

ISSN—0386—5878

土研資料第1940号

マイクロフィルム第 号

土木研究所資料

有限要素法による河川分流計算プログラム

昭和58年3月

建設省土木研究所

河川研究室

Copyright © 1982 fiscal year by P.W.R. I.

All rights reserved. No part of this book may be reproduced by any means, nor transmitted, nor transland into a machine language without the written permission of the Director of P.W.R.I.

この報告書は、土木研究所長の承認を得て刊行したものである。
したがって、本報告書の全部又は一部の転載、複写は土木研究所長の承認を得ずしてこれを行ってはならない。

は じ め に

河川流の水利計算は断面平均化された諸量についての一次元計算が主体である。実際、地図で見れば川は紐のようなものであり、一次元計算で十分な場合も多いと思われる。しかし時には河道内の流れの平面的な挙動が重要な場合もある。例えば分流部，合流部の水利設計を行う場合などである。

流れの平面的な挙動を知る手段として水利模型実験がある。相似条件に注意さえすれば非常に精密な情報を得ることが可能である。しかし一般に水利模型実験には多額の経費と労力とそして時間がかかる。特に設計の自由度が大きい場合にはよほど熟練した技術者でない限り最良案を得るまでに数年を要する。著者は江戸川流頭部水利模型実験においてこのことを痛感した。同実験は緩流分派点の形状を定めることを目的としているが、緩流であるがゆえに分流量を調整するために実に広大な面積の改造を必要とし、改造経費と改造時間がネックとなって思うように進められなかったのである。

そこで概略でもよいから流れの平面的な特性を手軽に知る手段を得たいと考え、平面流計算プログラムの開発に踏み切った。本プログラムは一応分流を対象としてはいるが境界条件の種類を自由に選択できるようになっているので、河川流一般に適用可能である。

昭和 58 年 3 月

河川研究室 須 賀 堯 三
石 川 忠 晴

目 次

は じ め に	
第1章 浅水流基礎方程式の有限要素法による定式化	1
1. 基礎方程式	1
2. Galerkin 法による有限要素定式化	2
3. 要素方程式	5
4. 境界条件	7
5. アイソパラメトリック四辺形要素	11
6. 初期条件について	13
7. 収束の判定	14
第2章 プログラム説明	17
1. プログラムの概要	17
2. グローバル・コモン	21
3. メモリ・アロケーション	23
4. 各モジュール説明	24
第3章 使用説明	28
第4章 テスト・ラン	37
1. 一次元流への適用によるチェック	37
2. 江戸川流頭部水理計算	38
お わ り に	48
付録1 矩形断面水路の一次元不等流方程式厳密解	51

第1章 浅水流基礎方程式の有限要素法による定式化

1. 基礎方程式

河川流は一般に水深に較べて幅が非常に広く，また重力によってエネルギーが補給されるので，河岸等の段差付近を除けば鉛直一層の平面的な流れとして近似できる。そこで次に示す浅水流の運動方程式と連続式を基礎方程式とする。

x 方向運動方程式

$$\begin{aligned} \frac{\partial u}{\partial t} + \frac{\partial}{\partial x} \left(\frac{u^2}{H} \right) + \frac{\partial}{\partial y} \left(\frac{uv}{H} \right) \\ = -gH \frac{\partial \eta}{\partial x} + \frac{\partial}{\partial x} \left(2\epsilon \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left\{ \epsilon \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right\} - \frac{g}{C^2} \frac{u(u^2 + v^2)^{1/2}}{H^2} \end{aligned} \quad \dots\dots (1)$$

y 方向運動方程式

$$\begin{aligned} \frac{\partial v}{\partial t} + \frac{\partial}{\partial x} \left(\frac{uv}{H} \right) + \frac{\partial}{\partial y} \left(\frac{v^2}{H} \right) \\ = -gH \frac{\partial \eta}{\partial y} + \frac{\partial}{\partial y} \left(2\epsilon \frac{\partial v}{\partial y} \right) + \frac{\partial}{\partial x} \left\{ \epsilon \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right\} - \frac{g}{C^2} \frac{v(u^2 + v^2)^{1/2}}{H^2} \end{aligned} \quad \dots\dots (2)$$

連続条件式

$$\frac{\partial \eta}{\partial t} + \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad \dots\dots\dots (3)$$

ここに，

$u = x$ 方向の単位巾あたりの流量

$v = y$ 方向の単位巾あたりの流量

$\eta =$ 水位 ($\eta = H +$ 河床高)

$H =$ 水深

$\epsilon =$ 渦動粘性係数

$C =$ シェジー係数

$g =$ 動力加速度

本研究の目的は流れの定常解を求めることにある。したがって時間微分項を落した方程式を解くのが普通であるが，従来発表されているその種の研究では収束性が非常に悪いようである。そこで本研究では時間微分項を残し，適当な初期条件から解の時間変化を追跡して定常解に収束させる方法をとる。

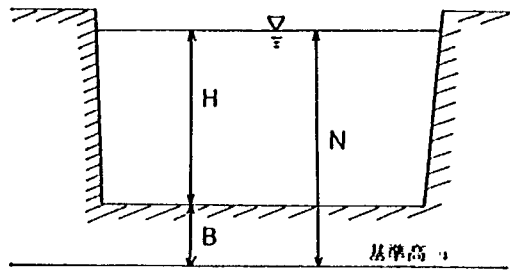


図1-1 水位と水深

2. Galerkin 法による有限要素定式化

(1)～(3)式は u , v , η に関する非線形偏微分方程式である。これを Galerkin 法で数値的に解くことにする。解くべき領域を D , 境界を S で表わし, D を有限要素に分割してその代表要素 e について式を立てる。要素内の未知量 u , v , η を要素を構成する節点 i での未知量 u_i , v_i , η_i で次のように表わす。

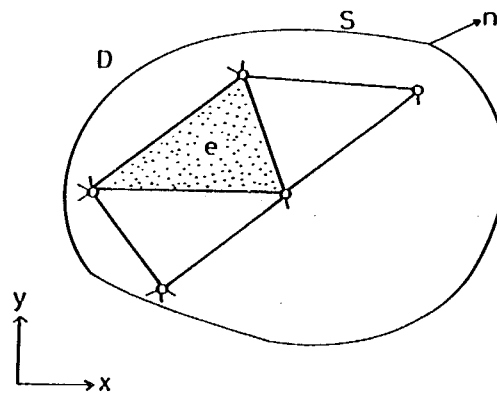


図1-2 領域と有限要素

$$u = \sum_{i=1}^m N_i u_i = \mathbf{N} \mathbf{u} \quad \dots\dots\dots (4)$$

$$v = \sum_{i=1}^m N_i v_i = \mathbf{N} \mathbf{v} \quad \dots\dots\dots (5)$$

$$\eta = \sum_{i=1}^m N_i \eta_i = \mathbf{N} \boldsymbol{\eta} \quad \dots\dots\dots (6)$$

ただし

$$\mathbf{N} = [N_1, N_2, \dots, N_m] \quad \dots\dots\dots (7)$$

$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_m \end{bmatrix} \quad \mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_m \end{bmatrix} \quad \boldsymbol{\eta} = \begin{bmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_m \end{bmatrix} \quad \dots\dots\dots (8)$$

ここに N_i は補完関数, m は要素を構成する節点数で, 例えば三角形要素では $m=3$ となる。 N_i を重み関数として, 重みをかけた平均的な意味で (1) ~ (3) 式を満足させるとすれば次式を得る。

$$\int_{D_e} N_i \left[\frac{\partial u}{\partial t} + gH \frac{\partial \eta}{\partial x} + F_x^L + F_x^N \right] dD = 0 \quad \dots\dots\dots (9)$$

$$\int_{D_e} N_i \left[\frac{\partial v}{\partial t} + gH \frac{\partial v}{\partial t} + F_y^L + F_y^N \right] dD = 0 \quad \dots\dots\dots (10)$$

$$\int_{D_e} N_i \left[\frac{\partial \eta}{\partial t} + \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right] dD = 0 \quad \dots\dots\dots (11)$$

ここに,

$$F_x^L = -\frac{\partial}{\partial x} \left(2\epsilon \frac{\partial u}{\partial x} \right) - \frac{\partial}{\partial y} \left\{ \epsilon \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right\} \quad \dots\dots\dots (12)$$

$$F_x^N = \frac{\partial}{\partial x} \left(\frac{u^2}{H} \right) + \frac{\partial}{\partial y} \left(\frac{uv}{H} \right) + \frac{g}{C^2} \frac{u(u^2+v^2)^{1/2}}{H^2} \quad \dots\dots\dots (13)$$

$$F_y^L = -\frac{\partial}{\partial y} \left(2\epsilon \frac{\partial v}{\partial y} \right) - \frac{\partial}{\partial x} \left\{ \epsilon \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right\} \quad \dots\dots\dots (14)$$

$$F_y^N = \frac{\partial}{\partial x} \left(\frac{uv}{H} \right) + \frac{\partial}{\partial y} \left(\frac{v^2}{H} \right) + \frac{g}{C^2} \frac{v(u^2+v^2)^{1/2}}{H^2} \quad \dots\dots\dots (15)$$

ただし $\int_{D_e} dD$ は要素 e での面積分を表わす。

(4), (5), (6) 式を (9) 式に代入すると各項は以下ようになる。

第 1 項

$$\int_{D_e} N_i \frac{\partial u}{\partial t} dD = \int_{D_e} N_i \frac{\partial u}{\partial t} dD \quad \dots\dots\dots (16)$$

第 2 項

$$\int_{D_e} N_i gH \frac{\partial \eta}{\partial x} dD = g \int_{D_e} N_i H \frac{\partial \eta}{\partial x} dD \quad \dots\dots\dots (17)$$

第 3 項

$$\int_{D_e} N_i F_x^L dD = - \int_{D_e} N_i \left[\frac{\partial}{\partial x} \left(2\epsilon \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left\{ \epsilon \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right\} \right] dD$$

(Green の定理を用いて)

$$\begin{aligned} &= - \int_S N_i \left[2\epsilon \frac{\partial u}{\partial x} \cos(n, x) + \epsilon \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \cos(n, y) \right] dS \\ &\quad + \int_{D_e} \left[\frac{\partial N_i}{\partial x} 2\epsilon \frac{\partial u}{\partial x} + \frac{\partial N_i}{\partial y} \epsilon \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right] dD \end{aligned}$$

(4), (5)式を用いて)

$$= - \int_s N_i \left[2\epsilon \frac{\partial u}{\partial x} \cos(n, x) + \epsilon \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \cos(n, x) \right] dS \\ + \int \epsilon \left[\left(2 \frac{\partial N_i}{\partial x} \frac{\partial N}{\partial x} + \frac{\partial N_i}{\partial y} \frac{\partial N}{\partial y} \right) u + \frac{\partial N_i}{\partial y} \frac{\partial N}{\partial x} w \right] dD \quad \dots\dots\dots (18)$$

ここに, n は境界での外向き法線方向を, また $\cos(n, x)$, $\cos(n, y)$ は各々 n と x および y 軸との方向余弦を表わす。

(16) ~ (18) 式を (9) 式に代入すると次式を得る。

$$\int_{D'} N_i N dD u + \int_{D'} \epsilon \left[2 \frac{\partial N_i}{\partial x} \frac{\partial N}{\partial x} + \frac{\partial N_i}{\partial y} \frac{\partial N}{\partial y} \right] dD u + \int_{D'} \epsilon \frac{\partial N_i}{\partial y} \frac{\partial N}{\partial x} dD w \\ + g \int_{D'} N_i H \frac{\partial N}{\partial x} dD \eta - \int_s \left[N_i \left\{ 2\epsilon \frac{\partial u}{\partial x} \cos(n, x) + \epsilon \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \cos(n, y) \right\} \right] ds \\ + \int_{D'} N_i F_x^N dD = 0 \quad \dots\dots\dots (19)$$

また(10)式に代入すると次式を得る。

$$\int_{D'} N_i N dD v + \int_{D'} \epsilon \left[\frac{\partial N_i}{\partial x} \frac{\partial N}{\partial x} + 2 \frac{\partial N_i}{\partial y} \frac{\partial N}{\partial y} \right] dD v + \int_{D'} \epsilon \frac{\partial N_i}{\partial x} \frac{\partial N}{\partial y} dD u \\ + g \int_{D'} N_i H \frac{\partial N}{\partial y} dD \eta - \int_s N_i \left\{ 2\epsilon \frac{\partial v}{\partial y} \cos(n, y) + \epsilon \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \cos(n, x) \right\} ds \\ + \int_{D'} N_i F_y^N dD = 0 \quad \dots\dots\dots (20)$$

一方連続条件式については, (11) 式に (4) ~ (6) 式を代入することにより,

$$\int N_i N dD \eta + \int N_i \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) dD = 0 \quad \dots\dots\dots (21)$$

ここで左辺第2項に Green の定理を用いると,

$$\int_{D'} N_i \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) dD \\ = \int_s N_i \left(u \cdot \cos(n, x) + v \cdot \cos(n, y) \right) ds - \int_{D'} \left(\frac{\partial N_i}{\partial x} u + \frac{\partial N_i}{\partial y} v \right) dD \\ = \int_s N_i u_n ds - \int_{D'} \frac{\partial N_i}{\partial x} N dD u - \int_{D'} \frac{\partial N_i}{\partial y} N dD v \quad \dots\dots\dots (22)$$

(22) 式を (21) 式に代入すると次式を得る。

$$\int_D N_i N dD \dot{\eta} - \int_D \frac{\partial N_i}{\partial x} N dD u - \int_D \frac{\partial N_i}{\partial y} N dD v + \int_S N_i u_n ds = 0 \quad \dots\dots (23)$$

3. 要素方程式

(19), (20), (23) 式は節点での未知ベクトル u , v , η に関する非線形の連立常微分方程式である。これを後退差分によって時間方向に離散化する。時刻 t までの量が既知で $t + \Delta t$ の時刻の諸量を解くものとする

$$\dot{u}^{K+1} = (u^{K+1} - u^K) / \Delta t \quad \dots\dots\dots (24)$$

$$\dot{v}^{K+1} = (v^{K+1} - v^K) / \Delta t \quad \dots\dots\dots (25)$$

$$\dot{\eta}^{K+1} = (\eta^{K+1} - \eta^K) / \Delta t \quad \dots\dots\dots (26)$$

一方 (19), (20) 式の左辺第 5 項は, (4), (5) 式から,

$$\begin{aligned} & \int_S N_i \left\{ 2\epsilon \frac{\partial u}{\partial x} l_x + \epsilon \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) l_y \right\} ds \\ &= \int_S N_i \left\{ 2\epsilon \frac{\partial N}{\partial x} l_x + \epsilon \frac{\partial N}{\partial y} l_y \right\} ds u + \int_S \epsilon l_y N_i \frac{\partial N}{\partial x} ds v \\ &= \int_S \epsilon \left\{ 2l_x N_i \frac{\partial N}{\partial x} + l_y N_i \frac{\partial N}{\partial y} \right\} ds u + \int_S \epsilon l_y N_i \frac{\partial N}{\partial x} ds v \quad \dots\dots\dots (27) \end{aligned}$$

$$\begin{aligned} & \int_S N_i \left\{ 2\epsilon \frac{\partial v}{\partial y} l_y + \epsilon \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) l_x \right\} ds \\ &= \int_S N_i \left\{ 2\epsilon \frac{\partial N}{\partial y} l_y + \epsilon \frac{\partial N}{\partial x} l_x \right\} ds v + \int_S N_i \epsilon \frac{\partial N}{\partial y} l_x ds u \\ &= \int_S \epsilon \left\{ 2l_y N_i \frac{\partial N}{\partial y} + l_x N_i \frac{\partial N}{\partial x} \right\} ds v + \int_S \epsilon l_x N_i \frac{\partial N}{\partial y} ds u \quad \dots\dots\dots (28) \end{aligned}$$

また (23) 式の左辺第 4 項は,

$$\begin{aligned} \int_S N_i u_n ds &= \int_S N_i \{ u l_x - v l_y \} ds \\ &= \int_S N_i N l_x ds u + \int_S N_i N l_y ds v \quad \dots\dots\dots (29) \end{aligned}$$

ただし $l_x = \cos(n, x)$, $l_y = \cos(n, y)$ である。(19), (20) 式の第 6 項については部分積分を行って,

$$\begin{aligned} & \int_D N_i F_x^N dD \\ &= \int_D N_i \left\{ \frac{\partial}{\partial x} \left(\frac{u^2}{H} \right) + \frac{\partial}{\partial y} \left(\frac{uv}{H} \right) + \frac{g}{C^2} \frac{u(u^2 - v^2)^{1/2}}{H^2} \right\} dD \\ &= \int_S N_i \left(\frac{u^2}{H} \right) l_x ds + \int_S N_i \left(\frac{uv}{H} \right) l_y ds - \int_D \left[\frac{\partial N_i}{\partial x} \frac{u^2}{H} + \frac{\partial N_i}{\partial y} \left(\frac{uv}{H} \right) \right] dD \\ & \quad + \int_D N_i \frac{g}{C^2} \frac{(u^2 + v^2)^{1/2}}{H^2} N dD u \end{aligned}$$

$$\begin{aligned}
&= \int_s N_i \frac{u}{H} l_x N ds u + \int_s N_i \frac{v}{H} l_y N ds u - \int_s \left[\frac{\partial N_i}{\partial x} \frac{u}{H} + \frac{\partial N_i}{\partial y} \frac{v}{H} \right] N dD u \\
&\quad + \int_D N_i \frac{g}{C^2} \frac{(u^2 + v^2)^{1/2}}{H^2} N dD u \\
&= \int_s \left(\frac{u}{H} l_x + \frac{v}{H} l_y \right) N_i N ds u - \int_D \left[\frac{\partial N_i}{\partial x} \frac{u}{H} + \frac{\partial N_i}{\partial y} \frac{v}{H} \right] N dD u \\
&\quad + \int_D \frac{g}{H} \frac{(u^2 + v^2)^{1/2}}{H^2} N_i N dD u \quad \dots\dots\dots (30)
\end{aligned}$$

$$\begin{aligned}
&\int N_i F_y^N dD \\
&= \int N_i \left\{ \frac{\partial}{\partial x} \left(\frac{uv}{H} \right) + \frac{\partial}{\partial y} \left(\frac{v^2}{H} \right) + \frac{g}{C^2} \frac{v(u^2 + v^2)^{1/2}}{H^2} \right\} dD \\
&= \int N_i \left(\frac{uv}{H} \right) l_x ds + \int N_i \left(\frac{v^2}{H} \right) l_y ds - \int \left[\frac{\partial N_i}{\partial x} \frac{uv}{H} + \frac{\partial N_i}{\partial y} \frac{v^2}{H} \right] dD \\
&\quad + \int N_i \frac{g}{C^2} \frac{(u^2 + v^2)^{1/2}}{H^2} N dD v \\
&= \int_s N_i \frac{u}{H} l_x N ds v + \int_s N_i \frac{v}{H} l_y N ds v - \int_D \left[\frac{\partial N_i}{\partial x} \frac{u}{H} + \frac{\partial N_i}{\partial y} \frac{v}{H} \right] N dD v \\
&\quad + \int_D N_i \frac{g}{C^2} \frac{(u^2 + v^2)^{1/2}}{H^2} N dD v \\
&= \int_s \left(\frac{u}{H} l_x + \frac{v}{H} l_y \right) N_i N ds v - \int_D \left(\frac{u}{H} \frac{\partial N_i}{\partial x} + \frac{v}{H} \frac{\partial N_i}{\partial y} \right) N dD v \\
&\quad + \int_D \frac{g}{C^2} \frac{(u^2 + v^2)^{1/2}}{H^2} N_i N dD v \quad \dots\dots\dots (31)
\end{aligned}$$

(24)(25)(26) 式を (19), (20), (23) 式に代入し, (27) ~ (30) 式を用いて整理すると要素方程式が得られる。

$$C^e X_{K+1}^e = P^e \quad \dots\dots\dots (32)$$

ここに,

$$X_{K+1}^e = \begin{Bmatrix} u_{K+1}^e \\ w_{K+1}^e \\ \eta_{K+1}^e \end{Bmatrix}$$

$$\begin{aligned}
C^e = & \left[\begin{array}{l} \int_0 \left\{ 1 + \Delta t \frac{g}{C^2} \frac{(u^2 + v^2)^{1/2}}{H^2} \right\} N^T N \, dD^{k+1} \\ + \Delta t \int_0 \left\{ 2 \frac{\partial N^T}{\partial x} \frac{\partial N}{\partial x} + \frac{\partial N^T}{\partial y} \frac{\partial N}{\partial y} \right\} dD \\ - \Delta t \int_0 \left\{ \frac{u}{H} \frac{\partial N^T}{\partial x} + \frac{v}{H} \frac{\partial N^T}{\partial y} \right\} N \, dD^{k+1} \\ + \Delta t \int_0 \left\{ \frac{u}{H} l_x + \frac{v}{H} l_y \right\} N^T N \, ds^{k+1} \\ - \Delta t \int_0 \left\{ 2 l_x N^T \frac{\partial N}{\partial x} + l_y N^T \frac{\partial N}{\partial y} \right\} ds \\ \Delta t \int_0 \left\{ \frac{\partial N^T}{\partial x} \frac{\partial N}{\partial y} \right\} dD \\ - \Delta t \int_0 l_x N^T \frac{\partial N}{\partial y} ds \\ - \Delta t \int_0 \frac{\partial N^T}{\partial x} N \, dD \\ + \Delta t \int_0 l_x N^T N \, ds \end{array} \right] \\
P = & \left[\begin{array}{l} \int_0 \left\{ 1 + \Delta t \frac{g}{C^2} \frac{(u^2 + v^2)^{1/2}}{H^2} \right\} N^T N \, dD^{k+1} \\ + \Delta t \int_0 \left\{ \frac{\partial N^T}{\partial x} \frac{\partial N}{\partial x} + 2 \frac{\partial N^T}{\partial y} \frac{\partial N}{\partial y} \right\} dD \\ - \Delta t \int_0 \left\{ \frac{u}{H} \frac{\partial N^T}{\partial x} + \frac{v}{H} \frac{\partial N^T}{\partial y} \right\} N \, dD^{k+1} \\ \Delta t \int_0 \left\{ \frac{u}{H} l_x + \frac{v}{H} l_y \right\} N^T N \, ds^{k+1} \\ - \Delta t \int_0 \left\{ 2 l_x N^T \frac{\partial N}{\partial y} + l_y N^T \frac{\partial N}{\partial x} \right\} ds \\ - \Delta t \int_0 \frac{\partial N^T}{\partial y} N \, dD \\ + \Delta t \int_0 l_y N^T N \, ds \end{array} \right] \\
P^* = & \left[\begin{array}{l} \int_0 N^T N \, dD \, u^A \\ \int_0 N^T N \, dD \, v^A \\ \int_0 N^T N \, dD \, \eta^A \\ - \Delta t \int_0 N^T u_x \, ds^{k+1} \end{array} \right]
\end{aligned}$$

(33) 式を領域内のすべての要素にわたって組み立てることにより全体方程式が得られる。

$$C X^{k+1} = P \quad \dots\dots\dots (34)$$

ここに、

$$C = \sum_e C^e \quad \dots\dots\dots (35)$$

$$P = \sum_e P^e \quad \dots\dots\dots (36)$$

(34) 式は節点での x 方向流量, y 方向流量, 水位 η に関する連立一次方程式になっており, 適当な境界条件のもとに解かれる。

4. 境界条件

水理的境界条件は 3 種に分類され, 各々以下のように取り扱う。

(1) 上流端境界条件

上流端断面では流量が与えられる。しかしその断面内の配分は計算区域外も含めた河道条件によって定まるものであり, 一義的には与えられない。そこで本計算プログラムでは上流端断面の各節点における流量ベクトル (u, v) を任意に与えられるようにした。すなわち,

$$(u, v) = \text{既知}$$

(2) 側岸境界条件

側岸は垂直壁とみなし法線流量ベクトルをゼロとする。

すなわち,

$$u_n = 0 \quad \dots\dots\dots (37)$$

(3) 下流端境界条件

下流端断面では一般に流量と水位の関係が与えられる。河道が大きく湾曲しない限りは横断面内の水位は一定とみなせる。したがって次のような条件式が与えられる。

$$\int_{S_i} u_n ds = f(\eta_i) \quad \dots\dots\dots (38)$$

$$\eta_1 = \eta_2 = \eta_3 = \dots\dots = \eta_m \quad \dots\dots\dots (39)$$

ここで u_n は下流端法線方向流量, η_i は下流端横断面上節点の水位である。また m は同断面内節点数である。分流計算の場合には各派川の下流端断面で各々独立に 38, 39 式を与える。

節点 i での法線流量を U_i と書くと,

$$c_1 U_1 + c_2 U_2 + \dots\dots c_m U_m = f(\eta_m) \quad \dots\dots\dots (40)$$

$f(\eta_m)$ は一般に非線形関数であり, 基礎方程式 (1), (2) 式と同じシェジャー式を用いれば

$f(\eta_m) \propto (\eta_m - B)^{3/2}$ となる。

しかしこのままの形では取り扱いにくいので, 解の近傍で展開して次式を仮定する。

$$f(\eta) = c_0 \eta + a_0 \quad \dots\dots\dots (41)$$

係数 c_0 , a_0 は, 概略の水位を予想してその近傍でシェジャー式の水位・流量関係に一致するように定める。一般に正解の水位にピッタリと一致することは無いので, 定常解が得られた後に又決めなおして再計算することになる。

さて, 38 ~ (41) 式より以下の条件式が得られる。

$$\begin{aligned} U_1 &= \tilde{c}_0 \eta_m + \tilde{c}_2 U_2 + \tilde{c}_3 U_3 + \dots\dots + \tilde{c}_m U_m + \tilde{a}_0 \\ \eta_1 &= \eta_m \\ \eta_2 &= \eta_m \quad \dots\dots\dots (42) \\ &\vdots \\ \eta_{m-1} &= \eta_m \end{aligned}$$

ここに,

$$\begin{aligned} \tilde{c}_0 &= \frac{c_0}{c_1} \\ \tilde{c}_2 &= -\frac{c_2}{c_1} \\ &\vdots \\ \tilde{c}_m &= -\frac{c_m}{c_1} \end{aligned}$$

すなわち下流端断面では $U_1, \eta_1, \dots, \eta_{m-1}$ が従属自由度に、 $U_2, \dots, U_m, \eta_m$ が独立自由度になる。

(4) 要素方程式への組みこみ

境界に接する要素 e の変数 X^e に関する制約条件は上記を合成して次のように書ける

$$X^e = A \bar{X} + B \quad \dots\dots\dots (43)$$

ここに A は制約条件長方形マトリックス、 X は独立な自由度である。(43) 式を前章の要素方程式 (32) 式に代入すると次式を得る。

$$\tilde{K}^e \bar{X} = P^e \quad \dots\dots\dots (44)$$

ここに、

$$\tilde{K}^e = A^T K A$$

$$\tilde{P}^e = A^T P^e - A^T K^e B$$

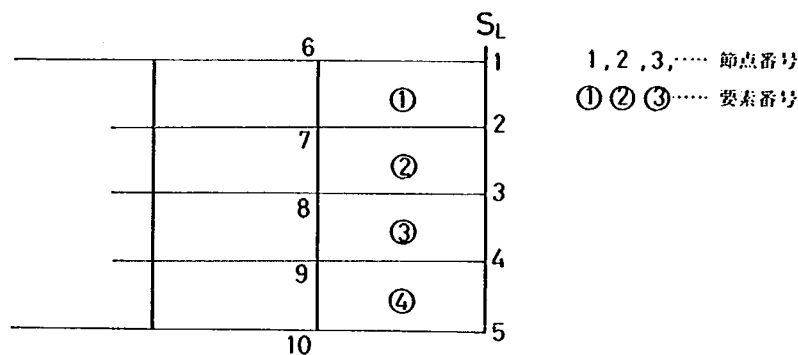


図 1-3 下流端境界条件の例

一例として図 1-3 に示す下流端断面の要素①要素②の場合を示す。ただし表記を簡単にするために流量を一次元的にとり U と書く。

要素①の場合

境界条件は次のようである。

$$\begin{matrix} X^1 \\ \left[\begin{matrix} U_1 \\ \eta_1 \\ U_6 \\ \eta_6 \\ U_7 \\ \eta_7 \\ U_2 \\ \eta_2 \end{matrix} \right] \end{matrix} = \begin{matrix} A \\ \left[\begin{matrix} \tilde{c}_2 & \tilde{c}_3 & \tilde{c}_4 & \tilde{c}_5 & 0 & 0 & 0 & 0 & \tilde{c}_0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{matrix} \right] \end{matrix} \cdot \begin{matrix} \bar{X} \\ \left[\begin{matrix} U_2 \\ U_3 \\ U_4 \\ U_5 \\ U_6 \\ \eta_6 \\ U_7 \\ \eta_7 \\ \eta_5 \end{matrix} \right] \end{matrix} \quad \dots\dots\dots (45)$$

一方要素マトリックスは次式のように書かれるので,

$$\begin{array}{c} U^e \\ \\ K^1 \\ (8 \times 8) \end{array} \begin{array}{c} \left[\begin{array}{c} U_1 \\ \eta_1 \\ U_6 \\ \eta_6 \\ U_7 \\ \eta_7 \\ U_2 \\ \eta_2 \end{array} \right] \end{array} = \begin{array}{c} P^1 \\ (8 \times 1) \end{array} \dots\dots\dots (46)$$

これに代入して $\tilde{K}^1$, $\tilde{P}^1$ が得られる。 $\tilde{K}^1$ の元数および $\tilde{P}^1$ の次数は 9 となり $\tilde{K}^1$ の元数および $\tilde{P}^1$ の次数 8 に対して 1 個増加する。

要素②の場合

同様にして次のように書かれる。

境界条件式

$$\begin{array}{c} X^2 \\ \\ \left[\begin{array}{c} U_2 \\ \eta_2 \\ U_7 \\ \eta_7 \\ U_8 \\ \eta_8 \\ U_3 \\ \eta_3 \end{array} \right] \end{array} = \begin{array}{c} A \\ \\ \left[\begin{array}{cccccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array} \right] \end{array} \begin{array}{c} \bar{X} \\ \\ \left[\begin{array}{c} U_2 \\ U_7 \\ \eta_7 \\ U_8 \\ \eta_8 \\ U_3 \\ \eta_5 \end{array} \right] \end{array} \dots\dots\dots (47)$$

要素方程式

$$K^2 \begin{array}{c} \left[\begin{array}{c} U_2 \\ \eta_2 \\ U_7 \\ \eta_7 \\ U_8 \\ \eta_8 \\ U_3 \\ \eta_3 \end{array} \right] \end{array} = P^2 \dots\dots\dots (48)$$

要素②の場合は $\bar{X}$ の次数が7であるため $\bar{K}^2$ の元数は7になる。

5. アイソパラメトリック四辺形要素

本計算プログラムではアイソパラメトリック四辺形要素を用いる。(図4) 節点 j の (x, y) 座標を (x_j, y_j) と書く。また要素内に ξ, η なる正規座標系を与え, (x, y) 座標を (ξ, η) の関数として次のように定義する。

$$\begin{aligned} x &= \sum_{j=1}^4 N_j x_j = \bar{N}x \\ y &= \sum_{j=1}^4 N_j y_j = \bar{N}y \end{aligned} \quad \dots\dots\dots (49)$$

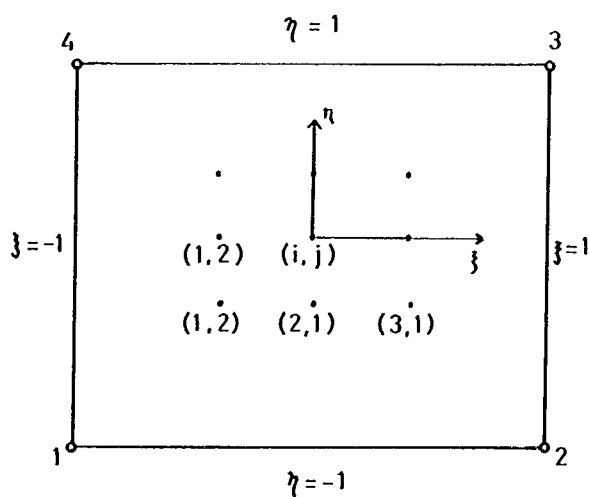


図1-4 アイソパラメトリック四辺形要素

ここに,

$$\begin{aligned} N_1 &= \frac{1}{4} (1-\eta) (1-\xi) \\ N_2 &= \frac{1}{4} (1-\eta) (1+\xi) \\ N_3 &= \frac{1}{4} (1+\eta) (1+\xi) \\ N_4 &= \frac{1}{4} (1+\eta) (1-\xi) \\ \bar{N} &= (N_1, N_2, N_3, N_4) \end{aligned} \quad \begin{aligned} x &= \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \\ y &= \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} \end{aligned} \quad \dots\dots\dots (50)$$

一方,

$$\begin{aligned} \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix} &= G \begin{pmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{pmatrix} & G &= \begin{pmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{pmatrix} = \begin{pmatrix} G_1 \\ G_2 \end{pmatrix} \\ G_1 &= \left(\frac{\partial \xi}{\partial x} \quad \frac{\partial \eta}{\partial x} \right) & G_2 &= \left(\frac{\partial \xi}{\partial y} \quad \frac{\partial \eta}{\partial y} \right) \dots\dots\dots (51) \end{aligned}$$

また,

$$J = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{pmatrix}$$

とおくと明らかに,

$$JG = I$$

より

$$G = J^{-1}$$

(51) 式より

$$\begin{aligned} \frac{\partial N_i}{\partial x} &= G_1 \begin{pmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \end{pmatrix} & (i=1, 4) & \dots\dots\dots (52) \\ \frac{\partial N_i}{\partial y} &= G_2 \begin{pmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \end{pmatrix} & (i=1, 4) & \end{aligned}$$

又 (50) 式から

$$\begin{aligned} \frac{\partial N_1}{\partial \xi} &= -\frac{1}{4} (1-\eta) & \frac{\partial N_1}{\partial \eta} &= -\frac{1}{4} (1-\xi) \\ \frac{\partial N_2}{\partial \xi} &= \frac{1}{4} (1-\eta) & \frac{\partial N_2}{\partial \eta} &= -\frac{1}{4} (1+\xi) \\ \frac{\partial N_3}{\partial \xi} &= \frac{1}{4} (1+\eta) & \frac{\partial N_3}{\partial \eta} &= \frac{1}{4} (1+\xi) \\ \frac{\partial N_4}{\partial \xi} &= -\frac{1}{4} (1+\eta) & \frac{\partial N_4}{\partial \eta} &= \frac{1}{4} (1-\xi) \end{aligned} \dots\dots\dots (53)$$

一方非線形項については Gauss 数値積分を行う。

$$\int_A f(x, y) dA = \sum_{j=1}^{n_p} \sum_{i=1}^{n_p} f(x_{ij}, y_{ij}) |J|_{ij} w_{ij} \dots\dots\dots (54)$$

i, j は積分点を表わす添字である。また w_{ij} は積分点 i および j での重み w_i, w_j の積であり、 n_p は一方向の積分点数である。

6. 初期条件について

(1) 初期条件についての基本的な考え方

初期条件をいかに与えるかは、収束のスピード、安定性、計算時間等に大きく影響し、事前に十分検討する必要がある。ここでは、初期値をセットするために次の3つの方法を検討する。

i) プログラム内で一次的に初期値を与える場合

ii) 他の何らかの方法で求めた初期値を与える場合

iii) 本計算で既に得られた結果を初期値として与える場合

ii) は i), iii) が適用できない場合に使用するが、2次元問題では非常に煩雑となり机上で考えるには実用的でない。iii) は1回の実行で収束しなかった場合のリスタート用として用いる。

(2) 一次的な初期値の与え方

一次元問題として考えた場合、河川の各断面について断面を通過する流量と水位がわかればよい。それらに対する最も適確な解は、境界条件を考慮した不等流計算を行うことにより得られるが、境界条件の組み合わせや派川をもつ場合等を考えれば、不等流計算を行うこと自体が1つの大きな作業になってしまう。

そこで、簡単に次のような方法で初期値を与えるものとする。

i) 初期水位 η

境界条件より上流端または下流端の水位を求める。直接求まらない場合は、境界での水位が等しいものとして条件式を解いてやる。あとは適当な水面勾配（たとえば河床勾配）で各横断面の水位を与える。水面勾配を無視して全断面同一の水位を与えてもよいが、この場合は定常解への収束がその分だけ遅くなる。

ii) 初期流量 Q

境界条件及び条件式より境界での流量を求める。あとは流量に関する連続条件に注意して各断面の流量を与える。

(3) 1次元から2次元への変換

• 初期水位 η の変換

横断面を構成する各節点水位がすべて等しいものとしてセットする。

• 初期流量 $Q \rightarrow u, v$ の変換

横断面を構成する各節点流量は、流速及び流れの向きがすべて等しいものとしてセットする。流れの向きは、横断面の下流法線方向とする。

尚、2つの断面によって定義された節点（分流または合流する場合）の水位及び流量は、それぞれの断面の水位及び流量の平均とする。

今、図1-5のような n 個の節点をもつ横断面を考える。

各節点の水深を H_i とすれば、断面積は、

$$A = \frac{1}{2} \left\{ (H_1 + H_2) \cdot \overline{P_1 P_2} + (H_2 + H_3) \cdot \overline{P_2 P_3} + \cdots + (H_{n-1} + H_n) \cdot \overline{P_{n-1} P_n} \right\} \quad (55)$$

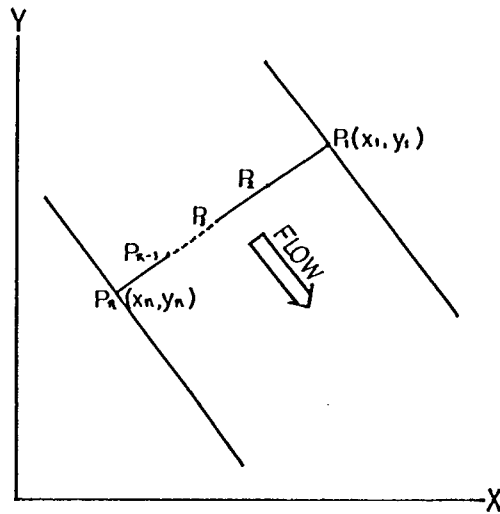


図1-5 二次問題における境界条件

従って断面流量を Q 、平均流速を V とすれば、

$$V = Q/A \quad (56)$$

このとき、各節点の流量の大きさ q_i は

$$q_i = H_i V = (\eta - B_i) \cdot V \quad (57)$$

次に、左岸節点 P_1 の座標を (x_1, y_1) 、右岸節点 P_n の座標を (x_n, y_n) とすれば、 X 方向流量及び Y 方向流量 v_i は

$$\begin{aligned} u_i &= q_i \cdot (y_1 - y_n) / \sqrt{(x_1 - x_n)^2 + (y_1 - y_n)^2} \\ v_i &= -q_i \cdot (x_1 - x_n) / \sqrt{(x_1 - x_n)^2 + (y_1 - y_n)^2} \end{aligned} \quad (58)$$

