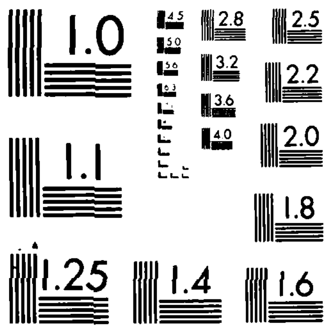
AD-A159 245 A FINITE ELEMENT SOLUTION OF THE TRANSPORT EQUATION(U) 1/3
AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL
OF ENGINEERING F A TARANTINO MAR 85 AFIT/GNE/PH/85M-19

UNCLASSIFIED F/G 12/1 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

REPRODUCED AT GOVERNMENT EXPENSE

AD-A159 245

A FINITE ELEMENT SOLUTION OF THE TRANSPORT EQUATION

THESIS

Frederick A, Tarantino
Captain IN, USA

AFIT/GNE/PH/85M-19

DTIC FILE COPY

This document has been approved for public release and sale; its distribution is unlimited.

DTIC
ELECTE
SEP 1 8 1985
S D
A

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

85 09 17 032

AFIT/GNE/PH/85M-19

A FINITE ELEMENT SOLUTION OF THE TRANSPORT EQUATION

THESIS

Presented to the Faculty of the School of Engineering of

the Air Force Institute of Technology

Air University

In Partial Fulfillment of the Requirement for

the Degree of Master of Science in Nuclear Science

Frederick A. Tarantino B.S.

Captain IN, USA

March 1985

Approved for public release; distribution unlimited

A FINITE ELEMENT SOLUTION OF THE
TRANSPORT EQUATION

THESIS

Frederick A. Tarantino
Captain IN, USA

AFIT/GNE/PH/85M-19

Approved for public release; distribution unlimited

DTIC
ELECTE
SEP 1 8 1985
S D
A

Abstract

Using a self adjoint form of the transport equation expressed as a variational integral, finite element equations for the one dimensional, one speed, homogeneous, time independent transport equation in slab geometry were derived and encoded in Fortran 77. The accuracy of C^0 and C^1 continuous fits was compared against an analytical solution for the case of noscatter. It was found that the C^0 fits require an excessive amount of mesh refinement. The C^1 fit is very accurate, and does not appear to be computationally excessive.

The finite element results were then compared, for the case of isotropic scatter, to a legendre polynomial solution, and the results of a recently developed code known as Ln. The methods accuracy was sufficiently verified with inexact scattering term evaluation. A technique of exact scattering integral evaluation is proposed that should reduce the amount of refinement required for convergence, and improve computational efficiency.

DTIC COPY INSPECTED

Table of Contents

Preface

The purpose of this study was to continue the work of a previous graduate student, (A.D. Goff GNE 84M) and demonstrate that a finite element solution of the transport equation would work. Using a self adjoint form of the transport equation expressed as a variational integral, finite element equations with C^0 and C^1 continuity were derived, encoded and compared to a spherical harmonic solution over a test case domain.

I have been extremely pleased with the graduate education provided by the AFIT GNE faculty. Dr.'s Charles Bridgman , George John and Bernard Kaplan all deserve my thanks. I would particularly like to thank Dr. Donn Shankland for his guidance and instruction throughout this study. He provided a challenging and exciting thesis topic, from which I have learned greatly. Finally I would like to thank my wife Jazmine, whose love and understanding never falter.

Frederick A. Tarantino

List of Figures

List of Tables

Notation

Coordinates

$x \; y \; y'$ - Cartesian coordinates

$\ell_1 \; \ell_2 \; \ell_3$ - Triangular coordinates

$L_1 \; L_2 \; L_3 \; L_4$ - Tetrahedral coordinates

Operators

d - Derivative

∂ - Partial derivative

∇ - Gradient

δ - First variation

L - SATE operator

$+$ - Adjoint

$\sim$ - Transpose

$\mathcal{L}$ - Transport operator

Σ - Summation

$!$ - Factorial

Variables

ϕ - Angular flux

Σ_t - Total macro cross section

Σ_s - Isotropic cross section for scatter

μ - Direction cosine in 1-D, before collision

μ' - Direction cosine in 1-D, after collision

$'$ - Primed refers to after-collision properties

- Unprimed refers to before-collision coordinates

I - Variational functional

A - Area

V - Volume

Matrices

$\underline{\underline{MG}}$ - Global matrix

$\underline{\underline{GT}}$ - Matrix of interpolating function constants

$\underline{\underline{I}}$ - Identity matrix

$\underline{\underline{MA}}$ - Absorbing matrix

$\underline{\underline{MS}}$ - Streaming matrix

$\underline{\underline{MB}}$ - Boundary matrix

$\underline{\underline{ML}}$ - Local matrix

$\underline{\underline{NML}}$ - Non local matrix

$\underline{\underline{O}}$ - Zero matrix

$\underline{\underline{LI}}$ - Local integral

$\underline{\underline{NLI}}$ - Non local integral

Vectors

$\underline{h}$ - Basis functions

$\underline{m}$ - Vector of natural coordinate polynomials, which together constitute a complete basis, of first, second or third order depending upon the fit being used.

$\underline{\varphi}$ - Vector of finite element interpolating nodes

$\underline{F}$ - Vector of product $\Phi\Phi'$

$\underline{G}$ - Vector of product $\Phi_x\Phi'$

$(1,0,0)$, $\Phi = \varphi_2$ at node 2, and $\Phi = \varphi_3$ at node 3. This fit has C^0 continuity since flux is continuous along element interfaces, but its first partial derivatives are not.

The quadratic fit for this element requires six degrees of freedom.

$$\Phi = \sum_{i=1}^{6} h_i \varphi_i \tag{2-12}$$

Where the h_i are basis functions and the φ_i are now the value of the flux at corner nodes and the first derivative with respect to x. Such that

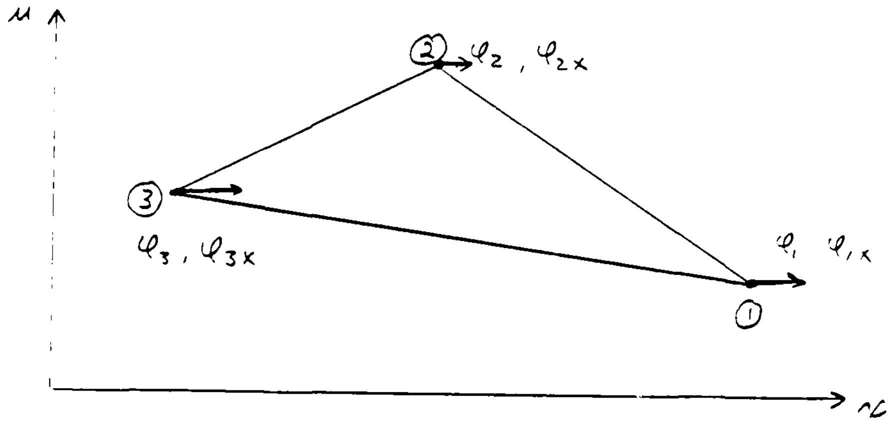


Figure 2-2
Quadratic Fit Using Derivatives at Corner Nodes

$$\Phi = \underline{\tilde{h}} \; \underline{\varphi} \tag{2-13}$$

where

$$\underline{\hat{\varphi}} = \begin{bmatrix} \varphi_1 & \varphi_2 & \varphi_3 & \varphi_{1x} & \varphi_{2x} & \varphi_{3x} \end{bmatrix} \tag{2-14}$$

and

$$\frac{\partial \Phi(1,0,0)}{\partial x} = \varphi_{1x} \qquad \frac{\partial \Phi(0,1,0)}{\partial x} = \varphi_{2x} \qquad \frac{\partial \Phi}{\partial x}(0,0,1) = \varphi_{3x}$$

node 2 is at (0,1,0) and node 3 at (0,0,1) .

