002300
C                                                    00002400
*INCLUDE PARAMF, FIXED, INSOURCE                                00002500
000002      PARAMETER      ( IMAXX = 100 , IMAXX2 = IMAXX + 2 )    00000010
000003      PARAMETER      (          , IMAXX3 = IMAXX + 3 )    00000020
000004      PARAMETER      (          , IMAXX4 = IMAXX + 4 )    00000030
000005      PARAMETER      ( JMAXX = 100 , JMAXX2 = JMAXX + 2 )  00000040
000006      PARAMETER      ( I2J2 =          , IMAXX2 * JMAXX2 )  00000050
000007      PARAMETER      ( LMAX = 100 , LMAX1 = LMAX + 1 )    00000060
*INCLUDE COMFIG, FIXED, INSOURCE                                00002600
000008      COMMON      /FIGPM/ NDIV, DH(5)                        00000010
*INCLUDE COMDSP, FIXED, INSOURCE                                00002700
000009      COMMON      /DSPDT/  RX(IMAXX3) , PHIX(JMAXX2) , RR(LMAX1)  00000010
      .                                                    , THT(IMAXX4, JMAXX2) , PSI(IMAXX3, JMAXX2)  00000020
      .                                                    , IMAX, JMAX, KMAX                                00000030
*INCLUDE COMARG, FIXED, INSOURCE                                00002800
000010      COMMON      /DSPAG/  RXT(IMAXX4)                        00000010
      .                                                    , VX(I2J2), VY(I2J2)                            00000020
      .                                                    , XA(I2J2), YA(I2J2)                            00000030
      .                                                    , RADA(LMAX), AM(LMAX)                          00000040
      .                                                    , RADA AV, AM AV                                00000050
000011      CHARACTER*8  IDDATA, IDDAT(3)                          00002900
000012      CHARACTER*31 CMTITL, CMLNUM(5), CMRADA, CMAM, CMROUT    00003000
C                                                    00003100
C*****      PROCEDURE      *****                                00003200
C                                                    00003300
000013      IMAX2 = IMAX + 2                                        00003400
000014      IMAX3 = IMAX + 3                                        00003500
000015      IMAX4 = IMAX + 4                                        00003600
000016      JMAX2 = JMAX + 2                                        00003700
000017      IX2Y2 = IMAX2*JMAX2                                    00003800
C                                                    00003900
000018      NDIV = -11                                             00004000
000019      NDIVS = 777                                            00004100
000020      ANGL1 = 30.0                                           00004200
000021      ANGL2 = .0                                             00004300
000022      ANGL3 = -45.                                          00004400
000023      VLNG = 400.0                                           00004500
000024      DLNG = 350.0                                           00004600
C                                                    00004700
000025      CMTITL = 'LAYER NO.      R      RA*DA      AM&'      00004800
000026      CMRADA = 'AVERAGE RA*DA :      &'                    00004900
000027      CMAM = 'AVERAGE AM :      &'                          00005000
000028      CMROUT = 'ROUT/RIN :      &'                          00005100
000029      IDDAT(1) = 'FLOW '                                       00005200
000030      IDDAT(2) = 'THERMP'                                       00005300
000031      IDDAT(3) = 'VECTOR'                                       00005400
C                                                    00005500
C                                                    00005600
000032      DO 4300 K = 1, KMAX                                       00005700
000033      CMLNUM(K) = '                                                    &' 00005800
000034      WRITE(CMLNUM(K)(4:5), '(I2)') K                            00005900
000035      WRITE(CMLNUM(K)(11:15), '(F5.1)') RR(K)                   00006000
000036      WRITE(CMLNUM(K)(19:24), '(F6.1)') RADA(K)                 00006100
000037      WRITE(CMLNUM(K)(26:30), '(F5.1)') AM(K)                   00006200
000038      4300 CONTINUE                                             00006300
C                                                    00006400
000039      WRITE(CMROUT(17:21), '(F5.1)') RR(KMAX+1)                 00006500
000040      WRITE(CMRADA(16:21), '(F6.1)') RADA AV                    00006600
000041      WRITE(CMAM(17:21), '(F5.1)') AM AV                          00006700

```

```

C
C***** OUTPUT FIGURE *****
C
000042 CALL ARGUS(0)
000043 IDDATA = IDDAT(IFLAG)
C
000044 IF( IFLAG.EQ.3 ) GO TO 3500
C
C*** PLANE FIGURE ***
C
000045 IF( IFLAG.EQ.1 ) THEN
000046 CALL STOR2N(IDDATA,-IMAX3,-JMAX2
. RX,PHIX,PSI,IMAXX3)
C
000047 CALL COMENT(IDDATA,1,'<STREAM-LINE>&'
. ,15.0,165.0,0.0,5.)
C
000048 ELSE
000049 CALL STOR2N(IDDATA,-IMAX4,-JMAX2
. ,RXT,PHIX,THT,IMAXX4)
C
000050 CALL COMENT(IDDATA,10,'<DIMENSIONLESS TEMP.>&'
. ,15.0,165.0,0.0,5.)