7. 収束の判定

(1) 収束の判定の基本的な考え方

定常解への収束の判定は、汎用性を考慮し、すべて相対誤差により行うものとする。

ここでは、収束を判定するために次の3つの方法を考えた。

収束条件1…領域内各節点での水深の時間微分項(変化率)の水深に対する割合の最大値が、ある許容

値以内になる。

収束条件 2…領域内の全エネルギーの時間微分項の全エネルギーに対する割合が、ある許容値以内になる。

収束条件 3…各派川での派川流出量の時間微分項の派川流出量に対する割合の最大値が、ある許容値以内になる。

(2) 収束条件 1

各節点での水位を η_i 、前時間の水位を η'_i 、単位時間を Δt とすれば、このときの誤差 E_i は

$$E_i = \frac{H_i - H'_i}{\Delta t \cdot H_i} = \frac{\eta_i - \eta'_i}{\Delta t (\eta_i - B_i)} \quad \dots\dots\dots (59)$$

$|E_i|$ の最大値を $|E_{\max}|$ 、誤差の許容値を ϵ として、次式が成り立つときに限り収束したものと判定する。

$$|E_{\max}| = \max \{|E_1|, |E_2|, \dots\dots\dots, |E_n|\} \leq \epsilon \quad \dots\dots\dots (60)$$

(3) 収束条件 2

各要素の運動エネルギー V^e は次式により求められる。

$$\begin{aligned} V^e &= \frac{1}{2} \int_{D^e} \frac{1}{H} (u^2 + v^2) dD \\ &= \frac{1}{2} \int_{D^e} \frac{1}{H} (u^T N^T N u + w^T N^T N w) dD \\ &= \frac{1}{2} u^T \int_{D^e} \frac{1}{H} N^T N dD u + \frac{1}{2} w^T \int_{D^e} \frac{1}{H} N^T N dD w \\ &\doteq \frac{1}{2\bar{H}} (u^T \int_{D^e} N^T N dD u + w^T \int_{D^e} N^T N dD w) \quad \dots\dots\dots (61) \end{aligned}$$

ただし、 $\bar{H}$ は要素の平均水深を示し、プログラムでは各節点の水深の平均値として処理している。

各要素のポテンシャル・エネルギー U^e は、次式により求められる。

$$\begin{aligned} U^e &= \int_{D^e} gH \eta dD = \int_{D^e} gH(H+B) dD \\ &= \int_{D^e} g(H^2 + BH) dD \\ &= g \int_{D^e} h^T N^T N h dD + g \int_{D^e} b^T N^T N h dD \\ &= gh^T \int_{D^e} N^T N dD h + gb^T \int_{D^e} N^T N dD h \quad \dots\dots\dots (62) \end{aligned}$$

ただし、 b 、 h はそれぞれ要素節点での河床高、水深を表わす。このとき、領域内の全運動エネルギー V 、全ポテンシャル・エネルギー U 、全エネルギー T は、

$$\begin{aligned} V &= \sum_e V^e, \quad U = \sum_e U^e \\ T &= V + U \end{aligned} \quad \dots\dots\dots (63)$$

次に、前時間の全エネルギーを T' とすれば、このときの誤差 E は、

$$E = \frac{T - T'}{\Delta t \cdot T} \quad \dots\dots\dots (64)$$

誤差の許容値を ϵ として、次式が成り立つとき収束したものと判定する。

$$|E| \leq \epsilon \quad \dots\dots\dots (65)$$

(4) 収束条件 3

各派川の流出量 Q_i は、派川を代表する接点水位を η_{ip} として、

$$Q_i = a_i \eta_{ip} + b_i \quad \dots\dots\dots (66)$$

ここで a_i 、 b_i は、あらかじめ与えられている定数である。前時間の派川流出量を Q'_i とすれば、このときの誤差 E_i は、

$$E_i = \frac{Q_i - Q'_i}{\Delta t \cdot Q_i} \quad \dots\dots\dots (67)$$

$|E_i|$ の最大値を $|E_{max}|$ 、誤差の許容値を ϵ として、次式が成り立つときに限り収束したものと判定する。

$$|E_{max}| = \max \{|E_1|, |E_2|, \dots, |E_n|\} \leq \epsilon \quad \dots\dots\dots (68)$$

(5) 誤差の許容値 ϵ の目安

次表に誤差の許容値 ϵ の目安を示す。ただし、これらはどんな場合でも適用できるという性質のものではなく、与えられた問題の要求する精度、計算時間等を考慮して、最終的に決定すべきである。

上記の ϵ の値は、後述の 3.1 で示した 1 次元の流れの解析で、計算開始 15 分後の流れを定常状態とみて、このときの ϵ を参考に決定したものである。

また、ある収束条件を無視して計算したいときは、該当する項目をblankとして入力すればよい。このときプログラムで $\epsilon = 1.0 \times 10^{10}$ にセットされ、収束条件は無条件に満足する。

表 1 - 1 許容誤差との目安

収 束 条 件	ϵ
1	0.25×10^{-5}
2	0.50×10^{-5}
3	0.25×10^{-5}

第 2 章 プログラム説明

1. プログラムの概要

(1) プログラムの基本構成

本プログラムは、有限要素法による平面流解析プログラムで、「1. 理論説明」で示した解析の手順を忠実にプログラミングしたものである。

プログラムは、1つのメイン・プログラムと、45このサブ・ルーチンからなる。

メイン・プログラムは形式的なものであり、サブ・ルーチンFEMが実質的なコントロールを行う。この関係を次に示す。

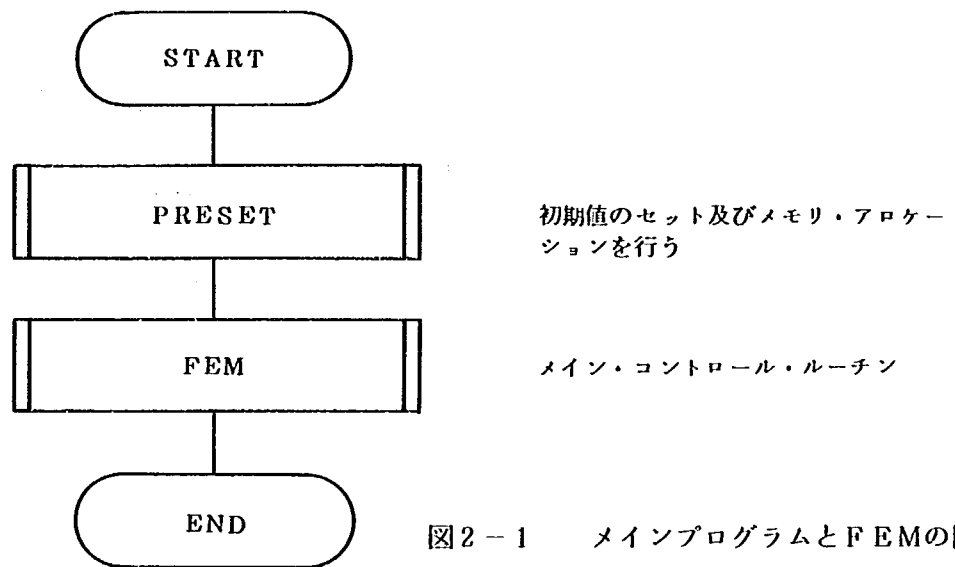


図 2 - 1 メインプログラムとFEMの関係

(2) 概略フロー・チャート

全体フロー・チャートを図 2 - 2 に、CLMAIN フロー・チャートを図 2 - 3 に、IVSET フロー・チャートを図 2 - 4 に示す。

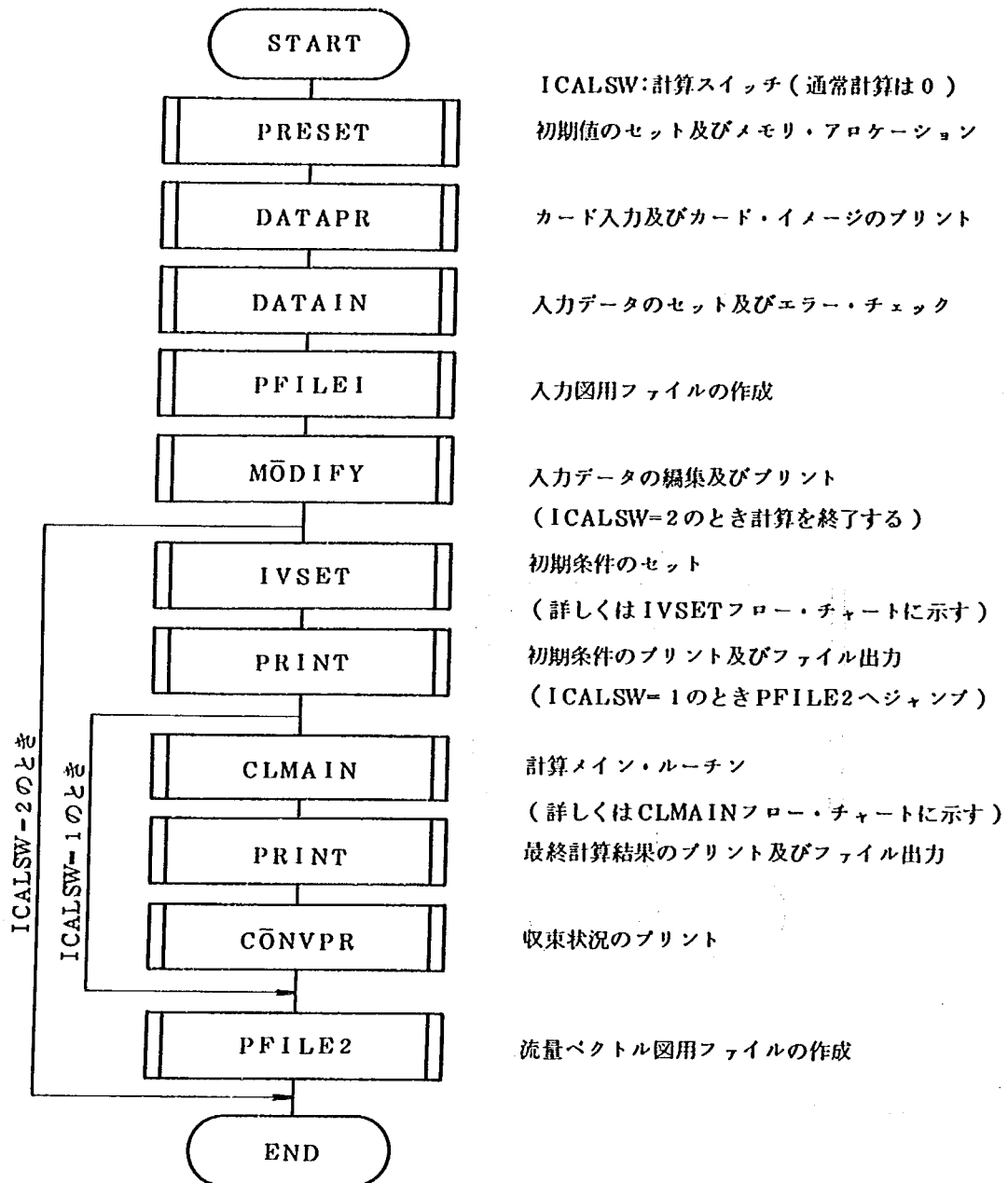


図 2-2 全体フローチャート

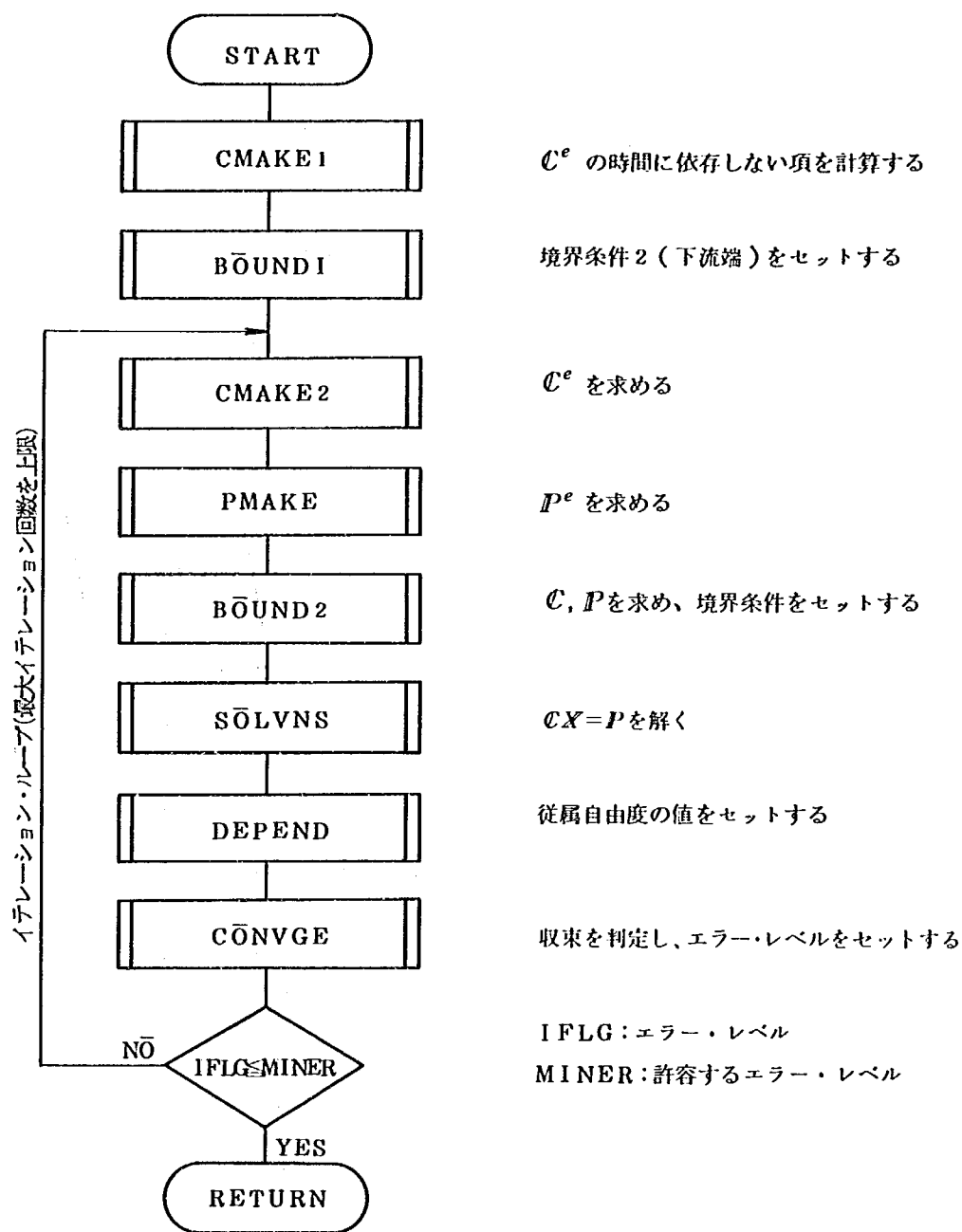


図2-3 CLMAINフロー・チャート

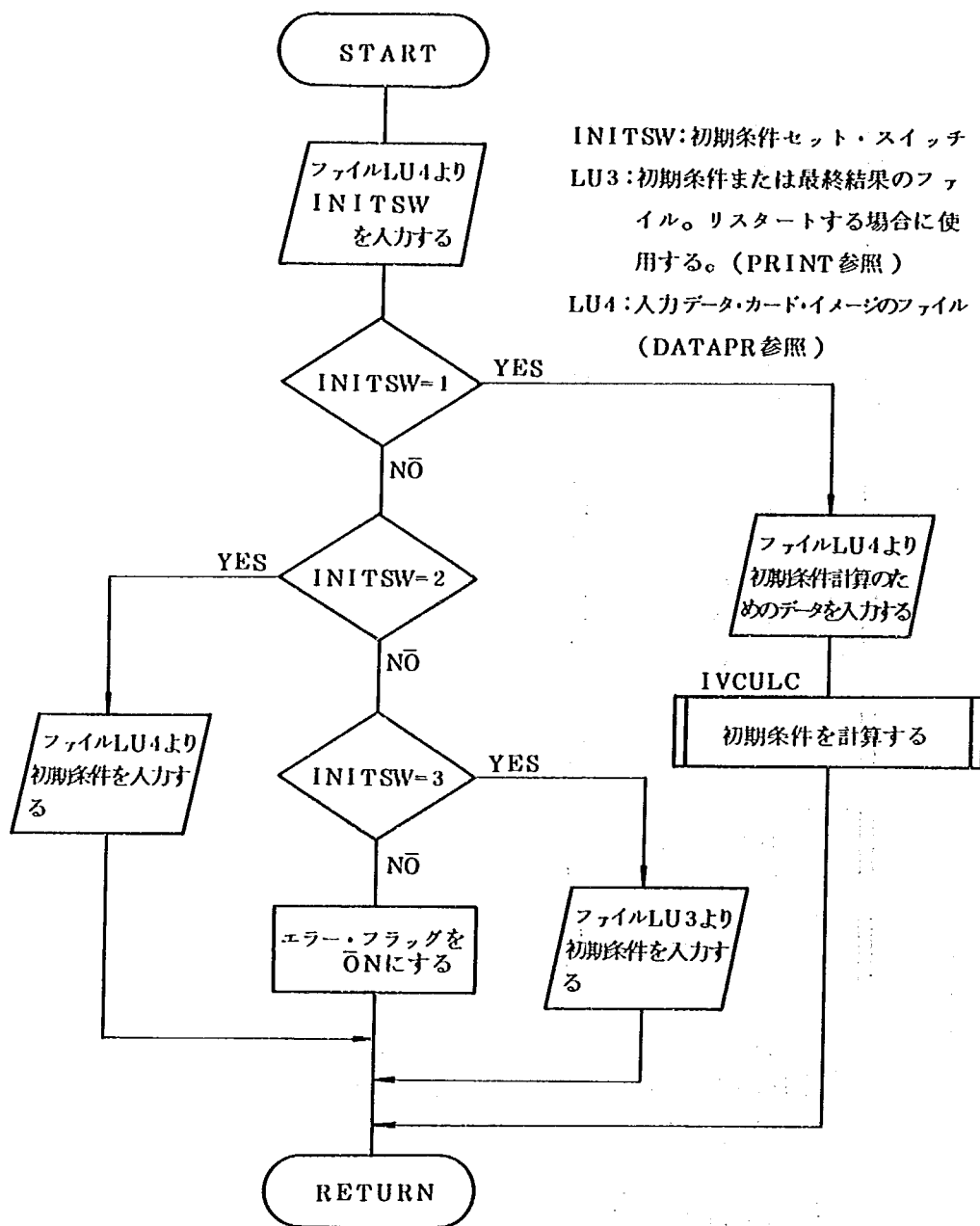


図2-4 IVSETフロー・チャート

(3) ファイル構成

各ファイルの装置番号は、プログラム内で次の値にセットしている (PRESET 参照)

LIN = 5 , LOUT = 6 , LU1 = 1 , LU2 = 2
LU3 = 3 , LU4 = 4 , LU10 = 10 , LU11 = 11
LU12 = 12 , LU13 = 13 , LU14 = 14 , LU15 = 15
LU16 = 16

各ファイルの働きを次に示す。

LIN : データ・カード入力ファイル。DATAPR で入力し、すぐに LU4 に出力する。

(入力ルーチン: PRESET 及び DATAPR , 出力ルーチン: DATAPR)

LOUT: プリント出力ファイル。カード・イメージ、編集された入力データ、初期条件、最終計算結果、チェック・プリント及びエラー・メッセージが出力される。

LU1 : 入力図用ファイル。(出力ルーチン: PFILE1)

LU2 : 流量ベクトル図用ファイル。(出力ルーチン: PFILE2)

LU3 : 初期条件または最終結果のファイル。

(入力ルーチン: IVSET , 出力ルーチン: PRINT , リスタートする場合に使用する)

LU4 : 入力データ・カード イメージ・ファイル。

(入力ルーチン: DATAPR , 出力ルーチン: DATAIN 及び IVSET)

LU10: 収束状況の情報ファイル。内部使用。

LU11: 要素方程式の C^e の時間に依存しない項のファイル。内部使用。

LU12: 制約条件式の A , B のファイル。内部使用。

LU13: 要素方程式の C^e のファイル。内部使用。

LU14: 要素方程式の P^e のファイル。内部使用。

LU15: $\int_{D^e} N^T N dD$ のファイル。内部使用。

2. グローバル・コモン

1) /ALLLOC/ IALC(100), NALC, NOWFL, MAXFL

メモリ・アロケーションの管理

IALC(i): 各エリアの先頭番地

NALC : 分割エリアの個数

NOWFL : 現在の最終番地

MAXFL : 最大エリア (= 700,000 にセット)

2) /MAXMAX/ NPMAX, NEMAX, NBMAX, NMMAX, NBNMX1, NBNMX2, NBNMX3

各変数エリアの最大値

NPMAX : 最大節点数 (デフォルトは 200)

NEMAX : 最大要素数 (" 200)

NBMAX	: 最大バンド幅	(デフォルトは 100)
NMMAX	: 最大マテリアル数	(" 5)
NBNMX 1	: 境界条件 1 の最大条件数	(" 100)
NBNMX 2	: 境界条件 2 の最大条件数	(" 20)
NBNMX 3	: 境界条件 2 の最大独立自由度数	(" 20)

3) /ERROR /IERROR

エラー・フラッグ

IERROR = 0 : エラーなし, = 1 : エラー有り

4) /FILE /LIN, LOUT, LU 1, LU 2, LU 3, LU 4, LU 10, LU 11, LU 12, LU 13, LU 14, LU 15

使用ファイルの装置番号

LIN	: データ・カード入力ファイル (= 5)
LOUT	: プリント出力ファイル (= 6)
LU 1	: 入力図用ファイル (= 1)
LU 2	: 流量ベクトル図用ファイル (= 2)
LU 3	: 初期条件または最終結果ファイル (= 3)

詳細は, 前述のファイル構成の項参照

5) /CHECK /ICHECK (10)

チェック・プリント・スイッチ

ICHECK (i) = 0 : チェック・プリントなし
 ≥ 1 : チェック・プリントあり

詳しくは, 入力フォーム (3 / 8) 参照

6) /CALSW /ICALSW

計算スイッチ

ICALSW = 0 : 通常の計算を行う
 = 1 : 初期条件チェック用
 = 2 : 入力データチェック用

7) /PAGEC /IPAGE, ILINE

プリント出力の管理

IPAGE : ページのカウンター
 ILINE : ラインのカウンター

8) /CNTRL /NP, NE, NB, NM, NBN 1, NBN 2

各変数のコントロール・データ

NP : 節点数

NE : 要素数
 NB : バンド幅
 NM : マテリアル数
 NBN1 : 境界条件1の条件数
 NBN2 : 境界条件2の条件数

9) / TITLE / ITIT (20)

計算タイトル (20 A 4)

3. メモリ・アロケーション

本プログラムのメモリ・アロケーションを表2-1に示す。

表2-1 メモリ・アロケーション

変数名	タイプ	使用メモリ数	変数の説明
X	D	NP MAX	節点のX座標
Y	D	NP MAX	節点のY座標
B	D	NP MAX	節点の河床高
EFIX	D	NP MAX	節点の既定水位
ETA	D	NP MAX	節点水位
U	D	NP MAX	X方向節点流量
V	D	NP MAX	Y方向節点流量
IELE	I	4 × NEMAX	要素を構成する節点番号
KELE	I	5 × NEMAX	要素の境界条件1, 形状に関する情報
MELE	I	NEMAX	要素のマテリアル番号
CMAT	D	2 × NM MAX	渦動粘性係数及びシェーger係数
IEBN	I	2 × NBNMX1	境界条件1を指定する要素及び辺番号
QBN	D	2 × NBNMX1	辺における法線方向流量
INBN	I	3 × NBNMX2	境界条件2を指定する節点及び自由度番号他
JNBN	I	2 × NBNMX3 × NBNMX2	独立節点及び自由度番号
CBN	D	NBNMX3 × NBNMX2	線型結合の係数
UX	D	NEMAX	X方向要素流量
VX	D	NEMAX	Y方向要素流量

4. 各モジュールの説明

(1) PRESET

・目的

プログラム実行前の各変数の初期値をセットし、ワーキング・エリアのアロケーションを行う。

・パラメータ

なし

(2) ALLOCT

・目的

必要な語数分のエリアを確保し、その変数の先頭番地をセットする。

・パラメータ

DUM : ダミー・パラメータ

N : 確保すべき語数

ITYPE : その変数のタイプ

ITYPE = 1 のとき 4 バイト変数

= 2 のとき 8 バイト変数

(3) FEM

・目的

実質上のメイン・プログラム。(図 2-2 全体フロー・チャート参照)

・パラメータ

X, Y : 節点座標

B : 節点の河床高

EFIX : 節点の既定水位

ETA : 節点水位

U, V : X, Y 方向節点流量

IELE : 要素を構成する節点番号

KELE : 要素の境界条件 1, 形状に関する情報

KELE (j, i), j = 1, 4 各辺の境界条件 1 番号

KELE (5, i) 要素を構成する節点数

MELE : 要素のマテリアル番号

CMAT : マテリアル

CMAT (1, i) 渦動粘性係数 ϵ

CMAT (2, i) シェジャー係数 C

IEBN : 境界条件 1 を指定する要素及び辺番号

IEBN (1, i) 要素番号

: IEBN (2 , i) 辺番号
 QBN : 辺における法線方向流量
 QBN (1 , i) 始点側流量 Q_1
 QBN (2 , i) 終点側流量 Q_2
 INBN : 境界条件 2 を指定する節点及び自由度番号他
 INBN (1 , i) 節点番号
 INBN (2 , i) 従属自由度番号
 INBN (3 , i) 独立自由度数
 JNBN : 独立節点及び自由度番号
 JNBN (1 , j , i) 節点番号
 JNBN (2 , j , i) 独立自由度番号
 CBN : 線型結合の係数
 UX , VX : X , Y 方向要素流量
 BBN : 線型結合の定数

(4) DATAPR

・目 的

データを入力し、データ・カード・イメージをプリントする。

・パラメータ：なし

・使用ファイル：LU 4

(5) DATAIN

・目 的

入力データを所定のエリアにセットし、入力時に検出できるエラーのチェックを行う。

・パラメータ

FEM の項参照

(6) PFILE 1

・目 的

入力図作成用のデータをファイルに出力する。

・パラメータ：FEM の項参照

・使用ファイル：LU 1

・ファイル構成 (Binary mode)

レコード 1.	タイトル	(ITIT (i) , i = 1, 20)
2.	節点数	NP
3.	X, Y 座標	(X (i) , Y (i) , i = 1, NP)
4.	要素数	NE

5. 要素構成節点 (CIELE(j, i), $j=1,4$), $i=1,NE$)

(7) MODIFY

• 目的

入力データを編集して、プリントする。

• パラメータ

FEM の項参照

(8) PEJECT

• 目的

必要があれば改ページを行い、見出しをプリントする。

• パラメータ

ISW : 見出しの項目のスイッチ

ISW = 1 節点データ, = 2 要素データ

= 3 マテリアル・データ

= 4 境界条件 1 データ

= 5 境界条件 2 データ

= 6 初期条件または計算結果

= 7 収束過程の表示

(9) IVSET

• 目的

指定された方法で初期条件をセットする。(図 2-4 IVSET フロー・チャート参照)

• パラメータ

FEM の項参照

(10) IVCULC

• 目的

1次元の初期条件を与えて、各節点の初期条件をセットする。

初期条件セット・スイッチ INITSW = 1 のとき、このルーチンが呼び出される。

• パラメータ

FEM の項参照

(11) PRINT

• 目的

初期条件または最終計算結果をプリントする。

• パラメータ

ISW : 見出しの項目のスイッチ

ISW = 1 初期条件

(12) CLMAIN

・目 的

計算メイン・ルーチン。(図2-3 CLMAIN フロー・チャート参照)

・パラメータ

FEM の項参照

(13) CONVPR

・目 的

イテレーション毎の収束状況をプリントする。

・パラメータ : なし

(14) PFILE2

・目 的

流量ベクトル図作成のデータをファイルに出力する。

・パラメータ : FEM の項参照

・使用ファイル : LU2

・ファイル構成 (Binary mode)

レコード 1. タイトル (ITIT (i) , i = 1, 20)

2. 節点数 NP

3. X, Y 座標, 水位

(X (i) , Y (i) , ETA (i) , i = 1, NP)

4. 要素数 NE

5. 要素構成節点及び要素流量

((IELE (j , i) , j = 1, 4) , UX (i) , VX (i) , i = 1, NE)

第3章 使用説明

入力項目は次の通りである。

- | | |
|-----------------------------|------|
| (1) メモリ・アロケーション・データ | (必須) |
| (2) タイトル | (必須) |
| (3) コントロール・データ | (必須) |
| (4) チェック・プリント・スイッチ・データ | (必須) |
| (5) 派川流出量算定データ | |
| (6) 節点データ | (必須) |
| (7) 要素データ | (必須) |
| (8) マテリアル・データ | (必須) |
| (9) 境界条件1のデータ(境界の法線方向流量の指定) | |
| (10) 境界条件2のデータ(従属自由度の指定) | |
| (11) 初期条件セット・スイッチ・データ | (必須) |
| (12) 初期条件データ | |

表3-1～8に入力フォームを示す。

表3-1 入力フォーム (1), (2)

[illegible]

表 3-2 入力フォーム (3)

1 2 3 4 5 6 7 8 9 10		11 12 13 14 15 16 17 18 19 20		21 22 23 24 25 26 27 28 29 30		31 32 33 34 35 36 37 38 39 40		41 42 43 44 45 46 47 48 49 50		51 52 53 54 55 56 57 58 59 60		61 62 63 64 65 66 67 68 69 70		71 72 73 74 75 76 77 78 79 80	
(3.)		コントロールデータ													
IN P	IN E	IN M	NBN	D T	ICALSW	NGS	MAXIT	NPOINT	MINER	EP:SLIN1	EP:SLIN2	EP:SLIN3			
				N.P		接点数				N.E		要素数			
				N.M		マテリアル数				N.B.N		境界条件数			
				D.T		計算単位時間 (S)									
				ICALSW		計算スイッチ									
				NGS		Gauss積分の積分点数 (≤2) のとき NGS=2 にセットする。									
				MAXIT		最大イテレーション回数				NPOINT		派川数 (≤10)			
				MINER		許容するエラーレベル									
				EP:SLIN1		収束条件1 (全節点での水深の最大変化率の水深に対する割合の許容値、デフォルトは 1.0E-10)									
				EP:SLIN2		収束条件2 (全エネルギー変化率の全エネルギーに対する割合の許容値、デフォルトは 1.0E-10)									
				EP:SLIN3		収束条件3 (派川流出量最大変化率の派川流出量に対する割合の許容値、デフォルトは 1.0E-10)									

図3-3 入力フォーム (4)

[illegible]

図 3-4 入力フォーム (5), (6)

[illegible]

表 3-5 入力フォーム (7), (8)

[illegible]

表3-6 入力データ・フォーム (9)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80																																								
(9) 境界条件1の										データ (境界条件1がない場合は、この項はスキップする。)																																																																																																													
1																				(1.5, 1.5X, 2, 1.5, 2.5, 1.0, 1.0)																																																																																																			
2										INOE										辺番号										Q1										Q2										(境界条件1の数を繰り返して入力する。)																																																																					
3																																																																																																																							
4																																																																																																																							
5																																																																																																																							
6																																																																																																																							
7																																																																																																																							
8																																																																																																																							
9																																																																																																																							
10																																																																																																																							
11																																																																																																																							
12																																																																																																																							
13																																																																																																																							
14																																																																																																																							
15																																																																																																																							
16																																																																																																																							
17																																																																																																																							
18																																																																																																																							
19																																																																																																																							
20																																																																																																																							
21																																																																																																																							
22																																																																																																																							
23																																																																																																																							
24																																																																																																																							
25																																																																																																																							
26																																																																																																																							
27																																																																																																																							
28																																																																																																																							
29																																																																																																																							
30																																																																																																																							
31																																																																																																																							
32																																																																																																																							
33																																																																																																																							
34																																																																																																																							
35																																																																																																																							
36																																																																																																																							
37																																																																																																																							
38																																																																																																																							
39																																																																																																																							
40																																																																																																																							
41																																																																																																																							
42																																																																																																																							
43																																																																																																																							
44																																																																																																																							
45																																																																																																																							
46																																																																																																																							
47																																																																																																																							
48																																																																																																																							
49																																																																																																																							
50																																																																																																																							
51																																																																																																																							
52																																																																																																																							
53																																																																																																																							
54																																																																																																																							
55																																																																																																																							
56																																																																																																																							
57																																																																																																																							
58																																																																																																																							
59																																																																																																																							
60																																																																																																																							
61																																																																																																																							
62																																																																																																																							
63																																																																																																																							
64																																																																																																																							
65																																																																																																																							
66																																																																																																																							
67																																																																																																																							
68																																																																																																																							
69																																																																																																																							
70																																																																																																																							
71																																																																																																																							
72																																																																																																																							
73																																																																																																																							
74																																																																																																																							
75																																																																																																																							
76																																																																																																																							
77																																																																																																																							
78																																																																																																																							
79																																																																																																																							
80																																																																																																																							

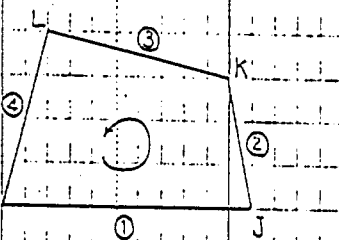
(1.5, 1.5X, 2, 1.5, 2.5, 1.0, 1.0)

(境界条件1の数を繰り返して入力する。)

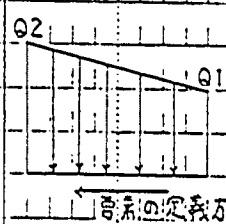
N.O.E. 境界条件1を指定する符号番号

Q.1 辺の始点側の法線方向流量

Q.2 辺の終点側の法線方向流量



① ~ ④ 辺番号



[illegible]

図3-8 入力フォーム (11), (12)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80																				
(11) 初期条件ビット:スイッチ・デコー																																																																																																			
INITSW										(11.5)																																																																																									
																				INITSW										初期条件ビット・スイッチ										=1.1のとき 流量と水位を指定して初期条件をセットする。																																																											
																																								=1.2のとき カードより通入される。																																																											
																																								=1.3のとき 12ビットの初期条件を読み込む。																																																											
(12) 初期条件デコー										(12.1) INITSW=1.3のとき (この項はスキップする)																																																																																									
(12.1) INITSW=1.1のとき																																																																																																			
										(12.1.5) 2F/0.01																																																																																									
*1										*2										ETA										Q										(断面数だけ繰り返して入力する)																																																											
																																								ETA: 断面水位, Q: 断面流量																																																											
																																								た場合は直前に入力したETA, Qがセットされる。																																																											
(12.1.1) INITSW=1.2のとき																																																																																																			
																				(12.1.5) 5X/3F/0.01																																																																															
NOP										U										V										ETA										(特定点だけ繰り返して入力する)																																																											
																				NOP										特定点番号																																																																					
																				U, V, ETA										特定点のX方向流量, Y方向流量と水位の初期値																																																																					

第4章 テスト・ラン

1. 一次元流れへの適用によるチェック

計算法の妥当性および収束性を調べるために、厳密解の得られる一次元流れの計算を行い比較した。図4-1に示す一定勾配の一樣直線流路を考える。過動粘性係数は0とおく。またシェジャー係数は50とする。

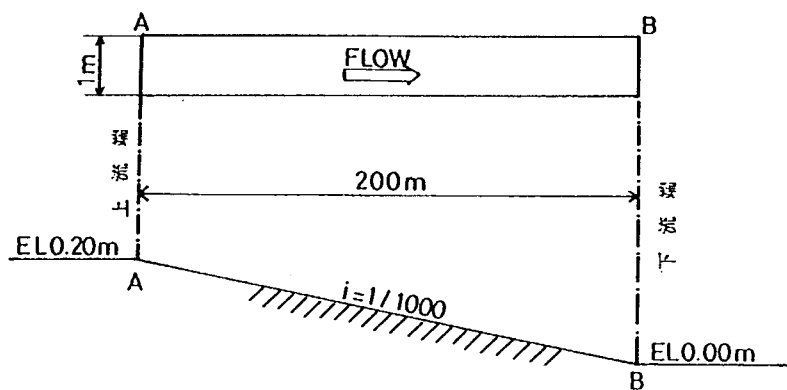


図4-1 水路諸元

次の2種類の境界条件について計算した。

ケース 1. : $Q_A = 1.0 \text{ m}^3/\text{s}$, $Q_B = 1.25 \eta_B$

ケース 2. : $Q_A = 1.0 \text{ m}^3/\text{s}$, $\eta_B = 0.80 \text{ m}$

厳密解の誘導過程と計算結果を付録に示す。これによれば上流端水位は 0.961 m となる。

有限要素モデルを図4-2のように設定した。要素数は20，接点数は42となる。境界条件式は以下のようになる。

上流端の法線方向流量 : $u_n = 1.0 \text{ m}^2/\text{s}$

水路側壁法線方向流量 : $u_n = 0$

下流端水位 : $\eta_{41} = \eta_{42} = 0.4 u_{41} + 0.4 u_{42}$ (ケース 1)

$\eta_{41} = \eta_{42} = 0.80 \text{ m}$ (ケース 2)

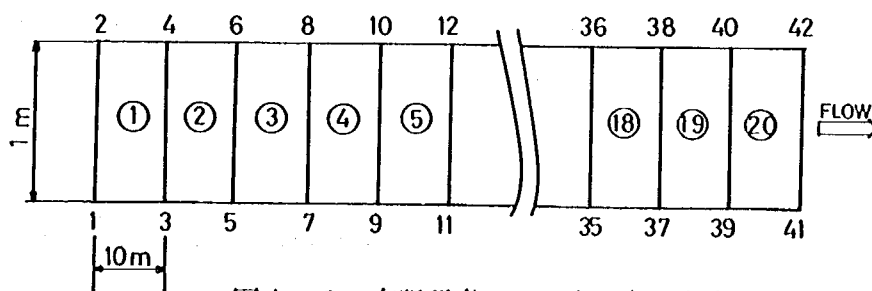


図4-2 有限要素モデル(一次元水路)

初期条件は次の2種類を考えた。

$$\begin{aligned}
 &u_1 = u_2 = \dots = u_{42} = 1.0 \text{ m}^3/\text{s} \\
 \text{ケース A} \quad &v_1 = v_2 = \dots = v_{42} = 0 \\
 &\eta_1 = \eta_2 = \dots = \eta_{42} = 0.80 \text{ m} \\
 &u_1 = u_2 = \dots = u_{42} = 1.0 \text{ m}^3/\text{s} \\
 \text{ケース B} \quad &v_1 = v_2 = \dots = v_{42} = 0 \\
 &\eta_1 = \eta_2 = 1.00 \text{ m} \quad \xrightarrow{1/1000} \quad \eta_{41} = \eta_{42} = 0.80 \text{ m}
 \end{aligned}$$