Over the area of a triangle the integral of natural coordinate powers is given (3:145) by

$$\int dA\, \ell_1^{p} \ell_2^{q} \ell_3^{r} = 2A \frac{p!\,q!\,r!}{(p+q+r+2)!} \tag{2-7}$$

where A is the area of the triangle and is equal to

$$2A = \begin{vmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ \mu_1 & \mu_2 & \mu_3 \end{vmatrix} \tag{2-8}$$

The simplest interpolant for a triangle is the linear fit. It express the field variable, ϕ ,(in this case particle angular flux) across the triangle as a linear combination of the flux at the corner nodes such that

$$\phi = \phi(x,\mu) = \phi(\ell_1\, \ell_2\, \ell_3) = \sum_{i=1}^{3} \ell_i\, \varphi_i \tag{2-9}$$

The partial derivatives of the flux are

$$\frac{\partial \phi}{\partial x} = \sum_{i=1}^{3} \frac{\partial \ell_i}{\partial x} \varphi_i = \sum_{i=1}^{3} g_i\, \varphi_i \tag{2-10}$$

$$\frac{\partial \phi}{\partial \mu} = \sum_{i=1}^{3} \frac{\partial \ell_i}{\partial \mu} \varphi_i = \sum_{i=1}^{3} f_i\, \varphi_i \tag{2-11}$$

where g_i and f_i are the partial derivatives of the three natural coordinates with respect to x and μ respectively. Within an element they are constant, but are different for each distinct triangle geometry. Note that $\frac{\partial \varphi_i}{\partial x} = \frac{\partial \varphi_i}{\partial \mu} = 0$

It is clear from this expression that the equation is satisfied at corner nodes, that is that $\Phi = \varphi_1$, at node 1 element (3:140). In two dimensions this system is often referred to as area coordinates, since it can easily be shown that the natural coordinates represent ratios of area.

Over a triangle one expresses the (x,u) coordinates in terms of three variables ℓ_1, ℓ_2 and ℓ_3 such that the sum of the natural coordinates is always one.

$$\ell_1 + \ell_2 + \ell_3 = 1 \tag{2-1}$$

The linear relation below exists between the cartesian and natural coordinates;

$$\ell_1 x_1 + \ell_2 x_2 + \ell_3 x_3 = x \tag{2-2}$$

$$\ell_1 u_1 + \ell_2 u_2 + \ell_3 u_3 = u \tag{2-3}$$

Written in matrix form the above relations become

$$\begin{bmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ u_1 & u_2 & u_3 \end{bmatrix} \begin{bmatrix} \ell_1 \\ \ell_2 \\ \ell_3 \end{bmatrix} = \begin{bmatrix} 1 \\ x \\ u \end{bmatrix} \tag{2-4}$$

It is easily shown from this expression that

$$\frac{\partial \ell_1}{\partial x} = \frac{u_2 - u_3}{2\,(\text{Area of triangle})} \tag{2-5}$$

and

$$\frac{\partial \ell_1}{\partial u} = \frac{x_3 - x_2}{2\,(\text{Area of triangle})} \tag{2-6}$$

and that the indices permutate cyclically. Note that the coordinates of node 1 in figure 2-1 are (ℓ_1, ℓ_2, ℓ_3) = (1,0,0,) compatibility requirement of elements. Without it, "gaps" may arise between elements from discontinuous derivatives, and unsolicited contributions to field variables can arise. Completeness is the term associated with the second requirement. It insures that the variational integral is well defined.

Standard finite element nomenclature is to express element continuity as a function of the highest order derivative held continuous on boundaries. C^0 continuity implies field variable values are continuous on element edges, C^1 continuity has variable and first derivative inter-element continuity, an so on.

B. The Triangle and Two Dimensional Interpolating Functions

The two dimensional element chosen for this study was the triangle. It is a simple element to refine, can be maneuvered easily to snugly fit irregular boundaries, and can be expediently described in terms of its natural coordinates

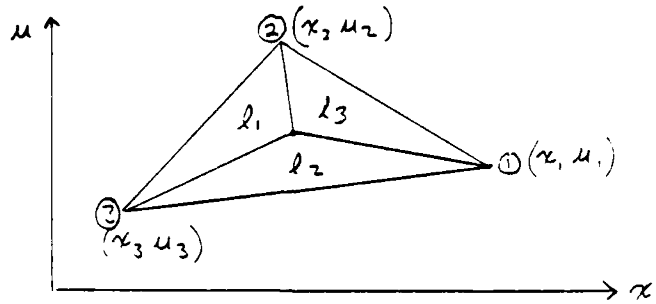


Figure 2-1
Cartesian and Triangular Coordinates

A natural coordinate system is a local coordinate system that relies upon the element geometry for its definition, and whose coordinates range in value from zero to one within an

2. Elements and Interpolating Functions

The study of interpolating functions, and the elements composing finite element meshes is an important one. The wrong choices can cause excessive computations, or worse prevent convergence from occuring. The elements and interpolating functions presented here are by no means inclusive. Finite element literature on the topic is extensive.

In this section general requirements for monotonic convergence of the finite element method will be presented Then, the two and three dimensional elements chosen for this study are described. Finally the interpolating functions used for each element are given, and their derivations are explained.

A. Compatibility and Completeness

Convergence of the finite element method is guaranteed if the elements composing the mesh, and the selected interpolation function satisfy two requirements (3:79). Namely, 1) along element boundaries the field variable and any of its partial derivatives up to one order less than the highest order derivative appearing in the variational functional must be continuous and 2) the field variable, and its partial derivatives up to the highest order appearing in the functional should be represented in the interpolation function as the limit of element size approaches zero.

The first of these requirements is the so called variable value at that node must be the same from each element possesing that node.

5) Solve the System Equations. The resulting coefficient matrix (referred to as the global matrix) is symmetric, banded and positive definite. System of equations with coefficient matrices of this type are best solved by Cholesky decomposition (7:13) and it is the algorithim used in this study. Solution of the system equations yields nodal values of the field variable, which together with the interpolating function defines piecewise approximations across the domain under scrutiny.

6) Make additional Computations if desired. With respect to the transport equation this step is not required, except in the case where penalties are desired for automatic mesh refinement.

D) Summary

Operating on the transport operator with the adjoint transport operator produces a self adjoint transport equation. This equation can be expressed variationally as a quadratic functional, that when minimized is equivilent to solving the SATE. The finite element method is best suited for solving this type of problem. It is a numerical technique that approximates field variable values with separate analytical expressions, of the same order, across a mesh of small interconnected elements. The resulting set of linear equations is positive definite, and can be solved quickly by direct means.

in step 3 below. Additionally, triangles are easily generated and refined, and fit irregular boundaries snugly. In general one should start with a sparse mesh composed of few elements. This allows a solution to be calculated, and regions of high penalty identified for mesh refinement. Constructing large meshes by hand is a time consuming and error prone process. References for automatic mesh generation are listed in Heubner (3:511).

2) Select interpolation functions. Chapter two is dedicated to this topic. Interpolating nodes must be assigned to each element, and interpolants chosen. The form of these functions depend upon the number of geometric nodes an element possesses, the number of unknowns at each node, and certain continuity requirements to be discussed in section 2-a.

3) Find the element properties. The interpolation functions are substituted for the field variable in the functional, and the integral is evaluated. When the resultant expressions are minimized with respect to nodal values, the remaining matrix equation describes element properties in terms of nodal variables. Element properties are expressed in terms of the coefficient matrices of this equation, referred to in this report as the local and nonlocal matrices.

4) Assemble element properties to obtain system equations. When local and nonlocal matrices are computed for each element, they are assembled globally to provide a set of simultaneous linear equations with nodal values as the unknowns. The foundation for this procedure lies on the fact that if a node is shared by more than one element the field

than a finite difference rectangular grid. An additional advantage of the method is that mesh refinement can occur easily, and that there is a built-in indicator to dictate where mesh refinement should occur. Local mesh refinement, cumbersome in a finite difference grid, consists only of subdividing a finite element into smaller ones. Elements over which this is necessary are discovered by evaluating the so-called penalty function. Since the variational integral is minimized, its value over a particular element is referred to as that element's penalty. Elements where large penalties occur are those where the interpolation function fit is poorest, and mesh refinement is required. These elements can be subdivided until penalties are equal across the mesh and further refinement fails to produce significant total penalty reduction.