C
000051 ENDIF
C
000052 CALL GRSIZE(IDDATA,0,200.0,150.0)
000053 CALL ORIGIN(IDDATA,-20.0,20.0)
000054 CALL HVALUE(IDDATA,NDIV,DH)
000055 CALL CVKIND(IDDATA,0,1,5)
000056 CALL APPROX(IDDATA,0)
000057 XL = 190.0
000058 YL = 145.0
000059 CALL COMENT(IDDATA,2,CMTITL,XL,YL,0.0,3.)
000060 KID = 3
000061 DO 4400 K = 1,KMAX
000062 YL = YL - 5.0
000063 KID = KID + 1
000064 CALL COMENT(IDDATA,KID,CMLNUM(K)
. ,XL,YL,0.0,3.)
000065 4400 CONTINUE
000066 YL = YL - 10.0
000067 CALL COMENT(IDDATA,20,CMROUT,XL,YL,0.0,3.)
000068 YL = YL - 5.0
000069 CALL COMENT(IDDATA,21,CMRADA,XL,YL,0.0,3.)
000070 YL = YL - 5.0
000071 CALL COMENT(IDDATA,22,CMAM,XL,YL,0.0,3.)
C
000072 CALL CMTOPT(3)
C
000073 CALL CONTOR(IDDATA,0)
C
C
C
C*** SOLID FIGURE ***
C
000074 IF( IFLAG.EQ.1 ) THEN
000075 DO 4040 JY = 1,JMAX+2
000076 DO 4040 IX = 1,IMAX+2
000077 PSI(IX,JY) = 1.0 - PSI(IX,JY)
000078 4040 CONTINUE
C
000079 IDDATA = 'FLOWSLID'
000080 CALL STOR2N(IDDATA,-IMAX3,-JMAX2
. ,RX,PHIX,PSI,IMAXX3)
C
000081 CALL COMENT(IDDATA,1000,'<STREAM-LINE>&'
. ,-15.0,195.0,0.0,5.)
C
000082 ELSE
000083 DO 4100 JY = 1,JMAX+2
000084 THT(1,JY) = .0
000085 THT(2,JY) = .0
000086 4100 CONTINUE
C
000087 IDDAT(2) = 'THERMS'
000088 IDDATA = IDDAT(2)
000089 CALL STOR2N(IDDATA,-IMAX4,-JMAX2
. ,RXT,PHIX,THT,IMAXX4)
C
000090 CALL COMENT(IDDATA,1100,'<DIMENSIONLESS TEMP.>&'
. ,-15.0,195.0,0.0,5.)
000091 ENDIF
C

```

```

000092      CALL      GRSIZE(IDDATA,0,250.0,200.0)      00015000
000093      CALL      ORIGIN(IDDATA,10.0,-10.0)         00015100
000094      CALL      VIEWP (IDDATA,ANGL1,ANGL2,ANGL3,VLNG,DLNG,4) 00015200
000095      CALL      OBJECT(IDDATA,150.0,150.0,60.0)   00015300
000096      CALL      HVALUE(IDDATA,NDIVS,DH)           00015400
000097      CALL      REALGH(IDDATA,0)                   00015500
C
000098      XL = 165.0      00015600
000099      YL = 175.0      00015700
000100      CALL      COMENT(IDDATA,5200,CMTITL,XL,YL,0.0,3.) 00015800
000101      KID = 5200      00015900
000102      DO 4500 K =1,KMAX      00016000
000103      YL = YL - 5.0      00016100
000104      KID = KID +1      00016200
000105      CALL      COMENT(IDDATA,KID,CMLNUM(K)         00016300
           ,XL,YL,0.0,3.)      00016400
C
000106      4500      CONTINUE      00016500
000107      YL = YL - 10.0     00016600
000108      CALL      COMENT(IDDATA,5300,CMROUT,XL,YL,0.0,3.) 00016700
000109      YL = YL - 5.0      00016800
000110      CALL      COMENT(IDDATA,5310,CMRADA,XL,YL,0.0,3.) 00016900
000111      YL = YL - 5.0      00017000
000112      CALL      COMENT(IDDATA,5320,CMAM,XL,YL,0.0,3.) 00017100
C
000113      CALL      CMTOPT(3)      00017200
C
000114      CALL      BRDEYE(IDDATA,0)      00017300
C
000115      GO TO 3900      00017400
C
C*** VELOCITY VECTOR ***      00017500
C
000116      3500 CONTINUE      00017600
000117      CALL      STFLZR(IDDAT(3),IX2Y2,XA,YA,VX,VY) 00017700
C
000118      CALL      GRSIZE(IDDATA,0,75.0,150.0)      00017800
000119      CALL      ORIGIN(IDDATA,80.0,20.0)         00017900
000120      CALL      XWYWGH(IDDATA,1)                 00018000
000121      CALL      COMENT(IDDATA,10,'<VELOCITY>&',' -85.,165.0,.0,5.) 00018100
C
000122      XL = 95.0      00018200
000123      YL = 145.0     00018300
000124      CALL      COMENT(IDDATA,100,CMTITL,XL,YL,0.0,3.) 00018400
000125      KID = 100      00018500
000126      DO 4600 K =1,KMAX      00018600
000127      YL = YL - 5.0      00018700
000128      KID = KID +1      00018800
000129      CALL      COMENT(IDDATA,KID,CMLNUM(K)         00018900
           ,XL,YL,0.0,3.)      00019000
C
000130      4600      CONTINUE      00019100
000131      YL = YL - 10.0     00019200
000132      CALL      COMENT(IDDATA,200,CMROUT,XL,YL,0.0,3.) 00019300
000133      YL = YL - 5.0      00019400
000134      CALL      COMENT(IDDATA,210,CMRADA,XL,YL,0.0,3.) 00019500
000135      YL = YL - 5.0      00019600
000136      CALL      COMENT(IDDATA,220,CMAM,XL,YL,0.0,3.) 00019700
C
000137      CALL      CMTOPT(3)      00019800
C
000138      CALL      ARGUS(888)      00019900
C
000139      CALL      FLOW2D(IDDATA,0)      00020000
C
000140      CALL      FLMDSP(RR,KMAX,LMAX1)      00020100
C
000141      CALL      ARGUS(666)      00020200
C
000142      3900 CONTINUE      00020300
C
000143      CALL      ARGUS(999)      00020400
C
000144      RETURN      00020500
C
000145      END      00020600

```

```