ケースAは水面を水平とした場合，ケースBは水面を河床勾配に平行においたものである。したがって計算ケースは表4-1に示す4ケースとなる。また計算条件は以下のように設定している。

計算単位時間 : $\Delta t = 20 \text{ s}$
 最大イテレーション回数 : $\text{MAXIT} = 50 \text{ 回}$
 許容するエラー・レベル : $\text{MINER} = 0$
 収束条件1の ϵ : $\text{EPSLN } 1 = 0.25 \times 10^{-5}$
 収束条件2の ϵ : $\text{EPSLN } 2 = 0.50 \times 10^{-5}$
 収束条件3の ϵ : $\text{EPSLN } 3 = 0.25 \times 10^{-5}$

ただし許容するエラー・レベル $\text{MINER} = 0$ とは，すべての収束条件を満足した時に収束したとみなすものである。

表4-1 計算ケースと収束状況

計算ケース	下流端の境界条件	初期水位	収束時の上流端水位	収束時間 (実物の時間)	摘 要
1-A	$Q_B = 1.25 \eta_B$	水面勾配0	0.9614	15分	収束条件3は無視する "
1-B	"	" 1/1000	0.9619	12	
2-A	$\eta_B = 0.80 \text{ m}$	水面勾配0	0.9616	8	
2-B	"	" 1/1000	0.9616	7	

上流端水位の収束状況を図4-3に示す。どのケースにおいても実物時間で10分（繰り返し計算回数30回）で厳密解にはほぼ一致した結果を与えることがわかる。

2. 江戸川流頭部水理計算

江戸川流頭部現計画低水路について計算を行った。同水路の平面形を図4-4に示す。また各区間の諸元は以下のとおりである。

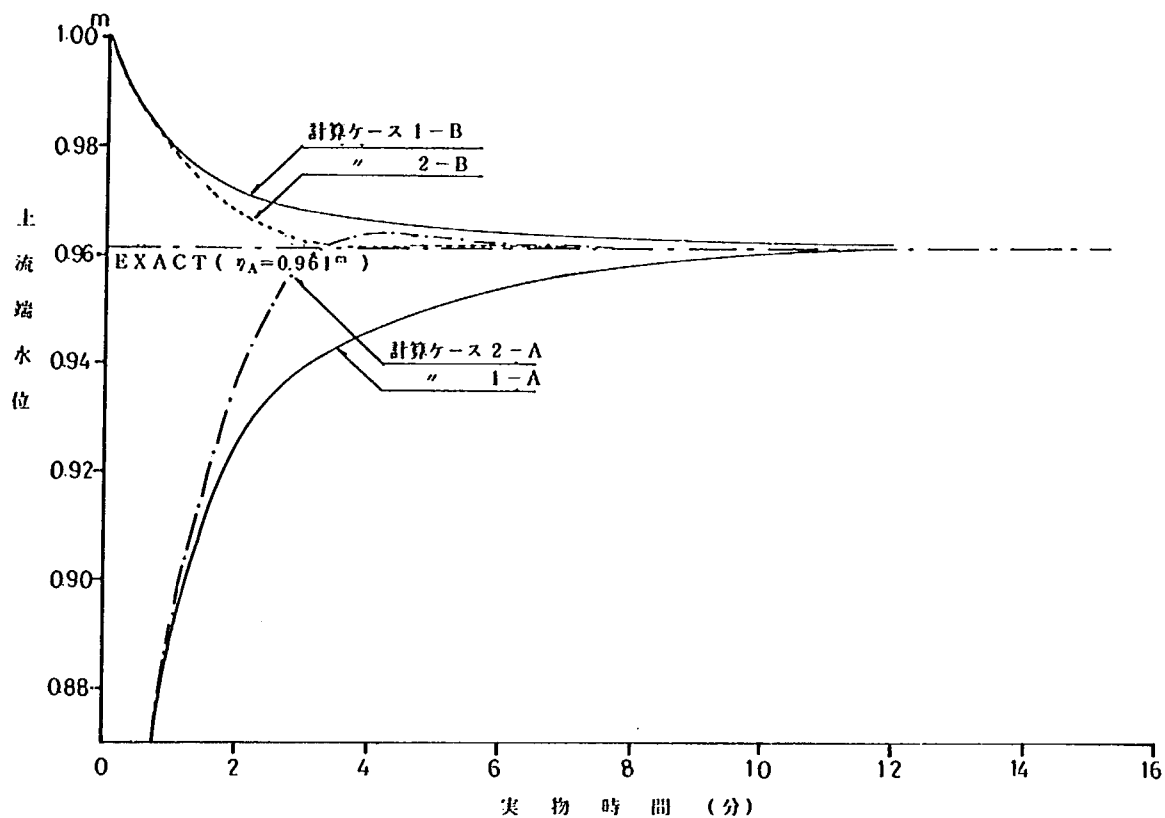


図4-3 上流端水位収束状況

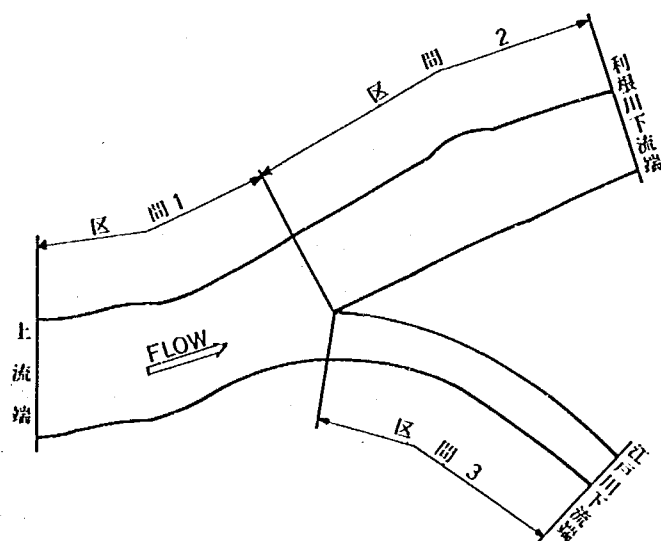


図4-4 江戸川流頭部低水路形状図

区間長	区間1	$l_1 \approx 950 \text{ m}$	河床勾配	区間1	$i_1 \approx 1/3300$
	区間2	$l_2 \approx 1000 \text{ m}$		区間2	$i_2 \approx 1/4000$
	区間3	$l_3 \approx 1100 \text{ m}$		区間3	$i_3 \approx 1/2700$
	計算延長	$l \approx 3050 \text{ m}$			

断面幅	区間1	$T_1 \approx 350 \text{ m}$
	区間2	$T_2 \approx 230 \text{ m}$
	区間3	$T_3 \approx 130 \text{ m}$

物理定数は全区間一定として次の値を与えた。

渦動粘性係数 : $\epsilon = 3.54$

シェジャー係数 : $C = 57$

上流端流量は $2,000 \text{ m}^3/\text{s}$ とする。

$$Q = 2000 \text{ m}^3/\text{s} \quad (\text{A})$$

また下流端境界条件は利根川および江戸川の現計画粗度から求まる水位流量関係より次の式を設定した。

$$\text{利根川} \quad Q_T = 347 \eta_T - 3316 \quad (\text{B})$$

$$\text{江戸川} \quad Q_E = 468 \eta_E - 4306 \quad (\text{C})$$

図4-5に有限要素モデルを示す。節点数 N_P は132, 要素数 N_E は100である。

(2) 境界条件の組み込み方法

前述の一次元計算では境界条件の与え方は容易に決定できたが、流れが二次元的になるといくつかの問題が生じてくる。そこで以下に境界条件の与え方について若干の検討をする。

(a) 上流端の境界条件

上流端の境界条件を有限要素モデルに組み入れる方法として、次の4つが考えられる。

- ① 上流端を構成する要素の各辺に対して、法線方向流量を与える。
- ② ①の条件に加えて、上流端を構成する節点の流れの向きを与える。
- ③ 上流端を構成する各節点に対して、 x 方向流量と y 方向流量を与える。
- ④ ③の条件に加えて、各節点水位が等しいものとする。

①の方法が最も自由度数が大きく、②、③、④の順に自由度数が小さくなる。このことは、上流端付近の水の流れに対して、①の方法が最も自由な挙動が可能であり、④の方法が最も制約が大きいことを意味している。

①の方法は、1次元の流れの解析で採用した方法であり、最も単純で理解しやすいが、2次元問題で上流端を構成する要素が複雑になった場合、上流端付近の水の挙動が不安定となる恐れがある。実際、江戸川流頭部の予備計算段階では、 x 、 y 方向流量が発散し、収束しなかった。また②についてもほぼ同様の結果を得た。

③の方法は、①、②に比べてかなり安定するが、予備計算では、ある時間経過後、上流端の水位の挙動が収束に対してネックとなることがわかった。

④の方法は、上流端の自由度が1つ（すなわちある節点の水位だけが自由であり、その他の節点の水位及び各節点の x 、 y 方向流量はすべて制約を受ける）であるが、予備計算では非常に良好な結果を得た。

以上のことから、上流端の境界条件を有限要素モデルに組み入れる方法として、④の方法を採用した。

具体的には、上流端を構成する各節点（節点番号1, 2, 3, 4）に対して、

$$x \text{ 方向流量} : u_1 = u_2 = u_3 = u_4 = 5.5446 \text{ m}^3/\text{s}$$

FEM PLOT SCALE 0.0 120.0

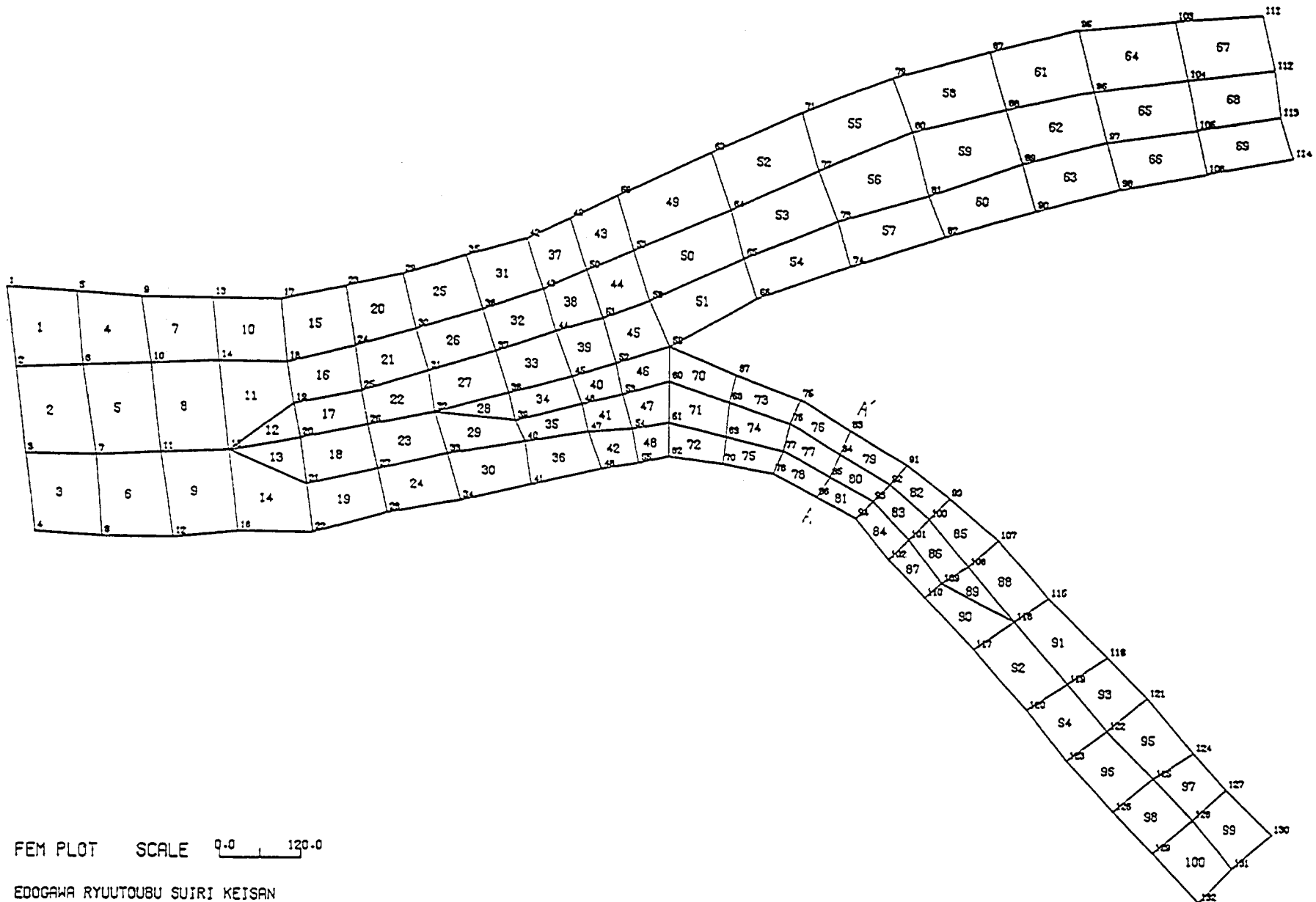


図 4 - 5 江戸川流頭部低水路有限要素モデル

$$\left. \begin{array}{l} y \text{ 方向流量} : v_1 = v_2 = v_3 = v_4 = 0.7028 \text{ m}^2/\text{s} \\ \text{水 位} : \eta_2 = \eta_1, \eta_3 = \eta_1, \eta_4 = \eta_1 \end{array} \right\} \cdots (A)'$$

ただし、プログラムの中では次のように変形している。

$$\left. \begin{array}{l} u_i = 0.0 \times \eta_1 + 5.5446 \quad (i = 1, 4) \\ v_i = 0.0 \times \eta_1 + 0.7028 \quad (i = 1, 4) \\ \eta_i = 1.0 \times \eta_1 + 0.0 \quad (i = 2, 4) \end{array} \right\} \cdots (A)''$$

(b) 下流端の境界条件

下流端の境界条件を有限要素モデルに組み入れるためには、若干の工夫を必要とする。
派川の下流端における流出量が、 a 、 b を定数として次式で与えられているものとする。
(ただし $a \neq 0$)

$$Q = a\eta + b \cdots \cdots \cdots (D)$$

これを η について解けば、

$$\eta = \frac{1}{a} Q - \frac{b}{a} \cdots \cdots \cdots (E)$$

図4-5において、左岸節点 P_1 の座標を (x_1, y_1) 、右岸節点 P_n の座標を (x_n, y_n) とすれば、各節点の x 方向流量 u_i 及び y 方向流量 v_i の下流端断面の下流法線方向の成分の和 u_{ni} は、

$$\begin{aligned} u_{ni} = & u_i \cdot (y_1 - y_n) / \sqrt{(x_1 - x_n)^2 + (y_1 - y_n)^2} \\ & - v_i \cdot (x_1 - x_n) / \sqrt{(x_1 - x_n)^2 + (y_1 - y_n)^2} \cdots \cdots \cdots (F) \end{aligned}$$

従って、下流端流量 Q は、式 (F) の u_{ni} を用いて

$$Q = \frac{1}{2} \left\{ (u_{n1} + u_{n2}) \cdot \overline{P_1 P_2} + (u_{n2} + u_{n3}) \cdot \overline{P_2 P_3} + \cdots \cdots \cdots + (u_{nn-1} + u_{nn}) \cdot \overline{P_{n-1} P_n} \right\} \cdots \cdots \cdots (G)$$

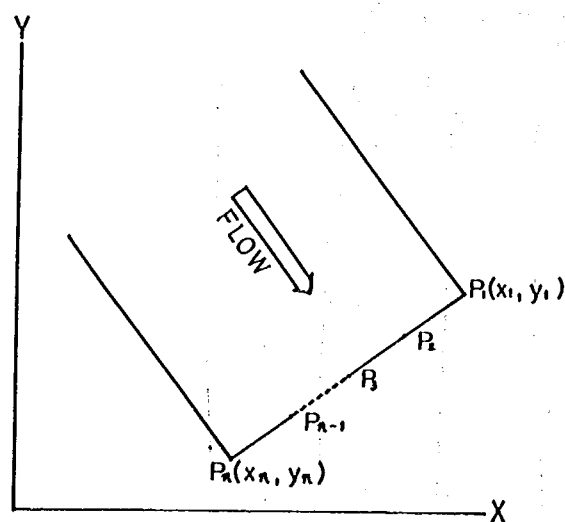


図4-6 二次元下流端境界条件

簡単に $\overline{P_1 P_2} \approx \overline{P_2 P_3} \approx \dots \approx \overline{P_{n-1} P_n}$ とすれば、

$$Q = \frac{\overline{P_1 P_n}}{2(n-1)} (u_{n1} + 2u_{n2} + \dots + 2u_{n,n-1} + u_{nn}) \quad \dots \dots \dots (G)'$$

式 (B) の境界条件より、具体的に有限要素モデルでの境界条件を導入してみよう。

左岸節点座標 (2000, 1440), 右岸節点座標 (2050, 1230) を式 (F) に代入して、

$$u_{ni} = 0.97281 u_i + 0.23162 v_i$$

$n = 4$ を式 (G) に代入して、

$$Q = 35.9784 (u_{n111} + 2u_{n112} + 2u_{n113} + u_{n114})$$

従って、利根川下流端水位の制約条件式は、式 (E) に $a = 347$, $b = -3316$ 及び以上の結果を代入して、

$$\begin{aligned} \eta_{111} \sim \eta_{114} = & 0.1009 u_{111} + 0.2017 u_{112} + 0.2017 u_{113} \\ & + 0.1009 u_{114} + 0.0240 v_{111} + 0.0480 v_{112} \\ & + 0.0480 v_{113} + 0.0240 v_{114} + 9.5562 \quad \dots \dots \dots (B)' \end{aligned}$$

同様にして、江戸川下流端の制約条件式は、

$$\begin{aligned} \eta_{130} \sim \eta_{132} = & 0.05342 u_{130} + 0.10684 u_{131} + 0.05342 u_{132} \\ & - 0.05876 v_{130} - 0.11752 v_{131} - 0.05876 v_{132} + 9.2009 \quad \dots \dots \dots (C)' \end{aligned}$$

(3) 初期条件の設定

初期条件は、水面勾配を無視した 1 次元問題として与える。実際の問題では、収束性や計算時間等から、できるだけ定常状態に近い状態を与えることが望ましいわけであろうが、テスト・ランでは収束過程や収束するまでの時間も重要な検討の対象となるため、初期値を最もシンプルな形で与えるものとした。

従って、本計算で使用した初期条件は、境界条件を満足するが、定常状態からはかなり離れている。

下流端における初期水位 η は、境界条件式 (B), (C) で $\eta_T = \eta_E$ とおいて

$$Q_T + Q_E = 815\eta - 7622$$

$$Q_T + Q_E = Q_A = 2000 \text{ m}^3/\text{s} \text{ であるから}$$

$$\eta = 11.8061 \approx 11.81 \text{ m}$$

水面勾配を無視した場合、すべての断面に対してこの値を初期値として与える。

各区間の初期流量は、境界条件式 (A) ~ (C) より

$$\text{区間 1} : Q_1 = 2000 \text{ m}^3/\text{s}$$

$$\text{区間 2} : Q_2 = 347 \times 11.81 - 3316 \approx 780 \text{ m}^3/\text{s}$$

$$\text{区間 3} : Q_3 = 468 \times 11.81 - 4306 \approx 1220 \text{ "}$$

河川各区間の断面毎の初期データを、表 4 - 2 に示す。

表 4 - 2 初期条件データ

区間	左岸 接点番号	右岸 接点番号	断 面 水 位 (m)	断 面 水 位 (m ³ / s)	摘 要
区 間 1	1	4	11.81	2000	
	5	8	"	"	
	9	12	"	"	
	13	16	"	"	
	17	22	"	"	
	23	28	"	"	
	29	34	"	"	
	35	41	"	"	
	42	48	"	"	
	49	55	"	"	
区 間 2	56	59	11.81	780	
	63	66	"	"	
	71	74	"	"	
	79	82	"	"	
	87	90	"	"	
	95	98	"	"	
	103	106	"	"	
	111	114	"	"	
区 間 3	59	62	11.81	1220	
	67	70	"	"	
	75	78	"	"	
	83	86	"	"	
	91	94	"	"	
	99	102	"	"	
	107	110	"	"	
	115	117	"	"	
	118	120	"	"	
	121	123	"	"	
	124	126	"	"	
	127	129	"	"	
	130	132	"	"	

(4) 計算結果

有限要素モデル，境界条件及び初期条件については前項までに示したが，その他の計算条件は次の通りである。

計算単位時間	: DT = 60S
最大イテレーション回数	: MAXIT = 120 回
許容するエラー・レベル	: MINER = 0
収束条件 1 の ϵ	: EPSLN 1 = 0.25×10^{-5}
収束条件 2 の ϵ	: EPSLN 2 = 0.50×10^{-5}
収束条件 3 の ϵ	: EPSLN 3 = 0.25×10^{-5}

図 4 - 7 に下流端流量の収束状況を，図 4 - 8 に上下流端水位の収束状況を示す。いずれも実時間で 80 分（繰り返し計算回数 80 回）で収束している。また収束後の流速ベクトル図を図 4 - 9 に示す。

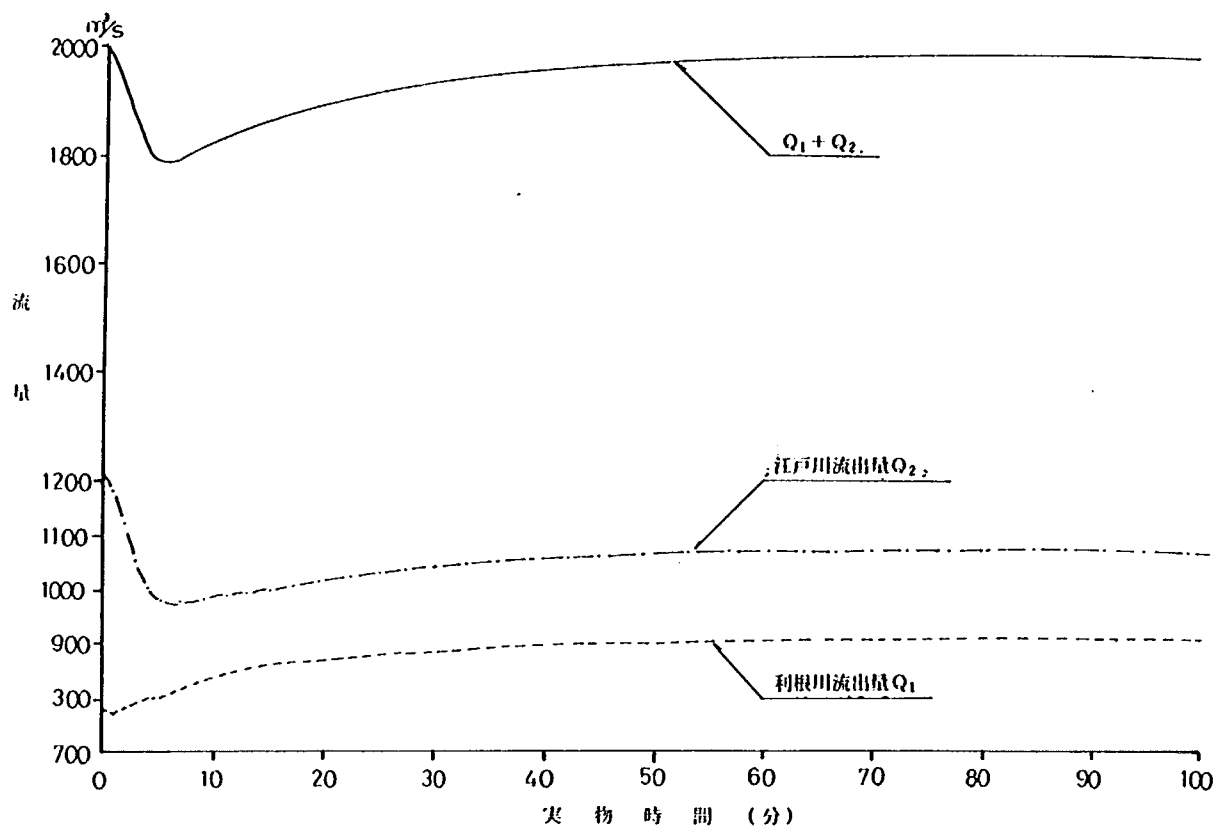


图4-7 下流端流量收束状况

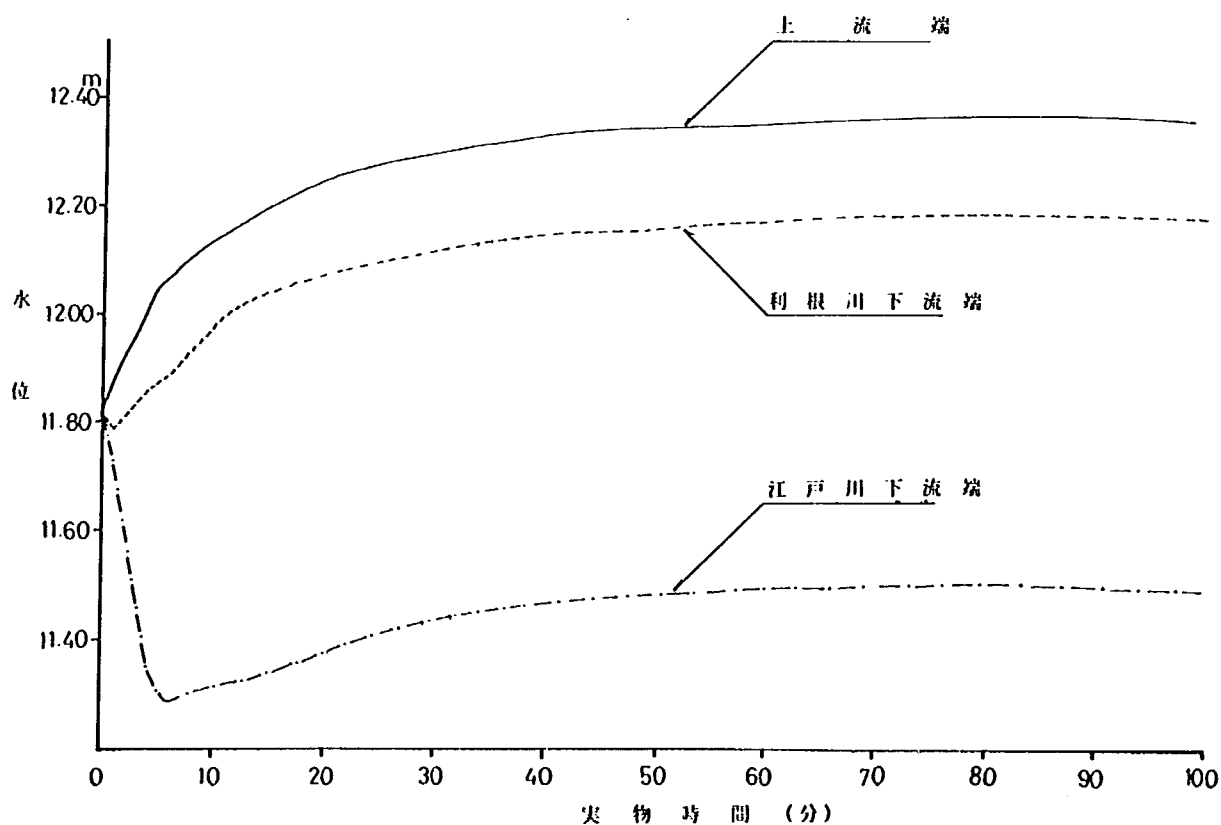
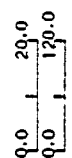


图4-8 上下流端水位收束状况



CURRENT
SCALE

FEM PLOT
SCALE

CURRENT SCALE 0.0 20.0
FEM PLOT SCALE 0.0 120.0

EDOGAWA RYUUTOUBU SUIRI KEISAN (PART 2)

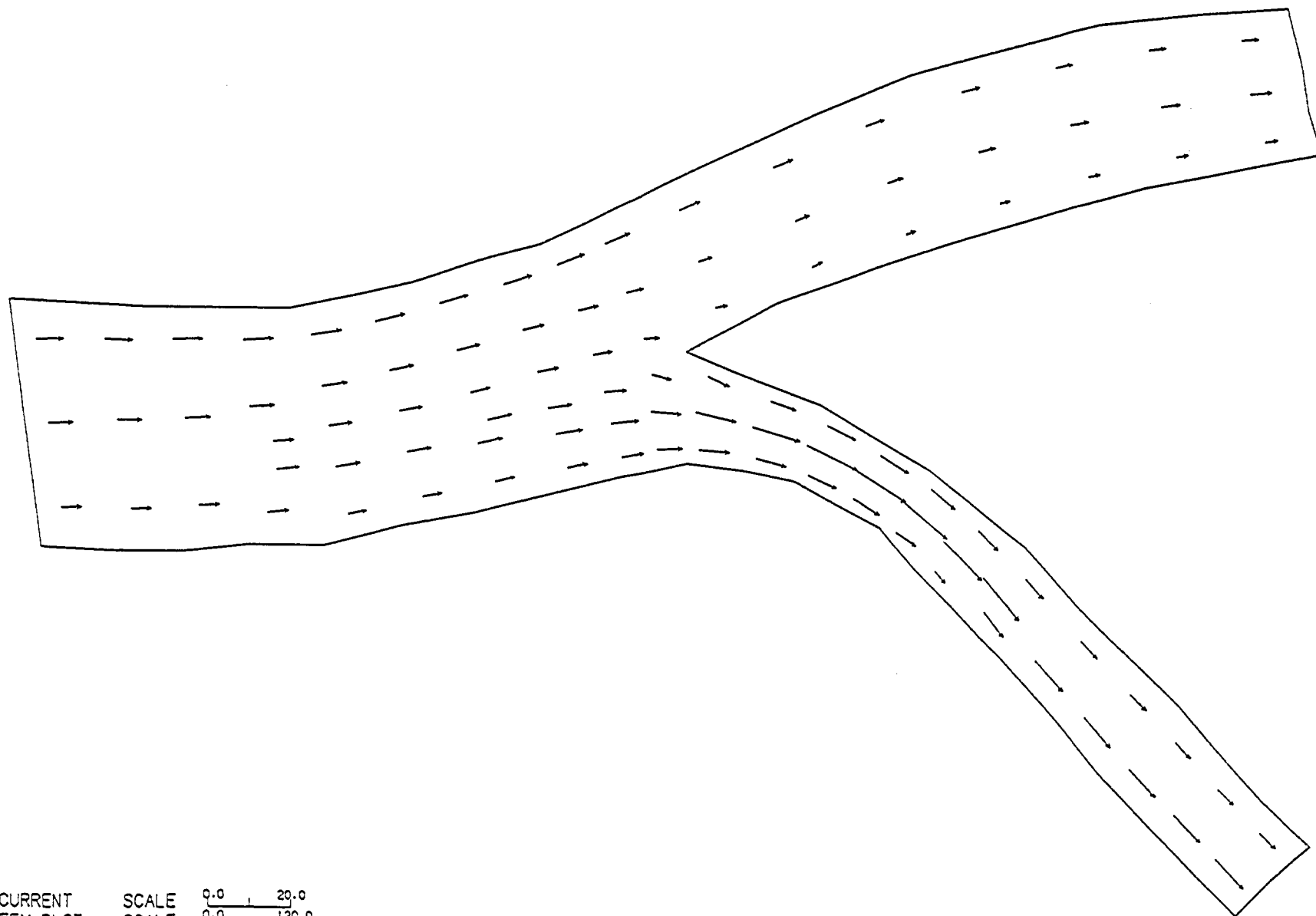


図 4 - 9 流速ベクトル図

お わ り に

今回作成したプログラムは単断面河道についてのものである。しかし現実には複断面河道が増えつつある。本プログラムでも、原理的にはメッシュの取り方で複断面河道を表わせる可能性もあるが、しかし一般に低水路河岸では浅水流方程式の成立しない恐れがあるので、局部的に別の方程式を挿入するか、又はそれを暗に考慮した形でのプログラム上の工夫が必要とされる。次回の検討項目としたい。

本プログラム作成にあたり、(株)センチュリリサーチセンタ、田辺誠氏他技士の方々に大変お世話になった。謝意を表する。

付録 1 短形断面水路の一次元 不等流方程式厳密解

2020 年 9 月 25 日 星期五

1. 2010年12月10日，在“2010年中国年度人物”颁奖典礼上，中国首位“冰上丝绸之路”破冰人——中国极地科考破冰船“雪龙”号船长刘建民，荣获“2010中国年度人物”称号。

1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 2072, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083, 2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 2095, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117, 2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 2129, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 2140, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151, 2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 2163, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 2174, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 2197, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 2208, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 2231, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253, 2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 2265, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 2276, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287, 2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 2299, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 2333, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 2344, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355, 2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 2367, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 2378, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389, 2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457, 2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491, 2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 2639, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 2650, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661, 2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 2673, 2674, 2675, 2676, 2677, 2678, 2679, 2680, 26

付録1 矩形断面水路の一次元不等流方程式の厳密解

一次元定常問題では(1)式は次のように簡略化される。

$$\frac{\partial}{\partial x} \left(\frac{u^2}{H} \right) = -gH \frac{\partial \eta}{\partial x} - \frac{g}{c^2} \frac{u^2}{H^2} \quad \dots\dots\dots (A)$$

流量 u は一定値であるから,

$$\frac{\partial}{\partial x} \left(\frac{u^2}{H} \right) = - \frac{u^2}{H^2} \frac{\partial H}{\partial x} \quad \dots\dots\dots (B)$$

一方水位を水深と河床高に分解することにより,

$$-gH \frac{\partial \eta}{\partial x} = -gH \frac{\partial H}{\partial x} + gHi \quad \dots\dots\dots (C)$$

ここに i は河床勾配である。(B), (C) 式を (A) 式に代入すると次式を得る。

$$- \frac{u^2}{H^2} \frac{\partial H}{\partial x} + gH \frac{\partial H}{\partial x} = gHi - \frac{g}{c^2} \frac{u^2}{H^2}$$

$$\frac{\partial H}{\partial x} = i \frac{H^3 - \frac{u^2}{c^2 i}}{H^3 - \frac{u^2}{g}} = i \frac{H^3 - H_0^3}{H^3 - H_c^3} \quad \dots\dots\dots (D)$$

ここに,

$$H_c = \sqrt[3]{\frac{u^2}{g}} \quad \dots\dots\dots \text{限界水深}$$

$$H_0 = \sqrt[3]{\frac{u^2}{c^2 i}} \quad \dots\dots\dots \text{等流水深} \quad \dots\dots\dots (E)$$

各量を等流水深 H_0 で無次元化する。

$$\int \frac{y^3 - y_c^3}{y^3 - 1} dy = i d\xi \quad \dots\dots\dots (F)$$

ここに, $y = H/H_0$, $y_c = H_c/H_0$, $\xi = x/H_0$ である。左辺は次のように分解される。

左辺は、

$$\begin{aligned}\frac{y^3 - y_c^3}{y^3 - 1} &= 1 + \frac{1 - y_c^3}{y^3 - 1} = 1 + \frac{1 - y_c^3}{(y - 1)(y^2 + y + 1)} \\ &= 1 + \frac{1 - y_c^3}{3} \left(\frac{1}{y - 1} - \frac{y + 2}{y^2 + y + 1} \right) \dots\dots\dots (G)\end{aligned}$$

() 内の 2 項の積分は次のようになる。

$$\begin{aligned}\int \frac{y}{y^2 + y + 1} dy &= \frac{1}{2} \ln(y^2 + y + 1) - \frac{1}{\sqrt{3}} \tan^{-1} \left(\frac{2y + 1}{\sqrt{3}} \right) \\ \int \frac{2}{y^2 + y + 1} dy &= \frac{4}{\sqrt{3}} \tan^{-1} \left(\frac{2y + 1}{\sqrt{3}} \right) \dots\dots\dots (H)\end{aligned}$$

(G), (H) 式を (F) 式に代入すると次の水面形方程式を得る。

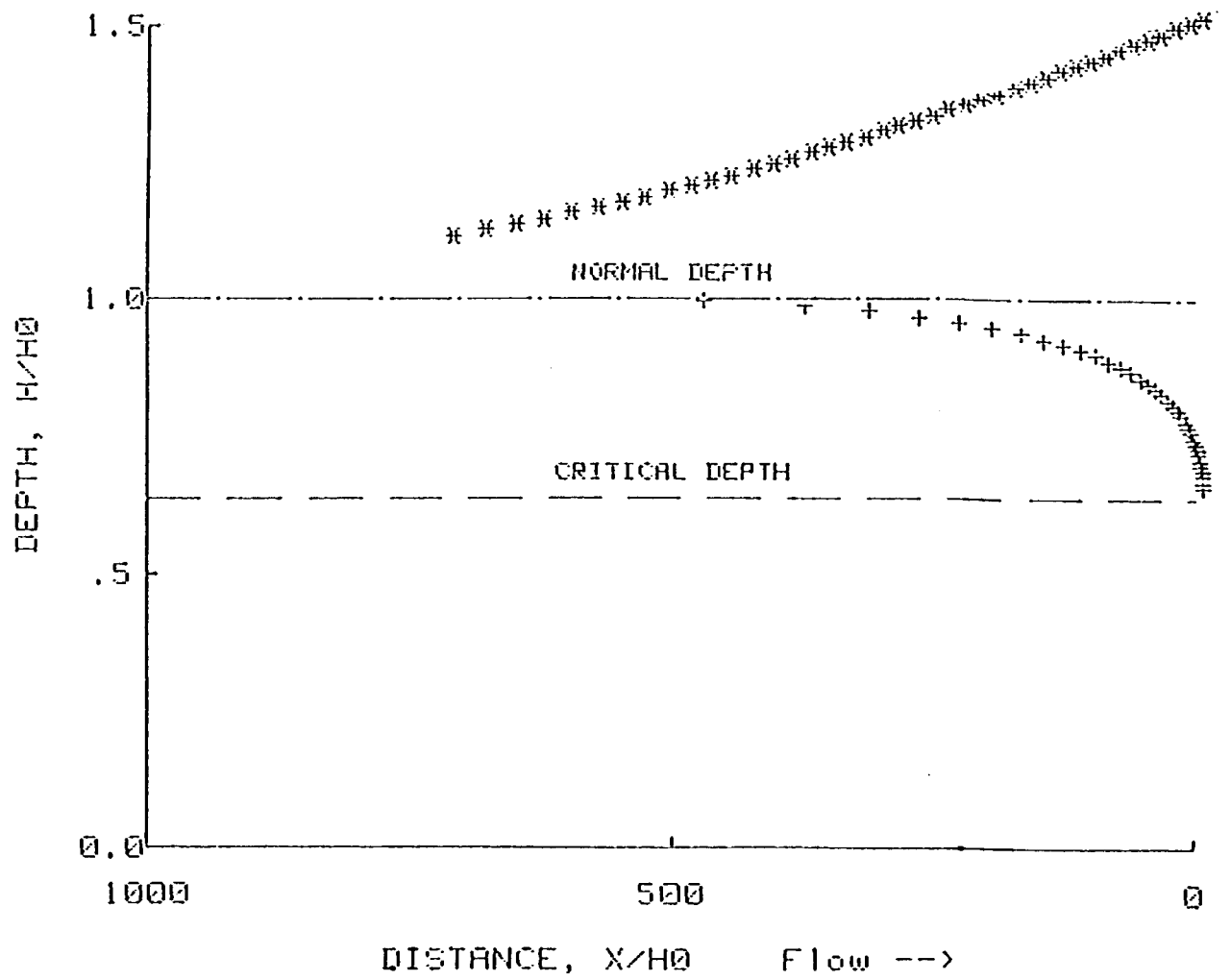
$$\begin{aligned}\int \frac{y^3 - y_c^3}{y^3 - 1} dy &= y + \frac{1 - y_c^3}{3} \left\{ \ln|y - 1| - \frac{1}{2} \ln(y^2 + y + 1) - \sqrt{3} \tan^{-1} \left(\frac{2y + 1}{\sqrt{3}} \right) \right\} \\ &= i\xi + \text{const.} \dots\dots\dots (I)\end{aligned}$$