Solution steps

These six steps are given by Heubner (3:7) as an orderly method for obtaining a finite element solution. They are described in general terms below, and are elaborated upon specifically with respect to the SATE in later sections.

1) Discretize the Continuum. The first step is to divide the domain under consideration into a set of interconnected elements. A multiformity of elements may be used. The triangle is very well suited for two dimensional problems, and it is the element used in this study. The ability to express interpolating functions in terms of triangular natural coordinates simplifies the evaluation of integrals appearing

Across each of these elements an assumed solution is prescribed, called an interpolating, or approximating function. This solution is written as a function of field variable values, and sometimes the variable's spatial derivatives, at element nodes. Solution requires choosing these nodal variables so as to minimize the variational statement of the problem and satisfy boundary conditions. If the operator acting upon the field variable is self-adjoint, then equations describing the variable values at element nodes can be assembled globally, (over the entire material) and an approximate solution to the partial differential equation can be calculated by solving the resulting set of simultaneous linear equations.

Consider a comparison of the finite element method with the well known finite difference method. The finite difference approximation is that a derivative can be approximated by a difference operation over a very small interval. The resulting solution is a set of grid points at which the field variable values are defined. Finite elements assumes an analytical expression for variable values over a small element. This approach results in a piecewise approximation, with field variable values given by separate analytic expressions across an array of small, interconnected elements, as well as at interpolation nodes.

Because the finite element mesh is composed of elements, they can be put together in a variety of ways. This makes the method well suited for problems with complex geometries. Small elements can be made to fit an irregular boundary much easier

$$I = \frac{1}{2} \int_D \left\{ \mu \frac{\partial \phi}{\partial x} + \Sigma_t \phi - \frac{\Sigma_s}{2} \int_{-1}^{1} d\mu' \phi' \right\}^2 dD$$

$$I = \frac{1}{2} \int_D \left\{ \mu^2 \left(\frac{\partial \phi}{\partial x}\right)^2 + \Sigma_t^2 \phi^2 + 2\mu \Sigma_t \frac{\partial \phi}{\partial x} \phi \right.$$

$$+ \left(\frac{\Sigma_s}{2} \int_{-1}^{1} d\mu' \phi' \right) \left(\frac{\Sigma_s}{2} \int_{-1}^{1} d\mu'' \phi'' \right) - \mu \Sigma_s \frac{\partial \phi}{\partial x} \int_{-1}^{1} d\mu' \phi'$$

$$\left. - \Sigma_t \Sigma_s \phi \int_{-1}^{1} d\mu' \phi' \right\} dD \qquad (1\text{-}16)$$

To be sure this is correct, the expressions given by equations (1-9) and (1-10) should be satisfied. Specifically

$$\frac{\partial (\mathcal{L}\phi)^2}{\partial \phi} - \frac{\partial}{\partial x} \frac{\partial (\mathcal{L}\phi)^2}{\partial \phi_x} = 0 \qquad (1\text{-}17)$$

should reduce to the self adjoint transport equation, and

$$\frac{\partial (\mathcal{L}\phi)^2}{\partial \phi_x} = 0 \qquad (1\text{-}18)$$

is the condition required along the boundary. Straightforward substitution verifies that (1-17) is the SATE, and that the natural boundary condition is the transport equation itself, certainly an acceptable requirement.

Up to this point the transport equation has been recast into a variational form, and it has been shown that minimizing this functional is equivalent to solving the SATE. In the next section is presented background on a numerical technique which has achieved the most success in solving this type of problem.

C) The Finite Element Method

The finite element method is a numerical technique used to solve partial differential equations. The region under scrutiny is divided up into a finite number of elements.

$$I = \frac{1}{2}\int_D (\mathcal{L}\phi - s)(\mathcal{L}\phi - s)\, dD \tag{1-11}$$

yields

$$I = \frac{1}{2}\int_D (\mathcal{L}\phi\mathcal{L}\phi - 2s\mathcal{L}\phi + s^2)\, dD \tag{1-12}$$

Using the definition of adjointness this functional can be expressed as

$$I = \frac{1}{2}\int_D (\phi\mathcal{L}^+\mathcal{L}\phi - 2\phi\mathcal{L}^+ s + s^2)\, dD \tag{1-13}$$

Setting the variation equal to zero, using the definition of adjointness and recalling that $\mathcal{L}^+\mathcal{L}$ is self adjoint gives

$$\delta I = \frac{1}{2}\int_D \{\delta\phi(\mathcal{L}^+\mathcal{L}\phi - 2\mathcal{L}^+ s) + \phi\mathcal{L}^+\mathcal{L}\delta\phi\}\, dD$$

$$\delta I = \int_D \delta\phi(\mathcal{L}^+\mathcal{L}\phi - \mathcal{L}^+ s)\, dD = 0$$

$$\delta I = \int_D \delta\phi\, \mathcal{L}^+(\mathcal{L}\phi - s)\, dD = 0 \tag{1-14}$$

Solution of which is identical to solving the SATE.

The One Speed, One Dimensional Functional

The one dimensional, one speed, time independent transport equation is given (1:76) by

$$\mu\frac{\partial\phi}{\partial x} + \Sigma_t\phi = \frac{\Sigma_s}{2}\int_{-1}^{1} d\mu'\, \phi(x,\mu') \tag{1-15}$$

in the case of isotropic scatter, with no sources, where

μ = cosine(angle particle is traveling)

$\phi = \phi(x,\mu)$, angular flux

Σ_s = scattering cross section

In this case $\mathcal{L} = \mu\frac{\partial}{\partial x} + \Sigma_t - \frac{\Sigma_s}{2}\int_{-1}^{1} d\mu'$

This is the form of the transport equation chosen to test the finite element solution of the quadratic transport functional. The expression requiring minimization now becomes where $\Phi = \phi(x,u)$.

The minimum of this functional is found in an analogous manner to finding the minimum of a function in ordinary calculus, by setting the variation to zero.

$$\delta I = \int_{u_1}^{u_2} \int_{x_1}^{x_2} \left[\frac{\partial F}{\partial \phi} \delta\phi + \frac{\partial F}{\partial \phi_x} \delta\phi_x \right] dx\, du = 0 \tag{1-7}$$

Integrating the second term by parts

$$\int w\, dv = vw - \int v\, dw$$

$$w = \frac{\partial F}{\partial \phi_x} \qquad v = \delta\phi$$

$$dw = \frac{\partial}{\partial x} \frac{\partial F}{\partial \phi_x} dx \qquad dv = \delta\phi_x\, dx$$

gives

$$\delta I = \int_{u_1}^{u_2} \int_{x_1}^{x_2} \left[\frac{\partial F}{\partial \phi} - \frac{\partial}{\partial x} \frac{\partial F}{\partial \phi_x} \right] \delta\phi\, dx\, du + \int_{u_1}^{u_2} \frac{\partial F}{\partial \phi_x} \delta\phi \Big|_{x_1}^{x_2} \tag{1-8}$$

Since $\delta\phi$ is an arbitrary admissible variable (H:551) δI can equal zero only if

$$\frac{\partial F}{\partial \phi} - \frac{\partial}{\partial x} \frac{\partial F}{\partial \phi_x} = 0 \tag{1-9}$$

and

$$\frac{\partial F}{\partial \phi_x} \Big|_{x_1}^{x_2} = 0 \tag{1-10}$$

The first term above is a simplified version of the Euler-Lagrange equation (3:551) and is the differential equation satisfied when I is minimized. The second expression is referred to as a natural boundary condition, since it specifies the solution form on the boundary, and since the functional can only be minimized when it is satisfied.

The Transport Functional

Expansion of the quadratic functional (2:15) repeated here for the purpose of document continuity.