000001      SUBROUTINE  FLMDSP(RR,KMAX,LMAX)
C
C*****  DECLARATION  *****
C
000002      REAL*4      RR(LMAX)
C
000003      PI = 3.141592654
000004      PIH = 3.141592654/2.0
000005      PIX = PI/50.
000006      XO = 0.0
000007      YO = 0.0
C
000008      DO 4000 L = 1,KMAX+1,KMAX
000009          X1 = XO
000010          Y1 = YO - RR(L)
C
000011          CALL      UPLOT(X1,Y1,3)
C
000012          DO 4100 I = 1,50
000013              THETA = PIX*FLOAT(I) - PIH
000014              X = XO + RR(L)*COS(THETA)
000015              Y = YO + RR(L)*SIN(THETA)
000016              CALL      UPLOT(X,Y,2)
C
000017      4100      CONTINUE
000018      4000      CONTINUE
C
000019          XE = XO
000020          YE = YO + RR(1)
000021          CALL      UPLOT(XE,YE,2)
C
000022          YE = YO - RR(1)
000023          CALL      UPLOT(XE,YE,3)
C
000024          YE = YO - RR(KMAX+1)
000025          CALL      UPLOT(XE,YE,2)
C
000026          CALL      UPLOT(0.0,0.0,3)
C
C
000027      RETURN
C
000028      END

```

```

00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000
00002100
00002200
00002300
00002400
00002500
00002600
00002700
00002800
00002900
00003000
00003100
00003200
00003300
00003400
00003500
00003600
00003700
00003800
00003900
00004000
00004100
00004200
00004300

```



```

C
000042          LCMTL = 'R-OUT'
000043          CALL PXCNTL( -1, ITITLL, IXTTLL, IYTTLL, 0.0, 180.0,
          .
          .
          .
          DMINO, DMAX0, 0, JMAX+2,
          PHIXD, ALAM(1,2), 2, LCMTL )
C
000044          GO TO 3900
C
C**** THERMAL DISTRIBUTION ****
C
C*** R = CONSTANT ***
C
000045          3400 CONTINUE
C
          THTMIN = 0.0
          THTMAX = 0.99
          ITITLR = 'DIMENSIONLESS TEMP. R=CONST.'
          IXTTLR = 'PHI( DEG)'
          IYTTLR = 'THETA'
          DO 4200 LR1 = 1, NR
C
          WRITE(CHRXX(LR1), '(F5.2)') CHRXX(LR1)
          CALL VALFIX( CHRXX(LR1), THTX, JMAX2, 1 )
C
          IFLM = 0
          IF( LR1 .EQ. NR ) IFLM = -1
          IF( LR1 .EQ. 1 ) IFLM = 1
          LCMTX = CHRXX/CHRXX(LR1)
          CALL PXCNTL( IFLM, ITITLR, IXTTLR, IYTTLR, 0.0, 180.0
          , THTMIN, THTMAX, 0, JMAX+2, PHIXD
          , THTX, LR1, LCMTX )
C
000059          4200 CONTINUE
C
000060          GO TO 3900
C
C*** PHI = CONSTANT ***
C
000061          3600 CONTINUE
          THTMIN = 0.0
          THTMAX = 0.99
          ITITLP = 'DIMENSIONLESS TEMP. PHI=CONST.'