計算例を付図 1 に示す。計算条件は第 4 章 2 節と同じである。

Chezy's Coefficient $C = 50$

Slope $I = 1/1000$

Ratio of H_c to H_0 $Y_c = .635$



付図1 一次元水路水面形（厳密解）

有限要素法による河川分流
計算プログラム

ISSN 0386-5878
土研資料第 1940 号
昭和 58 年 3 月

須賀堯三, 石川忠晴

キーワード 数値計算, 分流, 有限要素法

要 旨 河川分流部の流れを, 浅水流方程式に基づき有限要素法で
数値計算するプログラムを開発した。



郵便はがき

3 0 5 - □ □

40円切手
をはいて
下さい

茨城県筑波郡豊里町大字旭一番地

建設省土木研究所

河川研究室 御中

No. _____

No. _____

配布先氏名

配布先住所





土木研究所資料第 1940 号 部

上記のとおり受領しました。

所 属 (住所)

氏 名

印



プログラム・リスト

有限要素法による河川分流計算プログラム

昭和58年 3 月

建設省土木研究所

河 川 研 究 室

1	1.	PROGRAM MAIN	MAIN.2
2	C		MAIN.3
3	2.	COMMON / ALLOC / IALC(100),NALC ,NOHFL ,MAXFL	/ALLOC/.2
4	3.	COMMON / ERROR / IERROR	/ERROR/.2
5	4.	COMMON / MAXHAX / NPMAX ,NEMAX ,NBMAX ,NMHAX ,	/MAXHAX/.2
6		* NBNMX1 ,NBNMX2 ,NBNMX3	/MAXHAX/.3
7	5.	COMMON / / A(1)	//.2
8	C		MAIN.8
9	6.	CALL PRESET	MAIN.9
10	C		MAIN.10
11	7.	CALL FEM	MAIN.11
12		* (A(IALC(1)), A(IALC(2)), A(IALC(3)), A(IALC(4))	MAIN.12
13		* ,A(IALC(5)), A(IALC(6)), A(IALC(7)), A(IALC(8))	MAIN.13
14		* ,A(IALC(9)), A(IALC(10)), A(IALC(11)), A(IALC(12))	MAIN.14
15		* ,A(IALC(13)), A(IALC(14)), A(IALC(15)), A(IALC(16))	MAIN.15
16		* ,A(IALC(17)), A(IALC(18)), A(IALC(19)), A(IALC(20))	MAIN.16
17		* ,A(IALC(21)), A(IALC(22)), A(IALC(23)), A(IALC(24))	MAIN.17
18		* ,A(IALC(25)), A(IALC(26)), A(IALC(27)), A(IALC(28))	MAIN.18
19		* ,NPMAX, NEMAX ,NBMAX ,NMHAX ,NBNMX1	MAIN.19
20		* ,NBNMX2, NBNMX3)	MAIN.20
21	C		MAIN.21
22	8.	STOP	MAIN.22
23	9.	END	MAIN.23

24	1.	SUBROUTINE HSET1(IV,IA,N)	HSET1.2
25	C		HSET1.3
26	2.	DIMENSION IA(N)	HSET1.4
27	C		HSET1.5
28	3.	DO 10 I=1,N	HSET1.6
29	4.	IA(I) = IV	HSET1.7
30	5.	10 CONTINUE	HSET1.8
31	C		HSET1.9
32	6.	RETURN	HSET1.10
33	7.	END	HSET1.11

34	1.	SUBROUTINE HSET2(V,A,N)	HSET2.2
35	C		HSET2.3
36	2.	REAL*8 V,A	HSET2.4
37	3.	DIMENSION A(N)	HSET2.5
38	C		HSET2.6
39	4.	DO 10 I=1,N	HSET2.7
40	5.	A(I) = V	HSET2.8
41	6.	10 CONTINUE	HSET2.9
42	C		HSET2.10
43	7.	RETURN	HSET2.11
44	8.	END	HSET2.12

45	1.	SUBROUTINE MOVE1(IA,IB,N)	MOVE1.2
46	C		MOVE1.3
47	2.	DIMENSION IA(N),IB(N)	MOVE1.4
48	C		MOVE1.5
49	3.	DO 10 I=1,N	MOVE1.6
50	4.	IB(I) = IA(I)	MOVE1.7
51	5.	10 CONTINUE	MOVE1.8
52	C		MOVE1.9
53	6.	RETURN	MOVE1.10
54	7.	END	MOVE1.11

55	1.	SUBROUTINE MOVE2(A,B,N)	MOVE2.2
56	C		MOVE2.3
57	2.	REAL*8 A,B	MOVE2.4
58	3.	DIMENSION A(N),B(N)	MOVE2.5
59	C		MOVE2.6
60	4.	DO 10 I=1,N	MOVE2.7
61	5.	B(I) = A(I)	MOVE2.8
62	6.	10 CONTINUE	MOVE2.9
63	C		MOVE2.10
64	7.	RETURN	MOVE2.11
65	8.	END	MOVE2.12

66	1.	SUBROUTINE ADD(A,B,C,CA,CB,M,N)	ADD.2
67	C		ADD.3
68	2.	REAL*8 A,B,C,CA,CB	ADD.4
69	3.	DIMENSION A(M,N),B(M,N),C(M,N)	ADD.5
70	C		ADD.6
71	4.	DO 10 J=1,N	ADD.7
72	5.	DO 20 I=1,M	ADD.8
73	6.	C(I,J) = CA*A(I,J) + CB*B(I,J)	ADD.9
74	7.	20 CONTINUE	ADD.10
75	8.	10 CONTINUE	ADD.11
76	C		ADD.12
77	9.	RETURN	ADD.13
78	10.	END	ADD.14

79	1.	SUBROUTINE MULTI(A,B,C,L,M,N)	MULT.2
80	C		MULT.3
81	2.	REAL*8 A,B,C	MULT.4
82	3.	DIMENSION A(L,M),B(M,N),C(L,N)	MULT.5
83	C		MULT.6
84	4.	DO 10 I=1,L	MULT.7
85	5.	DO 20 K=1,N	MULT.8
86	6.	S = 0.0	MULT.9
87	C		MULT.10
88	7.	DO 30 J=1,M	MULT.11
89	8.	S = S + A(I,J)*B(J,K)	MULT.12
90	9.	30 CONTINUE	MULT.13
91	C		MULT.14
92	10.	C(I,K) = S	MULT.15
93	11.	20 CONTINUE	MULT.16
94	12.	10 CONTINUE	MULT.17
95	C		MULT.18
96	13.	RETURN	MULT.19
97	14.	END	MULT.20

98	1.	SUBROUTINE TRANS(A,B,H,H)	TRANS.2
99	C		TRANS.3
100	2.	REAL*8 A,B	TRANS.4
101	3.	DIMENSION A(H,H),B(H,H)	TRANS.5
102	C		TRANS.6
103	4.	DO 10 I=1,H	TRANS.7
104	5.	DO 20 J=1,N	TRANS.8
105	6.	B(J,I) = A(I,J)	TRANS.9
106	7.	20 CONTINUE	TRANS.10
107	8.	10 CONTINUE	TRANS.11
108	C		TRANS.12
109	9.	RETURN	TRANS.13
110	10.	END	TRANS.14

111	1.	SUBROUTINE MULT2(A,B,C,L,M,N,N1,N2,N3)	MULT2.2
112	C		MULT2.3
113	2.	REAL*8 A,B,C	MULT2.4
114	3.	DIMENSION A(N1,1),B(N2,1),C(N3,1)	MULT2.5
115	C		MULT2.6
116	4.	DO 10 I=1,L	MULT2.7
117	5.	DO 20 K=1,N	MULT2.8
118	6.	S = 0.0	MULT2.9
119	C		MULT2.10
120	7.	DO 30 J=1,M	MULT2.11
121	8.	S = S + A(I,J)*B(J,K)	MULT2.12
122	9.	30 CONTINUE	MULT2.13
123	C		MULT2.14
124	10.	C(I,K) = S	MULT2.15
125	11.	20 CONTINUE	MULT2.16
126	12.	10 CONTINUE	MULT2.17
127	C		MULT2.18
128	13.	RETURN	MULT2.19
129	14.	END	MULT2.20

130	1.	SUBROUTINE TRANS2(A,B,H,N,N1,N2)	TRANS2.2
131	C		TRANS2.3
132	2.	REAL*8 A,B	TRANS2.4
133	3.	DIMENSION A(N1,1),B(N2,1)	TRANS2.5
134	C		TRANS2.6
135	4.	DO 10 I=1,H	TRANS2.7
136	5.	DO 20 J=1,N	TRANS2.8
137	6.	B(J,I) = A(I,J)	TRANS2.9
138	7.	20 CONTINUE	TRANS2.10
139	8.	10 CONTINUE	TRANS2.11
140	C		TRANS2.12
141	9.	RETURN	TRANS2.13
142	10.	END	TRANS2.14

```

143 1. SUBROUTINE PRESET PRESET.1
144 C PRESET.3
145 2. IMPLICIT REAL*8 (A-H,O-Z) PRESET.4
146 C PRESET.5
147 3. COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 , /FILE/.2
148 * LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15 /FILE/.3
149 4. COMMON / PAGEC / IPAGE ,ILINE /PAGEC/.2
150 5. COMMON / ERROR / IERROR /ERROR/.2
151 6. COMMON / CHECK / ICHECK(10) /CHECK/.2
152 7. COMMON / MAXMAX / NPMAX ,NEMAX ,NBMAX ,NMMAX , /MAXMAX/.2
153 * NBNHX1 ,NBNHX2 ,NBNHX3 /MAXMAX/.3
154 8. COMMON / ALLOC / IALC(100),NALC ,NOWFL ,MAXFL /ALLOC/.2
155 9. COMMON / CONST / ZERO ,ONE ,TWO /CONST/.2
156 10. COMMON / GAUSS / NGS ,G(5,4) ,W(5,4) /GAUSS/.2
157 11. COMMON / GRAVITY / GRAV /GRAVITY/.2
158 12. COMMON / / A(1) //2
159 C PRESET.16
160 13. DATA G / PRESET.17
161 * -0.577350269189600 , 0.577350269189600 , 0.000 PRESET.18
162 * , 0.000 , 0.000 PRESET.19
163 * -0.774596669241500 , 0.000 , 0.774596669241500 PRESET.20
164 * , 0.000 , 0.000 PRESET.21
165 * -0.861136311594100 , -0.339981043584900 , 0.339981043584900 PRESET.22
166 * , 0.861136311594100 , 0.000 PRESET.23
167 * -0.906179845938700 , -0.538469310105700 , 0.000 PRESET.24
168 * , 0.538469310105700 , 0.906179845938700 PRESET.25
169 * / PRESET.26
170 14. DATA H / PRESET.27
171 * 1.000 , 1.000 , 0.000 PRESET.28
172 * , 0.000 , 0.000 PRESET.29
173 * 0.555555555555600 , 0.888888888888900 , 0.555555555555600 PRESET.30
174 * , 0.000 , 0.000 PRESET.31
175 * 0.347854845137500 , 0.652145154862500 , 0.652145154862500 PRESET.32
176 * , 0.347854845137500 , 0.000 PRESET.33
177 * 0.236926885056200 , 0.478628670499400 , 0.568888888888900 PRESET.34
178 * , 0.478628670499400 , 0.236926885056200 PRESET.35
179 * / PRESET.36
180 15. DATA GRAV / 9.800 / PRESET.37
181 C PRESET.38
182 16. DATA LIN ,LOUT,LU1 ,LU2 ,LU3 ,LU4 ,LU10,LU11,LU12,LU13,LU14,LU15 PRESET.39
183 * / 5 , 6 , 1 , 2 , 3 , 4 , 10 , 11 , 12 , 13 , 14 , 15 / PRESET.40
184 C PRESET.41
185 17. ZERO = 0.0 PRESET.42
186 18. ONE = 1.0 PRESET.43
187 19. TWO = 2.0 PRESET.44
188 C PRESET.45
189 20. IPAGE = 0 PRESET.46
190 21. ILINE = 0 PRESET.47
191 22. IERROR = 0 PRESET.48
192 23. CALL MSET1(0,ICHECK,10) PRESET.49
193 C PRESET.50
194 24. READ(LIN,501) NPMAX,NEMAX,NBMAX,NMMAX,NBNHX1,NBNHX2,NBNHX3 PRESET.51
195 25. 501 FORMAT( 7I5 ) PRESET.52
196 C PRESET.53
197 26. IF ( NPMAX .LE. 0 ) NPMAX = 200 PRESET.54
198 27. IF ( NEMAX .LE. 0 ) NEMAX = 200 PRESET.55
199 28. IF ( NBMAX .LE. 0 ) NBMAX = 100 PRESET.56
200 29. IF ( NMMAX .LE. 0 ) NMMAX = 5 PRESET.57
201 30. IF ( NBNHX1 .LE. 0 ) NBNHX1 = 100 PRESET.58
202 31. IF ( NBNHX2 .LE. 0 ) NBNHX2 = 20 PRESET.59
203 32. IF ( NBNHX3 .LE. 0 ) NBNHX3 = 20 PRESET.60
204 C PRESET.61
205 33. NALC = 0 PRESET.62
206 34. NOWFL = 0 PRESET.63
207 35. MAXFL = 700000 PRESET.64
208 C PRESET.65
209 36. CALL ALLOC(+X +,NPMAX,2) PRESET.66
210 37. CALL ALLOC(+Y +,NPMAX,2) PRESET.67
211 38. CALL ALLOC(+B +,NPMAX,2) PRESET.68
212 39. CALL ALLOC(+EFIX+,NPMAX,2) PRESET.69
213 40. CALL ALLOC(+ETA +,NPMAX,2) PRESET.70
214 41. CALL ALLOC(+U +,NPMAX,2) PRESET.71
215 42. CALL ALLOC(+V +,NPMAX,2) PRESET.72
216 43. CALL ALLOC(+IELE+,4*NEMAX,1) PRESET.73
217 44. CALL ALLOC(+KELE+,5*NEMAX,1) PRESET.74
218 45. CALL ALLOC(+MELE+,NEMAX,1) PRESET.75
219 46. CALL ALLOC(+CHAT+,2*NMMAX,2) PRESET.76
220 47. CALL ALLOC(+IEBN+,2*NBNHX1,1) PRESET.77
221 48. CALL ALLOC(+QBN +,2*NBNHX1,2) PRESET.78
222 49. CALL ALLOC(+INBN+,3*NBNHX2,1) PRESET.79
223 50. CALL ALLOC(+JBNB+,2*NBNHX3*NBNHX2,1) PRESET.80

```

224	51.	CALL ALLOC(*CBN + , NBNMX3 * NBNMX2 , 2)	PRESET.81
225	52.	CALL ALLOC(*UX + , NEMAX , 2)	PRESET.82
226	53.	CALL ALLOC(*VX + , NEMAX , 2)	PRESET.83
227	54.	CALL ALLOC(*ETA0 + , NPMAX , 2)	PRESET.84
228	55.	CALL ALLOC(*ETA1 + , NPMAX , 2)	PRESET.85
229	56.	CALL ALLOC(*U0 + , NPMAX , 2)	PRESET.86
230	57.	CALL ALLOC(*U1 + , NPMAX , 2)	PRESET.87
231	58.	CALL ALLOC(*V0 + , NPMAX , 2)	PRESET.88
232	59.	CALL ALLOC(*V1 + , NPMAX , 2)	PRESET.89
233	60.	CALL ALLOC(*TSH + , 3 * NPMAX * 2 * NBMAX , 2)	PRESET.90
234	61.	CALL ALLOC(*BMAT + , 3 * NPMAX , 2)	PRESET.91
235	62.	CALL ALLOC(*BBN + , NBNMX2 , 2)	PRESET.92
236			PRESET.93
237	63.	CALL REDUCE	PRESET.94
238	64.	CALL CHGFL1(A , NOWFL + 1024)	PRESET.95
239			PRESET.96
240	65.	RETURN	PRESET.97
241	66.	END	PRESET.98

242	1.	SUBROUTINE ALLOC(OUM , N , ITYPE)	ALLOC.2
243			ALLOC.3
244	2.	COMMON / ALLOC / IALC(100) , NALC , NOWFL , MAXFL	/ALLOC/.2
245	3.	COMMON / FILE / LIN , LOUT , LU1 , LU2 , LU3 , LU4 ,	/FILE/.2
246		* LU10 , LU11 , LU12 , LU13 , LU14 , LU15	/FILE/.3
247	4.	COMMON / ERROR / IERROR	/ERROR/.2
248			ALLOC.7
249	5.	DIMENSION IWDS(2)	ALLOC.8
250	6.	DATA IWDS / 1 , 1 /	ALLOC.9
251			ALLOC.10
252	7.	NALC = NALC + 1	ALLOC.11
253	8.	IF (NALC .LE. 100) GO TO 15	ALLOC.12
254	9.	IERROR = 1	ALLOC.13
255	10.	CALL LINE(2)	ALLOC.14
256	11.	WRITE(LOUT,601)	ALLOC.15
257	12.	601 FORMAT(/T21, ***** ERROR ***** SUB. ALLOC NALC .GT. + ,	ALLOC.16
258		* +100+)	ALLOC.17
259	13.	GO TO 999	ALLOC.18
260			ALLOC.19
261	14.	15 CONTINUE	ALLOC.20
262	15.	IALC(NALC) = NOWFL + 1	ALLOC.21
263	16.	NOWFL = NOWFL + N * IWDS(ITYPE)	ALLOC.22
264	17.	IF (NOWFL .LE. MAXFL) GO TO 25	ALLOC.23
265	18.	IERROR = 1	ALLOC.24
266	19.	CALL LINE(2)	ALLOC.25
267	20.	WRITE(LOUT,602) NALC , NOWFL , MAXFL	ALLOC.26
268	21.	602 FORMAT(/T21, ***** ERROR ***** SUB. ALLOC NOWFL .GT. +	ALLOC.27
269		* , +MAXFL NOWFL = + , I7 , + MAXFL = + , I7)	ALLOC.28
270	22.	GO TO 999	ALLOC.29
271			ALLOC.30
272	23.	25 CONTINUE	ALLOC.31
273	24.	IS = NALC + 1	ALLOC.32
274			ALLOC.33
275	25.	DO 10 I = IS , 100	ALLOC.34
276	26.	IALC(I) = NOWFL + 1	ALLOC.35
277	27.	10 CONTINUE	ALLOC.36
278			ALLOC.37
279	28.	999 CONTINUE	ALLOC.38
280			ALLOC.39
281	29.	RETURN	ALLOC.40
282	30.	END	ALLOC.41

```

283      1.      SUBROUTINE FEM                                FEH.2
284      * ( X ,Y ,B ,EFIX,ETA ,U ,V ,IELE,KELE,MELE,      FEH.3
285      *   CHAT,IEBN,QBN ,INBN,JNBN,CBN ,UX ,VX ,ETA0,ETA1,  FEH.4
286      *   U0 ,U1 ,V0 ,V1 ,TSH ,BMAT,BBN ,OUM1,           FEH.5
287      *   NPMAX ,NEMAX ,NBMAX ,NMMAX ,NBNMX1 ,           FEH.6
288      *   NBNMX2 ,NBNMX3 )                                FEH.7
289      C                                                    FEH.8
290      2.      IMPLICIT REAL*8 (A-H,O-Z)                   FEH.9
291      C                                                    FEH.10
292      3.      COMMON / CNTRL / NP ,NE ,NB ,NT ,           /CNTRL/.2
293      *   NBN1 ,NBN2                                     /CNTRL/.3
294      4.      COMMON / CALCSH / ICALSH                     /CALCSH/.2
295      5.      COMMON / ERROR / IERROR                      /ERROR/.2
296      C                                                    FEH.14
297      6.      DIMENSION X(NPMAX),Y(NPMAX),B(NPMAX),EFIX(NPMAX),ETA(NPMAX),  FEH.15
298      *   U(NPMAX),V(NPMAX),IELE(4,NEMAX),KELE(5,NEMAX),  FEH.16
299      *   MELE(NEMAX),CHAT(2,NMMAX),IEBN(2,NBNMX1),      FEH.17
300      *   QBN(2,NBNMX1),INBN(3,NBNMX2),JNBN(2,NBNMX3,NBNMX2),  FEH.18
301      *   CBN(NBNMX3,NBNMX2),UX(NEMAX),VX(NEMAX),        FEH.19
302      *   ETA0(NPMAX),ETA1(NPMAX),U0(NPMAX),U1(NPMAX),  FEH.20
303      *   V0(NPMAX),V1(NPMAX),TSH(3*NP,2*NBMAX),        FEH.21
304      *   BMAT(3*NP,2*NBMAX),BBN(NBNMX2)                FEH.22
305      C                                                    FEH.23
306      7.      CALL DATAPR                                  FEH.24
307      C                                                    FEH.25
308      8.      CALL DATAIN                                  FEH.26
309      * ( X ,Y ,B ,EFIX,IELE,KELE,MELE,CHAT,IEBN,QBN ,  FEH.27
310      *   INBN,JNBN,CBN ,BBN ,                           FEH.28
311      *   NPMAX ,NEMAX ,NBMAX ,NMMAX ,NBNMX1 ,           FEH.29
312      *   NBNMX2 ,NBNMX3 )                                FEH.30
313      9.      IF ( IERROR .GT. 0 ) GO TO 999              FEH.31
314      C                                                    FEH.32
315      10.     CALL PFILE1(X,Y,IELE,NP,NE,NPMAX,NEMAX)     FEH.33
316      C                                                    FEH.34
317      11.     CALL MODIFY                                  FEH.35
318      * ( X ,Y ,B ,EFIX,IELE,KELE,MELE,CHAT,IEBN,QBN ,  FEH.36
319      *   INBN,JNBN,CBN ,BBN ,                           FEH.37
320      *   NPMAX ,NEMAX ,NBMAX ,NMMAX ,NBNMX1 ,           FEH.38
321      *   NBNMX2 ,NBNMX3 )                                FEH.39
322      12.     IF ( ICALSH .GT. 1 ) GO TO 999              FEH.40
323      13.     IF ( IERROR .GT. 0 ) GO TO 999              FEH.41
324      C                                                    FEH.42
325      14.     CALL IVSET (X,Y,B,EFIX,ETA,U,V,CHAT,NPMAX,NMMAX)  FEH.43
326      15.     IF ( IERROR .GT. 0 ) GO TO 999              FEH.44
327      C                                                    FEH.45
328      16.     CALL PRINT (U,V,ETA,B,NPMAX,1)              FEH.46
329      17.     IF ( ICALSH .GT. 0 ) GO TO 10               FEH.47
330      C                                                    FEH.48
331      18.     CALL CLMAIN                                  FEH.49
332      * ( X ,Y ,B ,EFIX,ETA ,U ,V ,ETA0,ETA1,U0 ,      FEH.50
333      *   U1 ,V0 ,V1 ,IELE,MELE,CHAT,IEBN,QBN ,INBN,JNBN,  FEH.51
334      *   CBN ,TSH ,BMAT,BBN ,NP ,NE ,NB ,NM ,NBN1,NBN2,  FEH.52
335      *   NPMAX ,NEMAX ,NBMAX ,NMMAX ,NBNMX1 ,           FEH.53
336      *   NBNMX2 ,NBNMX3 ,KELE )                        FEH.54
337      C                                                    FEH.55
338      19.     CALL PRINT (U,V,ETA,B,NPMAX,2)              FEH.56
339      C                                                    FEH.57
340      20.     CALL CONVPR                                  FEH.58
341      C                                                    FEH.59
342      21.     10 CONTINUE                                  FEH.59
343      22.     CALL PFILE2(X,Y,U,V,ETA,IELE,KELE,UX,VX,NP,NE,NPMAX,NEMAX)  FEH.60
344      C                                                    FEH.61
345      23.     999 CONTINUE                                  FEH.62
346      C                                                    FEH.63
347      24.     RETURN                                       FEH.64
348      25.     END                                          FEH.65

```

```

349      1.      SUBROUTINE DATAPR                                DATAPR.2
350      C                                                    DATAPR.3
351      2.      COMMON / TITLE / ITIT(20)                   /TITLE/.2
352      3.      COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,  /FILE/.2
353      *   LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15              /FILE/.3
354      C                                                    DATAPR.6
355      4.      DIMENSION IBUF(20)                          DATAPR.7
356      C                                                    DATAPR.8
357      5.      REWIND LU4                                    DATAPR.9
358      C                                                    DATAPR.10
359      6.      READ(LIN,500,END=999) IBUF                  DATAPR.11
360      7.      500 FORMAT( 20A4 )                          DATAPR.12

```

361	8.	CALL MOVE1(IBUF,ITII,20)	DATA PR.13
362	C		DATA PR.14
363	9.	KNT = 1	DATA PR.15
364	C		DATA PR.16
365	10.	222 CONTINUE	DATA PR.17
366	11.	CALL PAGE	DATA PR.18
367	12.	CALL LINE(4)	DATA PR.19
368	13.	WRITE(LOUT,601)	DATA PR.20
369	14.	601 FORMAT(/T31, +INPUT DATA CARD IMAGE+)	DATA PR.21
370	15.	WRITE(LOUT,602) (I,I=10,80,5)	DATA PR.22
371	16.	602 FORMAT(/T6, +NO. 5+, 15(+...+, 12) /)	DATA PR.23
372	C		DATA PR.24
373	17.	111 CONTINUE	DATA PR.25
374	18.	CALL LINE(1)	DATA PR.26
375	19.	WRITE(LOUT,603) KNT,IBUF	DATA PR.27
376	20.	WRITE(LU4,500) IBUF	DATA PR.28
377	21.	603 FORMAT(I8, 2X, 20A4)	DATA PR.29
378	C		DATA PR.30
379	22.	READ(LIN,500,END=999) IBUF	DATA PR.31
380	C		DATA PR.32
381	23.	KNT = KNT + 1	DATA PR.33
382	24.	IF (MOD(KNT,50) .EQ. 1) GO TO 222	DATA PR.34
383	25.	GO TO 111	DATA PR.35
384	C		DATA PR.36
385	25.	999 CONTINUE	DATA PR.37
386	C		DATA PR.38
387	27.	RETURN	DATA PR.39
388	28.	END	DATA PR.40

389	1.	SUBROUTINE DATAIN	DATAIN.2
390		* (X , Y , B , EFIX,IELE,XELE,MELE,CHAT,IEBN,QBN ,	DATAIN.3
391		* INBN,JBN,CBN ,QBN ,	DATAIN.4
392		* NPMAX ,NEMAX ,NBMAX ,NMHAX ,NBNMX1 ,	DATAIN.5
393		* NBNMX2 ,NBNMX3)	DATAIN.6
394	C		DATAIN.7
395	2.	IMPLICIT REAL*8 (A-H,O-Z)	DATAIN.8
396	C		DATAIN.9
397	3.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 , /FILE/.2	
398		* LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15 /FILE/.3	
399	4.	COMMON / ERROR / IERROR	/ERROR/.2
400	5.	COMMON / CNTRL / NP ,NE ,NB ,NF ,	/CNTRL/.2
401		* NBN1 ,NBN2	/CNTRL/.3
402	6.	COMMON / CALCSH / ICALSH	/CALCSH/.2
403	7.	COMMON / ITERAT / MAXIT ,MINER ,IFLG(3) ,EPSLN(3)	/ITERAT/.2
404	8.	COMMON / ISTEP / ICOUNT ,DT ,TIME	/ISTEP/.2
405	9.	COMMON / GAUSS / NGS ,G(5,4) ,H(5,4)	/GAUSS/.2
406	10.	COMMON / CONST / ZERO ,ONE ,TWO	/CONST/.2
407	11.	COMMON / POINT / NPOINT ,IPL(10) ,IPR(10) ,CA(10) ,CB(10)	/POINT/.2
408	12.	COMMON / CHECK / ICHECK(10)	/CHECK/.2
409	C		DATAIN.20
410	13.	DIMENSION X(NPMAX) ,Y(NPMAX) ,B(NPMAX) ,EFIX(NPMAX) ,	DATAIN.21
411		* IELE(4,NEMAX) ,KELE(5,NEMAX) ,MELE(NEMAX) ,	DATAIN.22
412		* CHAT(2,NHMAX) ,IEBN(2,NBNMX1) ,QBN(2,NBNMX1) ,	DATAIN.23
413		* INBN(3,NBNMX2) ,JBN(2,NBNMX3,NBNMX2) ,	DATAIN.24
414		* CBN(NBNMX3,NBNMX2) ,QBN(NBNMX2)	DATAIN.25
415	C		DATAIN.26
416	14.	IZERO = 0	DATAIN.27
417	15.	CALL HSET2(ZERO,X ,NPMAX)	DATAIN.28
418	16.	CALL HSET2(ZERO,Y ,NPMAX)	DATAIN.29
419	17.	CALL HSET2(ZERO,B ,NPMAX)	DATAIN.30
420	18.	CALL HSET2(ZERO,EFIX,NPMAX)	DATAIN.31
421	19.	CALL HSET1(IZERO,IELE,4*NEMAX)	DATAIN.32
422	20.	CALL HSET1(IZERO,KELE,5*NEMAX)	DATAIN.33
423	21.	CALL HSET1(IZERO,MELE,NEMAX)	DATAIN.34
424	22.	CALL HSET2(ZERO,CHAT,2*NHMAX)	DATAIN.35
425	23.	CALL HSET1(IZERO,IEBN,2*NBNMX1)	DATAIN.36
426	24.	CALL HSET2(ZERO,QBN ,2*NBNMX1)	DATAIN.37
427	25.	CALL HSET1(IZERO,INBN,3*NBNMX2)	DATAIN.38
428	26.	CALL HSET1(IZERO,JBN,2*NBNMX3*NBNMX2)	DATAIN.39
429	27.	CALL HSET2(ZERO,CBN ,NBNMX3*NBNMX2)	DATAIN.40
430	28.	NBN1 = 0	DATAIN.41
431	29.	NBN2 = 0	DATAIN.42
432	C		DATAIN.43
433	30.	REWIND LU4	DATAIN.44
434	C		DATAIN.45
435	31.	READ(LU4,401) DUM	DATAIN.46
436	C		DATAIN.47
437	C	 CONTROL DATA	DATAIN.48
438	32.	READ (LU4,402) NP, NE, NM, NBN, DT, ICALSH, NGS, MAXIT,	DATAIN.49
439		* NPOINT, MINER, EPSLN, ICHECK, TOTALQ	DATAIN.50
440	33.	IF (NPOINT .GT. 0) READ (LU4,408)	DATAIN.51

441	*	(IPL(I), IPR(I), CA(I), CB(I), I=1,NPOINT)	DATAIN.52
442	34.	IF (NP .GE. 3 .AND. NP .LE. NPHAX) GO TO 15	DATAIN.53
443	35.	IERORR = 1	DATAIN.54
444	36.	CALL LINE(2)	DATAIN.55
445	37.	WRITE(LOUT,601) NP,NPHAX	DATAIN.56
446	38.	601 FORMAT(/I21, +***** ERROR ***** NP=+,I5, + NPHAX=+,	DATAIN.57
447	*	I5)	DATAIN.58
448	39.	15 CONTINUE	DATAIN.59
449	40.	IF (NE .GE. 1 .AND. NE .LE. NEMAX) GO TO 25	DATAIN.60
450	41.	IERORR = 1	DATAIN.61
451	42.	CALL LINE(2)	DATAIN.62
452	43.	WRITE(LOUT,602) NE,NEMAX	DATAIN.63
453	44.	602 FORMAT(/I21, +***** ERROR ***** NE=+,I5, + NEMAX=+,	DATAIN.64
454	*	I5)	DATAIN.65
455	45.	25 CONTINUE	DATAIN.66
456	46.	IF (NM .GE. 1 .AND. NM .LE. NMMAX) GO TO 35	DATAIN.67
457	47.	IERORR = 1	DATAIN.68
458	48.	CALL LINE(2)	DATAIN.69
459	49.	WRITE(LOUT,603) NM,NMMAX	DATAIN.70
460	50.	603 FORMAT(/I21, +***** ERROR ***** NM=+,I5, + NMMAX=+,	DATAIN.71
461	*	I5)	DATAIN.72
462	51.	35 CONTINUE	DATAIN.73
463	52.	IF (NBN .GE. 1) GO TO 45	DATAIN.74
464	53.	IERORR = 1	DATAIN.75
465	54.	CALL LINE(2)	DATAIN.76
466	55.	WRITE(LOUT,604) NBN	DATAIN.77
467	56.	604 FORMAT(/I21, +***** ERROR ***** NBN=+, I5)	DATAIN.78
468	57.	45 CONTINUE	DATAIN.79
469	58.	IF (DT .GT. 0.0) GO TO 55	DATAIN.80
470	59.	IERORR = 1	DATAIN.81
471	60.	CALL LINE(2)	DATAIN.82
472	61.	WRITE(LOUT,605) DT	DATAIN.83
473	62.	605 FORMAT(/I21, +***** ERROR ***** DT=+,F10.3)	DATAIN.84
474	63.	55 CONTINUE	DATAIN.85
475	64.	NGS = MAX0(2,NGS)	DATAIN.86
476	65.	IF (NGS .LE. 5) GO TO 65	DATAIN.87
477	66.	IERORR = 1	DATAIN.88
478	67.	CALL LINE(2)	DATAIN.89
479	68.	WRITE(LOUT,606) NGS	DATAIN.90
480	69.	606 FORMAT(/I21, +***** ERROR ***** NGS=+, I5)	DATAIN.91
481	70.	65 CONTINUE	DATAIN.92
482	71.	IF (IERORR .GT. 0) GO TO 999	DATAIN.93
483	C		DATAIN.94
484	72.	IFLG(1) = 1	DATAIN.95
485	73.	IFLG(2) = 1	DATAIN.96
486	74.	IFLG(3) = 1	DATAIN.97
487	75.	IF (EPSLN(1) .LE. 0.0) EPSLN(1) = 1.0010	DATAIN.98
488	76.	IF (EPSLN(2) .LE. 0.0) EPSLN(2) = 1.0010	DATAIN.99
489	77.	IF (EPSLN(3) .LE. 0.0) EPSLN(3) = 1.0010	DATAIN.100
490	C		DATAIN.101
491	C	 NODE DATA	DATAIN.102
492	78.	DO 10 I=1,NP	DATAIN.103
493	79.	READ(LU4,403) NO,XX,YY,BB,EE	DATAIN.104
494	80.	IF (NO .GE. 1 .AND. NO .LE. NP) GO TO 75	DATAIN.105
495	81.	IERORR = 1	DATAIN.106
496	82.	CALL LINE(2)	DATAIN.107
497	83.	WRITE(LOUT,607) NO,XX,YY	DATAIN.108
498	84.	607 FORMAT(/I21, +***** ERROR ***** NODE NO.=+, I5,	DATAIN.109
499	*	+ (X,Y)=+, 2F10.3)	DATAIN.110
500	85.	GO TO 10	DATAIN.111
501	86.	75 CONTINUE	DATAIN.112
502	87.	X(NO) = XX	DATAIN.113
503	88.	Y(NO) = YY	DATAIN.114
504	89.	B(NO) = BB	DATAIN.115
505	90.	EFIX(NO)=EE	DATAIN.116
506	91.	10 CONTINUE	DATAIN.117
507	C		DATAIN.118
508	C	 ELEMENT DATA	DATAIN.119
509	92.	DO 20 I=1,NE	DATAIN.120
510	93.	READ(LU4,404) NO,II,JJ,KK,LL,M	DATAIN.121
511	94.	K = 4	DATAIN.122
512	95.	IF (LL .EQ. 0) K = 3	DATAIN.123
513	96.	IF (LL .EQ. 0) LL = II	DATAIN.124
514	97.	M = MAX0(1,M)	DATAIN.125
515	98.	IF (NO .GE. 1 .AND. NO .LE. NE) GO TO 85	DATAIN.126
516	99.	IERORR = 1	DATAIN.127
517	100.	CALL LINE(2)	DATAIN.128
518	101.	WRITE(LOUT,608) NO,II,JJ,KK,LL	DATAIN.129
519	102.	608 FORMAT(/I21, +***** ERROR ***** ELEM NO.=+, I5,	DATAIN.130
520	*	+ (I,J,K,L)=+,4I5)	DATAIN.131