The Self-Adjoint Transport Equation (SATE)

Adjointness is defined (5:10) by the property

$$\int_D \phi \mathcal{L} \psi \, dD = \int_D \psi \mathcal{L}^{+} \phi \, dD \qquad (1\text{-}4)$$

where $\mathcal{L}^{+}$ is the adjoint of the operator $\mathcal{L}$. If $\mathcal{L} = \mathcal{L}^{+}$ then $\mathcal{L}$ is said to be self-adjoint.

Consider the operator $L = \mathcal{L}^{+}\mathcal{L}$ where $\mathcal{L}$ is the transport operator and $\mathcal{L}^{+}$ is its adjoint. Since

$$\int_D \phi L \psi \, dD = \int_D \phi \mathcal{L}^{+}\mathcal{L} \psi \, dD = \int_D \mathcal{L} \psi \mathcal{L} \phi \, dD$$

$$= \int_D \mathcal{L} \phi \mathcal{L} \psi \, dD = \int_D \psi \mathcal{L}^{+}\mathcal{L} \phi \, dD = \int_D \psi L \phi \, dD$$

L is self adjoint. If $\mathcal{L}^{+}$ is allowed to operate on the transport equation, the resultant expression

$$\mathcal{L}^{+}(\mathcal{L}\phi - S) = 0 \qquad (1\text{-}5)$$

is self adjoint. Solutions of this equation must satisfy the transport equation (2:16).

$L\phi - \mathcal{L}^{+}S = 0$ is a self adjoint operator equation, the solution of which always satisfies the transport equation. This expression is referred to as the self adjoint transport equation (SATE).

B) Variational Minimization of a Functional

The next task described in Goff's thesis was to find a variational functional, minimization of which would be equivalent to solving the SATE. Before reproducing that effort, consider the task of minimizing a functional, $I(\phi)$.

$$I = \int_{\mu_1}^{\mu_2} \int_{x_1}^{x_2} F(\phi, \phi_x) \, dx \, d\mu \qquad (1\text{-}6)$$

1. Introduction

A) The Transport Equation

The Boltzman transport equation, written in its general integro-differential form is

$$\frac{1}{v}\frac{\partial \phi}{\partial t} + \hat{\Omega}\cdot\vec{\nabla}\phi + \Sigma_t \phi =$$

$$S + \int_0^\infty dE' \int d\hat{\Omega}' \Sigma_s(E' \to E, \hat{\Omega}' \to \hat{\Omega})\, \phi(r, E', \hat{\Omega}', t) \qquad (1\text{-}1)$$

where

v is particle velocity

ϕ is particle angular flux

$\hat{\Omega}$ is particle direction

Σ_t is the transport cross section $\Sigma_t(\vec{r}, \hat{\Omega}, E, t)$

S is particle sources $S(\vec{r}, \hat{\Omega}, E, t)$

E is particle energy

$\Sigma_s(E' \to E, \hat{\Omega}' \to \hat{\Omega})$ is the scattering cross section from energy E' to E and angle $\hat{\Omega}'$ to $\hat{\Omega}$. The transport equation can be written as

$$\mathcal{L}\phi - S = 0 \qquad (1\text{-}2)$$

where the operator $\mathcal{L}$ is clearly

$$\frac{1}{v}\frac{\partial}{\partial t} + \hat{\Omega}\cdot\vec{\nabla} + \Sigma_t - \int_0^\infty dE' \int d\hat{\Omega}' \Sigma_s(E' \to E, \hat{\Omega}' \to \Omega) \qquad (1\text{-}3)$$

In this formulation, the operator $\mathcal{L}$ is non-self-adjoint. Finite element solutions of this equation have been tried (1 : 479) but without a self-adjoint operator, variational extremum principles do not exist, and the finite element method's power is not achieved.

Reformulating the transport operator into a self adjoint form has been accomplished (2:15) and its derivation is

The basis functions are found by requiring that at (1,0,0) = , and = . Four more identities are found by similar relations at nodes 2 and 3 . If the basis functions are considered to be the product

$$\underline{\tilde{h}} = \underline{\hat{m}} \, \underline{\underline{G}}^T \tag{2-15}$$

where

$$\underline{\hat{m}} = \left[\ell_1^2 \;\; \ell_2^2 \;\; \ell_3^2 \;\; \ell_1\ell_2 \;\; \ell_2\ell_3 \;\; \ell_1\ell_3 \right] \tag{2-16}$$

is a matrix of polynomials, which together represent a complete quadratic basis, then

$$\phi = \underline{\hat{m}} \, \underline{\underline{G}}^T \, \underline{\varphi} \tag{2-17}$$

and

$$\frac{\partial \phi}{\partial x} = \phi_x = \frac{\partial \underline{\hat{m}}}{\partial x} \, \underline{\underline{G}}^T \, \underline{\varphi} \tag{2-18}$$

then since

$$\frac{\partial \underline{\hat{m}}}{\partial x} = \underline{\hat{m}}_x = \left[2\ell_1 g_1 \;\; 2\ell_2 g_2 \;\; 2\ell_3 g_3 \;\; \ell_1 g_2 \cdot \ell_2 g_1 \;\; \ell_2 g_3 + \ell_3 g_2 \;\; \ell_1 g_3 + \ell_3 g_1 \right] \tag{2-19}$$

the relation below must hold

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ g_1 & 0 & 0 & g_2 & 0 & g_3 \\ 0 & g_2 & 0 & g_1 & g_3 & 0 \\ 0 & 0 & g_3 & 0 & g_2 & g_1 \end{bmatrix} \underline{\underline{GT}} = \underline{\underline{I}} \qquad (2\text{-}20)$$

Where $\underline{\underline{I}}$ is the identity matrix.

Finding $\underline{\underline{GT}}$ involves taking the inverse of the 6x6 matrix above. If this interpolating function matrix is partitioned into four 3x3 matrices, so that (2-20) can be written

$$\begin{bmatrix} \underline{\underline{I}} & \underline{\underline{0}} \\ \underline{\underline{A}} & \underline{\underline{B}} \end{bmatrix} \underline{\underline{GT}} = \underline{\underline{I}} \qquad (2\text{-}21)$$

then $\underline{\underline{GT}}$ can be found by

$$\underline{\underline{GT}} = \begin{bmatrix} \underline{\underline{I}} & \underline{\underline{0}} \\ \underline{\underline{C}} & \underline{\underline{D}} \end{bmatrix} \qquad (2\text{-}22)$$

where

$$\underline{\underline{C}} = \underline{\underline{B}}^{-1} \underline{\underline{A}} \qquad (2\text{-}23)$$

and

$$\underline{\underline{D}} = \underline{\underline{B}}^{-1} \qquad (2\text{-}24)$$

Now calculations are simplified since the inverse of only one 3x3 matrix must be found to invert the 6x6 interpolating function matrix. Flux at a point in the triangle is found by evaluating the matrix $\underline{m}$ with the natural coordinates of the point in question, finding the derivatives of the natural coordinates in the triangle with respect to x, and calculating $\underline{\underline{G}}^T$. Note that $\underline{\underline{G}}^T$ is a matrix of constants within a triangle, but since the g_i depend on triangle geometry, $\underline{\underline{G}}^T$ will also be different for each distinct geometry.

An unexpected discovery prevented utilization of the above quadratic interpolating function in this study. It is described here only because it may be of interest to other researchers, and because it explains why the more complicated cubic fit over a triangle eventually had to be used.

For reasons to be discussed in chapter 4, it would be extremely inconvenient to construct a finite element mesh for the transport equation without the use of right triangles. In the case of a right triangle the derivatives of natural coordinates with respect to x (and u) are constrained so that two are of equal magnitude but opposite sign, and the third is identically zero. In this case $\underline{\underline{B}}$ of equation (2-21) is singular, and the right triangle is in every instance pathological for a quadratic interpolant that uses 3 fluxes and 3 derivatives as degrees of freedom.