          IXTTLP = 'RADIUS (R(J)/RIN)'
          IYTTLP = 'THETA'
C
          DO 4400 LP1 = 1, NP
          WRITE(CHRXP(LP1), '(F5.1)') CHRXP(LP1)
          CALL VALFIX( CHRXP(LP1), THTX, IMAXX2, 2 )
C
          IFLM = 0
          IF( LP1 .EQ. NP ) IFLM = -1
          IF( LP1 .EQ. 1 ) IFLM = 1
          LCMTX = CHRXP/CHRXP(LP1)
          CALL PXCNTL( IFLM, ITITLP, IXTTLP, IYTTLP
          , RR(1), RR(KMAX+1)-0.1
          , THTMIN, THTMAX-0.1, 0, IMAX+2, RX(2)
          , THTX, LP1, LCMTX )
C
000075          4400 CONTINUE
C
C
C
000076          3900 CONTINUE
000077          CALL PLOT(0.0, 0.0, 999)
C
000078          RETURN
C
000079          END

```

```

000001      SUBROUTINE      VALFIX( DATAXX,THTXX,MMAX,IFLG )      00000100
C                                                    00000200
C*****                                                    00000300
C*                                                    *00000400
C*      FUNCTION ;      *00000500
C*      *00000600
C*      GET THE MOST REASONABLE PHI VALUE      *00000700
C*      *00000800
C*      DATE      ; 84/03/30      *00000900
C*      *00001000
C*      ARGUMENTS;      (TYPE,I/O,UNIT) *00001100
C*      *00001200
C*      (1) DATAXX --- INPUT R OR PHI DATA      (R*4,I,- OR *00001300
C*      RAD )      *00001400
C*      *00001500
C*      (2) THTXX --- OUTPUT THETA DATA      (R*4,O,-) *00001600
C*      (MMAX)      *00001700
C*      *00001800
C*      (3) MMAX --- DATA MAX DIMENSION      (I*4,I) *00001900
C*      ( IMAXX2 OR JMAXX2 )      *00002000
C*      *00002100
C*      (4) IFLG --- DATA SELECT CODE      (R*4,I,-) *00002200
C*      =1 ; R =CONST.      *00002300
C*      =2 ; PHI=CONST.      *00002400
C*      *00002500
C*****                                                    00002600
C                                                    00002700
C*INCLUDE PARAMF,FIXED,INSOURCE      00002800
000002      PARAMETER      ( IMAXX = 100 ,IMAXX2 = IMAXX + 2 )      00000010
000003      PARAMETER      ( IMAXX3 = IMAXX + 3 )      00000020
000004      PARAMETER      ( IMAXX4 = IMAXX + 4 )      00000030
000005      PARAMETER      ( JMAXX = 100 ,JMAXX2 = JMAXX + 2 )      00000040
000006      PARAMETER      ( I2J2 = IMAXX2 * JMAXX2 )      00000050
000007      PARAMETER      ( LMAX = 100 ,LMAX1 = LMAX + 1 )      00000060
C                                                    00002900
C*INCLUDE COMDSP,FIXED,INSOURCE      00003000
000008      COMMON /DSPDT/ RX(IMAXX3) ,PHIX(JMAXX2) ,RR(LMAX1)      00000010
      ,THT(IMAXX4,JMAXX2),PSI(IMAXX3,JMAXX2)      00000020
      ,IMAX,JMAX,KMAX      00000030
C                                                    00003100
000009      REAL*4 THTXX(MMAX)      00003200
000010      REAL*4 PHIXD(JMAXX2)      00003300
C                                                    00003400
C***** PROCEDURE *****      00003500
C                                                    00003600
C**** R = CONSTANT ****      00003700
C                                                    00003800
000011      IF( IFLG .EQ. 1 ) THEN      00003900
000012      RXX = DATAXX      00004000
000013      IF( RXX .EQ. RX(2) ) THEN      00004100
000014      IIX = 1      00004200
C                                                    00004300
000015      ELSEIF( RXX .EQ. RX(IMAX+3) ) THEN      00004400
000016      IIX = IMAX+2      00004500
C                                                    00004600
000017      ELSE      00004700
000018      DO 4200 I = 3,IMAX+2      00004800
000019      IF( RXX .LE. RX(I) ) THEN      00004900
000020      RXM = RXX - RX(I-1)      00005000
000021      RXP = RX(I) - RXX      00005100
000022      IF( RXM .GT. RXP ) THEN      00005200
000023      IIX = I - 1      00005300
000024      GO TO 3100      00005400
000025      ELSE      00005500
000026      IIX = I - 2      00005600
000027      GO TO 3100      00005700
000028      ENDIF      00005800
000029      ENDIF      00005900
000030      4200 CONTINUE      00006000
C                                                    00006100
000031      3100 CONTINUE      00006200
000032      ENDIF      00006300
C                                                    00006400
C*** THETA SETTING ***      00006500
C                                                    00006600
000033      II = IIX      00006700
000034      DO 4300 JJ = 1,JMAX+2      00006800
000035      THTXX(JJ) = THT(II,JJ)      00006900
000036      4300 CONTINUE      00007000
C                                                    00007100

```



```