521	103.	85	CONTINUE						DATAIN.132
522	104.		IF ((II .GE. 1 .AND. II .LE. NP) .AND.						DATAIN.133
523		*	(JJ .GE. 1 .AND. JJ .LE. NP) .AND.						DATAIN.134
524		*	(KK .GE. 1 .AND. KK .LE. NP) .AND.						DATAIN.135
525		*	(LL .GE. 1 .AND. LL .LE. NP))			GO TO 95			DATAIN.136
526	105.		IERROR = 1						DATAIN.137
527	106.		CALL LINE(2)						DATAIN.138
528	107.		WRITE(LOUT,608) NO,II,JJ,KK,LL						DATAIN.139
529	108.	95	CONTINUE						DATAIN.140
530	109.		IF (H .GE. 1 .AND. H .LE. NMHAX)			GO TO 105			DATAIN.141
531	110.		IERROR = 1						DATAIN.142
532	111.		CALL LINE(2)						DATAIN.143
533	112.		WRITE(LOUT,609) NO,H						DATAIN.144
534	113.	609	FORMAT(/T21, +***** ERROR *****			ELEM NO.=+,	I5,		DATAIN.145
535		*	+ H=+,	I5	)				DATAIN.146
536	114.	105	CONTINUE						DATAIN.147
537	115.		IF (IERROR .GT. 0)			GO TO 20			DATAIN.148
538	116.		IELE(1,NO) = II						DATAIN.149
539	117.		IELE(2,NO) = JJ						DATAIN.150
540	118.		IELE(3,NO) = KK						DATAIN.151
541	119.		IELE(4,NO) = LL						DATAIN.152
542	120.		KELE(5,NO) = K						DATAIN.153
543	121.		MELE(NO) = H						DATAIN.154
544	122.	20	CONTINUE						DATAIN.155
545		C							DATAIN.156
546		C	 MATERIAL DATA						DATAIN.157
547	123.		DO 30 I=1,NM						DATAIN.158
548	124.		READ(LU4,403) NO,EPS,C						DATAIN.159
549	125.		IF (NO .GE. 1 .AND. NO .LE. NMHAX)			GO TO 115			DATAIN.160
550	126.		IERROR = 1						DATAIN.161
551	127.		CALL LINE(2)						DATAIN.162
552	128.		WRITE(LOUT,610) NO,NMHAX						DATAIN.163
553	129.	610	FORMAT(/T21, +***** ERROR *****			HAT NO.=+,I5,			DATAIN.164
554		*	+ NMHAX=+,	I5	)				DATAIN.165
555	130.		GO TO 30						DATAIN.166
556	131.	115	CONTINUE						DATAIN.167
557	132.		CHAT(1,NO) = EPS						DATAIN.168
558	133.		CHAT(2,NO) = C						DATAIN.169
559	134.	30	CONTINUE						DATAIN.170
560		C							DATAIN.171
561		C	 BOUNDARY CONDITION DATA						DATAIN.172
562	135.		DO 40 I=1,NBN						DATAIN.173
563	136.		READ(LU4,404) ISW						DATAIN.174
564	137.		IF (ISW .EQ. 1 .OR. ISW .EQ. 2)			GO TO 125			DATAIN.175
565	138.		IERROR = 1						DATAIN.176
566	139.		CALL LINE(2)						DATAIN.177
567	140.		WRITE(LOUT,611) ISW						DATAIN.178
568	141.	611	FORMAT(/T21, +***** ERROR *****			BOUNDARY CONDITION+,			DATAIN.179
569		*	+ SWITCH=+,	I5	)				DATAIN.180
570	142.		GO TO 999						DATAIN.181
571	143.	125	CONTINUE						DATAIN.182
572	144.		BACKSPACE LU4						DATAIN.183
573		C							DATAIN.184
574	145.		GO TO (11,22) , ISW						DATAIN.185
575	146.	11	CONTINUE						DATAIN.186
576	147.		READ(LU4,405) IE,IS,Q1,Q2						DATAIN.187
577	148.		IF (IE .GE. 1 .AND. IE .LE. NE)			GO TO 135			DATAIN.188
578	149.		IERROR = 1						DATAIN.189
579	150.		CALL LINE(2)						DATAIN.190
580	151.		WRITE(LOUT,612) IE						DATAIN.191
581	152.	612	FORMAT(/T21, +***** ERROR *****			BOUNDARY CONDITION+,			DATAIN.192
582		*	+ ELEM NO.=+,	I5	)				DATAIN.193
583	153.	135	CONTINUE						DATAIN.194
584	154.		IF (IS .GE. 1 .AND. IS .LE. 4)			GO TO 145			DATAIN.195
585	155.		IERROR = 1						DATAIN.196
586	156.		CALL LINE(2)						DATAIN.197
587	157.		WRITE(LOUT,613) IE,IS						DATAIN.198
588	158.	613	FORMAT(/T21, +***** ERROR *****			BOUNDARY CONDITION+,			DATAIN.199
589		*	+ ELEM NO.=+,	I5,	+ SURFACE NO.=+,	I5	)		DATAIN.200
590	159.	145	CONTINUE						DATAIN.201
591	160.		NBN1 = NBN1 + 1						DATAIN.202
592	161.		IF (NBN1 .LE. NBNHX1)			GO TO 155			DATAIN.203
593	162.		IERROR = 1						DATAIN.204
594	163.		CALL LINE(2)						DATAIN.205
595	164.		WRITE(LOUT,614) NBN1,NBNHX1						DATAIN.206
596	165.	614	FORMAT(/T21, +***** ERROR *****			NBN1=+,	I5,		DATAIN.207
597		*	+ NBNHX1=+,	I5	)				DATAIN.208
598	166.		GO TO 999						DATAIN.209
599	167.	155	CONTINUE						DATAIN.210
600	168.		IEGN(1,NBN1) = IE						DATAIN.211

601	169.	IEBN(2,NBN1) = IS	DATAIN.212
602	170.	QBN(1,NBN1) = Q1	DATAIN.213
603	171.	QBN(2,NBN1) = Q2	DATAIN.214
604	172.	GO TO 40	DATAIN.215
605			DATAIN.216
606	173.	22 CONTINUE	DATAIN.217
607	174.	READ(LU4,406) IONO,IOFD,NO,VAL	DATAIN.218
608	175.	IF (IONO .GE. 1 .AND. IONO .LE. NP) GO TO 165	DATAIN.219
609	176.	IERROR = 1	DATAIN.220
610	177.	CALL LINE(2)	DATAIN.221
611	178.	WRITE(LOUT,615) IONO	DATAIN.222
612	179.	615 FORMAT(/T21, +***** ERROR ***** IONO=+, 15)	DATAIN.223
613	180.	165 CONTINUE	DATAIN.224
614	181.	IF (IOFD .GE. 1 .AND. IOFD .LE. 3) GO TO 175	DATAIN.225
615	182.	IERROR = 1	DATAIN.226
616	183.	CALL LINE(2)	DATAIN.227
617	184.	WRITE(LOUT,616) IOFU	DATAIN.228
618	185.	616 FORMAT(/T21, +***** ERROR ***** IOFD=+, 15)	DATAIN.229
619	186.	175 CONTINUE	DATAIN.230
620	187.	IF (NO .GE. 1 .AND. NO .LE. NBNMX3) GO TO 185	DATAIN.231
621	188.	IERROR = 1	DATAIN.232
622	189.	CALL LINE(2)	DATAIN.233
623	190.	WRITE(LOUT,617) NO,NBNMX3	DATAIN.234
624	191.	617 FORMAT(/T21, +***** ERROR ***** NO=+,15, + NBNMX3=+,	DATAIN.235
625		15)	DATAIN.236
626	192.	GO TO 999	DATAIN.237
627	193.	185 CONTINUE	DATAIN.238
628	194.	NBN2 = NBN2 + 1	DATAIN.239
629	195.	IF (NBN2 .GE. 1 .AND. NBN2 .LE. NBNMX2) GO TO 195	DATAIN.240
630	196.	IERROR = 1	DATAIN.241
631	197.	CALL LINE(2)	DATAIN.242
632	198.	WRITE(LOUT,618) NBN2,NBNMX2	DATAIN.243
633	199.	618 FORMAT(/T21, +***** ERROR ***** NBN2=+, 15,	DATAIN.244
634		+ NBNMX2=+, 15)	DATAIN.245
635	200.	GO TO 999	DATAIN.246
636	201.	195 CONTINUE	DATAIN.247
637	202.	INBN(1,NBN2) = IONO	DATAIN.248
638	203.	INBN(2,NBN2) = IOFD	DATAIN.249
639	204.	INBN(3,NBN2) = NO	DATAIN.250
640	205.	UBN(NBN2) = VAL	DATAIN.251
641			DATAIN.252
642	206.	READ(LU4,407) ((JNBN(K,J,NBN2),K=1,2),CBN(J,NBN2),J=1,NO)	DATAIN.253
643	207.	40 CONTINUE	DATAIN.254
644			DATAIN.255
645	208.	DO 50 J=1,NO	DATAIN.256
646	209.	INNO = JNBN(1,J,NBN2)	DATAIN.257
647	210.	INFO = JNBN(2,J,NBN2)	DATAIN.258
648	211.	IF (INNO .GE. 1 .AND. INNO .LE. NP) GO TO 205	DATAIN.259
649	212.	IERROR = 1	DATAIN.260
650	213.	CALL LINE(2)	DATAIN.261
651	214.	WRITE(LOUT,619) INNO	DATAIN.262
652	215.	619 FORMAT(/T21, +***** ERROR ***** INNO=+, 15)	DATAIN.263
653	216.	205 CONTINUE	DATAIN.264
654	217.	IF (INFO .GE. 1 .AND. INFO .LE. 3) GO TO 50	DATAIN.265
655	218.	IERROR = 1	DATAIN.266
656	219.	CALL LINE(2)	DATAIN.267
657	220.	WRITE(LOUT,620) INFO	DATAIN.268
658	221.	620 FORMAT(/T21, +***** ERROR ***** INFO=+, 15)	DATAIN.269
659	222.	50 CONTINUE	DATAIN.270
660			DATAIN.271
661	223.	IF (NPOINT .LE. 0) GO TO 999	DATAIN.272
662	224.	CALL BOUND3 (X, Y, INBN, BBN, JNBN, CBN, NPMAX, NBNMX2, NBNMX3)	DATAIN.273
663			DATAIN.274
664	225.	999 CONTINUE	DATAIN.275
665	226.	401 FORMAT(20A4)	DATAIN.276
666	227.	402 FORMAT(4I5, F5.0, 5I5, 3F10.0 / 10I1, F10.0)	DATAIN.277
667	228.	403 FORMAT(15, 5X, 4F10.0)	DATAIN.278
668	229.	404 FORMAT(15, 5X, 5I5)	DATAIN.279
669	230.	405 FORMAT(10X, 2I5, 2F10.0)	DATAIN.280
670	231.	406 FORMAT(10X, 3I5, 5X, F10.0)	DATAIN.281
671	232.	407 FORMAT(4(2I5,F10.0))	DATAIN.282
672	233.	408 FORMAT(2I5, 2F10.0)	DATAIN.283
673			DATAIN.284
674	234.	RETURN	DATAIN.285
675	235.	END	DATAIN.286

676	1.	SUBROUTINE BOUND3	BOUND3.2
677		*(X, Y, INBN, BBN, JBN, CBN, NPHAX, NBNMX2, NBNMX3)	BOUND3.3
678	C		BOUND3.4
679	2.	IMPLICIT REAL*8 (A-H, O-Z)	BOUND3.5
680	C		BOUND3.6
681	3.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
682		* COMMON / ERROR / IERROR	/ERROR/.2
683	4.	COMMON / CNTRL / NP ,NE ,NO ,NM ,	/CNTRL/.2
684	5.	* COMMON / POINT / NBN1 ,NBN2 ,NBN(10) ,IPR(10) ,CA(10) ,CB(10)	/POINT/.2
685	6.		BOUND3.11
686	C		BOUND3.12
687	7.	DIMENSION X(NPHAX), Y(NPHAX), INBN(3,NBNMX2), BBN(NBNMX2),	BOUND3.13
688		* JBN(2,NBNMX3,NBNMX2), CBN(NBNMX3,NBNMX2)	BOUND3.14
689	C		BOUND3.15
690	8.	DO 999 I = 1, NPOINT	BOUND3.16
691	C		BOUND3.17
692	9.	IPL = IPL(I)	BOUND3.18
693	10.	IIPR = IPR(I)	BOUND3.19
694	11.	CCA = CA(I)	BOUND3.20
695	12.	CCB = CB(I)	BOUND3.21
696	13.	IF (IPL.LT.1 .OR. IPL.GT.NP) IERROR = 1	BOUND3.22
697	14.	IF (IIPR.LT.1 .OR. IIPR.GT.NP) IERROR = 1	BOUND3.23
698	15.	IF (IPL .EQ. IIPR) IERROR = 1	BOUND3.24
699	16.	IF (CCA .EQ. 0.0) IERROR = 1	BOUND3.25
700	17.	IF (IERROR .EQ. 0) GO TO 10	BOUND3.26
701	C		BOUND3.27
702	18.	CALL LINE(2)	BOUND3.28
703	19.	WRITE(LOUT,601) I, IPL, IIPR, CCA, CCB	BOUND3.29
704	20.	601 FORMAT(/T21, +***** ERROR ***** HASEN NO.=+, I5, + IPL=+,	BOUND3.30
705		* I5, + IPR=+, I5, + CA=+, F10.3, + CB=+, F10.3)	BOUND3.31
706	21.	GO TO 999	BOUND3.32
707	C		BOUND3.33
708	22.	10 CONTINUE	BOUND3.34
709	23.	N = IABS(IPL - IIPR) + 1	BOUND3.35
710	24.	NO = N * 2	BOUND3.36
711	25.	IF (NO .LE. NBNMX3) GO TO 20	BOUND3.37
712	26.	IERROR = 1	BOUND3.38
713	27.	CALL LINE(2)	BOUND3.39
714	28.	WRITE(LOUT,602) NO, NBNMX3	BOUND3.40
715	29.	602 FORMAT(/T21, +***** ERROR ***** NO=+, I5, + NBNMX3=+, I5)	BOUND3.41
716	30.	GO TO 999	BOUND3.42
717	C		BOUND3.43
718	31.	20 CONTINUE	BOUND3.44
719	32.	VAL = - CCB / CCA	BOUND3.45
720	33.	CU = (Y(IPL) - Y(IIPR)) / (2.0 * FLOAT(N-1) * CCA)	BOUND3.46
721	34.	CV = - (X(IPL) - X(IIPR)) / (2.0 * FLOAT(N-1) * CCA)	BOUND3.47
722	35.	IP = IPL	BOUND3.48
723	36.	IF (IPL.LT. IIPR) M = 1	BOUND3.49
724	37.	IF (IPL.GT. IIPR) M = -1	BOUND3.50
725	C		BOUND3.51
726	38.	DO 70 J = 1, N	BOUND3.52
727	39.	NBN2 = NBN2 + 1	BOUND3.53
728	40.	IF (NBN2 .LE. NBNMX2) GO TO 30	BOUND3.54
729	41.	IERROR = 1	BOUND3.55
730	42.	CALL LINE(2)	BOUND3.56
731	43.	WRITE(LOUT,603) NBN2, NBNMX2	BOUND3.57
732	44.	603 FORMAT(/T21, +***** ERROR ***** NBN2=+, I5, + NBNMX2=+, I5)	BOUND3.58
733	45.	GO TO 999	BOUND3.59
734	C		BOUND3.60
735	46.	30 CONTINUE	BOUND3.61
736	47.	INBN(1,NBN2) = IP	BOUND3.62
737	48.	INBN(2,NBN2) = 3	BOUND3.63
738	49.	INBN(3,NBN2) = NO	BOUND3.64
739	50.	BBN (NBN2) = VAL	BOUND3.65
740	C		BOUND3.66
741	51.	IIP = IPL	BOUND3.67
742	52.	DO 60 K = 1, N	BOUND3.68
743	53.	JNBN(1,K ,NBN2) = IIP	BOUND3.69
744	54.	JNBN(1,K+N,NBN2) = IIP	BOUND3.70
745	55.	JNBN(2,K ,NBN2) = 1	BOUND3.71
746	56.	JNBN(2,K+N,NBN2) = 2	BOUND3.72
747	57.	IF (K.EQ.1 .OR. K.EQ.N) GO TO 40	BOUND3.73
748	58.	CBN (K ,NBN2) = 2.0 * CU	BOUND3.74
749	59.	CBN (K+N,NBN2) = 2.0 * CV	BOUND3.75
750	60.	GO TO 50	BOUND3.76
751	61.	40 CONTINUE	BOUND3.77
752	62.	CBN (K ,NBN2) = 1.0 * CU	BOUND3.78
753	63.	CBN (K+N,NBN2) = 1.0 * CV	
754			

755	64.	50 CONTINUE	BOUND3.79
756	65.	IIP = IIP + M	BOUND3.80
757	66.	60 CONTINUE	BOUND3.81
758	C		BOUND3.82
759	67.	IP = IP + M	BOUND3.83
760	68.	70 CONTINUE	BOUND3.84
761	C		BOUND3.85
762	69.	999 CONTINUE	BOUND3.86
763	C		BOUND3.87
764	70.	RETURN	BOUND3.88
765	71.	END	BOUND3.89

766	1.	SUBROUTINE PFILE1(X,Y,IELE,NP,NE,NPHAX,NEHAX)	PFILE1.2
767	C		PFILE1.3
768	2.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
769	*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
770	3.	COMMON / TITLE / ITIT(20)	/TITLE/.2
771	C		PFILE1.6
772	4.	REAL*8 X,Y	PFILE1.7
773	5.	DIMENSION X(NPHAX),Y(NPHAX),IELE(4,NEHAX)	PFILE1.8
774	C		PFILE1.9
775	6.	REWIND LU1	PFILE1.10
776	C		PFILE1.11
777	7.	WRITE(LU1) ITIT	PFILE1.12
778	8.	WRITE(LU1) NP	PFILE1.13
779	9.	WRITE(LU1) (X(I),Y(I),I=1,NP)	PFILE1.14
780	10.	WRITE(LU1) NE	PFILE1.15
781	11.	WRITE(LU1) ((IELE(J,I),J=1,4),I=1,NE)	PFILE1.16
782	C		PFILE1.17
783	12.	REWIND LU1	PFILE1.18
784	C		PFILE1.19
785	13.	RETURN	PFILE1.20
786	14.	END	PFILE1.21

787	1.	SUBROUTINE MODIFY	MODIFY.2
788	*	(X , Y , 0 , EFIX,IELE,KELE,MELE,CHAT,IERN,QBN ,	MODIFY.3
789	*	INBN,JBN,CBN ,BBN ,	MODIFY.4
790	*	NPHAX ,NEMAX ,NBMAX ,NMHAX ,NBNHX1 ,	MODIFY.5
791	*	NBNHX2 ,NBNHX3)	MODIFY.6
792	C		MODIFY.7
793	2.	IMPLICIT REAL*8 (A-H,O-Z)	MODIFY.8
794	C		MODIFY.9
795	3.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
796	*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
797	4.	COMMON / CNTRL / NP ,NE ,NB ,NH ,	/CNTRL/.2
798	*	NBN1 ,NBN2	/CNTRL/.3
799	5.	COMMON / CHECK / ICHECK(10)	/CHECK/.2
800	6.	COMMON / ERROR / IERROR	/ERROR/.2
801	C		MODIFY.14
802	7.	DIMENSION X(NPHAX),Y(NPHAX),B(NPHAX),EFIX(NPHAX),	MODIFY.15
803	*	IELE(4,NEMAX),KELE(5,NEMAX),MELE(NEMAX),	MODIFY.16
804	*	CHAT(2,NMHAX),IERN(2,NBNHX1),QBN(2,NBNHX1),	MODIFY.17
805	*	INBN(3,NBNHX2),JBN(2,NBNHX3),NBNHX2,	MODIFY.18
806	*	CBN(NBNHX3,NBNHX2),BBN(NBNHX2)	MODIFY.19
807	C		MODIFY.20
808	8.	ICBK = ICHECK(1)	MODIFY.21
809	C		MODIFY.22
810	9.	CALL PAGE	MODIFY.23
811	10.	CALL PEJECT(1,1,3)	MODIFY.24
812	C		MODIFY.25
813	11.	DO 10 I=1,NP	MODIFY.26
814	12.	CALL PEJECT(1,0,1)	MODIFY.27
815	13.	WRITE(LOUT,601) I,X(I),Y(I),B(I),EFIX(I)	MODIFY.28
816	14.	601 FORMAT(I20, I4, I31, 4F10.3)	MODIFY.29
817	15.	10 CONTINUE	MODIFY.30
818	C		MODIFY.31
819	16.	NBM = 0	MODIFY.32
820	17.	CALL PEJECT(2,1,4)	MODIFY.33
821	C		MODIFY.34
822	18.	DO 20 I=1,NE	MODIFY.35
823	19.	K = KELE(5,I)	MODIFY.36
824	20.	IF (K .GT. 0)	MODIFY.37
825	21.	IERROR = 1	MODIFY.38
826	22.	CALL PEJECT(2,0,2)	MODIFY.39
827	23.	WRITE(LOUT,602) I	MODIFY.40
828	24.	602 FORMAT(/I21, +***** ERROR ***** UNDEFINED ELEMENT NO. = I,	MODIFY.41
829	*	15)	MODIFY.42

830	25.	GO TO 20	MODIFY.43
831	26.	15 CONTINUE	MODIFY.44
932	27.	CALL PEJECT(2,0,1)	MODIFY.45
833	28.	IF (K .EQ. 3) WRITE(LOUT,603) I,(IELE(J,I),J=1,3),HELE(I)	MODIFY.46
834	29.	IF (K .EQ. 4) WRITE(LOUT,604) I,(IELE(J,I),J=1,4),HELE(I)	MODIFY.47
835	30.	603 FORMAT(T20, I4, T30, 3I6, I63, I1)	MODIFY.48
836	31.	604 FORMAT(T20, I4, T30, 4I6, I63, I1)	MODIFY.49
837	C		MODIFY.50
838	32.	IPMAX = 0	MODIFY.51
839	33.	IPMIN = 99999	MODIFY.52
840	34.	00 220 I1 = 1, K	MODIFY.53
841	35.	IP = IELE(I1,I)	MODIFY.54
842	36.	IF (IP .GT. IPMAX) IPMAX = IP	MODIFY.55
843	37.	IF (IP .LT. IPMIN) IPMIN = IP	MODIFY.56
844	38.	00 210 JJ = 1, NBN2	MODIFY.57
845	39.	IF (IP .NE. INBN(1,JJ)) GO TO 210	MODIFY.58
846	40.	NO = INBN(3,JJ)	MODIFY.59
847	41.	00 200 KK = 1, NO	MODIFY.60
848	42.	IIP = JNBN(1,KK,JJ)	MODIFY.61
849	43.	IF (IIP .GT. IPMAX) IPMAX = IIP	MODIFY.62
850	44.	IF (IIP .LT. IPMIN) IPMIN = IIP	MODIFY.63
851	45.	200 CONTINUE	MODIFY.64
852	46.	GO TO 220	MODIFY.65
853	47.	210 CONTINUE	MODIFY.66
854	48.	220 CONTINUE	MODIFY.67
855	C		MODIFY.68
856	49.	HB = (IPMAX - IPMIN + 1) * 3	MODIFY.69
857	50.	IF (ICHK .LE. 1) GO TO 230	MODIFY.70
858	51.	CALL LINE(1)	MODIFY.71
859	52.	WRITE(LOUT,900) IPMIN, IPMAX, HB	MODIFY.72
860	53.	900 FORMAT(+ *CHECK*+, 60X, +IPMIN=+, I4, + IPMAX=+, I4, +	MODIFY.73
861		+ * HB=+, I4)	MODIFY.74
862	C		MODIFY.75
863	54.	230 CONTINUE	MODIFY.76
864	55.	HBM = MAX0(HBM,HB)	MODIFY.77
865	56.	IF (HB .LE. NBMAX) GO TO 20	MODIFY.78
866	57.	IERROR = 1	MODIFY.79
867	58.	CALL PEJECT(2,0,2)	MODIFY.80
868	59.	WRITE(LOUT,605) MB	MODIFY.81
869	60.	605 FORMAT(/I21, +***** ERROR ***** BAND WIDTH OVER MB=+,	MODIFY.82
870		+ I5)	MODIFY.83
871	61.	20 CONTINUE	MODIFY.84
872	C		MODIFY.85
873	62.	NB = HBM	MODIFY.86
874	C		MODIFY.87
875	63.	IF (ICHK .EQ. 0) GO TO 55	MODIFY.88
876	64.	CALL LINE(2)	MODIFY.89
877	65.	WRITE(LOUT,901) NB	MODIFY.90
878	66.	901 FORMAT(+0*CHECK* BAND WIDTH =+, I4)	MODIFY.91
879	C		MODIFY.92
880	67.	55 CONTINUE	MODIFY.93
881	68.	CALL PEJECT(3,1,4)	MODIFY.94
882	C		MODIFY.95
883	69.	DO 30 I=1,NM	MODIFY.96
884	70.	CALL PEJECT(3,0,1)	MODIFY.97
885	71.	WRITE(LOUT,606) I,(CHAT(J,I),J=1,2)	MODIFY.98
886	72.	606 FORMAT(T20, I4, T31, 2F10.3)	MODIFY.99
887	73.	30 CONTINUE	MODIFY.100
888	C		MODIFY.101
889	74.	IF (NBN1 .GT. 0) CALL PEJECT(4,1,6)	MODIFY.102
890	75.	IF (NBN1 .EQ. 0) GO TO 25	MODIFY.103
891	C		MODIFY.104
892	76.	DO 40 I=1,NBN1	MODIFY.105
893	77.	IE = IEEN(1,I)	MODIFY.106
894	78.	IS = IEEN(2,I)	MODIFY.107
895	79.	IF (IS.GE.1 .AND. IS.LE.KELE(5,IE)) GO TO 35	MODIFY.108
896	80.	IERROR = 1	MODIFY.109
897	81.	CALL PEJECT(4,0,2)	MODIFY.110
898	82.	WRITE(LOUT,607) IE,IS	MODIFY.111
899	83.	607 FORMAT(/I21, +***** ERROR ***** ELEM NO.=+, I5,	MODIFY.112
900		+ * SURF NO.=+, I5)	MODIFY.113
901	84.	GO TO 40	MODIFY.114
902	85.	35 CONTINUE	MODIFY.115
903	86.	IP1 = IELE(IS,IE)	MODIFY.116
904	87.	JS = IS + 1	MODIFY.117
905	88.	IF (JS .EQ. 5) JS = 1	MODIFY.118
906	89.	IP2 = IELE(JS,IE)	MODIFY.119
907	90.	KELE(IS,IE) = 1	MODIFY.120
908	91.	CALL PEJECT(4,0,1)	MODIFY.121
909	92.	WRITE(LOUT,608) IE,IP1,QUN(1,I),IP2,QBN(2,I)	MODIFY.122
910	93.	608 FORMAT(T24, I4, T38, I4, F12.3, T63, I4, F12.3)	MODIFY.123

911	94.	40 CONTINUE	MODIFY.124
912	C		MODIFY.125
913	95.	IF (ICHK .LE. 1) GO TO 25	MODIFY.126
914	96.	CALL LINE(2)	MODIFY.127
915	97.	WRITE(LOUT,902)	MODIFY.128
916	98.	902 FORMAT(+0*CHECK* KELE (I=1,4 - UN1 NO. I=5 - ELEMENT FORM)+)	MODIFY.129
917	99.	DO 240 I = 1, NE	MODIFY.130
918	100.	CALL LINE(1)	MODIFY.131
919	101.	WRITE(LOUT,903) 1, (KELE(IS,I), IS=1,5)	MODIFY.132
920	102.	903 FORMAT(I20 ,I4 ,I30 ,5I6)	MODIFY.133
921	103.	240 CONTINUE	MODIFY.134
922	C		MODIFY.135
923	104.	25 CONTINUE	MODIFY.136
924	105.	IF (NBN2 .GT. 0) CALL PEJECT(5,1,4)	MODIFY.137
925	106.	IF (NBN2 .EQ. 0) GO TO 45	MODIFY.138
926	C		MODIFY.139
927	107.	DO 50 I=1,NBN2	MODIFY.140
928	108.	NL = (INBN(3,I)-1) / 2 + 1	MODIFY.141
929	C		MODIFY.142
930	109.	DO 60 J=1,NL	MODIFY.143
931	110.	CALL PEJECT(5,0,1)	MODIFY.144
932	111.	I1 = J*2 - 1	MODIFY.145
933	112.	I2 = MIN(I1+1,INBN(3,I))	MODIFY.146
934	113.	WRITE(LOUT,609) ((JNBN(K,L,I),K=1,2),CBN(L,I),L=I1,I2)	MODIFY.147
935	114.	609 FORMAT(I51, I7, I9, F9.3, I81, I7, I9, F9.3)	MODIFY.148
936	115.	IF (J .EQ. 1) WRITE(LOUT,610) (INBN(K,I),K=1,2),UBN(I)	MODIFY.149
937	116.	610 FORMAT(I1+, I21, I7, I9, F10.3)	MODIFY.150
938	117.	60 CONTINUE	MODIFY.151
939	118.	50 CONTINUE	MODIFY.152
940	C		MODIFY.153
941	119.	45 CONTINUE	MODIFY.154
942	C		MODIFY.155
943	120.	RETURN	MODIFY.156
944	121.	END	MODIFY.157

945	1.	SUBROUTINE PEJECT(ISH,JSH,ILN)	PEJECT.2
946	C		PEJECT.3
947	2.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
948	*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
949	3.	COMMON / PAGEC / IPAGE ,ILINE	/PAGEC/.2
950	C		PEJECT.6
951	4.	ILINE = ILINE + ILN	PEJECT.7
952	5.	IF (ILINE .GT. 63) CALL PAGE	PEJECT.8
953	6.	IF (ILINE .GT. 61 .AND.	PEJECT.9
954	*	JSH .EQ. 1) CALL PAGE	PEJECT.10
955	C		PEJECT.11
956	7.	IF (ILINE .GT. 5 .AND. JSH .EQ. 0) GO TO 999	PEJECT.12
957	C		PEJECT.13
958	8.	IF (ILINE .GT. 5) CALL LINE(1)	PEJECT.14
959	9.	IF (ILINE .GT. 5) WRITE(LOUT,600)	PEJECT.15
960	10.	600 FORMAT(/)	PEJECT.16
961	C		PEJECT.17
962	11.	GO TO (1, 2, 3, 4, 5, 6, 7) , ISH	PEJECT.18
963	12.	1 CONTINUE	PEJECT.19
964	13.	WRITE(LOUT,601)	PEJECT.20
965	14.	601 FORMAT(/I11, +NODE+, I21, +NO.+, I37, +X+, I47, +Y+,	PEJECT.21
966	*	I57, +B+, I66, +EFIX+ /)	PEJECT.22
967	15.	JLN = 3	PEJECT.23
968	16.	GO TO 15	PEJECT.24
969	17.	2 CONTINUE	PEJECT.25
970	18.	WRITE(LOUT,602)	PEJECT.26
971	19.	602 FORMAT(/I11, +ELEMENT NO.+, I35, +I J K L+,	PEJECT.27
972	*	+ MAT NO.+ /)	PEJECT.28
973	20.	JLN = 3	PEJECT.29
974	21.	GO TO 15	PEJECT.30
975	22.	3 CONTINUE	PEJECT.31
976	23.	WRITE(LOUT,603)	PEJECT.32
977	24.	603 FORMAT(/I11, +MATERIAL NO.+, I34, +EPSILON C+, /)	PEJECT.33
978	25.	JLN = 3	PEJECT.34
979	26.	GO TO 15	PEJECT.35
980	27.	4 CONTINUE	PEJECT.36
981	28.	WRITE(LOUT,604)	PEJECT.37
982	29.	604 FORMAT(/I11, +BOUNDARY CONDITION+,	PEJECT.38
983	*	//I21, +ELEM NO.+, I36, +NODE.1+, I49, +Q1+,	PEJECT.39
984	*	I61, +NODE.2+, I74, +Q2+ /)	PEJECT.40
985	30.	JLN = 5	PEJECT.41
986	31.	GO TO 15	PEJECT.42

987	32.	5 CONTINUE	PEJECT.43
988	33.	WRITE(LOUT,605)	PEJECT.44
989	34.	605 FORMAT(/T21, +NODE NO. FRD NO. BETA+,	PEJECT.45
990	*	T51, 2(+NODE NO. FRD NO. ALPHA+,5X) /)	PEJECT.46
991	35.	JLN = 3	PEJECT.47
992	36.	GO TO 15	PEJECT.48
993	37.	6 CONTINUE	PEJECT.49
994	38.	WRITE(LOUT,606)	PEJECT.50
995	39.	606 FORMAT(/T8, +NO.+, T21, +U+, T35, +V+, T48, +ETA+,	PEJECT.51
996	*	T63, +H+ /)	PEJECT.52
997	40.	JLN = 3	PEJECT.53
998	41.	GO TO 15	PEJECT.54
999	42.	7 CONTINUE	PEJECT.55
1000	43.	WRITE(LOUT,607)	PEJECT.56
1001	44.	607 FORMAT(1H0, 10X, +ITERATION+ / 1H, 14X,	PEJECT.57
1002	*	+COUNT 01 TIME+, 10X, +Q(1)+, 8X, +Q(2)+, 4X,	PEJECT.58
1003	*	+Q(1)+Q(2)+, 7X, +ERROR1 ERROR2 ERROR3+ /)	PEJECT.59
1004	45.	JLN = 4	PEJECT.60
1005	C		PEJECT.61
1006	46.	15 CONTINUE	PEJECT.62
1007	47.	IF (ILINE .EQ. 5) ILINE = ILINE + JLN	PEJECT.63
1008	48.	999 CONTINUE	PEJECT.64
1009	C		PEJECT.65
1010	49.	RETURN	PEJECT.66
1011	50.	END	PEJECT.67
1012	1.	SUBROUTINE IVSET (X, Y, B, EFIX, ETA, U, V, CHAT, NPMAX, NMHAX)	IVSET.2
1013	C		IVSET.3
1014	2.	IMPLICIT REAL*8 (A-H, O-Z)	IVSET.4
1015	C		IVSET.5
1016	3.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
1017	*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15 ,	/FILE/.3
1018	4.	COMMON / CNTRL / NP ,NE ,NB ,NM ,	/CNTRL/.2
1019	*	NBN1 ,NBN2 ,	/CNTRL/.3
1020	5.	COMMON / CONST / ZERO ,ONE ,TWO	/CONST/.2
1021	6.	COMMON / SECT / NSECT ,ISL(200) ,ISR(200) ,SH(200) ,SQ(200)	/SECT/.2
1022	7.	COMMON / ERROR / IERROR	/ERROR/.2
1023	C		IVSET.11
1024	8.	DIMENSION X(NPMAX), Y(NPMAX), B(NPMAX), U(NPMAX), V(NPMAX),	IVSET.12
1025	*	EFIX(NPMAX), ETA(NPMAX), CHAT(2,NMHAX)	IVSET.13
1026	C		IVSET.14
1027	C		IVSET.15
1028	C	INITIAL VALUE SET SWITCH INITSW = 1 ... SET IN IVCALC	IVSET.16
1029	C	INITSW = 2 ... INPUT FROM LU4	IVSET.17
1030	C	INITSW = 3 ... INPUT FROM LU3	IVSET.18
1031	C		IVSET.19
		CALL MSET 2 (ZERO,U,NP)	
		CALL MSET 2 (ZERO,V,NP)	
		CALL MSET 2 (ZERO,ETA,NP)	
	C		
1032	9.	READ (LU4,401,END=10) INITSW	IVSET.20
1033	10.	401 FORMAT(15)	IVSET.21
1034	C		IVSET.22
1035	11.	IF (INITSW .EQ. 1) GO TO 100	IVSET.23
1036	12.	IF (INITSW .EQ. 2) GO TO 200	IVSET.24
1037	13.	IF (INITSW .EQ. 3) GO TO 300	IVSET.25
1038	C		IVSET.26
1039	14.	10 CONTINUE	IVSET.27
1040	15.	IERROR = 1	IVSET.28
1041	16.	CALL LINE(2)	IVSET.29
1042	17.	WRITE(LOUT,601)	IVSET.30
1043	18.	601 FORMAT(/T21, +***** ERROR ***** INITIAL VALUE NOT DEFINED+)	IVSET.31
1044	19.	GO TO 999	IVSET.32
1045	C		IVSET.33
1046	20.	100 CONTINUE	IVSET.34
1047	21.	NSECT = 0	IVSET.35
1048	22.	110 CONTINUE	IVSET.36
1049	23.	READ (LU4,403,END=150) IISL, IISR, SSH, SSQ	IVSET.37
1050	24.	403 FORMAT(2I5, 2F10.0)	IVSET.38
1051	25.	NSECT = NSECT + 1	IVSET.39
1052	26.	IF (NSECT .LE. 200) GO TO 120	IVSET.40
1053	27.	IERROR = 1	IVSET.41
1054	28.	CALL LINE(2)	IVSET.42
1055	29.	WRITE(LOUT,604) NSECT	IVSET.43
1056	30.	604 FORMAT(/T21, +***** ERROR ***** SECT DATA DIMENSION OVER+,	IVSET.44
1057	*	+ NSECT=+, 15)	IVSET.45
1058	31.	GO TO 150	IVSET.46
1059	C		IVSET.47

1060	32.	120 CONTINUE		IVSET.48
1061	33.	IF (IISL.LI.1 .OR. IISL.GT.NP)	IERROR = 1	IVSET.49
1062	34.	IF (IISR.LI.1 .OR. IISR.GT.NP)	IERROR = 1	IVSET.50
1063	35.	IF (IISL .EQ. IISR)	IERROR = 1	IVSET.51
1064	36.	IF (IERROR .EQ. 0)	GO TO 130	IVSET.52
1065	37.	CALL LINE(2)		IVSET.53
1066	38.	WRITE(LOUT,605) NSECT, IISL, IISR		IVSET.54
1067	39.	605 FORMAT(/I21, +***** ERROR ***** SECT NO.=+, I5, + ISL=+,		IVSET.55
1068		* I5, + ISR=+, I5)		IVSET.56
1069		C		IVSET.57
1070	40.	130 CONTINUE		IVSET.58
1071	41.	IF (NSECT .EQ. 1)	GO TO 140	IVSET.59
1072	42.	IF (SSH .EQ. 0.0)	SSH = SH(NSECT-1)	IVSET.60
1073	43.	IF (SSQ .EQ. 0.0)	SSQ = SQ(NSECT-1)	IVSET.61
1074	44.	140 CONTINUE		IVSET.62
1075	45.	ISL(NSECT) = IISL		IVSET.63
1076	46.	ISR(NSECT) = IISR		IVSET.64
1077	47.	SH (NSECT) = SSH		IVSET.65
1078	48.	SQ (NSECT) = SSQ		IVSET.66
1079	49.	GO TO 110		IVSET.67
1080		C		IVSET.68
1081	50.	150 CONTINUE		IVSET.69
1082	51.	IF (IERROR .GT. 0)	GO TO 999	IVSET.70
1083		C		IVSET.71
1084	52.	CALL IVCULC (X, Y, B, ETA, U, V, NPMAX)		IVSET.72
1085	53.	GO TO 999		IVSET.73
1086		C		IVSET.74
1087	54.	200 CONTINUE		IVSET.75
1088	55.	~~CALL MSET2 (ZERO, ETA, NP)~~		IVSET.76
1089	56.	DO 220 I = 1, NP		IVSET.77
1090	57.	REAU (LU4,402,END=230) NO, UU, VV, EE		IVSET.78
1091	58.	402 FORMAT(I5, 5X, 3F10.0)		IVSET.79
1092	59.	IF (NO.GE.1 .AND. NC.LE.NP)	GO TO 210	IVSET.80
1093	60.	IERROR = 1		IVSET.81
1094	61.	CALL LINE(2)		IVSET.82
1095	62.	WRITE(LOUT,602) NO		IVSET.83
1096	63.	602 FORMAT(/I21, +***** ERROR ***** NODE NO.=+, I5)		IVSET.84
1097	64.	GO TO 220		IVSET.85
1098		C		IVSET.86
1099	65.	210 CONTINUE		IVSET.87
1100	66.	U(NO) = UU		IVSET.88
1101	67.	V(NO) = VV		IVSET.89
1102	68.	ETA(NO) = EE		IVSET.90
1103	69.	220 CONTINUE		IVSET.91
1104		C		IVSET.92
1105	70.	230 CONTINUE		IVSET.93
1106	71.	DO 240 I = 1, NP		IVSET.94
1107	72.	IF (ETA(I) .NE. ZERO)	GO TO 240	IVSET.95
1108	73.	IERROR = 1		IVSET.96
1109	74.	CALL LINE(2)		IVSET.97
1110	75.	WRITE(LOUT,603) I		IVSET.98
1111	76.	603 FORMAT(/I21, +***** ERROR ***** UNDEFINED NODE NO.=+, I5)		IVSET.99
1112	77.	240 CONTINUE		IVSET.100
1113	78.	GO TO 999		IVSET.101
1114		C		IVSET.102
1115	79.	300 CONTINUE		IVSET.103
1116	80.	REWIND LU3		IVSET.104
1117		C		IVSET.105
1118	81.	READ (LU3) (U(I), I=1,NP)		IVSET.106
1119	82.	READ (LU3) (V(I), I=1,NP)		IVSET.107
1120	83.	READ (LU3) (ETA(I), I=1,NP)		IVSET.108
1121		C		IVSET.109
1122	84.	REWIND LU3		IVSET.110
1123		C		IVSET.111
1124	85.	999 CONTINUE		IVSET.112
1125		C		IVSET.113
1126	86.	RETURN		IVSET.114
1127	87.	END		IVSET.115