Used instead was another quadratic interpolant, that uses values of flux at nodes and boundary centers to represent six degrees of freedom. In this case

$$\varphi = \begin{bmatrix} \varphi_1 & \varphi_2 & \varphi_3 & \varphi_4 & \varphi_5 & \varphi_6 \end{bmatrix} \tag{2-25}$$

Where the natural co-ordinates of nodes 4,5 and 6 are as

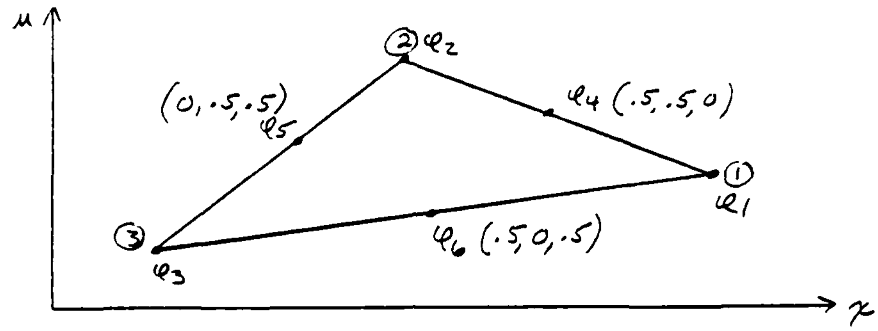


Figure 2-3
Numbering for a C^0 Quadratic Interpolant over a Triangle

given in figure 2-3. In this case $\underline{m}$ is the same, but the matrix $\underline{\underline{GT}}$ is no longer singular, and can now be found from the relation

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ .25 & .25 & 0 & .25 & 0 & 0 \\ 0 & .25 & .25 & 0 & .25 & 0 \\ .25 & 0 & .25 & 0 & 0 & .25 \end{bmatrix} \underline{\underline{GT}} = \underline{\underline{I}} \tag{2-26}$$

and the basis functions for this interpolant are

$$\underline{\hat{h}} = \underline{\hat{m}} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ -1 & -1 & 0 & 4 & 0 & 0 \\ 0 & -1 & -1 & 0 & 4 & 0 \\ -1 & 0 & -1 & 0 & 0 & 4 \end{bmatrix} \tag{2-27}$$

The cubic fit over a triangle requires that 10 degrees of freedom be specified. Chosen were flux values, and both partial derivatives at corner nodes, as well as the triangle centroid field variable value. Assigning numbers to the degrees of freedom as per below simplifies notation.

$$\hat{\underline{\varphi}} = \left[\phi_1 \quad \phi_{1x} \quad \phi_{1u} \quad \phi_2 \quad \phi_{2x} \quad \phi_{2u} \quad \phi_3 \quad \phi_{3x} \quad \phi_{3u} \quad \phi_4 \right]$$

$$= \left[\varphi_1 \quad \varphi_2 \quad \varphi_3 \quad \varphi_4 \quad \varphi_5 \quad \varphi_6 \quad \varphi_7 \quad \varphi_8 \quad \varphi_9 \quad \varphi_{10} \right] \qquad (2\text{-}28)$$

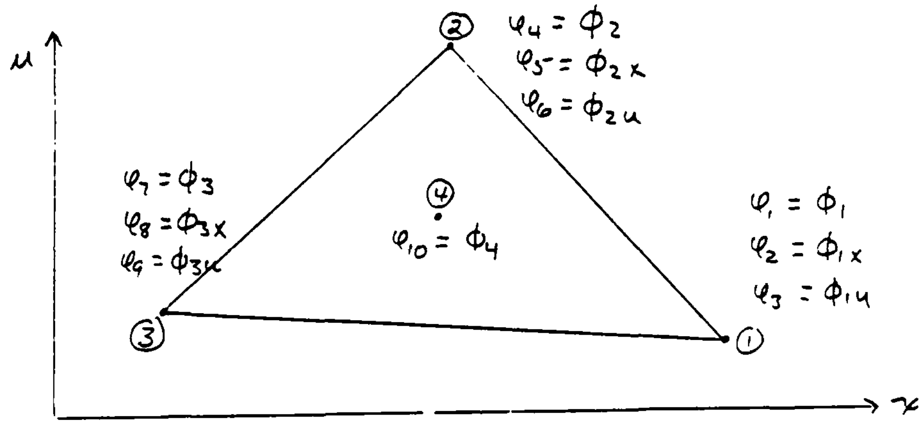


Figure 2-4
Numbering for Cubic Fit Over a Triangle

The basis functions are again given by (2-15) except now

$$\hat{\underline{m}} = \left[l_1^3 \quad l_1^2 l_2 \quad l_1^2 l_3 \quad l_2^3 \quad l_2^2 l_3 \quad l_2^2 l_1 \quad l_3^3 \quad l_3^2 l_1 \quad l_3^2 l_2 \quad l_1 l_2 l_3 \right] \qquad (2\text{-}29)$$

$$\hat{\underline{m}}_l = \Big[3l_1^2 g_1 \quad 2l_1 l_2 g_1 - l_1^2 g_2 \quad 2l_1 l_3 g_1 + l_1^2 g_3 \quad 3l_2^2 g_2$$
$$2l_2 l_3 g_2 + l_2^2 g_3 \quad 2l_2 l_1 g_2 + l_2^2 g_1 \quad 3l_3^2 g_3 \quad 2l_3 l_1 g_3 + l_3^2 g_1 \tag{2-30}$$
$$2l_3 l_2 g_3 + l_3^2 g_2 \quad g_1 l_2 l_3 + g_2 l_1 l_3 + g_3 l_1 l_2 \Big]$$

$$\hat{\underline{m}}_u = \Big[3l_1^2 f_1 \quad 2l_1 l_2 f_1 + l_1^2 f_2 \quad 2l_1 l_3 f_1 + l_1^2 f_3 \quad 3l_2^2 f_2$$
$$2l_2 l_3 f_2 + l_2^2 f_3 \quad 2l_2 l_1 f_2 + l_2^2 f_1 \quad 3l_3^2 f_3 \quad 2l_3 l_1 f_3 + l_3^2 f_1 \tag{2-31}$$
$$2l_3 l_2 f_3 + l_3^2 f_2 \quad f_1 l_2 l_3 + f_2 l_1 l_3 + f_3 l_1 l_2 \Big]$$

Evaluating this model at each of the 10 nodes yields the expression for $\underline{\underline{G}}^T$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3g_1 & g_2 & g_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3f_1 & f_2 & f_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3g_2 & g_3 & g_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3f_2 & f_3 & f_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3g_3 & g_1 & g_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3f_3 & f_1 & f_2 & 0 \\ \frac{1}{27} & \frac{1}{27} & \frac{1}{27} & \frac{1}{27} & \frac{1}{27} & \frac{1}{27} & \frac{1}{27} & \frac{1}{27} & \frac{1}{27} & \frac{1}{27} \end{bmatrix} \underline{\underline{G}}^T = \underline{\underline{I}} \tag{2-32}$$

It is not necessary to invert the 10 x 10 matrix above if it is partitioned into 9 3 x 3 matrices and several 3 x 1 vectors as below

$$\begin{bmatrix} \underline{\underline{A}} & \underline{\underline{0}} & \underline{\underline{0}} & \begin{smallmatrix} 0 \\ 0 \\ 0 \end{smallmatrix} \\ \underline{\underline{0}} & \underline{\underline{B}} & \underline{\underline{0}} & \begin{smallmatrix} 0 \\ 0 \\ 0 \end{smallmatrix} \\ \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{C}} & \begin{smallmatrix} 0 \\ 0 \\ 0 \end{smallmatrix} \\ \hat{\underline{s}} & \hat{\underline{s}} & \hat{\underline{s}} & \frac{1}{27} \end{bmatrix} \underline{\underline{G}}^T = \underline{\underline{I}} \tag{2-33}$$

if $\hat{\underline{s}} = \left[\frac{1}{27}\ \frac{1}{27}\ \frac{1}{27}\right]$ and $\underline{\underline{G}}^T$ is partitioned similarly then

$$\underline{\underline{G}}^T = \begin{bmatrix} \underline{\underline{D}} & \underline{\underline{0}} & \underline{\underline{0}} & \begin{smallmatrix}0\\0\\0\end{smallmatrix} \\ \underline{\underline{0}} & \underline{\underline{E}} & \underline{\underline{0}} & \begin{smallmatrix}0\\0\\0\end{smallmatrix} \\ \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{F}} & \begin{smallmatrix}0\\0\\0\end{smallmatrix} \\ \hat{\underline{p}} & \hat{\underline{q}} & \hat{\underline{r}} & 27 \end{bmatrix} \tag{2-34}$$

with

$$\begin{aligned} \underline{\underline{D}} &= \underline{\underline{A}}^{-1} & \hat{\underline{p}} &= -27\,\hat{\underline{s}}\,\underline{\underline{D}} \\ \underline{\underline{E}} &= \underline{\underline{B}}^{-1} & \hat{\underline{q}} &= -27\,\hat{\underline{s}}\,\underline{\underline{E}} \\ \underline{\underline{F}} &= \underline{\underline{C}}^{-1} & \hat{\underline{r}} &= -27\,\hat{\underline{s}}\,\underline{\underline{F}} \end{aligned} \tag{2-35}$$

The chore of inverting a 10 X 10 matrix is simplified. The basis functions, and the degrees of freedom for this cubic fit are specified. C^1 continuity is achieved with this fit. The only drawback is that the basis functions depend upon triangle geometry, and therefor must be recomputed for each unique triangle configuration.