C**** PHI = CONSTANT ****
C
000037     ELSE
000038     PI      = 3.141592654
000039     PIH     = PI/2.
000040     DGTORD  = PI/180.
000041     PHIXX   = DATAXX*DGTORD
000042     DO 4400 JJ = 1,JMAX+2
000043         PHIXD(JJ) = PHIX(JJ)+PIH
000044 4400 CONTINUE
000045     IF( PHIXX .EQ. PHIXD(1) ) THEN
000046         JJY = 1
C
000047     ELSEIF( PHIXX .EQ. PHIXD(JMAX+2) ) THEN
000048         JJY = JMAX+2
C
000049     ELSE
000050         DO 4000 J = 2,JMAX+1
000051             IF( PHIXX .LE. PHIXD(J) ) THEN
000052                 PHIXM = PHIXX - PHIXD(J-1)
000053                 PHIXP = PHIXD(J) - PHIXX
000054                 IF( PHIXM .GT. PHIXP ) THEN
000055                     JJY = J
000056                                     GO TO 3000
000057                 ELSE
000058                     JJY = J - 1
000059                                     GO TO 3000
000060                 ENDIF
000061             ENDIF
000062 4000 CONTINUE
C
000063 3000 CONTINUE
000064     ENDIF
C
C**** THETA SETTING ***
C
000065     JJ = JJY
000066     DO 4100 II = 1,IMAX+2
000067         THTXX(II) = THT(II+2,JJ)
000068 4100 CONTINUE
C
000069     ENDIF
C
000070     RETURN