1128	1.	SUBROUTINE IVCULC (X, Y, B, ETA, U, V, NPMAX)	IVCULC.2
1129	C		IVCULC.3
1130	2.	IMPLICIT REAL*8 (A-H, O-Z)	IVCULC.4
1131	C		IVCULC.5
1132	3.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 , /FILE/.2	IVCULC.10
1133	*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15 , /FILE/.3	IVCULC.11
1134	4.	COMMON / ERROR / IERROR	IVCULC.12
1135	5.	COMMON / CNTRL / NP ,NE ,NB ,NM , /CNTRL/.2	IVCULC.13
1136	*	NBN1 ,NBN2 , /CNTRL/.3	IVCULC.14
1137	6.	COMMON / SECT / NSECT ,ISL(200) ,ISR(200) ,SH(200) ,SQ(200) /SECT/.2	IVCULC.15
1138	C		IVCULC.16
1139	7.	DIMENSION X(NPMAX), Y(NPMAX), B(NPMAX), ETA(NPMAX),	IVCULC.17
1140	*	U(NPMAX), V(NPMAX)	IVCULC.18
1141	C		IVCULC.19
1142	8.	DO 50 I = 1, NSECT	IVCULC.20
1143	C		IVCULC.21
1144	9.	IISL = ISL(I)	IVCULC.22
1145	10.	IISR = ISR(I)	IVCULC.23
1146	11.	SSH = SH(I)	IVCULC.24
1147	12.	SSQ = SQ(I)	IVCULC.25
1148	13.	N = IABS(IISL - IISR)	IVCULC.26
1149	14.	A = 0.0	IVCULC.27
1150	15.	IS = IISL	IVCULC.28
1151	16.	IF (IISL .LT. IISR) M = 1	IVCULC.29
1152	17.	IF (IISL .GT. IISR) M = -1	IVCULC.30
1153	C		IVCULC.31
1154	18.	DO 10 J = 1, N	IVCULC.32
1155	19.	HH = ((SSH-B(IS)) + (SSH-B(IS+M))) / 2.0	IVCULC.33
1156	20.	A = A + HH + DSQRT((X(IS)-X(IS+M))**2 + (Y(IS)-Y(IS+M))**2)	IVCULC.34
1157	21.	IS = IS + M	IVCULC.35
1158	22.	10 CONTINUE	IVCULC.36
1159	C		IVCULC.37
1160	23.	CL = DSQRT((X(IISL)-X(IISR))**2 + (Y(IISL)-Y(IISR))**2)	IVCULC.38
1161	24.	CU = SSQ / A * (Y(IISL)-Y(IISR)) / CL	IVCULC.39
1162	25.	CV = - SSQ / A * (X(IISL)-X(IISR)) / CL	IVCULC.40
1163	26.	N2 = N + 1	IVCULC.41
1164	27.	IS = IISL	IVCULC.42
1165	28.	DO 40 J = 1, N2	IVCULC.43
1166	29.	IF (ETA(IS) .NE. 0.0) GO TO 20	IVCULC.44
1167	30.	ETA(IS) = SSH	IVCULC.45
1168	31.	U(IS) = (SSH - B(IS)) * CU	IVCULC.46
1169	32.	V(IS) = (SSH - B(IS)) * CV	IVCULC.47
1170	33.	GO TO 30	IVCULC.48
1171	C		IVCULC.49
1172	34.	20 CONTINUE	IVCULC.50
1173	35.	ETA(IS) = (ETA(IS) + SSH) / 2.0	IVCULC.51
1174	36.	U(IS) = (U(IS) + (SSH - B(IS)) * CU) / 2.0	IVCULC.52
1175	37.	V(IS) = (V(IS) + (SSH - B(IS)) * CV) / 2.0	IVCULC.53
1176	38.	30 CONTINUE	IVCULC.54
1177	39.	IS = IS + M	IVCULC.55
1178	40.	40 CONTINUE	IVCULC.56
1179	41.	50 CONTINUE	IVCULC.57
1180	C		IVCULC.58
1181	C *** CHECK ***		IVCULC.59
1182	42.	DO 60 I = 1, NP	IVCULC.60
1183	43.	IF (ETA(I) .NE. 0.0) GO TO 60	IVCULC.61
1184	44.	IERROR = 1	IVCULC.62
1185	45.	CALL LINE(2)	IVCULC.63
1186	46.	WRITE(LOUT,601) I	
1187	47.	601 FORMAT(/I21, '***** ERROR ***** UNDEFINED NODE NO.=', I5)	
1188	48.	60 CONTINUE	
1189	C		
1190	49.	RETURN	
1191	50.	END	

1192	1.	SUBROUTINE CLMAIN	CLMAIN.2
1193		* (X , Y , U , EFIX,ETA , U , V , ETA0,ETA1,U0 ,	CLMAIN.3
1194		* U1 ,V0 ,V1 , IELE,MELE,CHAT,IEBN,QUN , INBN,JNBN,	CLMAIN.4
1195		* CBN ,TSH ,BMT,BBN ,NP ,NE ,NB ,NM ,NBN1,NBN2,	CLMAIN.5
1196		* NPMAX ,NEMAX ,NBMAX ,NMHAX ,NBNMX1 ,	CLMAIN.6
1197		* NBNMX2 ,NBNMX3 ,KELE)	CLMAIN.7
1198	C		CLMAIN.8
1199	2.	IMPLICIT REAL*8 (A-H,O-Z)	CLMAIN.9
1200	C		CLMAIN.10
1201	3.	DIMENSION X(NPMAX) ,Y(NPMAX) ,B(NPMAX) ,EFIX(NPMAX),	CLMAIN.11
1202		ETA(NPMAX) ,KELE(5,NEMAX) ,BBN(NBNMX2) ,	CLMAIN.12
1203		U(NPMAX) ,V(NPMAX) ,ETA0(NPMAX) ,ETA1(NPMAX) ,	CLMAIN.13
1204		U0(NPMAX) ,U1(NPMAX) ,V0(NPMAX) ,V1(NPMAX) ,	CLMAIN.14
1205		IELE(4,NEMAX) ,MELE(NEMAX) ,CHAT(2,NMHAX) ,	CLMAIN.15
1206		IEBN(2,NBNMX1) ,QBN(2,NBNMX1) ,INBN(3,NBNMX2) ,	CLMAIN.16
1207		JNBN(2,NBNMX3,NBNMX2) ,CBN(NBNMX3,NBNMX2) ,	CLMAIN.17
1208		TSH(3*NP,2*NB-1) ,BMT(3*NP)	CLMAIN.18
1209	C		CLMAIN.19
1210	4.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
1211		LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
1212	5.	COMMON / ITERAT / MAXIT ,MINER ,IFLG(3) ,EPSLN(3)	/ITERAT/.2
1213	6.	COMMON / CONST / ZERO ,ONE ,TWO	/CONST/.2
1214	7.	COMMON / ISTEP / ICOUNT ,DT ,TIME	/ISTEP/.2
1215	8.	COMMON / ERROR / IERROR	/ERROR/.2
1216	9.	COMMON / CHECK / ICHECK(10)	/CHECK/.2
1217	10.	COMMON / TITLE / ITIT(20)	/TITLE/.2
1218	11.	COMMON / POINT / NPOINT ,IPL(10) ,IPR(10) ,CA(10) ,CB(10)	/POINT/.2
1219	C		CLMAIN.28
1220	12.	REWIND LU10	CLMAIN.29
1221	13.	WRITE(LU10) ITIT	CLMAIN.30
1222	14.	WRITE(LU10) EPSLN, NPOINT	CLMAIN.31
1223	C		CLMAIN.32
1224	15.	ICHK1 = ICHECK(3)	CLMAIN.33
1225	16.	ICHK2 = ICHECK(7)	CLMAIN.34
1226	17.	IF (ICHK1+ICHK2 .GT. 0) CALL PAGE	CLMAIN.35
1227	C		CLMAIN.36
1228	C	 INITIAL SET	CLMAIN.37
1229	C		CLMAIN.38
1230	18.	CALL MOVE2 (U, U1, NP)	CLMAIN.39
1231	19.	CALL MOVE2 (V, V1, NP)	CLMAIN.40
1232	20.	CALL MOVE2 (ETA, ETA1, NP)	CLMAIN.41
1233	C ***	(K-1) CONDITION LEFT FOR FUTURITY ***	CLMAIN.42
1234	21.	CALL MOVE2 (U, U0, NP)	CLMAIN.43
1235	22.	CALL MOVE2 (V, V0, NP)	CLMAIN.44
1236	23.	CALL MOVE2 (ETA, ETA0, NP)	CLMAIN.45
1237	C		CLMAIN.46
1238	C		CLMAIN.47
1239	24.	CALL CHAKE1 (X,Y,IELE,KELE,MELE,CHAT,NPMAX,NEMAX,NMHAX)	CLMAIN.48
1240	25.	CALL BOUND1 (IELE,INBN,JNBN,CBN,BBN,NEMAX,NBNMX2,NBNMX3)	CLMAIN.49
1241	26.	IF (IERROR .GT. 0) GO TO 999	CLMAIN.50
1242	C		CLMAIN.51
1243	27.	ICOUNT = 0	CLMAIN.52
1244	28.	TIME = 0.0	CLMAIN.53
1245	C		CLMAIN.54
1246	C	 ITERATION LOOP	CLMAIN.55
1247	C		CLMAIN.56
1248	29.	10 CONTINUE	CLMAIN.57
1249	30.	ICOUNT = ICOUNT + 1	CLMAIN.58
1250	31.	TIME = TIME + DT	CLMAIN.59
1251	C		CLMAIN.60
1252	32.	IF (ICHK1 .EQ. 0) GO TO 15	CLMAIN.61
1253	33.	CALL LINE(2)	CLMAIN.62
1254	34.	WRITE(LOUT,601) ICOUNT,DT,TIME	CLMAIN.63
1255	35.	601 FORMAT('0*CHECK* ITERATION COUNT+', I4, 5X, 'DT=+', F8.3,	CLMAIN.64
1256		' 3X, 'TIME=+', F10.3, 3X, 60('+='))	CLMAIN.65
1257	C		CLMAIN.66
1258	36.	15 CONTINUE	CLMAIN.67
1259	37.	IF (ICOUNT .LE. MAXIT) GO TO 25	CLMAIN.68
1260	38.	CALL LINE(2)	CLMAIN.69
1261	39.	WRITE(LOUT,602) MAXIT	CLMAIN.70
1262	40.	602 FORMAT(/20X, '***** ERROR ***** ITERATION COUNT OVER. ',	CLMAIN.71
1263		' *COUNT=+', I4)	CLMAIN.72
1264	41.	IERROR = 1	CLMAIN.73
1265	42.	GO TO 999	CLMAIN.74
1266	C		CLMAIN.75
1267	43.	25 CONTINUE	CLMAIN.76
1268	44.	CALL CHAKE2 (X, Y, B, U1, V1, ETA1, IELE, KELE, MELE, CHAT,	CLMAIN.77
1269		NPMAX, NEMAX, NMHAX)	CLMAIN.78
1270	45.	CALL PHAKE (X, Y, U1, V1, ETA1, IELE, KELE, QBN,	CLMAIN.79

1271		*	NPHAX, NEHAX, NBNHX1)	CLMAIN.80
1272	C			CLMAIN.81
1273	46.	CALL	BOUND2 (IELE, INBN, EFIX, BHAT, TSH, NP, NE, NB, NBN2)	CLMAIN.82
1274	47.	CALL	SOLVNS (TSH, BHAT, 3*NP, NB)	CLMAIN.83
1275	C			CLMAIN.84
1276	48.	CALL	DEPEND (BHAT, NP)	CLMAIN.85
1277	C			CLMAIN.86
1278	49.	DO	30 I=1,NP	CLMAIN.87
1279	50.	U(I)	= BHAT(3*I-2)	CLMAIN.88
1280	51.	V(I)	= BHAT(3*I-1)	CLMAIN.89
1281	52.	ETA(I)	= BHAT(3*I)	CLMAIN.90
1282	53.	30	CONTINUE	CLMAIN.91
1283	C			CLMAIN.92
1284	54.	IF (ICHK2 .GT. 0)	CALL CHKP6(U ,NP,+U ,ICOUNT)	CLMAIN.93
1285	55.	IF (ICHK2 .GT. 0)	CALL CHKP6(V ,NP,+V ,ICOUNT)	CLMAIN.94
1286	56.	IF (ICHK2 .GT. 0)	CALL CHKP6(ETA,NP,+ETA+,ICOUNT)	CLMAIN.95
1287	C			CLMAIN.96
1288	57.	CALL	CONVGE (B,U,U1,V,V1,ETA,ETA1,IELE,NPHAX,NEHAX,IIFLG)	CLMAIN.97
1289	58.	IF (IIFLG .LE. MINER)	GO TO 999	CLMAIN.98
1290	C			CLMAIN.99
1291	59.	DO	50 I=1,NP	CLMAIN.100
1292	60.	IF (ETA(I) .GT. B(I))	GO TO 50	CLMAIN.101
1293	61.	IERROR	= 1	CLMAIN.102
1294	62.	CALL	LINE(2)	CLMAIN.103
1295	63.	WRITE(LOUT,604)	I, ETA(I), B(I)	CLMAIN.104
1296	64.	604	FORMAT(/T21, +***** ERROR ***** ETA .LE. B NODE NO.,	CLMAIN.105
1297			I5, + ETA=+, F8.3, + B=+,F8.3)	CLMAIN.106
1298	65.	50	CONTINUE	CLMAIN.107
1299	66.	IF (IERROR .GT. 0)	GO TO 999	CLMAIN.108
1300	C			CLMAIN.109
1301	67.	CALL	MOVE2(U1 ,U0 ,NP)	CLMAIN.110
1302	68.	CALL	MOVE2(U ,U1 ,NP)	CLMAIN.111
1303	69.	CALL	MOVE2(V1 ,V0 ,NP)	CLMAIN.112
1304	70.	CALL	MOVE2(V ,V1 ,NP)	CLMAIN.113
1305	71.	CALL	MOVE2(ETA1,ETA0,NP)	CLMAIN.114
1306	72.	CALL	MOVE2(ETA ,ETA1,NP)	CLMAIN.115
1307	73.	GO	TO 10	CLMAIN.116
1308	C			CLMAIN.117
1309	74.	999	CONTINUE	CLMAIN.118
1310	C			CLMAIN.119
1311	75.	RETURN		CLMAIN.120
1312	76.	END		CLMAIN.121

1313	1.	SUBROUTINE	CHAKE1 (X,Y,IELE,KELE,MELE,CHAT,NPHAX,NEHAX,NMHAX)	CHAKE1.2
1314	C			CHAKE1.3
1315	2.	IMPLICIT	REAL*8 (A-H, O-Z)	CHAKE1.4
1316	C			CHAKE1.5
1317	3.	COMMON / FILE	/ LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
1318			LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
1319	4.	COMMON / CHECK	/ ICHECK(10)	/CHECK/.2
1320	5.	COMMON / CNTRL	/ NP ,NE ,NB ,NM ,	/CNTRL/.2
1321			NBN1 ,NBN2	/CNTRL/.3
1322	6.	COMMON / GAUSS	/ NGS ,G(5,4) ,H(5,4)	/GAUSS/.2
1323	7.	COMMON / ISTEP	/ ICOUNT ,OI ,TIME	/ISTEP/.2
1324	8.	COMMON / CONST	/ ZERO ,ONE ,TWO	/CONST/.2
1325	C			CHAKE1.12
1326	9.	DIMENSION	X(NPHAX), Y(NPHAX), IELE(4,NEHAX), KELE(5,NEHAX),	CHAKE1.13
1327			MELE(NEHAX), CHAT(2,NMHAX)	CHAKE1.14
1328	10.	DIMENSION	XX(4), YY(4), RN(4), DN(4), ONY(4), CSX(4), CSY(4),	CHAKE1.15
1329			EK0(4,4), EK1(4,4), EK2(4,4), EK4(4,4), EK5(4,4),	CHAKE1.16
1330			EK7(4,4), EK8(4,4), EK9(4,4), WK1(4,4), WK2(4,4),	CHAKE1.17
1331			EM(12,12)	CHAKE1.18
1332	C			CHAKE1.19
1333	11.	DATA	PAI / 3.141592653589800 /	CHAKE1.20
1334	C			CHAKE1.21
1335			I I	CHAKE1.22
1336	C		I (1) (2) () I	CHAKE1.23
1337	C	MAI	RX C = I (4) (5) () I	CHAKE1.24
1338	C		I (7) (8) (9) I	CHAKE1.25
1339	C		I I	CHAKE1.26
1340	C			CHAKE1.27
1341	12.	REWIND	LU11	CHAKE1.28
1342	13.	REWIND	LU15	CHAKE1.29
1343	C			CHAKE1.30
1344	14.	ICHK	= ICHECK(2)	CHAKE1.31
1345	15.	IF (ICHK .GT. 0)	CALL PAGE	CHAKE1.32

1346	C			CHAKE1.33
1347	C			CHAKE1.34
1348	16.	DO 999 IE = 1, NE		CHAKE1.35
1349	C			CHAKE1.36
1350	17.	DO 10 I = 1, 4		CHAKE1.37
1351	18.	IP = IELE(I, IE)		CHAKE1.38
1352	19.	XX(I) = X(IP)		CHAKE1.39
1353	20.	YY(I) = Y(IP)		CHAKE1.40
1354	21.	10 CONTINUE		CHAKE1.41
1355	C			CHAKE1.42
1356	22.	N = KELE(5, IE)		CHAKE1.43
1357	23.	DO 20 IS = 1, N		CHAKE1.44
1358	24.	JS = IS + 1		CHAKE1.45
1359	25.	IF (JS .EQ. 5) JS = 1		CHAKE1.46
1360	26.	RAD = DATAN2 (YY(JS)-YY(IS), XX(JS)-XX(IS)) - PAI/2.0		CHAKE1.47
1361	27.	CSX(IS) = UCOS(RAD)		CHAKE1.48
1362	28.	CSY(IS) = DSIN(RAD)		CHAKE1.49
1363	29.	20 CONTINUE		CHAKE1.50
1364	C			CHAKE1.51
1365	30.	EPS = CMAT(1, HELE(IE))		CHAKE1.52
1366	C			CHAKE1.53
1367	C		***** (1) *****	CHAKE1.54
1368	31.	CALL MSET2 (ZERO, EK0, 4*4)		CHAKE1.55
1369	32.	DO 120 I = 1, NGS		CHAKE1.56
1370	33.	CALL GHSET (I, XSI, WI)		CHAKE1.57
1371	34.	DO 110 J = 1, NGS		CHAKE1.58
1372	35.	CALL GHSET (J, ET, WJ)		CHAKE1.59
1373	C			CHAKE1.60
1374	36.	CALL DNDXY (XSI, ET, XX, YY, DNX, DNY, DETJ)		CHAKE1.61
1375	37.	CALL MULT (DNX, UNX, HK1, 4, 1, 4)		CHAKE1.62
1376	38.	CALL MULT (DNY, DNY, HK2, 4, 1, 4)		CHAKE1.63
1377	39.	CALL ADD (HK1, HK2, HK2, TWO, ONE, 4, 4)		CHAKE1.64
1378	40.	CALL ADD (HK2, EK0, EK0, EPS*HI*WJ*DETJ, ONE, 4, 4)		CHAKE1.65
1379	41.	110 CONTINUE		CHAKE1.66
1380	42.	120 CONTINUE		CHAKE1.67
1381	C			CHAKE1.68
1382	43.	CALL MSET2 (ZERO, EK1, 4*4)		CHAKE1.69
1383	44.	DO 140 IS = 1, N		CHAKE1.70
1384	45.	COSX = CSX(IS)		CHAKE1.71
1385	46.	COSY = CSY(IS)		CHAKE1.72
1386	47.	DO 130 I = 1, NGS		CHAKE1.73
1387	48.	CALL GHSET (I, XSI, WI)		CHAKE1.74
1388	C			CHAKE1.75
1389	49.	CALL SFSET (XSI, IS, XX, YY, RN, DNX, DNY, DETJ)		CHAKE1.76
1390	50.	CALL MULT (RN, DNX, HK1, 4, 1, 4)		CHAKE1.77
1391	51.	CALL MULT (RN, DNY, HK2, 4, 1, 4)		CHAKE1.78
1392	52.	CALL ADD (HK1, HK2, HK2, TWO*COSX, COSY, 4, 4)		CHAKE1.79
1393	53.	CALL ADD (HK2, EK1, EK1, EPS*HI*DETJ, ONE, 4, 4)		CHAKE1.80
1394	54.	130 CONTINUE		CHAKE1.81
1395	55.	140 CONTINUE		CHAKE1.82
1396	C			CHAKE1.83
1397	56.	CALL ADD (EK0, EK1, EK1, DT, -DT, 4, 4)		CHAKE1.84
1398	C			CHAKE1.85
1399	C		***** (2) *****	CHAKE1.86
1400	57.	CALL MSET2 (ZERO, EK0, 4*4)		CHAKE1.87
1401	58.	DO 220 I = 1, NGS		CHAKE1.88
1402	59.	CALL GHSET (I, XSI, WI)		CHAKE1.89
1403	60.	DO 210 J = 1, NGS		CHAKE1.90
1404	61.	CALL GHSET (J, ET, WJ)		CHAKE1.91
1405	C			CHAKE1.92
1406	62.	CALL DNDXY (XSI, ET, XX, YY, DNX, DNY, DETJ)		CHAKE1.93
1407	63.	CALL MULT (DNY, DNX, HK1, 4, 1, 4)		CHAKE1.94
1408	64.	CALL ADD (HK1, EK0, EK0, EPS*HI*WJ*DETJ, ONE, 4, 4)		CHAKE1.95
1409	65.	210 CONTINUE		CHAKE1.96
1410	66.	220 CONTINUE		CHAKE1.97
1411	C			CHAKE1.98
1412	67.	CALL MSET2 (ZERO, EK2, 4*4)		CHAKE1.99
1413	68.	DO 240 IS = 1, N		CHAKE1.100
1414	69.	COSY = CSY(IS)		CHAKE1.101
1415	70.	DO 230 I = 1, NGS		CHAKE1.102
1416	71.	CALL GHSET (I, XSI, WI)		CHAKE1.103
1417	C			CHAKE1.104
1418	72.	CALL SFSET (XSI, IS, XX, YY, RN, DNX, DNY, DETJ)		CHAKE1.105
1419	73.	CALL MULT (RN, DNX, HK1, 4, 1, 4)		CHAKE1.106
1420	74.	CALL ADD (HK1, EK2, EK2, EPS*COSY*HI*DETJ, ONE, 4, 4)		CHAKE1.107
1421	75.	230 CONTINUE		CHAKE1.108
1422	76.	240 CONTINUE		CHAKE1.109
1423	C			CHAKE1.110
1424	77.	CALL ADD (EK0, EK2, EK2, DT, -DT, 4, 4)		CHAKE1.111
1425	C			CHAKE1.112
1426	C		***** (4) *****	CHAKE1.113
1427	78.	CALL MSET2 (ZERO, EK0, 4*4)		CHAKE1.114

1428	79.	DO 420 I = 1, NGS	CHAKE1.115
1429	80.	CALL GHSET (I, XSI, WI)	CHAKE1.116
1430	81.	DO 410 J = 1, NGS	CHAKE1.117
1431	82.	CALL GHSET (J, ET, WJ)	CHAKE1.118
1432	C		CHAKE1.119
1433	83.	CALL DNOXY (XSI, ET, XX, YY, ONX, ONY, DETJ)	CHAKE1.120
1434	84.	CALL MULT (ONX, ONY, WK1, 4, 1, 4)	CHAKE1.121
1435	85.	CALL ADD (WK1, EK0, EK0, EPS*WI*WJ*DETJ, ONE, 4, 4)	CHAKE1.122
1436	86.	410 CONTINUE	CHAKE1.123
1437	87.	420 CONTINUE	CHAKE1.124
1438	C		CHAKE1.125
1439	88.	CALL HSET2 (ZERO, EK4, 4*4)	CHAKE1.126
1440	89.	DO 440 IS = 1, N	CHAKE1.127
1441	90.	COSX = CSX(IS)	CHAKE1.128
1442	91.	DO 430 I = 1, NGS	CHAKE1.129
1443	92.	CALL GHSET (I, XSI, WI)	CHAKE1.130
1444	C		CHAKE1.131
1445	93.	CALL SFSET (XSI, IS, XX, YY, RN, ONX, ONY, DETJ)	CHAKE1.132
1446	94.	CALL MULT (RN, ONY, WK1, 4, 1, 4)	CHAKE1.133
1447	95.	CALL ADD (WK1, EK4, EK4, EPS*COSX*WI*DETJ, ONE, 4, 4)	CHAKE1.134
1448	96.	430 CONTINUE	CHAKE1.135
1449	97.	440 CONTINUE	CHAKE1.136
1450	C		CHAKE1.137
1451	98.	CALL ADD (EK0, EK4, EK4, DT, -DT, 4, 4)	CHAKE1.138
1452	C		CHAKE1.139
1453	C		***** (5) *****
1454	99.	CALL HSET2 (ZERO, EK0, 4*4)	CHAKE1.140
1455	100.	DO 520 I = 1, NGS	CHAKE1.141
1456	101.	CALL GHSET (I, XSI, WI)	CHAKE1.142
1457	102.	DO 510 J = 1, NGS	CHAKE1.143
1458	103.	CALL GHSET (J, ET, WJ)	CHAKE1.144
1459	C		CHAKE1.145
1460	104.	CALL DNOXY (XSI, ET, XX, YY, ONX, ONY, DETJ)	CHAKE1.146
1461	105.	CALL MULT (ONX, ONX, WK1, 4, 1, 4)	CHAKE1.147
1462	106.	CALL MULT (ONY, ONY, WK2, 4, 1, 4)	CHAKE1.148
1463	107.	CALL ADD (WK1, WK2, WK2, ONE, TWO, 4, 4)	CHAKE1.149
1464	108.	CALL ADD (WK2, EK0, EK0, EPS*WI*WJ*DETJ, ONE, 4, 4)	CHAKE1.150
1465	109.	510 CONTINUE	CHAKE1.151
1466	110.	520 CONTINUE	CHAKE1.152
1467	C		CHAKE1.153
1468	111.	CALL HSET2 (ZERO, EK5, 4*4)	CHAKE1.154
1469	112.	DO 540 IS = 1, N	CHAKE1.155
1470	113.	COSX = CSX(IS)	CHAKE1.156
1471	114.	COSY = CSY(IS)	CHAKE1.157
1472	115.	DO 530 I = 1, NGS	CHAKE1.158
1473	116.	CALL GHSET (I, XSI, WI)	CHAKE1.159
1474	C		CHAKE1.160
1475	117.	CALL SFSET (XSI, IS, XX, YY, RN, ONX, ONY, DETJ)	CHAKE1.161
1476	118.	CALL MULT (RN, ONY, WK1, 4, 1, 4)	CHAKE1.162
1477	119.	CALL MULT (RN, ONX, WK2, 4, 1, 4)	CHAKE1.163
1478	120.	CALL ADD (WK1, WK2, WK2, TWO*COSY, COSX, 4, 4)	CHAKE1.164
1479	121.	CALL ADD (WK2, EK5, EK5, EPS*WI*DETJ, ONE, 4, 4)	CHAKE1.165
1480	122.	530 CONTINUE	CHAKE1.166
1481	123.	540 CONTINUE	CHAKE1.167
1482	C		CHAKE1.168
1483	124.	CALL ADD (EK0, EK5, EK5, DT, -DT, 4, 4)	CHAKE1.169
1484	C		CHAKE1.170
1485	C		***** (7) *****
1486	125.	CALL HSET2 (ZERO, EK0, 4*4)	CHAKE1.171
1487	126.	DO 720 I = 1, NGS	CHAKE1.172
1488	127.	CALL GHSET (I, XSI, WI)	CHAKE1.173
1489	128.	DO 710 J = 1, NGS	CHAKE1.174
1490	129.	CALL GHSET (J, ET, WJ)	CHAKE1.175
1491	C		CHAKE1.176
1492	130.	CALL DNOXY (XSI, ET, XX, YY, ONX, ONY, DETJ)	CHAKE1.177
1493	131.	CALL HSET (XSI, ET, RN)	CHAKE1.178
1494	132.	CALL MULT (ONX, RN, WK1, 4, 1, 4)	CHAKE1.179
1495	133.	CALL ADD (WK1, EK0, EK0, WI*WJ*DETJ, ONE, 4, 4)	CHAKE1.180
1496	134.	710 CONTINUE	CHAKE1.181
1497	135.	720 CONTINUE	CHAKE1.182
1498	C		CHAKE1.183
1499	136.	CALL HSET2 (ZERO, EK7, 4*4)	CHAKE1.184
1500	137.	DO 740 IS = 1, N	CHAKE1.185
1501	138.	IF (KELE(IS, IE) .NE. 0) GO TO 740	CHAKE1.186
1502	139.	COSX = CSX(IS)	CHAKE1.187
1503	140.	DO 730 I = 1, NGS	CHAKE1.188
1504	141.	CALL GHSET (I, XSI, WI)	CHAKE1.189
1505	C		CHAKE1.190
			CHAKE1.191
			CHAKE1.192

1506	142.	CALL SFSET (XSI, IS, XX, YY, RN, ONX, ONY, DETJ)	CHAKE1.193
1507	143.	CALL MULT (RN, RN, WK1, 4, 1, 4)	CHAKE1.194
1508	144.	CALL ADD (WK1, EK7, EK7, COSX*WI*DETJ, ONE, 4, 4)	CHAKE1.195
1509	145.	730 CONTINUE	CHAKE1.196
1510	146.	740 CONTINUE	CHAKE1.197
1511	C		CHAKE1.198
1512	147.	CALL ADD (EK0, EK7, EK7, -OT, OT, 4, 4)	CHAKE1.199
1513	C		CHAKE1.200
1514	C		CHAKE1.201
1515	148.	CALL MSET2 (ZERO, EK0, 4*4)	CHAKE1.202
1516	149.	DO 820 I = 1, NGS	CHAKE1.203
1517	150.	CALL GWSET (I, XSI, WI)	CHAKE1.204
1518	151.	DO 810 J = 1, NGS	CHAKE1.205
1519	152.	CALL GWSET (J, ET, WJ)	CHAKE1.206
1520	C		CHAKE1.207
1521	153.	CALL UNDXY (XSI, ET, XX, YY, ONX, ONY, DETJ)	CHAKE1.208
1522	154.	CALL NSET (XSI, ET, RN)	CHAKE1.209
1523	155.	CALL MULT (ONY, RN, WK1, 4, 1, 4)	CHAKE1.210
1524	156.	CALL ADD (WK1, EK0, EK0, WI*WJ*DETJ, ONE, 4, 4)	CHAKE1.211
1525	157.	810 CONTINUE	CHAKE1.212
1526	158.	820 CONTINUE	CHAKE1.213
1527	C		CHAKE1.214
1528	159.	CALL MSET2 (ZERO, EK8, 4*4)	CHAKE1.215
1529	160.	DO 840 IS = 1, N	CHAKE1.216
1530	161.	IF (KELE(IS,IE) .NE. 0) GO TO 840	CHAKE1.217
1531	162.	COSY = CSY(IS)	CHAKE1.218
1532	163.	DO 830 I = 1, NGS	CHAKE1.219
1533	164.	CALL GWSET (I, XSI, WI)	CHAKE1.220
1534	C		CHAKE1.221
1535	165.	CALL SFSET (XSI, IS, XX, YY, RN, ONX, ONY, DETJ)	CHAKE1.222
1536	166.	CALL MULT (RN, RN, WK1, 4, 1, 4)	CHAKE1.223
1537	167.	CALL ADD (WK1, EK8, EK8, COSY*WI*DETJ, ONE, 4, 4)	CHAKE1.224
1538	168.	830 CONTINUE	CHAKE1.225
1539	169.	840 CONTINUE	CHAKE1.226
1540	C		CHAKE1.227
1541	170.	CALL ADD (EK0, EK8, EK8, -OT, OT, 4, 4)	CHAKE1.228
1542	C		CHAKE1.229
1543	C		CHAKE1.230
1544	171.	CALL MSET2 (ZERO, EK9, 4*4)	CHAKE1.231
1545	172.	DO 920 I = 1, NGS	CHAKE1.232
1546	173.	CALL GWSET (I, XSI, WI)	CHAKE1.233
1547	174.	DO 910 J = 1, NGS	CHAKE1.234
1548	175.	CALL GWSET (J, ET, WJ)	CHAKE1.235
1549	C		CHAKE1.236
1550	176.	CALL UNDXY (XSI, ET, XX, YY, ONX, ONY, DETJ)	CHAKE1.237
1551	177.	CALL NSET (XSI, ET, RN)	CHAKE1.238
1552	178.	CALL MULT (RN, RN, WK1, 4, 1, 4)	CHAKE1.239
1553	179.	CALL ADD (WK1, EK9, EK9, WI*WJ*DETJ, ONE, 4, 4)	CHAKE1.240
1554	180.	910 CONTINUE	CHAKE1.241
1555	181.	920 CONTINUE	CHAKE1.242
1556	C		CHAKE1.243
1557	182.	WRITE(LU15) EK9	CHAKE1.244
1558	C		CHAKE1.245
1559	C		CHAKE1.246
1560	183.	DO 60 I = 1, 4	CHAKE1.247
1561	184.	DO 50 J = 1, 4	CHAKE1.248
1562	185.	EM(I, J) = EK1(I, J)	CHAKE1.249
1563	186.	EM(I, J+4) = EK2(I, J)	CHAKE1.250
1564	187.	EM(I, J+8) = 0.0	CHAKE1.251
1565	188.	EM(I+4, J) = EK4(I, J)	CHAKE1.252
1566	189.	EM(I+4, J+4) = EK5(I, J)	CHAKE1.253
1567	190.	EM(I+4, J+8) = 0.0	CHAKE1.254
1568	191.	EM(I+8, J) = EK7(I, J)	CHAKE1.255
1569	192.	EM(I+8, J+4) = EK8(I, J)	CHAKE1.256
1570	193.	EM(I+8, J+8) = EK9(I, J)	CHAKE1.257
1571	194.	50 CONTINUE	CHAKE1.258
1572	195.	60 CONTINUE	CHAKE1.259
1573	C		CHAKE1.260
1574	196.	WRITE(LU11) EM	CHAKE1.261
1575	C		CHAKE1.262
1576	197.	IF (ICHK .GT. 0) CALL CHKP2 (IE, EM)	CHAKE1.263
1577	C		CHAKE1.264
1578	198.	999 CONTINUE	CHAKE1.265
1579	C		CHAKE1.266
1580	199.	RETURN	CHAKE1.267
1581	200.	END	CHAKE1.268