C. The Tetrahedron and Three Dimensional Interpolation Functions

In three dimensions the simplest element is the four node tetrahedron. Four volume coordinates, (L_1, L_2 L_3 L_4) can be used to describe this element. A straight-forward linear relation exists between x,u,u' co-ordinates and a tetrahedron's natural coordinates:

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ \chi_1 & \chi_2 & \chi_3 & \chi_4 \\ \mu_1 & \mu_2 & \mu_3 & \mu_4 \\ \mu_1' & \mu_2' & \mu_3' & \mu_4' \end{bmatrix} \begin{bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \end{bmatrix} = \begin{bmatrix} 1 \\ \chi \\ \mu \\ \mu' \end{bmatrix} \tag{2-36}$$

(2-36) holds in a right handed system, if nodes are numbered such that nodes 1,2 and 3 progress counterclockwise when viewed from node 4.

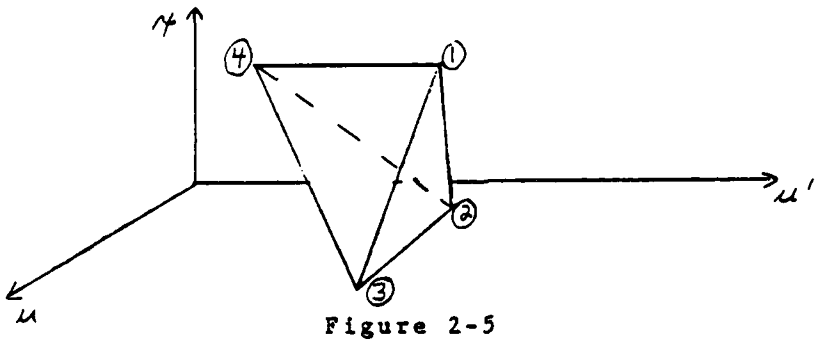


Figure 2-5
Four Node Tetrahedron Numbering in a
Right Handed Co-ordinate System

The partial derivatives of natural co-ordinates with respect to the x,y,z spatial variables will be needed in the next section to derive interpolating functions. If (2-36) is re-written as

$$\underline{\underline{A}} \begin{bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \end{bmatrix} = \begin{bmatrix} 1 \\ \chi \\ \mu \\ \mu' \end{bmatrix} \tag{2-37}$$

and the equation is differentiated with respect to x, then

$$\begin{bmatrix} \frac{\partial L_1}{\partial x} \\ \frac{\partial L_2}{\partial x} \\ \frac{\partial L_3}{\partial x} \\ \frac{\partial L_4}{\partial x} \end{bmatrix} = \underline{\underline{A}}^{-1} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \tag{2-38}$$

and the $\frac{\partial L_i}{\partial x}$ are the second column of $\underline{\underline{A}}^{-1}$. Similarly $\frac{\partial L_i}{\partial u}$ and $\frac{\partial L_i}{\partial u'}$ are the third and fouth columns of $\underline{\underline{A}}^{-1}$ respectively.

Integration of tetrahedral coordinates over a volume is conveniently given by (3: 148)

$$\int dV\, L_1^p L_2^q L_3^r L_4^s = 6V \frac{p!\, q!\, r!\, s!}{(p+q+r+s+3)!} \tag{2-39}$$

where V is the element volume given by

$$6V = \begin{vmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ u_1 & u_2 & u_3 & u_4 \\ u_1' & u_2' & u_3' & u_4' \end{vmatrix} \tag{2-40}$$

To prescribe a linear fit over this element 4 degrees of freedom must be specified. These can be the values of the flux at corner nodes so that

$$\phi = \sum_{i=1}^{4} L_i \varphi_i \tag{2-41}$$

A cubic requires twenty degrees of freedom in three dimensions. These can be nodal values of the flux, and the three directional derivatives at each node as well as face centered values of the flux, as per figure 2-6, with F as the field variable. Nodes 5 , 6, 7, and 8 are face centered across from nodes 1, 2, 3 and 4 respectively.

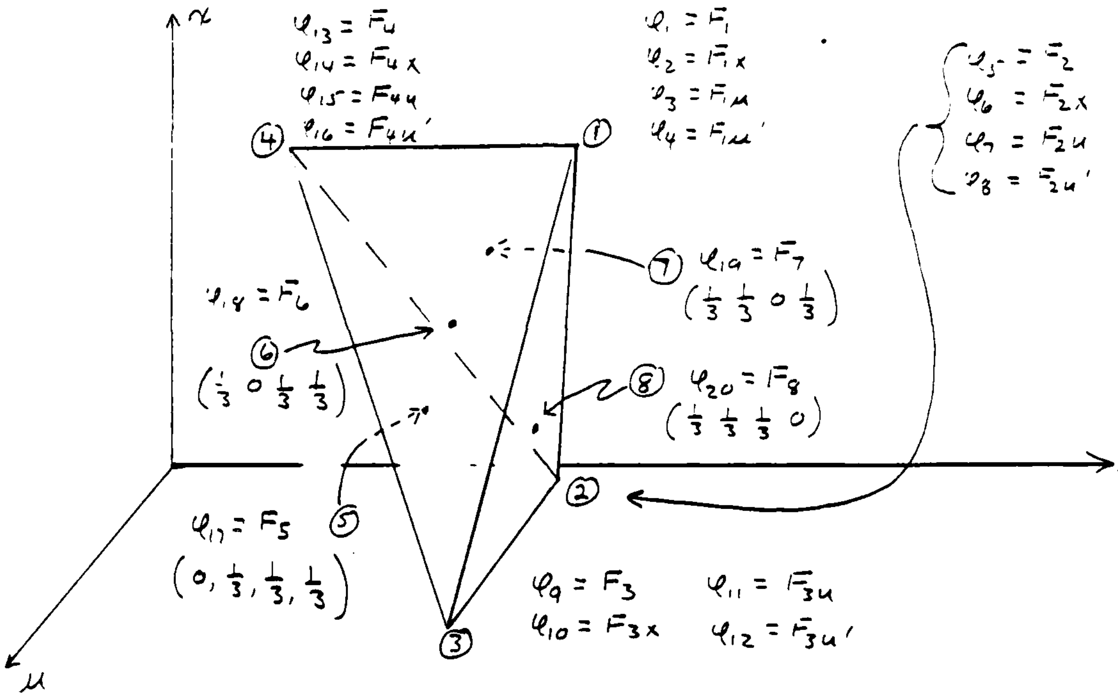


Figure 2-6
Numbering of 20 Degrees of Freedom for a Cubic Fit on a 4 Node Tetrahedron