C
000071     END
00007200
00007300
00007400
00007500
00007600
00007700
00007800
00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00009400
00009500
00009600
00009700
00009800
00009900
00010000
00010100
00010200
00010300
00010400
00010500
00010600
00010700
00010800
00010900
00011000
00011100
00011200
00011300
00011400
00011500
00011600
00011700

```

7. 計算結果

本計算コードを用いて、2層モデルの流れ場と温度分布を計算した例を紹介する。入力データは次のとおりである。

1) 制御データカード： IOPT 1 = 1

2) 形状データカード：

断熱層数	KMAX = 2
ϕ 方向分割総数	IMAX = 50
内外壁半径比 (r_o/r_i)	RMAX = 2

3) 層パラメータカード (各1枚)：

第1層：R方向分割総数	JMAX = 30
境界半径比	R = 1.0
ラーリー数	RA = 1.0×10^8
ダルシー数	DA = 1.0×10^{-8}
温度伝導率比	AM = 1.0
第2層：R方向分割総数	JMAX = 20
境界半径比 (r_2/r_i)	R = 1.6
ラーリー数	RA = 1.0×10^8
ダルシー数	DA = 5.0×10^{-6}
温度伝導率比 (a_{m2}/a_{m1})	AM = 1.0

4) 収束計算パラメータカード

反復回数の制限値	ITMAX = 1000
$\psi_{i,j}$ の収束判定値 (ϵ_1)	ESP 1 = 1.0×10^{-5}
$\theta_{i,j}$ の収束判定値 (ϵ_2)	ESP 2 = 1.0×10^{-5}

上記パラメータを用いて、流れ関数 $\psi_{i,j}$ 及び無次元温度 $\theta_{i,j}$ を計算した結果を、図形処理プログラムを用いて作図する。入力データは次のとおりである。

1) 流れ関数の等高線図及び鳥かん図用カード： A

2) 無次元温度の等高線図及び鳥かん図用カード： B

3) 速度ベクトル図用カード： C

4) 内外壁実効熱伝導率の ϕ 方向分布用カード： D

5) ϕ 方向温度分布用カード： E 及び出力個数 = 4, 半径比 (r/r_i) R = 1.2, 1.4, 1.6, 1.8