1582	1.	SUBROUTINE BOUND1 (IELE, INBN, JNBN, CBN, BBN, NEMAX, NBNHX2, NBNHX3)	BOUND1.2
1583	C		BOUND1.3
1584	2.	IMPLICIT REAL*8 (A-H, O-Z)	BOUND1.4
1585	C		BOUND1.5
1586	3.	COMMON / FILE / LIN , LOUT , LU1 , LU2 , LU3 , LU4 ,	/FILE/.2
1587	*	LU10 , LU11 , LU12 , LU13 , LU14 , LU15	/FILE/.3
1588	4.	COMMON / CHECK / ICHECK(10)	/CHECK/.2
1589	5.	COMMON / CNTRL / NP , NE , NB , NT ,	/CNTRL/.2
1590	*	NBN1 , NBN2	/CNTRL/.3
1591	6.	COMMON / CONST / ZERO , ONE , TWO	/CONST/.2
1592	7.	COMMON / ERROR / IERROR	/ERROR/.2
1593	C		BOUND1.11
1594	8.	DIMENSION IELE(4, NEMAX), INBN(3, NBNHX2), JNBN(2, NBNHX3, NBNHX2),	BOUND1.12
1595	*	CBN(NBNHX3, NBNHX2), BBN(NBNHX2)	BOUND1.13
1596	9.	DIMENSION IFONO(12), IFOSH(12), JFONO(50), A(12, 50), B(12)	BOUND1.14
1597	C		BOUND1.15
1598	10.	REWIND LU12	BOUND1.16
1599	C		BOUND1.17
1600	11.	ICLK = ICHECK(5)	BOUND1.18
1601	12.	IF (ICLK .GT. 0) CALL PAGE	BOUND1.19
1602	C		BOUND1.20
1603	13.	DO 999 IE = 1, NE	BOUND1.21
1604	C		BOUND1.22
1605	14.	NBN = 0	BOUND1.23
1606	15.	DO 20 I = 1, 4	BOUND1.24
1607	16.	DO 10 J = 1, NBN2	BOUND1.25
1608	17.	IF (IELE(I, IE) .EQ. INBN(1, J)) NBN = NBN + 1	BOUND1.26
1609	18.	10 CONTINUE	BOUND1.27
1610	19.	20 CONTINUE	BOUND1.28
1611	20.	IF (NBN .EQ. 0) GO TO 999	BOUND1.29
1612	C		BOUND1.30
1613	21.	DO 40 I = 1, 4	BOUND1.31
1614	22.	DO 30 J = 1, 3	BOUND1.32
1615	23.	IFD = 3 * (I-1) + J	BOUND1.33
1616	24.	IFONO(IFD) = 3 * (IELE(I, IE) - 1) + J	BOUND1.34
1617	25.	IFOSH(IFD) = 0	BOUND1.35
1618	26.	30 CONTINUE	BOUND1.36
1619	27.	40 CONTINUE	BOUND1.37
1620	C		BOUND1.38
1621	C	***** SET JFONO *****	BOUND1.39
1622	28.	KN = 0	BOUND1.40
1623	29.	DO 70 I = 1, 12	BOUND1.41
1624	30.	DO 50 J = 1, NBN2	BOUND1.42
1625	31.	IF (IFONO(I) .EQ. 3*(INBN(1, J)-1)+INBN(2, J)) GO TO 60	BOUND1.43
1626	32.	50 CONTINUE	BOUND1.44
1627	33.	KN = KN + 1	BOUND1.45
1628	34.	JFONO(KN) = IFONO(I)	BOUND1.46
1629	35.	GO TO 70	BOUND1.47
1630	C		BOUND1.48
1631	36.	60 CONTINUE	BOUND1.49
1632	37.	IFOSH(I) = J	BOUND1.50
1633	38.	70 CONTINUE	BOUND1.51
1634	C		BOUND1.52
1635	39.	DO 120 I = 1, 12	BOUND1.53
1636	40.	IFD = IFOSH(I)	BOUND1.54
1637	41.	IF (IFD .EQ. 9) GO TO 120	BOUND1.55
1638	C		BOUND1.56
1639	42.	NO = INBN(3, IFD)	BOUND1.57
1640	43.	DO 110 J = 1, NO	BOUND1.58
1641	44.	JFD = 3 * (JNBN(1, J, IFD) - 1) + JNBN(2, J, IFD)	BOUND1.59
1642	45.	IF (KN .EQ. 0) GO TO 90	BOUND1.60
1643	46.	DO 80 K = 1, KN	BOUND1.61
1644	47.	IF (JFONO(K) .EQ. JFD) GO TO 110	BOUND1.62
1645	48.	80 CONTINUE	BOUND1.63
1646	C		BOUND1.64
1647	49.	90 CONTINUE	BOUND1.65
1648	50.	KN = KN + 1	BOUND1.66
1649	51.	IF (KN .LE. 50) GO TO 100	BOUND1.67
1650	52.	IERROR = 1	BOUND1.68
1651	53.	CALL LINE(2)	BOUND1.69
1652	54.	WRITE(LOUT, 900) IE	BOUND1.70
1653	55.	900 FORMAT(/T21, ***** ERROR ***** HPC DIMENSION OVER +,	BOUND1.71
1654	*	+ELEMENT NO.=, I4)	BOUND1.72
1655	56.	GO TO 999	BOUND1.73
1656	C		BOUND1.74
1657	57.	100 CONTINUE	BOUND1.75
1658	58.	JFONO(KN) = JFD	BOUND1.76
1659	59.	110 CONTINUE	BOUND1.77
1660	60.	120 CONTINUE	BOUND1.78

```

1661      C
1662      C
1663      61.      CALL MSET2 (ZERO, A, 12*50)
1664      62.      CALL MSET2 (ZERO, B, 12)
1665      C
1666      63.      DO 170 I = 1, 12
1667      64.      IFD = IFDSH(I)
1668      65.      IF ( IFD .EQ. 0 )
1669      C
1670      66.      B(I) = BBN(IFD)
1671      67.      NO = INBN(3,IFD)
1672      68.      DO 140 J = 1, NO
1673      69.      JFO = 3 * (JBN(1,J,IFD) - 1) + JBN(2,J,IFD)
1674      70.      DO 130 K = 1, KN
1675      71.      IF ( JFONO(K) .NE. JFO )
1676      72.      A(I,K) = CBN(J,IFD)
1677      73.      130 CONTINUE
1678      74.      140 CONTINUE
1679      75.      GO TO 170
1680      C
1681      76.      150 CONTINUE
1682      77.      DO 160 J = 1, KN
1683      78.      IF ( IFONO(I) .EQ. JFONO(J) )
1684      79.      160 CONTINUE
1685      80.      170 CONTINUE
1686      C
1687      81.      WRITE(LU12) IE, KN
1688      82.      WRITE(LU12) IFONO
1689      83.      WRITE(LU12) JFONO
1690      84.      WRITE(LU12) A
1691      85.      WRITE(LU12) 3
1692      C
1693      86.      IF ( ICHK .GT. 0 )
1694      C
1695      87.      999 CONTINUE
1696      C
1697      88.      RETURN
1698      89.      END

```

***** SET A, B *****

BOUND1.79
BOUND1.80
BOUND1.81
BOUND1.82
BOUND1.83
BOUND1.84
BOUND1.85
BOUND1.86
BOUND1.87
BOUND1.88
BOUND1.89
BOUND1.90
BOUND1.91
BOUND1.92
BOUND1.93
BOUND1.94
BOUND1.95
BOUND1.96
BOUND1.97
BOUND1.98
BOUND1.99
BOUND1.100
BOUND1.101
BOUND1.102
BOUND1.103
BOUND1.104
BOUND1.105
BOUND1.106
BOUND1.107
BOUND1.108
BOUND1.109
BOUND1.110
BOUND1.111
BOUND1.112
BOUND1.113
BOUND1.114
BOUND1.115
BOUND1.116

```

1699      1.      SUBROUTINE CHAKE2 (X, Y, B, U1, V1, ETA1, IELE, KELE, HELE, CHAT,
1700      *
1701      C
1702      2.      IMPLICIT REAL*8 (A-H, O-Z)
1703      C
1704      3.      COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 , /FILE/.2
1705      *
1706      4.      COMMON / CHECK / ICHECK(10)
1707      5.      COMMON / CNTRL / NP ,NE ,NB ,NH , /CNTRL/.2
1708      *
1709      6.      COMMON / GAUSS / NGS ,G(5,4) ,H(5,4)
1710      7.      COMMON / ISTEP / ICOUNT ,OI ,TIME
1711      8.      COMMON / CONST / ZERO ,ONE ,TWO
1712      9.      COMMON / GRAVITY / GRAV
1713      C
1714      10.      DIMENSION X(NPHAX),Y(NPHAX),B(NPHAX),U1(NPHAX),V1(NPHAX),
1715      *
1716      *
1717      11.      DIMENSION XX(4),YY(4),UU(4),VV(4),HH(4),RN(4),ONX(4),
1718      *
1719      *
1720      C
1721      12.      DATA PA1 / 3.141592653589800 /
1722      C
1723      C
1724      C
1725      C
1726      C
1727      C
1728      C
1729      13.      REWIND LU11
1730      14.      REWIND LU13
1731      C
1732      15.      ICHK = ICHECK(8)
1733      16.      IF ( ICHK .GT. 0 )
1734      17.      DIG = OI * GRAV
1735      C

```

CHAKE2.2
CHAKE2.3
CHAKE2.4
CHAKE2.5
CHAKE2.6
/FILE/.2
/FILE/.3
/CHECK/.2
/CNTRL/.2
/CNTRL/.3
/GAUSS/.2
/ISTEP/.2
/CONST/.2
/GRAVITY/.2
CHAKE2.14
CHAKE2.15
CHAKE2.16
CHAKE2.17
CHAKE2.18
CHAKE2.19
CHAKE2.20
CHAKE2.21
CHAKE2.22
CHAKE2.23
CHAKE2.24
CHAKE2.25
CHAKE2.26
CHAKE2.27
CHAKE2.28
CHAKE2.29
CHAKE2.30
CHAKE2.31
CHAKE2.32
CHAKE2.33
CHAKE2.34
CHAKE2.35
CHAKE2.36

1736	C			CMAKE2.37
1737	18.	00 999	IE = 1, NE	CMAKE2.38
1738	C			CMAKE2.39
1739	19.	00 10	I = 1, 4	CMAKE2.40
1740	20.	IP	= IELE(I, IE)	CMAKE2.41
1741	21.	XX(I)	= X(IP)	CMAKE2.42
1742	22.	YY(I)	= Y(IP)	CMAKE2.43
1743	23.	UU(I)	= U1(IP)	CMAKE2.44
1744	24.	VV(I)	= V1(IP)	CMAKE2.45
1745	25.	HH(I)	= ETA1(IP) - B(IP)	CMAKE2.46
1746	26.	10	CONTINUE	CMAKE2.47
1747	C			CMAKE2.48
1748	27.	C	= CHAT(2, MELE(IE))	CMAKE2.49
1749	28.	DTGC	= DTG / (C+C)	CMAKE2.50
1750	C			CMAKE2.51
1751	C			CMAKE2.52
1752	29.	CALL	MSET2 (ZERO, EK0, 4*4) ***** (1) *****	CMAKE2.53
1753	30.	00 120	I = 1, NGS	CMAKE2.54
1754	31.	CALL	GHSET (I, XSI, WI)	CMAKE2.55
1755	32.	00 110	J = 1, NGS	CMAKE2.56
1756	33.	CALL	GHSET (J, ET, WJ)	CMAKE2.57
1757	C			CMAKE2.58
1758	34.	CALL	DNDXY (XSI, ET, XX, YY, DNX, DNY, DETJ)	CMAKE2.59
1759	35.	CALL	NSET (XSI, ET, RN)	CMAKE2.60
1760	36.	CALL	MULT (RN, UU, UUU, 1, 4, 1)	CMAKE2.61
1761	37.	CALL	MULT (RN, VV, VVV, 1, 4, 1)	CMAKE2.62
1762	38.	CALL	MULT (RN, HH, HHH, 1, 4, 1)	CMAKE2.63
1763	39.	SC	= 1.0 + DTGC * DSQRT(UUU*UUU+VVV*VVV) / (HHH*HHH)	CMAKE2.64
1764	40.	CALL	MULT (RN, RN, HK1, 4, 1, 4)	CMAKE2.65
1765	41.	CALL	ADD (WK1, EK0, EK0, SC*WI*WJ*DETJ, ONE, 4, 4)	CMAKE2.66
1766	42.	110	CONTINUE	CMAKE2.67
1767	43.	120	CONTINUE	CMAKE2.68
1768	C			CMAKE2.69
1769	44.	CALL	MSET2 (ZERO, EK1, 4*4)	CMAKE2.70
1770	45.	00 140	I = 1, NGS	CMAKE2.71
1771	46.	CALL	GHSET (I, XSI, WI)	CMAKE2.72
1772	47.	00 130	J = 1, NGS	CMAKE2.73
1773	48.	CALL	GHSET (J, ET, WJ)	CMAKE2.74
1774	C			CMAKE2.75
1775	49.	CALL	DNDXY (XSI, ET, XX, YY, DNX, DNY, DETJ)	CMAKE2.76
1776	50.	CALL	NSET (XSI, ET, RN)	CMAKE2.77
1777	51.	CALL	MULT (RN, UU, UUU, 1, 4, 1)	CMAKE2.78
1778	52.	CALL	MULT (RN, VV, VVV, 1, 4, 1)	CMAKE2.79
1779	53.	CALL	MULT (RN, HH, HHH, 1, 4, 1)	CMAKE2.80
1780	54.	CALL	MULT (DNX, RN, HK1, 4, 1, 4)	CMAKE2.81
1781	55.	CALL	MULT (DNY, RN, HK2, 4, 1, 4)	CMAKE2.82
1782	56.	CALL	ADD (WK1, WK2, WK2, UUU/HHH, VVV/HHH, 4, 4)	CMAKE2.83
1783	57.	CALL	ADD (WK2, EK1, EK1, WI*WJ*DETJ, ONE, 4, 4)	CMAKE2.84
1784	58.	130	CONTINUE	CMAKE2.85
1785	59.	140	CONTINUE	CMAKE2.86
1786	C			CMAKE2.87
1787	60.	CALL	ADD (EK0, EK1, EK1, ONE, -DI, 4, 4)	CMAKE2.88
1788	C			CMAKE2.89
1789	61.	CALL	MSET2 (ZERO, EK0, 4*4)	CMAKE2.90
1790	62.	N	= KELE(5, IE)	CMAKE2.91
1791	63.	00 160	IS = 1, N	CMAKE2.92
1792	64.	JS	= IS + 1	CMAKE2.93
1793	65.	IF (JS .EQ. 5)	JS = 1	CMAKE2.94
1794	66.	RAO	= DATAN2 (YY(JS)-YY(IS), XX(JS)-XX(IS)) - PAI/2.0	CMAKE2.95
1795	67.	COSX	= DCOS(RAO)	CMAKE2.96
1796	68.	COSY	= DSIN(RAO)	CMAKE2.97
1797	69.	00 150	I = 1, NGS	CMAKE2.98
1798	70.	CALL	GHSET (I, XSI, WI)	CMAKE2.99
1799	C			CMAKE2.100
1800	71.	CALL	SFSEI (XSI, IS, XX, YY, RN, DNX, DNY, DETJ)	CMAKE2.101
1801	72.	CALL	MULT (RN, UU, UUU, 1, 4, 1)	CMAKE2.102
1802	73.	CALL	MULT (RN, VV, VVV, 1, 4, 1)	CMAKE2.103
1803	74.	CALL	MULT (RN, HH, HHH, 1, 4, 1)	CMAKE2.104
1804	75.	SC	= UUU/HHH * COSX + VVV/HHH * COSY	CMAKE2.105
1805	76.	CALL	MULT (RN, RN, WK1, 4, 1, 4)	CMAKE2.106
1806	77.	CALL	ADD (WK1, EK0, EK0, SC*WI*DETJ, ONE, 4, 4)	CMAKE2.107
1807	78.	150	CONTINUE	CMAKE2.108
1808	79.	160	CONTINUE	CMAKE2.109
1809	C			CMAKE2.110
1810	80.	CALL	ADD (EK0, EK1, EK1, DT, ONE, 4, 4)	CMAKE2.111
1811	C			CMAKE2.112
1812	C			CMAKE2.113
1813	81.	CALL	MSET2 (ZERO, EK3, 4*4) ***** (3) *****	CMAKE2.114

1814	82.	DO 320 I = 1, NGS	CHAKE2.115
1815	83.	CALL GHSET (I, XSI, WI)	CHAKE2.116
1816	84.	DO 310 J = 1, NGS	CHAKE2.117
1817	85.	CALL GHSET (J, ET, WJ)	CHAKE2.118
1818	C		CHAKE2.119
1819	86.	CALL ONDXY (XSI, ET, XX, YY, ONX, ONY, DETJ)	CHAKE2.120
1820	87.	CALL NSET (XSI, ET, RN)	CHAKE2.121
1821	88.	CALL MULT (RN, HH, HHH, 1, 4, 1)	CHAKE2.122
1822	89.	CALL MULT (RN, ONX, WK1, 4, 1, 4)	CHAKE2.123
1823	90.	CALL ADD (WK1, EK3, EK3, DTG*HHH*WI*WJ*DETJ, ONE, 4, 4)	CHAKE2.124
1824	91.	310 CONTINUE	CHAKE2.125
1825	92.	320 CONTINUE	CHAKE2.126
1826	C		CHAKE2.127
1827	C		***** (5) *****
1828	93.	CALL MOVE2 (EK1, EK5, 4*4)	CHAKE2.128
1829	C		CHAKE2.129
1830	C		***** (6) *****
1831	94.	CALL MSET2 (ZERO, EK6, 4*4)	CHAKE2.130
1832	95.	DO 620 I = 1, NGS	CHAKE2.131
1833	96.	CALL GHSET (I, XSI, WI)	CHAKE2.132
1834	97.	DO 610 J = 1, NGS	CHAKE2.133
1835	98.	CALL GHSET (J, ET, WJ)	CHAKE2.134
1836	C		CHAKE2.135
1837	99.	CALL ONDXY (XSI, ET, XX, YY, ONX, ONY, DETJ)	CHAKE2.136
1838	100.	CALL NSET (XSI, ET, RN)	CHAKE2.137
1839	101.	CALL MULT (RN, HH, HHH, 1, 4, 1)	CHAKE2.138
1840	102.	CALL MULT (RN, ONY, WK1, 4, 1, 4)	CHAKE2.139
1841	103.	CALL ADD (WK1, EK6, EK6, DTG*HHH*WI*WJ*DETJ, ONE, 4, 4)	CHAKE2.140
1842	104.	610 CONTINUE	CHAKE2.141
1843	105.	620 CONTINUE	CHAKE2.142
1844	C		CHAKE2.143
1845	C		***** MAKE C *****
1846	106.	CALL MSET2 (ZERO, EH, 12*12)	CHAKE2.144
1847	107.	DO 60 I = 1, 4	CHAKE2.145
1848	108.	DO 50 J = 1, 4	CHAKE2.146
1849	109.	EH(I, J) = EK1(I, J)	CHAKE2.147
1850	110.	EH(I, J+8) = EK3(I, J)	CHAKE2.148
1851	111.	EH(I+4, J+4) = EK5(I, J)	CHAKE2.149
1852	112.	EH(I+4, J+8) = EK6(I, J)	CHAKE2.150
1853	113.	50 CONTINUE	CHAKE2.151
1854	114.	60 CONTINUE	CHAKE2.152
1855	C		CHAKE2.153
1856	115.	IF (ICHK .GT. 1) CALL CHKP2 (IE, EH)	CHAKE2.154
1857	C		CHAKE2.155
1858	116.	READ (LU11) EMO	CHAKE2.156
1859	C		CHAKE2.157
1860	117.	CALL ADD (EMO, EH, EH, ONE, ONE, 12, 12)	CHAKE2.158
1861	C		CHAKE2.159
1862	118.	WRITE(LU13) EH	CHAKE2.160
1863	C		CHAKE2.161
1864	119.	IF (ICHK .GT. 0) CALL CHKP2 (IE, EH)	CHAKE2.162
1865	C		CHAKE2.163
1866	120.	999 CONTINUE	CHAKE2.164
1867	C		CHAKE2.165
1868	121.	RETURN	CHAKE2.166
1869	122.	END	CHAKE2.167
			CHAKE2.168
			CHAKE2.169
			CHAKE2.170

1870	1.	SUBROUTINE PHAKE (X, Y, U1, V1, ETA1, IELE, KELE, QBN,	PHAKE.2
1871	*	NPHAX, NEMAX, NONHX1)	PHAKE.3
1872	C		PHAKE.4
1873	2.	IMPLICIT REAL*8 (A-H, O-Z)	PHAKE.5
1874	C		PHAKE.6
1875	3.	COMMON / FILE / LIN, LOUT, LU1, LU2, LU3, LU4, /FILE/.2	
1876	*	LU10, LU11, LU12, LU13, LU14, LU15, /FILE/.3	
1877	4.	COMMON / CHECK / ICHECK(10) /CHECK/.2	
1878	5.	COMMON / CNTRL / NP, NE, NB, NM, /CNTRL/.2	
1879	*	NBN1, NBN2, /CNTRL/.3	
1880	6.	COMMON / GAUSS / NGS, G(5,4), H(5,4) /GAUSS/.2	
1881	7.	COMMON / TSTEP / ICOUNT, OT, TIME /TSTEP/.2	
1882	8.	COMMON / CONST / ZERO, ONE, TWO /CONST/.2	
1883	C		PHAKE.13
1884	9.	DIMENSION X(NPHAX), Y(NPHAX), U1(NPHAX), V1(NPHAX), ETA1(NPHAX),	PHAKE.14
1885	*	IELE(4, NEMAX), KELE(5, NEMAX), QBN(2, NONHX1)	PHAKE.15
1886	10.	DIMENSION XX(4), YY(4), UU(4), VV(4), EE(4), RN(4), ONX(4),	PHAKE.16
1887	*	ONY(4), EK(4,4), EK0(4), EK11(4), EK12(4), EK13(4),	PHAKE.17
1888	*	P(12)	PHAKE.18

1889		C								PHAKE.19
1890	11.	C	DATA	PAI	/ 3.141592653589800 /					PHAKE.20
1891		C								PHAKE.21
1892		C				I		I		PHAKE.22
1893		C				I	(11)	I		PHAKE.23
1894		C			MATRIX P =	I	(12)	I		PHAKE.24
1895		C				I	(13)	I		PHAKE.25
1896		C				I		I		PHAKE.26
1897		C								PHAKE.27
1898	12.	C	REWIND	LU14						PHAKE.28
1899	13.	C	REWIND	LU15						PHAKE.29
1900		C								PHAKE.30
1901	14.	C	ICLK	=	ICHECK(4)					PHAKE.31
1902	15.	C	IF (ICHK .GT. 0)	CALL	PAGE					PHAKE.32
1903		C								PHAKE.33
1904	16.	C	DO 999	IE = 1, NE						PHAKE.34
1905		C								PHAKE.35
1906	17.	C	READ (LU15)	EK						PHAKE.36
1907		C								PHAKE.37
1908	18.	C	DO 10	I = 1, 4						PHAKE.38
1909	19.	C	IP	=	IELE(I,IE)					PHAKE.39
1910	20.	C	XX(I)	=	X(IP)					PHAKE.40
1911	21.	C	YY(I)	=	Y(IP)					PHAKE.41
1912	22.	C	UU(I)	=	U1(IP)					PHAKE.42
1913	23.	C	VV(I)	=	V1(IP)					PHAKE.43
1914	24.	C	EE(I)	=	ETA1(IP)					PHAKE.44
1915	25.	C	10	CONTINUE						PHAKE.45
1916		C								PHAKE.46
1917		C								PHAKE.47
1918	26.	C	CALL	MULT	(EK, UU, EK11, 4, 4, 1)		*****	(11)	*****	PHAKE.48
1919		C								PHAKE.49
1920		C					*****	(12)	*****	PHAKE.50
1921	27.	C	CALL	MULT	(EK, VV, EK12, 4, 4, 1)					PHAKE.51
1922		C								PHAKE.52
1923		C					*****	(13)	*****	PHAKE.53
1924	28.	C	CALL	MULT	(EK, EE, EK13, 4, 4, 1)					PHAKE.54
1925		C								PHAKE.55
1926	29.	C	CALL	MSET2	(ZERO, EK0, 4)					PHAKE.56
1927	30.	C	N	=	KELE(5,IE)					PHAKE.57
1928	31.	C	DO 310	IS = 1, N						PHAKE.58
1929	32.	C	K	=	KELE(IS,IE)					PHAKE.59
1930	33.	C	IF (K .EQ. 0)		GO TO 310					PHAKE.60
1931		C								PHAKE.61
1932	34.	C	DO 300	I = 1, NGS						PHAKE.62
1933	35.	C	CALL	GWSET	(I, XSI, WI)					PHAKE.63
1934	36.	C	CALL	SFSET	(XSI, IS, XX, YY, RN, ONX, ONY, DETJ)				PHAKE.64	
1935	37.	C	QQBN	= - (QBN(1,K)*QBN(2,K)) / 2.0 * (WI*DETJ)						PHAKE.65
1936	38.	C	CALL	ADD	(RN, EK0, EK0, QQBN, ONE, 4, 1)				PHAKE.66	
1937	39.	C	300	CONTINUE						PHAKE.67
1938	40.	C	310	CONTINUE						PHAKE.68
1939		C								PHAKE.69
1940	41.	C	CALL	ADD	(EK0, EK13, EK13, -OT, ONE, 4, 1)				PHAKE.70	
1941		C								PHAKE.71
1942		C					*****	MAKE P	*****	PHAKE.72
1943	42.	C	CALL	MOVE2	(EK11, P(1), 4)					PHAKE.73
1944	43.	C	CALL	MOVE2	(EK12, P(5), 4)					PHAKE.74
1945	44.	C	CALL	MOVE2	(EK13, P(9), 4)					PHAKE.75
1946		C								PHAKE.76
1947	45.	C	WRITE(LU14)	P						PHAKE.77
1948		C								PHAKE.78
1949	46.	C	IF (ICHK .GT. 0)	CALL	CHKP3 (IE, P)					PHAKE.79
1950		C								PHAKE.80
1951	47.	C	999	CONTINUE						PHAKE.81
1952		C								PHAKE.82
1953	48.	C	RETURN							PHAKE.83
1954	49.	C	END							PHAKE.84

1955	1.	SUBROUTINE BOUND2 (IELE, INBN, EFIX, BMAT, TSM, NP, NE, NB, NBN2)	BOUND2.2
1956	C		BOUND2.3
1957	2.	IMPLICIT REAL*8 (A-H, O-Z)	BOUND2.4
1958	C		BOUND2.5
1959	3.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 , /FILE/,2	
1960	*	COMMON / CHECK / LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15 , /FILE/,3	
1961	4.	COMMON / CHECK / ICHECK(10)	/CHECK/,2
1962	5.	COMMON / CONST / ZERO ,ONE ,THO	/CONST/,2
1963	C		BOUND2.9
1964	6.	DIMENSION IELE(4,NE) ,INBN(3,NBN 2),EFIX(NP) ,BMAT(3*NP),	BOUND2.10
1965	*	TSM(3*NP,2*NB-1)	BOUND2.11
1966	7.	DIMENSION IFONO(12), JFONO(50), A(12,50), B(12), KND(12),	BOUND2.12
1967	*	EH(12,12), P(12), AT(50,12), EHL(50,50), PL(50),	BOUND2.13
1968	*	BUF1(50,12), BUF2(50)	BOUND2.14
1969	8.	DATA KND / 1, 4, 7, 10, 2, 5, 8, 11, 3, 6, 9, 12 /	BOUND2.15
1970	C		BOUND2.16
1971	9.	REWINO LU12	BOUND2.17
1972	10.	REWINO LU13	BOUND2.18
1973	11.	REWINO LU14	BOUND2.19
1974	C		BOUND2.20
1975	12.	ICHK1 = ICHECK(5)	BOUND2.21
1976	13.	ICHK2 = ICHECK(6)	BOUND2.22
1977	14.	IF (ICHK1+ICHK2 .GT. 1) CALL PAGE	BOUND2.23
1978	C		BOUND2.24
1979	15.	N = 3 * NP	BOUND2.25
1980	16.	M = 2 * NB - 1	BOUND2.26
1981	17.	CALL MSET2 (ZERO, BMAT, N)	BOUND2.27
1982	18.	CALL MSET2 (ZERO, TSM, N*M)	BOUND2.28
1983	C		BOUND2.29
1984	19.	DO 999 IE = 1, NE	BOUND2.30
1985	C		BOUND2.31
1986	20.	READ (LU13) EM	BOUND2.32
1987	21.	READ (LU14) P	BOUND2.33
1988	C		BOUND2.34
1989	22.	DO 20 I = 1, 12	BOUND2.35
1990	23.	B(KND(I)) = P(I)	BOUND2.36
1991	24.	DO 10 J = 1, 12	BOUND2.37
1992	25.	A(KND(I),KND(J)) = EH(I,J)	BOUND2.38
1993	26.	10 CONTINUE	BOUND2.39
1994	27.	20 CONTINUE	BOUND2.40
1995	28.	CALL MOVE2 (A, EH, 12*12)	BOUND2.41
1996	29.	CALL MOVE2 (B, P, 12)	BOUND2.42
1997	C		BOUND2.43
1998	30.	KN = 12	BOUND2.44
1999	31.	DO 40 I = 1, KN	BOUND2.45
2000	32.	PL(I) = P(I)	BOUND2.46
2001	33.	DO 30 J = 1, KN	BOUND2.47
2002	34.	EHL(I,J) = EH(I,J)	BOUND2.48
2003	35.	30 CONTINUE	BOUND2.49
2004	36.	40 CONTINUE	BOUND2.50
2005	C		BOUND2.51
2006	37.	NBN = 0	BOUND2.52
2007	38.	DO 60 I = 1, 4	BOUND2.53
2008	39.	DO 50 J = 1, NBN2	BOUND2.54
2009	40.	IF (IELE(I,IE) .EQ. INBN(1,J)) NBN = NBN + 1	BOUND2.55
2010	41.	50 CONTINUE	BOUND2.56
2011	42.	60 CONTINUE	BOUND2.57
2012	C		BOUND2.58
2013	43.	DO 80 I = 1, 4	BOUND2.59
2014	44.	DO 70 J = 1, 3	BOUND2.60
2015	45.	JFO = 3 * (I-1) + J	BOUND2.61
2016	46.	JFONO(JFO) = 3 * (IELE(I,IE)-1) + J	BOUND2.62
2017	47.	70 CONTINUE	BOUND2.63
2018	48.	80 CONTINUE	BOUND2.64
2019	49.	IF (NBN .EQ. 0) GO TO 100	BOUND2.65
2020	C		BOUND2.66
2021	50.	READ (LU12) IIE, KN	BOUND2.67
2022	51.	READ (LU12) IFONO	BOUND2.68
2023	52.	READ (LU12) JFONO	BOUND2.69
2024	53.	READ (LU12) A	BOUND2.70
2025	54.	READ (LU12) B	BOUND2.71
2026	C		BOUND2.72
2027	55.	CALL TRANS2 (A, AT, 12, KN, 12, 50)	BOUND2.73
2028	56.	CALL MULT2 (AT, EM, BUF1, KN, 12, 12, 50, 12, 50)	BOUND2.74
2029	57.	CALL MULT2 (BUF1, A, EHL, KN, 12, KN, 50, 12, 50)	BOUND2.75
2030	58.	CALL MULT2 (AT, P, PL, KN, 12, 1, 50, 12, 50)	BOUND2.76
2031	59.	CALL MULT2 (AT, EM, BUF1, KN, 12, 12, 50, 12, 50)	BOUND2.77
2032	60.	CALL MULT2 (BUF1, B, BUF2, KN, 12, 1, 50, 12, 50)	BOUND2.78
2033	61.	CALL ADD (PL, BUF2, PL, ONE, -ONE, KN, 1)	BOUND2.79

2034	C				BOUND2.80
2035	62.	100	CONTINUE		BOUND2.81
2036	63.		IF (ICHK1 .GT. 1)	CALL CHKP8 (EML, JFONO, IE, KN, KN, 50)	BOUND2.82
2037	64.		IF (ICHK1 .GT. 1)	CALL CHKP6 (PL, KN, *PL+, 0)	BOUND2.83
2038	C				BOUND2.84
2039	65.		DO 120 I = 1, KN		BOUND2.85
2040	66.		IND = JFONO(I)		BOUND2.86
2041	67.		BHAT(IND) = BHAT(IND) + PL(I)		BOUND2.87
2042	68.		IF (DABS(BHAT(IND)) .LT. 1.00-10)	BHAT(IND) = 0.0	BOUND2.88
2043	69.		DO 110 J = 1, KN		BOUND2.89
2044	70.		JND = JFONO(J) + NB - IND		BOUND2.90
2045	71.		TSH(IND,JND) = TSH(IND,JND) + EHL(I,J)		BOUND2.91
2046	72.		IF (DABS(TSH(IND,JND)) .LT. 1.00-10)	TSH(IND,JND) = 0.0	BOUND2.92
2047	73.	110	CONTINUE		BOUND2.93
2048	74.	120	CONTINUE		BOUND2.94
2049	C				BOUND2.95
2050	75.	999	CONTINUE		BOUND2.96
2051	C				BOUND2.97
2052	76.		DO 210 I = 1, N		BOUND2.98
2053	77.		IF (TSH(I,NB) .EQ. 0.0)	TSH(I,NB) = 1.0	BOUND2.99
2054	78.	210	CONTINUE		BOUND2.100
2055	C				BOUND2.101
2056	C ***		SET EFIX ***		BOUND2.102
2057	79.		DO 340 I = 1, NP		BOUND2.103
2058	80.		IF (EFIX(I) .LE. 0.0)	GO TO 340	BOUND2.104
2059	81.		IFD = 3 * I		BOUND2.105
2060	82.		JS = NB + IFD		BOUND2.106
2061	C				BOUND2.107
2062	83.		DO 310 J = 1, N		BOUND2.108
2063	84.		JS = JS - 1		BOUND2.109
2064	85.		IF (JS .GT. M)	GO TO 310	BOUND2.110
2065	86.		IF (JS .EQ. 0)	GO TO 320	BOUND2.111
2066	87.		BHAT(J) = BHAT(J) - TSH(J,JS)*EFIX(I)		BOUND2.112
2067	88.		TSH(J,JS) = 0.0		BOUND2.113
2068	89.	310	CONTINUE		BOUND2.114
2069	C				BOUND2.115
2070	90.	320	CONTINUE		BOUND2.116
2071	91.		DO 330 J = 1, M		BOUND2.117
2072	92.		TSH(IFD,J) = 0.0		BOUND2.118
2073	93.	330	CONTINUE		BOUND2.119
2074	94.		TSH(IFD,NB) = 1.0		BOUND2.120
2075	95.		BHAT(IFD) = EFIX(I)		BOUND2.121
2076	96.	340	CONTINUE		BOUND2.122
2077	C				BOUND2.123
2078	97.		IF (ICHK2 .GT. 0)	CALL CHKP5 (TSH, N, M)	BOUND2.124
2079	98.		IF (ICHK2 .GT. 0)	CALL CHKP6 (BHAT, N, *BHAT+, 0)	BOUND2.125
2080	C				BOUND2.126
2081	99.		RETURN		BOUND2.127
2082	100.		END		BOUND2.128

2083	1.		SUBROUTINE GHSET (I, GG, HH)		GHSET.2
2084	C				GHSET.3
2085	2.		IMPLICIT REAL*8 (A-H, O-Z)		GHSET.4
2086	3.		COMMON / GAUSS / NGS ,G(5,4) ,H(5,4)		/GAUSS/.2
2087	C				GHSET.6
2088	4.		GG = G(I, NGS-1)		GHSET.7
2089	5.		HH = H(I, NGS-1)		GHSET.8
2090	C				GHSET.9
2091	6.		RETURN		GHSET.10
2092	7.		END		GHSET.11

2093	1.		SUBROUTINE NSET(X,E,RN)		NSET.2
2094	C				NSET.3
2095	2.		IMPLICIT REAL*8 (A-H, O-Z)		NSET.4
2096	3.		DIMENSION RN(4)		NSET.5
2097	C				NSET.6
2098	4.		RN(1) = 0.25*(1.0-E)*(1.0-X)		NSET.7
2099	5.		RN(2) = 0.25*(1.0-E)*(1.0*X)		NSET.8
2100	6.		RN(3) = 0.25*(1.0+E)*(1.0*X)		NSET.9
2101	7.		RN(4) = 0.25*(1.0+E)*(1.0-X)		NSET.10
2102	C				NSET.11
2103	8.		RETURN		NSET.12
2104	9.		END		NSET.13

2105	1.	SUBROUTINE ONOXY (X, E, XX, YY, ONX, ONY, DETJ)	ONOXY.2
2106	C		ONOXY.3
2107	2.	IMPLICIT REAL*8 (A-H,O-Z)	ONOXY.4
2108	3.	DIMENSION XX(4), YY(4), ONX(4), ONY(4), OXY(2,4)	ONOXY.5
2109	4.	DIMENSION RJ(2,2), RJI(2,2), I1(2,4), I2(4,2)	ONOXY.6
2110	C		ONOXY.7
2111	5.	I1(1,1) = -0.25*(1.0-E)	ONOXY.8
2112	6.	I1(1,2) = 0.25*(1.0-E)	ONOXY.9
2113	7.	I1(1,3) = 0.25*(1.0+E)	ONOXY.10
2114	8.	I1(1,4) = -0.25*(1.0+E)	ONOXY.11
2115	9.	I1(2,1) = -0.25*(1.0-X)	ONOXY.12
2116	10.	I1(2,2) = -0.25*(1.0+X)	ONOXY.13
2117	11.	I1(2,3) = 0.25*(1.0+X)	ONOXY.14
2118	12.	I1(2,4) = 0.25*(1.0-X)	ONOXY.15
2119	C		ONOXY.16
2120	13.	CALL MOVE2(XX, I2(1,1), 4)	ONOXY.17
2121	14.	CALL MOVE2(YY, I2(1,2), 4)	ONOXY.18
2122	C		ONOXY.19
2123	15.	CALL MULT(I1, I2, RJ, 2, 4, 2)	ONOXY.20
2124	C		ONOXY.21
2125	16.	DETJ = RJ(1,1)*RJ(2,2) - RJ(1,2)*RJ(2,1)	ONOXY.22
2126	C		ONOXY.23
2127	17.	RJI(1,1) = RJ(2,2) / DETJ	ONOXY.24
2128	18.	RJI(1,2) = -RJ(1,2) / DETJ	ONOXY.25
2129	19.	RJI(2,1) = -RJ(2,1) / DETJ	ONOXY.26
2130	20.	RJI(2,2) = RJ(1,1) / DETJ	ONOXY.27
2131	C		ONOXY.28
2132	21.	CALL MULT(RJI, I1, OXY, 2, 2, 4)	ONOXY.29
2133	22.	DO 10 I = 1, 4	ONOXY.30
2134	23.	ONX(I) = OXY(1, I)	ONOXY.31
2135	24.	ONY(I) = OXY(2, I)	ONOXY.32
2136	25.	10 CONTINUE	ONOXY.33
2137	C		ONOXY.34
2138	26.	RETURN	ONOXY.35
2139	27.	END	ONOXY.36

2140	1.	SUBROUTINE SFSET (XE, IS, XX, YY, RN, ONX, ONY, DETJ)	SFSET.2
2141	C		SFSET.3
2142	2.	IMPLICIT REAL*8 (A-H,O-Z)	SFSET.4
2143	C		SFSET.5
2144	3.	DIMENSION XX(4), YY(4), RN(4), ONX(4), ONY(4)	SFSET.6
2145	4.	DIMENSION XXX(5), YYY(5)	SFSET.7
2146	C		SFSET.8
2147	5.	CALL MOVE2(XX, XXX, 4)	SFSET.9
2148	6.	CALL MOVE2(YY, YYY, 4)	SFSET.10
2149	7.	XXX(5) = XX(1)	SFSET.11
2150	8.	YYY(5) = YY(1)	SFSET.12
2151	C		SFSET.13
2152	9.	GO TO (1, 2, 3, 4), IS	SFSET.14
2153	C		SFSET.15
2154	10.	1 CONTINUE	SFSET.16
2155	11.	X = XE	SFSET.17
2156	12.	E = -1.0	SFSET.18
2157	13.	GO TO 5	SFSET.19
2158	14.	2 CONTINUE	SFSET.20
2159	15.	X = 1.0	SFSET.21
2160	16.	E = XE	SFSET.22
2161	17.	GO TO 5	SFSET.23
2162	18.	3 CONTINUE	SFSET.24
2163	19.	X = XE	SFSET.25
2164	20.	E = 1.0	SFSET.26
2165	21.	GO TO 5	SFSET.27
2166	22.	4 CONTINUE	SFSET.28
2167	23.	X = -1.0	SFSET.29
2168	24.	E = XE	SFSET.30
2169	C		SFSET.31
2170	25.	5 CONTINUE	SFSET.32
2171	26.	CALL NSET(X, E, RN)	SFSET.33
2172	27.	CALL ONOXY (X, E, XX, YY, ONX, ONY, DETJ)	SFSET.34
2173	C		SFSET.35
2174	28.	X1 = XXX(IS)	SFSET.36
2175	29.	Y1 = YYY(IS)	SFSET.37
2176	30.	X2 = XXX(IS+1)	SFSET.38
2177	31.	Y2 = YYY(IS+1)	SFSET.39
2178	C		SFSET.40
2179	32.	DETJ = DSQRT((X1-X2)**2 + (Y1-Y2)**2) / 2.0	SFSET.41
2180	C		SFSET.42
2181	33.	RETURN	SFSET.43
2182	34.	END	SFSET.44