The basis functions for this fit are again given by (2-15) with

$$\hat{\underline{m}} = \Big[L_1^3 \quad L_1^2 L_2 \quad L_1^2 L_3 \quad L_1^2 L_4 \quad L_2^3 \quad L_2^2 L_1 \quad L_2^2 L_3 \quad L_2^2 L_4 \quad L_3^3 \quad L_3^2 L_1 \quad L_3^2 L_2 \quad L_3^2 L_4 \quad L_4^3 \quad L_4^2 L_1 \quad L_4^2 L_2 \quad L_4^2 L_3 \quad L_2 L_3 L_4 \quad L_1 L_3 L_4 \quad L_1 L_2 L_4 \quad L_1 L_2 L_3 \Big] \qquad (2\text{-}42)$$

$\underline{\underline{GT}}$ is now a 20 X 20 matrix found by inverting

$$
\begin{bmatrix}
\underline{\underline{M1}} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{0}} \\
\underline{\underline{0}} & \underline{\underline{M2}} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{0}} \\
\underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{M3}} & \underline{\underline{0}} & \underline{\underline{0}} \\
\underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{M4}} & \underline{\underline{0}} \\
\underline{\underline{M5}} & \underline{\underline{M6}} & \underline{\underline{M7}} & \underline{\underline{M8}} & \underline{\underline{M9}}
\end{bmatrix} \tag{2-43}
$$

where

$$
\underline{\underline{M1}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 3e_1 & e_2 & e_3 & e_4 \\ 3f_1 & f_2 & f_3 & f_4 \\ 3g_1 & g_2 & g_3 & g_4 \end{bmatrix}
\qquad
\underline{\underline{M4}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 3e_4 & e_1 & e_2 & e_3 \\ 3f_4 & f_1 & f_2 & f_3 \\ 3g_4 & g_1 & g_2 & g_3 \end{bmatrix}
\qquad
\underline{\underline{M7}} = \begin{bmatrix} \frac{1}{27} & 0 & \frac{1}{27} & \frac{1}{27} \\ \frac{1}{27} & \frac{1}{27} & 0 & \frac{1}{27} \\ 0 & 0 & 0 & 0 \\ \frac{1}{27} & \frac{1}{27} & \frac{1}{27} & 0 \end{bmatrix}
$$

$$
\underline{\underline{M2}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 3e_2 & e_1 & e_3 & e_4 \\ 3f_1 & f_2 & f_3 & f_4 \\ 3g_1 & g_2 & g_3 & g_4 \end{bmatrix}
\qquad
\underline{\underline{M5}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1/27 & 0 & 1/27 & 1/27 \\ 1/27 & 1/27 & 0 & 1/27 \\ 1/27 & 1/27 & 1/27 & 0 \end{bmatrix}
\qquad
\underline{\underline{M8}} = \begin{bmatrix} \frac{1}{27} & 0 & \frac{1}{27} & \frac{1}{27} \\ \frac{1}{27} & \frac{1}{27} & 0 & \frac{1}{27} \\ \frac{1}{27} & \frac{1}{27} & \frac{1}{27} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}
$$

$$
\underline{\underline{M3}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 3e_3 & e_1 & e_2 & e_4 \\ 3f_3 & f_1 & f_2 & f_4 \\ 3g_3 & g_1 & g_2 & g_4 \end{bmatrix}
\qquad
\underline{\underline{M6}} = \begin{bmatrix} 1/27 & 0 & 1/27 & \frac{1}{27} \\ 0 & 0 & 0 & 0 \\ \frac{1}{27} & \frac{1}{27} & 0 & \frac{1}{27} \\ \frac{1}{27} & \frac{1}{27} & \frac{1}{27} & 0 \end{bmatrix}
\qquad
\underline{\underline{M9}} = \begin{bmatrix} \frac{1}{27} & 0 & 0 & 0 \\ 0 & \frac{1}{27} & 0 & 0 \\ 0 & 0 & \frac{1}{27} & 0 \\ 0 & 0 & 0 & \frac{1}{27} \end{bmatrix}
$$

and $e_1 = \frac{\partial L_1}{\partial x}$, $f_1 = \frac{\partial L_1}{\partial u}$, $g_1 = \frac{\partial L_1}{\partial u'}$
and so on. If $\underline{\underline{GT}}$ is partitioned into

$$
\underline{\underline{GT}} = \begin{vmatrix}
\underline{\underline{M10}} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{0}} \\
\underline{\underline{0}} & \underline{\underline{M11}} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{0}} \\
\underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{M12}} & \underline{\underline{0}} & \underline{\underline{0}} \\
\underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{M13}} & \underline{\underline{0}} \\
\underline{\underline{M14}} & \underline{\underline{M15}} & \underline{\underline{M16}} & \underline{\underline{M17}} & \underline{\underline{M18}}
\end{vmatrix} \tag{2-44}
$$

then

$$\underline{\underline{M10}} = \underline{\underline{M1}}^{-1} \qquad (2\text{-}45)$$

$$\underline{\underline{M11}} = \underline{\underline{M2}}^{-1} \qquad (2\text{-}46)$$

$$\underline{\underline{M12}} = \underline{\underline{M3}}^{-1} \qquad (2\text{-}47)$$

$$\underline{\underline{M13}} = \underline{\underline{M4}}^{-1} \qquad (2\text{-}48)$$

$$\underline{\underline{M18}} = 27\,\underline{\underline{I}} \qquad (2\text{-}49)$$

$$\underline{\underline{M14}} = -\underline{\underline{M18}}\ \underline{\underline{M5}}\ \underline{\underline{M10}} \qquad (2\text{-}50)$$

$$\underline{\underline{M15}} = -\underline{\underline{M18}}\ \underline{\underline{M6}}\ \underline{\underline{M11}} \qquad (2\text{-}51)$$

$$\underline{\underline{M16}} = -\underline{\underline{M18}}\ \underline{\underline{M7}}\ \underline{\underline{M12}} \qquad (2\text{-}52)$$

and

$$\underline{\underline{M17}} = -\underline{\underline{M18}}\ \underline{\underline{M8}}\ \underline{\underline{M13}} \qquad (2\text{-}53)$$

The basis functions for this fit are defined. They depend upon element geometry, and require that 4 separate 4 X 4 matrices be inverted.

C. Summary

This study uses two elements to construct finite element meshes They are the triangle and the four node tetrahedon. Describing these elements in terms of their natural coordinates is straightfoward, and will be seen to simplify later calculations.

Four interpolating functions, one linear, two quadratic and one cubic were investigated over a triangle. The quadratic that uses partial derivatives as degrees of freedom turns out to be singular for right triangles, so a quadratic C^0 fit was substituted. Two fits were done on the tetrahedron, a linear and a cubic. Any fit that uses field variable derivatives as interpolants has geometry dependent basis functions. These increase the number of calculations required since they must be found for each distinct element geometry.

3. <u>The Case of no Scatter</u>

With scattering cross section of zero, the expression to be digitized (1-16) becomes

$$I = \frac{1}{2}\int dx\,du\,\mu^2 \left(\frac{\partial\phi}{\partial x}\right)^2 + \frac{1}{2}\int dx\,du\,\Sigma_t^2 \phi^2 + \frac{1}{2}\int dx\,du\,2\mu\Sigma_t \phi \frac{\partial\phi}{\partial x} \quad (3\text{-}1)$$

Only particle streaming and absorption is occuring. The first integral of (3-1) is referred to as the streaming term, since it represents particle streaming, the second term is called the absorbing term for the analagous reason, and the third term is the boundary term. This last integral results from the cross product of streaming and absorbing terms and is referred to as the boundary contribution since without it the natural boundary conditions which arise from the integration by parts in equation (1-12) are not satisfied. These three terms are referred to as local since they fit field variable values limited to the triangle under scrutiny. In this section, a description of these terms' derivation and preparation for digitization will occur. Since the quadratic and cubic fits involve extremely long derivations, their results only are presented in appendices. Lastly, a test case to which an analytical solution exists is described, and numerical results of the various fits' digitization are presented.