6) R 方向温度分布用カード： F 及び出力個数 = 5, 角度 $\phi = 30, 60, 90, 120, 150$ (単位 degree)

図 7.1 及び 7.2 に流れ関数の等高線図及び鳥かん図を示す。これらの図に示される流れ関数は、内

外壁で1.0, 流れ関数が最小となる点で0となるように正規化されたものである。鳥かん図については、見易いように上に凸の三次元図に変換してある。なお、図中の AVERAGE RA*DA 及び AVERAGE AM は、それぞれ断熱層の平均 Ra*Da 及び平均熱伝導率比を表す。等高線図に見られるように、外層で自然対流による強い流れを生じており、それが境界面を通過して内層にまで流入している様子が示されている。また、自然対流による流れは頂部付近で特に強いことが分かる。

図7.3及び7.4に無次元温度の等高線図及び鳥かん図を示す。鳥かん図で内壁側に見られる幾つものピークは、等高線を描く際の演算過程で生じたノイズによる。等高線図において、頂部付近に温度分布の大きなゆがみが見られ、自然対流が温度場に強く影響していることが分かる。

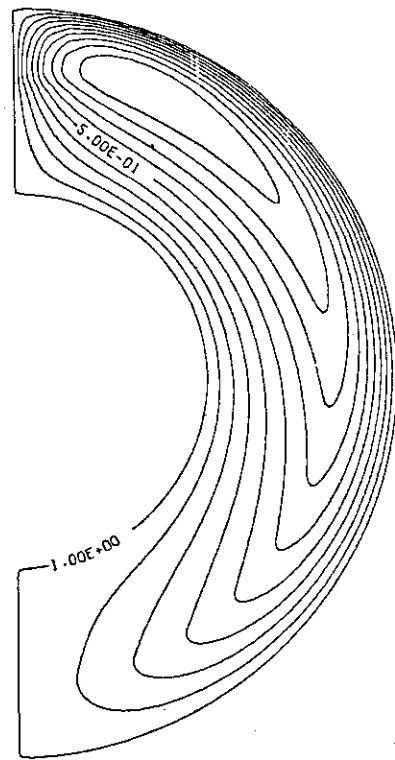
図7.5に速度ベクトル図を示す。矢印は、正規化した流れ関数より算出した速度を表しており、矢印が長いほど速度は大きい。断熱層内を循環する流れの挙動が明りょうに示されている。

図7.6に内外壁実効熱伝導率分布を示す。実線が内壁における実効熱伝導率比 (λ_e / λ_{m1}) の分布を表しており、破線が外壁における (λ_e / λ_{m2}) の分布を表している。外壁側の実効熱伝導率比 (λ_e / λ_{m2}) が頂部付近で急上昇しており、熱の移動が特に大きいことを示している。

図7.7に ϕ 方向温度分布を示す。図中の各線は、それぞれの半径比Rの位置での無次元温度分布を表している。断熱層の底部付近では僅かな温度上昇が見られるが、 ϕ が増加して頂部に近づくにつれ、急速に温度が上昇し、内壁温度に近くなっている。

図7.8にR方向温度分布を示す。図中の各線は、それぞれの角度 ϕ の位置での温度分布を表している。内層側の温度勾配は外層に比べかなり大きい。しかし、頂部付近の $\phi = 150^\circ$ では、自然対流の影響が強くなるため、外壁近傍で大きな温度勾配を生じている。

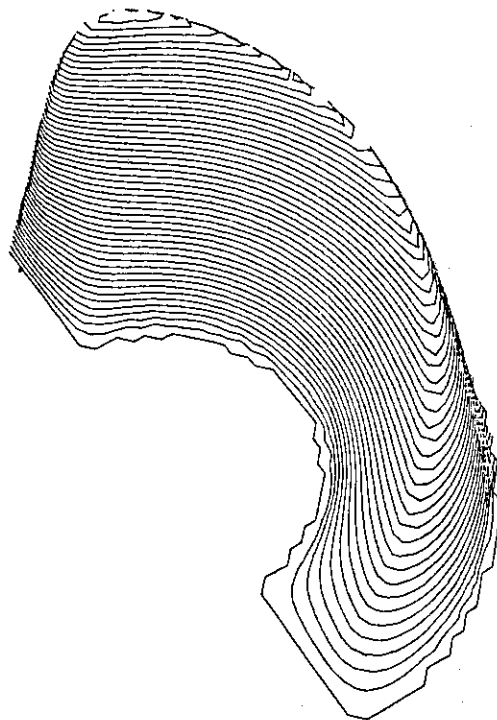
計算結果の詳細については次報で述べる予定である。



LAYER NO.	R	RA*DA	AM
1	1.0	1.0	1.0
2	1.6	500.0	1.0

ROUT/RIN : 2.0