2183	1.	SUBROUTINE SOLVNS(A , B ,NEQ,HB)	SOLVNS.2
2184	C		SOLVNS.3
2185	2.	IMPLICIT REAL*8 (A-H,O-Z)	SOLVNS.4
2186	C		SOLVNS.5
2187	3.	DIMENSION A(NEQ,1),B(NEQ)	SOLVNS.6
2188	C		SOLVNS.7
2189	4.	DO 100 I=1,NEQ	SOLVNS.8
2190	5.	IF(I.EQ.1) GO TO 100	SOLVNS.9
2191	6.	JS=1	SOLVNS.10
2192	7.	JE=NEQ	SOLVNS.11
2193	8.	IF(I.GT.HB) JS=I-HB+1	SOLVNS.12
2194	9.	IF(I.LE.NEQ-HB) JE=I+HB-1	SOLVNS.13
2195	10.	DO 40 J=JS,JE	SOLVNS.14
2196	11.	JJ=J-I+HB	SOLVNS.15
2197	12.	IF(I.LE.J) GO TO 42	SOLVNS.16
2198	13.	KS=I-HB+1	SOLVNS.17
2199	14.	KE=J-1	SOLVNS.18
2200	15.	GO TO 44	SOLVNS.19
2201	16.	42 CONTINUE	SOLVNS.20
2202	17.	KS=J-HB+1	SOLVNS.21
2203	18.	KE=I-1	SOLVNS.22
2204	19.	44 CONTINUE	SOLVNS.23
2205	20.	IF(KS.LE.0) KS=1	SOLVNS.24
2206	C		SOLVNS.25
2207	21.	IF(KS.GT.KE) GO TO 20	SOLVNS.26
2208	22.	S=0	SOLVNS.27
2209	23.	DO 10 K=KS,KE	SOLVNS.28
2210	24.	S=S+A(I,K-I+HB)*A(K,J-K+HB)	SOLVNS.29
2211	25.	10 CONTINUE	SOLVNS.30
2212	26.	A(I,JJ)=A(I,JJ)-S	SOLVNS.31
2213	27.	20 CONTINUE	SOLVNS.32
2214	C		SOLVNS.33
2215	28.	IF(I.LE.J) GO TO 30	SOLVNS.34
2216	29.	A(I,JJ)=A(I,JJ)/A(J,HB)	SOLVNS.35
2217	30.	B(I)=B(I)-A(I,JJ)*B(J)	SOLVNS.36
2218	31.	30 CONTINUE	SOLVNS.37
2219	32.	40 CONTINUE	SOLVNS.38
2220	33.	100 CONTINUE	SOLVNS.39
2221	C		SOLVNS.40
2222	34.	DO 200 K=1,NEQ	SOLVNS.41
2223	35.	I=NEQ-K+1	SOLVNS.42
2224	36.	JS=I+1	SOLVNS.43
2225	37.	JE=NEQ	SOLVNS.44
2226	38.	IF(I.LE.NEQ-HB) JE=I+HB-1	SOLVNS.45
2227	39.	IF(JS.GT.JE) GO TO 120	SOLVNS.46
2228	40.	DO 110 J=JS,JE	SOLVNS.47
2229	41.	JJ=J-I+HB	SOLVNS.48
2230	42.	B(I)=B(I)-A(I,JJ)*B(J)	SOLVNS.49
2231	43.	110 CONTINUE	SOLVNS.50
2232	44.	120 CONTINUE	SOLVNS.51
2233	45.	B(I)=B(I)/A(I,HB)	SOLVNS.52
2234	46.	200 CONTINUE	SOLVNS.53
2235	C		SOLVNS.54
2236	47.	RETURN	SOLVNS.55
2237	48.	END	SOLVNS.56

2238	1.	SUBROUTINE DEPEND (BMAT, NP)	DEPEND.2
2239	C		DEPEND.3
2240	2.	IMPLICIT REAL*8 (A-H,O-Z)	DEPEND.4
2241	C		DEPEND.5
2242	3.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 , /FILE/.2	
2243		* LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15 , /FILE/.3	
2244	4.	COMMON / CONST / ZERO ,ONE ,TWO , /CONST/.2	
2245	C		DEPEND.8
2246	5.	DIMENSION BMAT(3*NP)	DEPEND.9
2247	6.	DIMENSION JFONO(50),A(12,50),B(12),IFONO(12),P(12),PL(50)	DEPEND.10
2248	C		DEPEND.11
2249	7.	REWIND LU12	DEPEND.12
2250	C		DEPEND.13
2251	8.	10 CONTINUE	DEPEND.14
2252	9.	READ (LU12,END=999) IE, KN	DEPEND.15
2253	10.	READ (LU12) IFONO	DEPEND.16
2254	11.	READ (LU12) JFONO	DEPEND.17

2255	12.	READ (LU12) A	DEPEND.18
2256	13.	READ (LU12) B	DEPEND.19
2257	C		DEPEND.20
2258	14.	DO 30 I=1,KN	DEPEND.21
2259	15.	IF0 = JF0NO(I)	DEPEND.22
2260	16.	PL(I) = BMAT(IF0)	DEPEND.23
2261	17.	30 CONTINUE	DEPEND.24
2262	C		DEPEND.25
2263	18.	CALL MULT(A,PL,P,12,KN,1)	DEPEND.26
2264	19.	CALL ADD(P,B,P,ONE,ONE,12,1)	DEPEND.27
2265	C		DEPEND.28
2266	20.	DO 40 I=1,12	DEPEND.29
2267	21.	IF0 = IF0NO(I)	DEPEND.30
2268	22.	BMAT(IF0) = P(I)	DEPEND.31
2269	23.	40 CONTINUE	DEPEND.32
2270	C		DEPEND.33
2271	24.	GO TO 10	DEPEND.34
2272	C		DEPEND.35
2273	25.	999 CONTINUE	DEPEND.36
2274	C		DEPEND.37
2275	26.	RETURN	DEPEND.38
2276	27.	END	DEPEND.39

2277	1.	SUBROUTINE CONVEGE (B,U,U1,V,V1,ETA,ETA1,IELE,NPHAX,NEMAX,IIFLG)	CONVEGE.2
2278	C		CONVEGE.3
2279	2.	IMPLICIT REAL*8 (A-H, O-Z)	CONVEGE.4
2280	C		CONVEGE.5
2281	3.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 , /FILE/.2	
2282	*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15 /FILE/.3	
2283	4.	COMMON / CHECK / ICHECK(10)	/CHECK/.2
2284	5.	COMMON / CNTRL / NP ,NE ,NB ,NM , /CNTRL/.2	
2285	*	NBN1 ,NBN2 , /CNTRL/.3	
2286	6.	COMMON / ISTEP / ICOUNT ,DT ,TIME /ISTEP/.2	
2287	7.	COMMON / CONST / ZERO ,ONE ,TWO /CONST/.2	
2288	8.	COMMON / ITERAT / MAXIT ,MINER ,IFLG(3) ,EPSLN(3) /ITERAT/.2	
2289	9.	COMMON / POINT / NPOINT ,IPL(10) ,IPR(10) ,CA(10) ,CB(10) /POINT/.2	
2290	10.	COMMON / GRAVITY / GRAV /GRAVITY/.2	
2291	C		CONVEGE.14
2292	11.	DIMENSION B(NPHAX) ,U(NPHAX) ,U1(NPHAX) ,V(NPHAX) ,V1(NPHAX) ,	CONVEGE.15
2293	*	ETA(NPHAX) ,ETA1(NPHAX) ,IELE(4,NEMAX)	CONVEGE.16
2294	12.	DIMENSION BR(4) ,UU(4) ,VV(4) ,HH(4) ,EK(4,4) ,HK(4) ,Q(10) ,	CONVEGE.17
2295	*	Q1(10)	CONVEGE.18
2296	C		CONVEGE.19
2297	13.	REHIND LU15	CONVEGE.20
2298	C		CONVEGE.21
2299	14.	ICLK = ICHECK(3)	CONVEGE.22
2300	C		CONVEGE.23
2301	15.	IIFLG = 0	CONVEGE.24
2302	16.	WRITE(LU10) ICOUNT, DT, TIME	CONVEGE.25
2303	C		CONVEGE.26
2304	C	***** EPSLN(1) *****	CONVEGE.27
2305	17.	ERHX1 = 0.0	CONVEGE.28
2306	18.	DO 110 I = 1, NP	CONVEGE.29
2307	19.	ER = (ETA(I) - ETA1(I)) / (DT * (ETA(I)-B(I)))	CONVEGE.30
2308	20.	IF (DABS(ER) .LE. DABS(ERHX1)) GO TO 110	CONVEGE.31
2309	21.	IP = I	CONVEGE.32
2310	22.	ERHX1 = ER	CONVEGE.33
2311	23.	110 CONTINUE	CONVEGE.34
2312	24.	IF (DABS(ERHX1) .GT. EPSLN(1)) IIFLG = IIFLG + IFLG(1)	CONVEGE.35
2313	C		CONVEGE.36
2314	25.	IF (ICHK .EQ. 0) GO TO 200	CONVEGE.37
2315	26.	CALL LINE(2)	CONVEGE.38
2316	27.	WRITE(LOUT,601) IP, ETA1(IP), ETA(IP), ERHX1	CONVEGE.39
2317	C		CONVEGE.40
2318	C	***** EPSLN(2) *****	CONVEGE.41
2319	28.	200 CONTINUE	CONVEGE.42
2320	29.	IF (ICOUNT .GT. 1) GO TO 230	CONVEGE.43
2321	30.	VE1 = 0.0	CONVEGE.44
2322	31.	UE1 = 0.0	CONVEGE.45
2323	32.	DO 220 IE = 1, NE	CONVEGE.46
2324	C		CONVEGE.47
2325	33.	READ (LU15) EK	CONVEGE.48
2326	C		CONVEGE.49
2327	34.	DO 210 I = 1, 4	CONVEGE.50

2328	35.	IP = IELE(I,IE)	CONVGE.51
2329	36.	BB(I) = B(IP)	CONVGE.52
2330	37.	UU(I) = U(IP)	CONVGE.53
2331	38.	VV(I) = V(IP)	CONVGE.54
2332	39.	HH(I) = ETA1(IP) - BB(I)	CONVGE.55
2333	40.	210 CONTINUE	CONVGE.56
2334	41.	HBAR = (HH(1) + HH(2) + HH(3) + HH(4)) / 4.0	CONVGE.57
2335	C		CONVGE.58
2336	42.	CALL MULT (UU, EK, HK, 1, 4, 4)	CONVGE.59
2337	43.	CALL MULT (HK, UU, UUU, 1, 4, 1)	CONVGE.60
2338	44.	CALL MULT (VV, EK, HK, 1, 4, 4)	CONVGE.61
2339	45.	CALL MULT (HK, VV, VVV, 1, 4, 1)	CONVGE.62
2340	46.	VE1 = VE1 + 0.5/HBAR * (UUU+VVV)	CONVGE.63
2341	C		CONVGE.64
2342	47.	CALL MULT (HH, EK, HK, 1, 4, 4)	CONVGE.65
2343	48.	CALL MULT (HK, HH, HHH, 1, 4, 1)	CONVGE.66
2344	49.	CALL MULT (BB, EK, HK, 1, 4, 4)	CONVGE.67
2345	50.	CALL MULT (HK, HH, BBB, 1, 4, 1)	CONVGE.68
2346	51.	UE1 = UE1 + GRAV * (HHH+BBB)	CONVGE.69
2347	52.	220 CONTINUE	CONVGE.70
2348	C		CONVGE.71
2349	53.	TE1 = VE1 + UE1	CONVGE.72
2350	54.	REWIND LU15	CONVGE.73
2351	C		CONVGE.74
2352	55.	IF (ICHK .EQ. 0) GO TO 230	CONVGE.75
2353	56.	CALL LINE(2)	CONVGE.76
2354	57.	WRITE(LOUT,602) VE1, UE1, TE1	CONVGE.77
2355	C		CONVGE.78
2356	58.	230 CONTINUE	CONVGE.79
2357	59.	VE = 0.0	CONVGE.80
2358	60.	UE = 0.0	CONVGE.81
2359	61.	DO 250 IE = 1, NE	CONVGE.82
2360	C		CONVGE.83
2361	62.	READ (LU15) EK	CONVGE.84
2362	C		CONVGE.85
2363	63.	DO 240 I = 1, 4	CONVGE.86
2364	64.	IP = IELE(I,IE)	CONVGE.87
2365	65.	BB(I) = B(IP)	CONVGE.88
2366	66.	UU(I) = U(IP)	CONVGE.89
2367	67.	VV(I) = V(IP)	CONVGE.90
2368	68.	HH(I) = ETA (IP) - BB(I)	CONVGE.91
2369	69.	240 CONTINUE	CONVGE.92
2370	70.	HBAR = (HH(1) + HH(2) + HH(3) + HH(4)) / 4.0	CONVGE.93
2371	C		CONVGE.94
2372	71.	CALL MULT (UU, EK, HK, 1, 4, 4)	CONVGE.95
2373	72.	CALL MULT (HK, UU, UUU, 1, 4, 1)	CONVGE.96
2374	73.	CALL MULT (VV, EK, HK, 1, 4, 4)	CONVGE.97
2375	74.	CALL MULT (HK, VV, VVV, 1, 4, 1)	CONVGE.98
2376	75.	VE = VE + 0.5/HBAR * (UUU+VVV)	CONVGE.99
2377	C		CONVGE.100
2378	76.	CALL MULT (HH, EK, HK, 1, 4, 4)	CONVGE.101
2379	77.	CALL MULT (HK, HH, HHH, 1, 4, 1)	CONVGE.102
2380	78.	CALL MULT (BB, EK, HK, 1, 4, 4)	CONVGE.103
2381	79.	CALL MULT (HK, HH, BBB, 1, 4, 1)	CONVGE.104
2382	80.	UE = UE + GRAV * (HHH+BBB)	CONVGE.105
2383	81.	250 CONTINUE	CONVGE.106
2384	C		CONVGE.107
2385	82.	TE = VE + UE	CONVGE.108
2386	83.	ERMx2 = (TE - TE1) / (OI * TE)	CONVGE.109
2387	84.	IF (DABS(ERMx2) .GT. EPSLN(2)) IIFLG = IIFLG + IFLG(2)	CONVGE.110
2388	C		CONVGE.111
2389	85.	IF (ICHK .EQ. 0) GO TO 260	CONVGE.112
2390	86.	CALL LINE(1)	CONVGE.113
2391	87.	WRITE(LOUT,603) VE, UE, TE, ERMx2	CONVGE.114
2392	C		CONVGE.115
2393	88.	260 CONTINUE	CONVGE.116
2394	89.	TE1 = TE	CONVGE.117
2395	C		CONVGE.118
2396	C		CONVGE.119
2397	90.	ERMx3 = 0.0	CONVGE.120
2398	91.	IF (NPOINT .LE. 0) GO TO 400	CONVGE.121
2399	92.	IF (ICOUNT .GT. 1) GO TO 320	CONVGE.122
2400	C		CONVGE.123
2401	93.	DO 310 I = 1, NPOINT	CONVGE.124
2402	94.	IP = IPL(I)	CONVGE.125
2403	95.	Q1(I) = CA(I) * ETA1(IP) + CB(I)	CONVGE.126
2404	96.	310 CONTINUE	CONVGE.127
2405	C		CONVGE.128
2406	97.	WRITE(LU10) (Q1(I), I=1,NPOINT)	CONVGE.129

***** EPSLN(3) *****

2407	C			CONVGE.130
2408	98.	320	CONTINUE	CONVGE.131
2409	99.	DO 330	I = 1, NPOINT	CONVGE.132
2410	100.	IP	= IPL(I)	CONVGE.133
2411	101.	Q (I)	= CA(I) * ETA (IP) + CB(I)	CONVGE.134
2412	102.	ER	= (Q(I) - Q1(I)) / (Q1 + Q(I))	CONVGE.135
2413	103.	IF (DABS(ER) .LE. DABS(ERMX3))	GO TO 330	CONVGE.136
2414	104.	IIP	= I	CONVGE.137
2415	105.	ERMX3	= ER	CONVGE.138
2416	106.	330	CONTINUE	CONVGE.139
2417	107.	IF (DABS(ERMX3) .GT. EPSLN(3))	IIFLG = IIFLG + IFLG(3)	CONVGE.140
2418	C			CONVGE.141
2419	108.	WRITE(LU10)	(Q (I), I=1,NPOINT)	CONVGE.142
2420	C			CONVGE.143
2421	109.	IF (ICHK .EQ. 0)	GO TO 340	CONVGE.144
2422	110.	CALL LINE(1)		CONVGE.145
2423	111.	WRITE(LOUT,604)	IIP, Q1(IIP), Q(IIP), ERMX3	CONVGE.146
2424	C			CONVGE.147
2425	112.	340	CONTINUE	CONVGE.148
2426	113.	CALL	MOVE2 (Q, Q1, NPOINT)	CONVGE.149
2427	C			CONVGE.150
2428	114.	400	CONTINUE	CONVGE.151
2429	C			CONVGE.152
2430	115.	WRITE(LU10)	ERMx1, ERMx2, ERMx3	CONVGE.153
2431	C			CONVGE.154
2432	116.	601	FORMAT(+D*CHECK* (ETA-ETA1)/DT ERROR NODE NO.+, I5, 6X,	CONVGE.155
2433		*	+ETA1 =+, F10.4, 5X, +ETA =+, F10.4, 5X, +ERMx1 =+, E11.3)	CONVGE.156
2434	117.	602	FORMAT(+D*CHECK* INITIAL ENERGY VE =+, E10.3, 6X,	CONVGE.157
2435		*	+UE =+, E10.3, 5X, +IE =+, E10.3)	CONVGE.158
2436	118.	603	FORMAT(+ *CHECK* TOTAL ENERGY ERROR VE =+, E10.3, 6X,	CONVGE.159
2437		*	+UE =+, E10.3, 5X, +IE =+, E10.3, 5X, +ERMx2 =+, E11.3)	CONVGE.160
2438	119.	604	FORMAT(+ *CHECK* (Q - Q1)/DT ERROR POINT NO.+, I4, 8X,	CONVGE.161
2439		*	+Q1 =+, F10.2, 5X, + Q =+, F10.2, 5X, +ERMx3 =+, E11.3)	CONVGE.162
2440	120.	RETURN		CONVGE.163
2441	121.	END		CONVGE.164

2442	1.	SUBROUTINE	PRINT (U, V, ETA, B, NPHAX, ISW)	PRINT.2
2443	C			PRINT.3
2444	2.	IMPLICIT	REAL*8 (A-H,O-Z)	PRINT.4
2445	C			PRINT.5
2446	3.	COMMON / FILE	/ LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 , /FILE/.2	
2447		*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15 , /FILE/.3	
2448	4.	COMMON / CNTRL	/ NP ,NE ,NB ,NM , /CNTRL/.2	
2449		*	NBN1 ,NBN2 , /CNTRL/.3	
2450	C			PRINT.8
2451	5.	DIMENSION	U(NPHAX),V(NPHAX),ETA(NPHAX),B(NPHAX)	PRINT.9
2452	C			PRINT.10
2453	6.	CALL	PAGE	PRINT.11
2454	7.	CALL	LINE(2)	PRINT.12
2455	8.	IF (ISW .EQ. 1)	WRITE(LOUT,61)	PRINT.13
2456	9.	IF (ISW .EQ. 2)	WRITE(LOUT,62)	PRINT.14
2457	10.	61	FORMAT(1H0, 10X, *** INITIAL CONCITION ***)	PRINT.15
2458	11.	62	FORMAT(1H0, 10X, *** FINAL CONDITION ***)	PRINT.16
2459	12.	CALL	PEJECT(6,1,4)	PRINT.17
2460	C			PRINT.18
2461	13.	DO 10	I=1,NP	PRINT.19
2462	14.	H	= ETA(I) - B(I)	PRINT.20
2463	15.	CALL	PEJECT(6,0,1)	PRINT.21
2464	16.	WRITE(LOUT,601)	I,U(I),V(I),ETA(I),H	PRINT.22
2465	17.	601	FORMAT(17, 14, 1X, 4F14.4)	PRINT.23
2466	18.	10	CONTINUE	PRINT.24
2467	C			PRINT.25
2468	19.	REWIND	LU3	PRINT.26
2469	C			PRINT.27
2470	20.	WRITE(LU3)	(U(I), I=1,NP)	PRINT.28
2471	21.	WRITE(LU3)	(V(I), I=1,NP)	PRINT.29
2472	22.	WRITE(LU3)	(ETA(I), I=1,NP)	PRINT.30
2473	C			PRINT.31
2474	23.	REWIND	LU3	PRINT.32
2475	C			PRINT.33
2476	24.	RETURN		PRINT.34
2477	25.	END		PRINT.35

2478	1.	SUBROUTINE CONVPR	CONVPR.2
2479	C		CONVPR.3
2480	2.	IMPLICIT REAL*8 (A-H, O-Z)	CONVPR.4
2481	C		CONVPR.5
2482	3.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
2483	*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
2484	4.	DIMENSION ITIT(20), EPSLN(3), Q(10), ERROR(3), OUM(10)	CONVPR.7
2485	C		CONVPR.8
2486	5.	REWIND LU10	CONVPR.9
2487	6.	READ (LU10) ITIT	CONVPR.10
2488	7.	READ (LU10) (EPSLN(1) , I=1,3), NPOINT	CONVPR.11
2489	C		CONVPR.12
2490	8.	CALL PAGE	CONVPR.13
2491	9.	CALL LINE(6)	CONVPR.14
2492	10.	WRITE(LOUT,601) EPSLN	CONVPR.15
2493	11.	CALL PEJECT (7, 1, 5)	CONVPR.16
2494	C		CONVPR.17
2495	12.	DO 10 I = 1, 10	CONVPR.18
2496	13.	Q(I) = 0.0	CONVPR.19
2497	14.	10 CONTINUE	CONVPR.20
2498	C		CONVPR.21
2499	15.	15 CONTINUE	CONVPR.22
2500	16.	READ (LU10, ENQ=999) ICOUNT, DT , TIME	CONVPR.23
2501	17.	IF (ICOUNT .GT. 1) GO TO 20	CONVPR.24
2502	18.	IF (NPOINT .LE. 0) GO TO 20	CONVPR.25
2503	19.	READ (LU10) (Q(I) , I=1,NPOINT)	CONVPR.26
2504	20.	IQ = Q(1) + Q(2)	CONVPR.27
2505	21.	ICNT0 = 0	CONVPR.28
2506	22.	TIME0 = 0.0	CONVPR.29
2507	23.	CALL LINE(1)	CONVPR.30
2508	24.	WRITE(LOUT,602) ICNT0, TIME0, Q(1), Q(2), IQ	CONVPR.31
2509	C		CONVPR.32
2510	25.	20 CONTINUE	CONVPR.33
2511	26.	IF (NPOINT .LE. 0) GO TO 30	CONVPR.34
2512	27.	READ (LU10) (Q(I) , I=1,NPOINT)	CONVPR.35
2513	28.	30 CONTINUE	CONVPR.36
2514	29.	READ (LU10) (ERROR(I) , I=1,3)	CONVPR.37
2515	C		CONVPR.38
2516	30.	IQ = Q(1) + Q(2)	CONVPR.39
2517	31.	CALL PEJECT (7, 0, 1)	CONVPR.40
2518	32.	WRITE(LOUT,603) ICOUNT, DT, TIME, Q(1), Q(2), IQ, ERROR	CONVPR.41
2519	33.	GO TO 15	CONVPR.42
2520	C		CONVPR.43
2521	34.	999 CONTINUE	CONVPR.44
2522	35.	REWIND LU10	CONVPR.45
2523	C		CONVPR.46
2524	36.	601 FORMAT(1H0, 10X, *** CONVERGENT PROCESS *** /	CONVPR.47
2525	*	1H0, 40X, *EPSLN1 =+, E12.3 /	CONVPR.48
2526	*	1H , 40X, *EPSLN2 =+, E12.3 /	CONVPR.49
2527	*	1H , 40X, *EPSLN3 =+, E12.3)	CONVPR.50
2528	37.	602 FORMAT(1H , 118, 10X, F10.2, 2X, 3F12.2)	CONVPR.51
2529	38.	603 FORMAT(1H , 118, 2F10.2, 2X, 3F12.2, 3X, 3E12.3)	CONVPR.52
2530	39.	RETURN	CONVPR.53
2531	40.	END	CONVPR.54

2532	1.	SUBROUTINE PFILE2(X,Y,U,V,ETA,IELE,KELE,UX,VX,NP,NE,NPMAX,NEMAX)	PFILE2.2
2533	C		PFILE2.3
2534	2.	IMPLICIT REAL*8 (A-H,O-Z)	PFILE2.4
2535	C		PFILE2.5
2536	3.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
2537	*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
2538	4.	COMMON / TITLE / ITIT(20)	/TITLE/.2
2539	5.	COMMON / CONST / ZERO ,ONE ,TWO	/CONST/.2
2540	C		PFILE2.9
2541	6.	DIMENSION X(NPMAX),Y(NPMAX),U(NPMAX),V(NPMAX),ETA(NPMAX),	PFILE2.10
2542	*	IELE(4,NEMAX),KELE(5,NEMAX),UX(NEMAX),VX(NEMAX)	PFILE2.11
2543	C		PFILE2.12
2544	7.	REWIND LU2	PFILE2.13
2545	C		PFILE2.14
2546	8.	CALL HSET2(ZERO,UX,NE)	PFILE2.15
2547	9.	CALL HSET2(ZERO,VX,NE)	PFILE2.16
2548	C		PFILE2.17

2549	10.	DO 10 I=1,NE	PFILE2.18
2550	11.	K = KELE(5,I)	PFILE2.19
2551	C		PFILE2.20
2552	12.	DO 20 J=1,K	PFILE2.21
2553	13.	IP = IELE(J,I)	PFILE2.22
2554	14.	UX(I) = UX(I) + U(IP) / FLOAT(K)	PFILE2.23
2555	15.	VX(I) = VX(I) + V(IP) / FLOAT(K)	PFILE2.24
2556	16.	20 CONTINUE	PFILE2.25
2557	17.	10 CONTINUE	PFILE2.26
2558	C		PFILE2.27
2559	18.	WRITE(LU2) ITIT	PFILE2.28
2560	19.	WRITE(LU2) NP	PFILE2.29
2561	20.	WRITE(LU2) (X(I),Y(I),ETA(I),I=1,NP)	PFILE2.30
2562	21.	WRITE(LU2) NE	PFILE2.31
2563	22.	WRITE(LU2) ((IELE(J,I),J=1,4),UX(I),VX(I),I=1,NE)	PFILE2.32
2564	C		PFILE2.33
2565	23.	REWIND LU2	PFILE2.34
2566	C		PFILE2.35
2567	24.	RETURN	PFILE2.36
2568	25.	END	PFILE2.37

2569	1.	SUBROUTINE PAGE	PAGE.2
2570	C		PAGE.3
2571	2.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
2572	*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
2573	3.	COMMON / TITLE / ITIT(20)	/TITLE/.2
2574	4.	COMMON / PAGEC / IPAGE ,ILINE	/PAGEC/.2
2575	C		PAGE.7
2576	5.	IPAGE = IPAGE + 1	PAGE.8
2577	6.	ILINE = 5	PAGE.9
2578	7.	WRITE(LOUT,600) ITIT,IPAGE	PAGE.10
2579	8.	600 FORMAT(1H1, I21, 20A4, I121, #PAGE-#, I3 /)	PAGE.11
2580	C		PAGE.12
2581	9.	RETURN	PAGE.13
2582	10.	END	PAGE.14

2583	1.	SUBROUTINE LINE(ICNT)	LINE.2
2584	C		LINE.3
2585	2.	COMMON / PAGEC / IPAGE ,ILINE	/PAGEC/.2
2586	C		LINE.5
2587	3.	ILINE = ILINE + ICNT	LINE.6
2588	4.	IF (ILINE .GT. 63) CALL PAGE	LINE.7
2589	5.	IF (ILINE .EQ. 5) ILINE = ILINE + ICNT	LINE.8
2590	C		LINE.9
2591	6.	RETURN	LINE.10
2592	7.	END	LINE.11

2593	1.	SUBROUTINE CHKP1(IE,NO,EK)	CHKP1.2
2594	C		CHKP1.3
2595	2.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
2596	*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
2597	C		CHKP1.5
2598	3.	REAL*8 EK	CHKP1.6
2599	4.	DIMENSION EK(4,4)	CHKP1.7
2600	C		CHKP1.8
2601	5.	CALL LINE(6)	CHKP1.9
2602	6.	WRITE(LOUT,601) IE,NO,(I,I=1,4)	CHKP1.10
2603	7.	601 FORMAT(+0*CHECK* ELEH NO.+, I4, + (+, I1, +)+,	CHKP1.11
2604	*	I35, I1, J115)	CHKP1.12
2605	C		CHKP1.13
2606	8.	DO 10 I=1,4	CHKP1.14
2607	9.	WRITE(LOUT,602) I,(EK(I,J),J=1,4)	CHKP1.15
2608	10.	602 FORMAT(I26, I1, 4E15.5)	CHKP1.16
2609	11.	10 CONTINUE	CHKP1.17
2610	C		CHKP1.18
2611	12.	RETURN	CHKP1.19
2612	13.	END	CHKP1.20

2613	1.	SUBROUTINE CHKP2(IE,EK)	CHKP2.2
2614	C		CHKP2.3
2615	2.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
2616		* LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
2617	C		CHKP2.5
2618	3.	REAL*8 EK	CHKP2.6
2619	4.	DIMENSION EK(12,12)	CHKP2.7
2620	C		CHKP2.8
2621	5.	CALL LINE(4)	CHKP2.9
2622	6.	WRITE(LOUT,601) IE,(I,I=1,12)	CHKP2.10
2623	7.	601 FORMAT(+0*CHECK* ELEM NO.+, I4, I31, +EK+,	CHKP2.11
2624		* //I8, 12I10)	CHKP2.12
2625	C		CHKP2.13
2626	8.	DO 10 I=1,12	CHKP2.14
2627	9.	CALL LINE(1)	CHKP2.15
2628	10.	WRITE(LOUT,602) I,(EK(I,J),J=1,12)	CHKP2.16
2629	11.	602 FORMAT(I7, 3X, 12E10.3)	CHKP2.17
2630	12.	10 CONTINUE	CHKP2.18
2631	C		CHKP2.19
2632	13.	RETURN	CHKP2.20
2633	14.	END	CHKP2.21

2634	1.	SUBROUTINE CHKP3(IE,P)	CHKP3.2
2635	C		CHKP3.3
2636	2.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
2637		* LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
2638	C		CHKP3.5
2639	3.	REAL*8 P	CHKP3.6
2640	4.	DIMENSION P(12)	CHKP3.7
2641	C		CHKP3.8
2642	5.	CALL LINE(6)	CHKP3.9
2643	6.	WRITE(LOUT,601) IE	CHKP3.10
2644	7.	601 FORMAT(+0*CHECK* ELEM NO.+, I4, I31, +P+ /)	CHKP3.11
2645	8.	WRITE(LOUT,602) (I,P(I),I=1,12)	CHKP3.12
2646	9.	602 FORMAT((I11,4(I7,E13.5)))	CHKP3.13
2647	C		CHKP3.14
2648	10.	RETURN	CHKP3.15
2649	11.	END	CHKP3.16

2650	1.	SUBROUTINE CHKP4 (IE, N, NOI, NO, A, B)	CHKP4.2
2651	C		CHKP4.3
2652	2.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
2653		* LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
2654	C		CHKP4.5
2655	3.	REAL*8 A	CHKP4.6
2656	4.	DIMENSION NOI(12), NO(N), A(12,N), B(12)	CHKP4.7
2657	C		CHKP4.8
2658	5.	CALL LINE(6)	CHKP4.9
2659	6.	WRITE (LOUT,601) IE, (I,I=1,12), NOI	CHKP4.10
2660	7.	601 FORMAT(+0*CHECK* ELEM NO.+, I4, I31, +PPC BOUNDARY MATRIX+,	CHKP4.11
2661		* //I8, 12I10 /I8, 12I10)	CHKP4.12
2662	C		CHKP4.13
2663	8.	DO 10 I=1,N	CHKP4.14
2664	9.	CALL LINE(1)	CHKP4.15
2665	10.	WRITE(LOUT,602) NO(I),(A(J,I),J=1,12)	CHKP4.16
2666	11.	602 FORMAT(I7, 3X, 12F10.3)	CHKP4.17
2667	12.	10 CONTINUE	CHKP4.18
2668	C		CHKP4.19
2669	13.	CALL LINE(2)	CHKP4.20
2670	14.	WRITE (LOUT,603) B	CHKP4.21
2671	15.	603 FORMAT(+0 B =+, 3X, 12F10.3)	CHKP4.22
2672	C		CHKP4.23
2673	16.	RETURN	CHKP4.24
2674	17.	END	CHKP4.25

2675	1.	SUBROUTINE CHKP5(TSH,M,N)	CHKP5.2
2676	C		CHKP5.3
2677	2.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
2678	*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
2679	C		CHKP5.5
2680	3.	REAL*8 TSH	CHKP5.6
2681	4.	DIMENSION TSH(M,N)	CHKP5.7
2682	C		CHKP5.8
2683	5.	NN = (N-1)/10 + 1	CHKP5.9
2684	C		CHKP5.10
2685	6.	CALL LINE(2)	CHKP5.11
2686	7.	WRITE(LOUT,601)	CHKP5.12
2687	8.	601 FORMAT('0*CHECK*+', T31, 'BK')	CHKP5.13
2688	C		CHKP5.14
2689	9.	DO 10 I=1,NN	CHKP5.15
2690	10.	IS = 10*(I-1) + 1	CHKP5.16
2691	11.	IE = MIN0(N,IS+9)	CHKP5.17
2692	C		CHKP5.18
2693	12.	CALL LINE(2)	CHKP5.19
2694	13.	WRITE(LOUT,602) (J,J=IS,IE)	CHKP5.20
2695	14.	602 FORMAT(/7X, 10I12)	CHKP5.21
2696	C		CHKP5.22
2697	15.	DO 20 J=1,M	CHKP5.23
2698	16.	CALL LINE(1)	CHKP5.24
2699	17.	WRITE(LOUT,603) J,(TSH(J,K),K=IS,IE)	CHKP5.25
2700	18.	603 FORMAT(17, T11, 10E12.5)	CHKP5.26
2701	19.	20 CONTINUE	CHKP5.27
2702	20.	10 CONTINUE	CHKP5.28
2703	C		CHKP5.29
2704	21.	RETURN	CHKP5.30
2705	22.	END	CHKP5.31

2706	1.	SUBROUTINE CHKP6(X,N,NAME,IC)	CHKP6.2
2707	C		CHKP6.3
2708	2.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
2709	*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
2710	C		CHKP6.5
2711	3.	REAL*8 X	CHKP6.6
2712	4.	DIMENSION X(N)	CHKP6.7
2713	C		CHKP6.8
2714	5.	NN = (N-1)/10 + 1	CHKP6.9
2715	C		CHKP6.10
2716	6.	CALL LINE(2)	CHKP6.11
2717	7.	WRITE(LOUT,601) NAME,IC	CHKP6.12
2718	8.	601 FORMAT('0*CHECK*+', T31, A4, T41, 'COUNT*+', I3)	CHKP6.13
2719	9.	CALL LINE(2)	CHKP6.14
2720	10.	WRITE(LOUT,602) (J,J=1,10)	CHKP6.15
2721	11.	602 FORMAT(/7X, 10I12)	CHKP6.16
2722	C		CHKP6.17
2723	12.	DO 10 I=1,NN	CHKP6.18
2724	13.	IS = 10*(I-1) + 1	CHKP6.19
2725	14.	IE = MIN0(N,IS+9)	CHKP6.20
2726	C		CHKP6.21
2727	15.	J = I - 1	CHKP6.22
2728	16.	CALL LINE(1)	CHKP6.23
2729	17.	WRITE(LOUT,603) J,(X(K),K=IS,IE)	CHKP6.24
2730	18.	603 FORMAT(17, T11, 10E12.5)	CHKP6.25
2731	19.	10 CONTINUE	CHKP6.26
2732	C		CHKP6.27
2733	20.	RETURN	CHKP6.28
2734	21.	END	CHKP6.29

2735	1.	SUBROUTINE CHKP7(VC,NAME,IE)	CHKP7.2
2736	C		CHKP7.3
2737	2.	COMMON / FILE / LIN ,LOUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
2738	*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
2739	C		CHKP7.5
2740	3.	REAL*8 VC	CHKP7.6
2741	4.	DIMENSION VC(4)	CHKP7.7
2742	C		CHKP7.8
2743	5.	CALL LINE(2)	CHKP7.9
2744	6.	WRITE(LOUT,601) NAME,IE,VC	CHKP7.10
2745	7.	601 FORMAT(+0*CHECK+ P-VECTOR (+, A4, +) ELEM NO.+, 14,	CHKP7.11
2746	*	151, 4E13.5)	CHKP7.12
2747	C		CHKP7.13
2748	8.	RETURN	CHKP7.14
2749	9.	END	CHKP7.15

2750	1.	SUBROUTINE CHKP8(A,NO,IM,M,N,L)	CHKP8.2
2751	C		CHKP8.3
2752	2.	COMMON / FILE / LIN ,LCUT ,LU1 ,LU2 ,LU3 ,LU4 ,	/FILE/.2
2753	*	LU10 ,LU11 ,LU12 ,LU13 ,LU14 ,LU15	/FILE/.3
2754	C		CHKP8.5
2755	3.	REAL*8 A	CHKP8.6
2756	4.	DIMENSION A(L,N),NO(L)	CHKP8.7
2757	C		CHKP8.8
2758	5.	NN = (N-1) / 12 + 1	CHKP8.9
2759	C		CHKP8.10
2760	6.	CALL LINE(2)	CHKP8.11
2761	7.	WRITE(LOUT,601) IM	CHKP8.12
2762	8.	601 FORMAT(+0*CHECK+, 131, +ELEMENT MATRIX ELEM NO.=+,	CHKP8.13
2763	*	15)	CHKP8.14
2764	C		CHKP8.15
2765	9.	DO 10 I=1,NN	CHKP8.16
2766	10.	IS = 12*(I-1) + 1	CHKP8.17
2767	11.	IE = MIN0(N,IS+11)	CHKP8.18
2768	C		CHKP8.19
2769	12.	CALL LINE(2)	CHKP8.20
2770	13.	WRITE(LOUT,602) (NO(J),J=IS,IE)	CHKP8.21
2771	14.	602 FORMAT(/T8, 12I10)	CHKP8.22
2772	C		CHKP8.23
2773	15.	DO 20 J=1,M	CHKP8.24
2774	16.	CALL LINE(1)	CHKP8.25
2775	17.	WRITE(LOUT,603) NO(J),(A(J,K),K=IS,IE)	CHKP8.26
2776	18.	603 FORMAT(17, 3X, 12E10.3)	CHKP8.27
2777	19.	20 CONTINUE	CHKP8.28
2778	20.	10 CONTINUE	CHKP8.29
2779	C		CHKP8.30
2780	21.	RETURN	CHKP8.31
2781	22.	END	CHKP8.32