 AVARAGE RA*DA : 240.5
 AVARAGE AM : 1.0

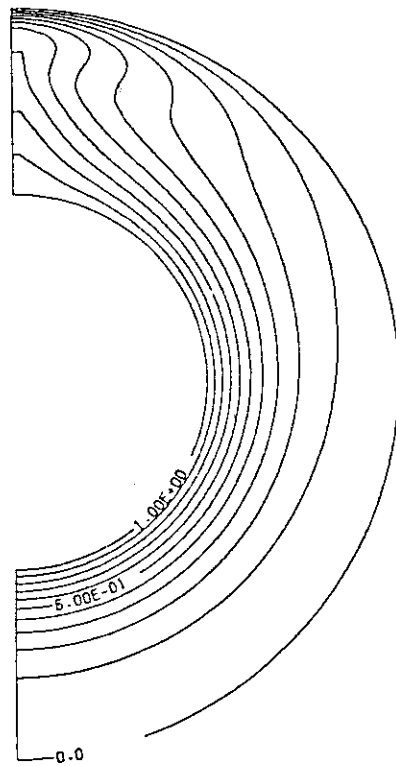
図 7.1 流れ関数の等高線図



LAYER NO.	R	RA*DA	AM
1	1.0	1.0	1.0
2	1.6	500.0	1.0

ROUT/RIN : 2.0
 AVARAGE RA*DA : 240.5
 AVARAGE AM : 1.0

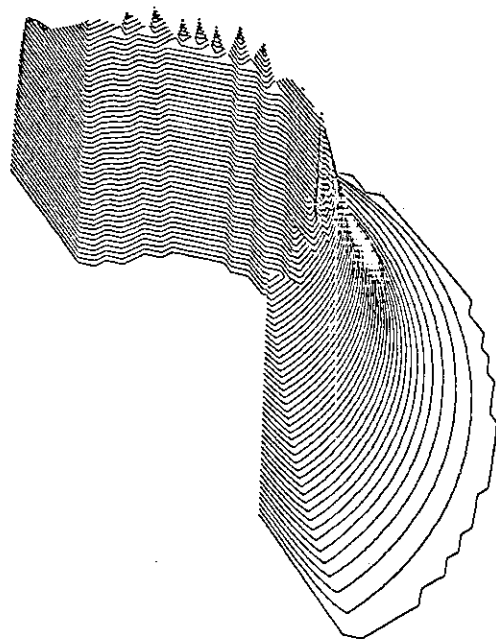
図 7.2 流れ関数の鳥かん図



LAYER NO.	R	RA*DA	AM
1	1.0	1.0	1.0
2	1.6	500.0	1.0

ROUT/RIN : 2.0
 AVARAGE RA*DA : 240.5
 AVARAGE AM : 1.0

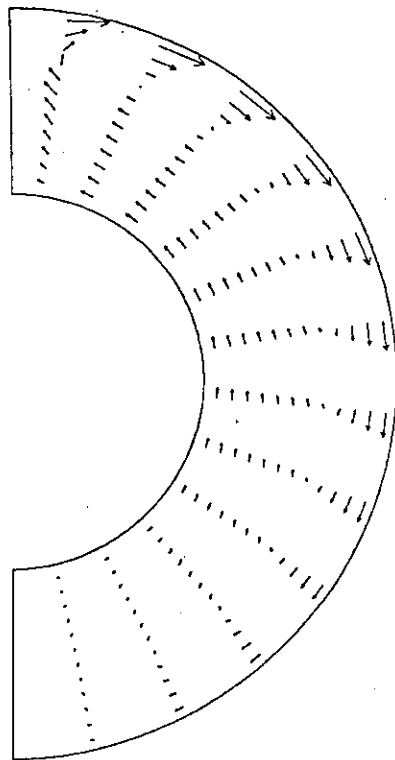
図 7.3 無次元温度の等高線図



LAYER NO.	R	RA*DA	AM
1	1.0	1.0	1.0
2	1.6	500.0	1.0

ROUT/RIN : 2.0
 AVARAGE RA*DA : 240.5
 AVARAGE AM : 1.0

図 7.4 無次元温度分布の鳥かん図



LAYER NO.	R	RA*DA	AM
1	1.0	1.0	1.0
2	1.6	500.0	1.0

ROUT/RIN : 2.0
 AVERAGE RA*DA : 240.5
 AVERAGE AM : 1.0

図 7.5 速度ベクトル図

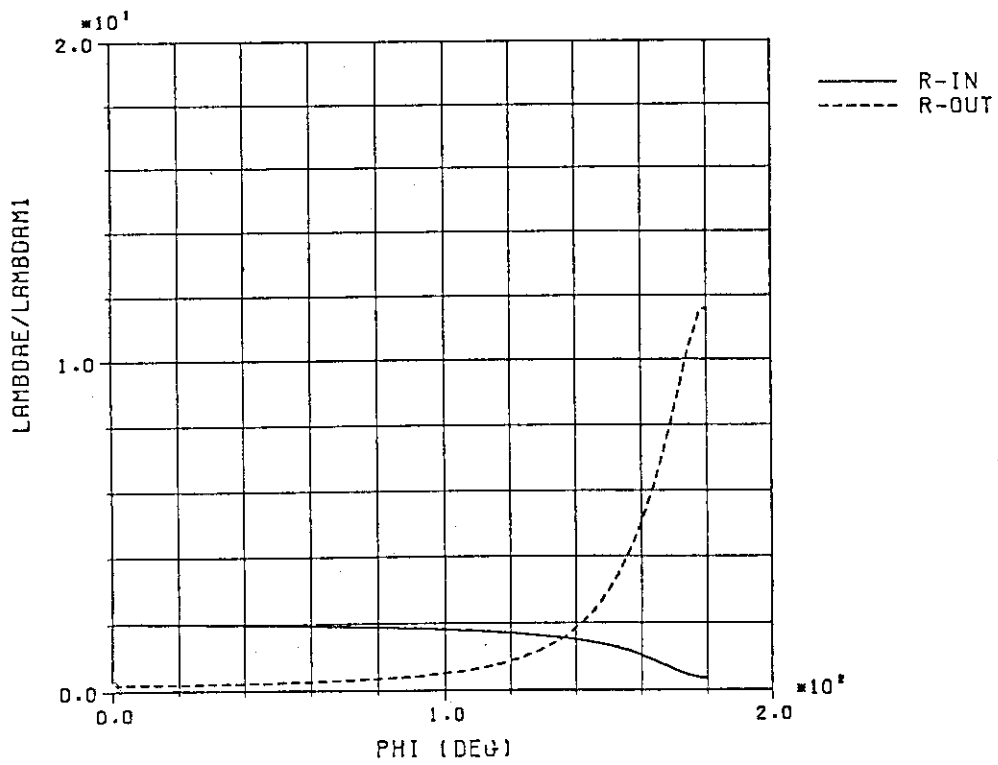


図 7.6 実効熱伝導率分布

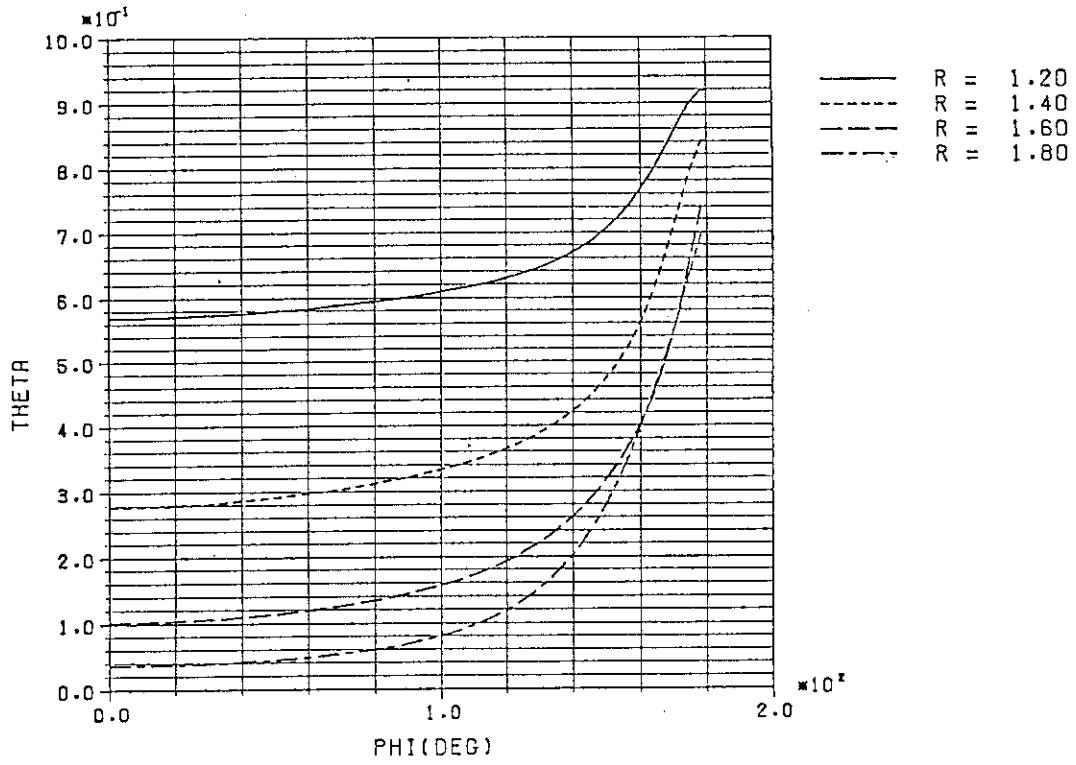


图 7.7 角度方向温度分布

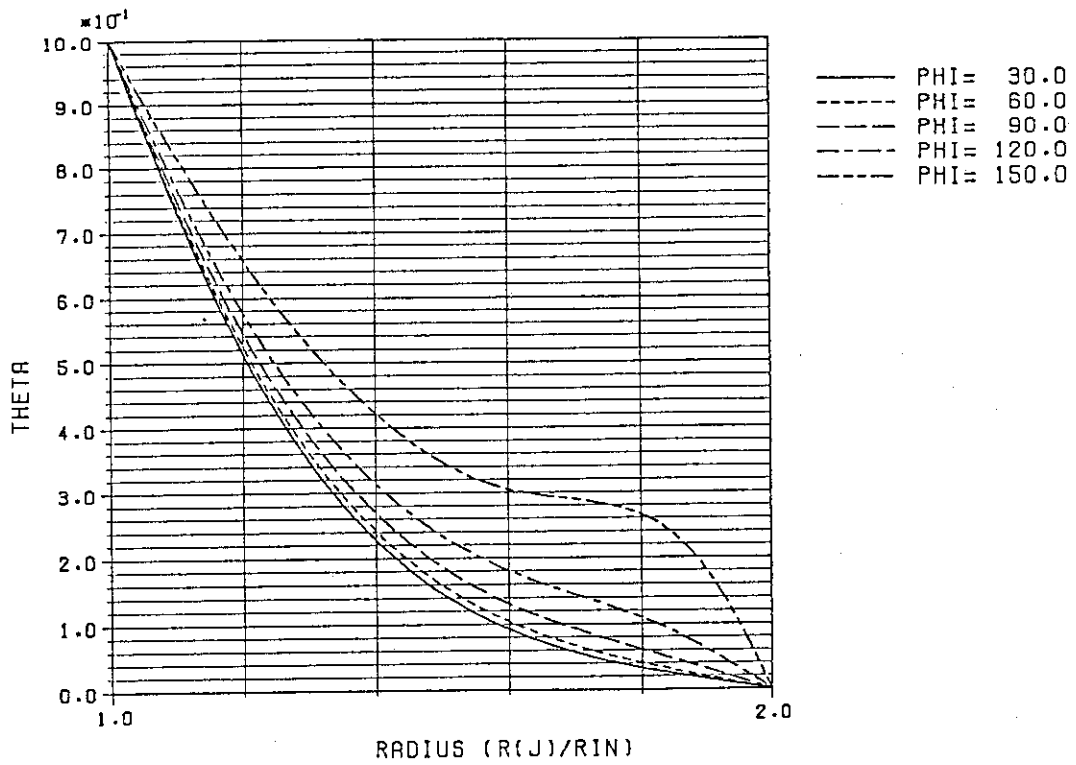


图 7.8 半径方向温度分布

参 考 文 献

- (1) 下村寛昭, 岡本芳三: 日本原子力学会誌, Vol. 20, No.8, P. 583-590 (1978).
- (2) 越後亮三ほか4名: 日本機械学会論文集 (B編), Vol. 45, No.389, P. 91-98 (1979).
- (3) 増岡隆士, 坂本直樹, 勝原哲治: 日本機械学会論文集 (B編), Vol. 46, No.405, P. 919-926 (1980).
- (4) J. P. Caltagirone: J. Fluid Mech., Vol. 76, part 2, P. 337-362 (1976).