A. The Local Terms

1. The Absorbing Term. Recalling (2-17) the interpolating function for Φ

$$\Phi = \underline{\hat{m}}\ \underline{\underline{G}}^T\ \underline{\varphi} \quad (2\text{-}17)$$

or

$$\Phi = \underline{\tilde{\varphi}}\ \underline{\underline{\tilde{G}}}^T\ \underline{m} \quad (3\text{-}1)$$

$$\int_{\mu \in \Delta_i} d\mu\, \phi_i = \left(\frac{\bar{\mu}_i - \underline{\mu}_i}{6}\right) \tilde{\underline{\varphi}}_i\, \tilde{\underline{\underline{GT}}}_i \Big[\underline{m}(l_1 \underline{l_2} l_3) + 4 * \underline{m}(l_1 \underline{\bar{l}_2} l_3) + \underline{m}(l_1 \bar{l}_2 l_3) \Big] \qquad (4\text{-}11)$$

$$= \tilde{\underline{\varphi}}_i\, \underline{\underline{LI}}_{x=c} \qquad (4\text{-}12)$$

Since the integration over X involves 3 points, LI (local integral) (of dimension 10x1) and NLI (of dimension 1x10) must be evaluated at x=a, b and c, then

$$\underline{\underline{NLM}}(i,j,k,l) = \left(\frac{a-c}{6}\right) * \alpha\, \hat{\underline{\varphi}}_i \Big[\underline{LI}_a \underline{NLI}_a + 4.0 * \underline{LI}_b \underline{NLI}_b + \underline{LI}_c \underline{NLI}_c \Big] \underline{\varphi}_j \qquad (4\text{-}13)$$

where NLM is the (10 X 10) (k x l) non local matrix reflecting triangle i scattering to triangle j. The five triangle column produces 25 such non local matrices, all of which must be assembled globally, and must be saved if triangle penalties are desired as mesh refinement indicators.

In appendix G, subroutine LCORD finds the natural coordinates of the points (49 for Weddle's n=6) needed on each triangle for integration, and evaluates $\underline{m}$ and $\underline{m}_\chi$ (needed to evaluate the second scattering integral) at each of these points. ANING perfoms the angle integrals, finding matrices $\underline{LI}$ and $\underline{NLI}$. Finally SPING calculates the space integral across the width of the triangle. The subroutines are well documented, with a list

Simpson's method yields

$$\int_{\mu' \in \Delta_j} d\mu' \, \phi(x,\mu') \cong \frac{(\bar{\mu}_j - \underline{\mu}_j)}{6} \Big[\phi(x, \underline{\mu}_j) + 4 * \phi(x, \underline{\bar{\mu}}_j) + \phi(x, \bar{\mu}_j) \Big] \qquad (4\text{-}8)$$

where $\bar{\mu}_j$ = u of triangle j, upper (x fixed)

$\underline{\bar{\mu}}_j$ = u of triangle j, middle

$\underline{\mu}_j$ = u of triangle j, lower

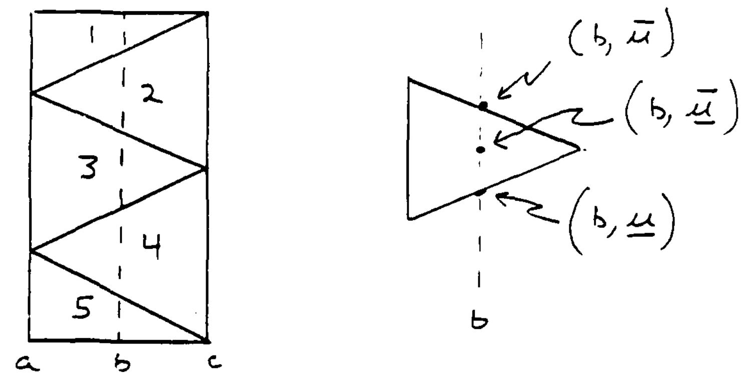


Figure 4-3
Sample 5 triangle column with U upper, lower and center of triangle 3 for X = B

which, in terms of the interpolation functions is

$$= \left(\frac{\bar{\mu}_j - \underline{\mu}_j}{6}\right) \Big[\underline{\tilde{m}}\,(\underline{l_1 l_2 l_3}) + 4 * \underline{\tilde{m}}\,(\underline{\overline{l_1 l_2} l_3}) + \underline{\tilde{m}}\,(\overline{l_1 l_2 l_3}) \Big] * \underline{\underline{G}}^T_j \, \underline{\varphi}_j' \qquad (4\text{-}9)$$

$$= \underline{NLI}_{x=c} \, \underline{\varphi}_j' \qquad (4\text{-}10)$$

where $\underline{\tilde{m}}\,(\overline{l_1 l_2 l_3})$ represents $\underline{\tilde{m}}$ evaluated at the natural coordinates of $(c, \bar{\mu}_j)$ referred to as $(\overline{l_1 \; l_2 \; l_3})$, and NLIx=c is the non local integral of triangle j at x = c.

Similarly, over the local triangle

triangles, was abandoned.

C. Numerical Evaluation of the Scattering Integrals

The first attempt to evaluate the scattering terms was to calculate the integrals with numerical approximations. Since the cubic fit provides such high accurary, it was felt that the very good streaming and absorbing approximations, with less accurate scattering, would provide solutions properly reflecting the physics involved in a problem.

Simpsons rule integration, Weddle's, and Weddle's rule for n=6 were sequentially tried. This section will describe Simpson's integration for one of the scattering terms,since it is the simplest to write out. The second integral and the other two techniques are straightforward extensions, and appendix H contains subroutines used numerically to evaluate both scattering integrals with Weddle's rule for n=6.

In the case of integral A

$$\int_a^c dx \int_{-1}^{1} du \int_{-1}^{1} du' \, \alpha \, \phi \phi' \tag{4-6}$$

where $\alpha = \left(\frac{\Sigma_s^2}{2} - \Sigma_s \Sigma_t\right)$, summation over the 5 triangle column of figure 4-3, is represented as

$$= \alpha \int_a^c dx \sum_{i=1}^{5} \int_{\mu \in \Delta_i} du \, \phi(x,u) \sum_{j=1}^{5} \int_{\mu' \in \Delta_j} du' \, \phi(x,u') \tag{4-7}$$

where $\mu \in \Delta_i$ represents integration over u in triangle i.

problem to simplify evaluation of the integral in this manner.

If meshes are constrained columnarly as in figure 4-2, then for example, integral A is

$$\left(\frac{\varepsilon_s^2}{2} - \varepsilon_s \varepsilon_t\right) \int_a^b dx \int_{-1}^{1}\int_{-1}^{1} du\, du'\, \phi\, \phi'$$

$$= \left(\frac{\varepsilon_s^2}{2} - \varepsilon_s \varepsilon_t\right) \int_a^b dx \int_{-1}^{1}\int_{-1}^{1} du\, du' \sum_{i=1}^{n} \sum_{j=1}^{n} \phi_i\, \phi_j' \qquad (4\text{-}5)$$

where n is the number of triangles in a column. Integration over u, can proceed from the column's top triangle to the

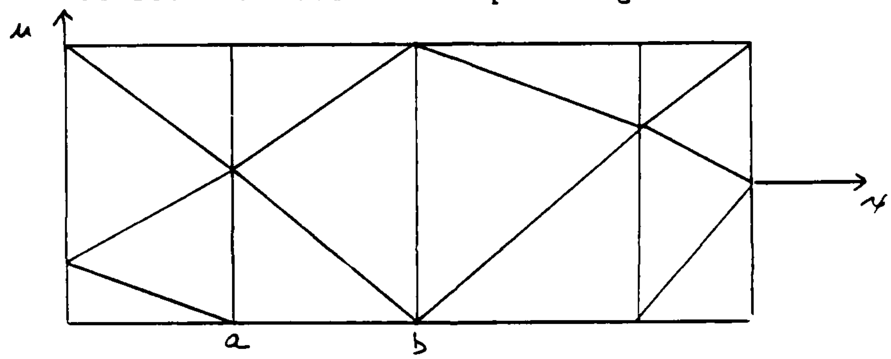


Figure 4-2
Colummar Finite Element Mesh

bottom triangle, one element at a time, halting at each triangle to integrate over u' for all elements in a column. Since n is the number of triangles in a column, n*n angle integrals are evaluated per column. Each integral is integrated over space separately and results in a non local matrix that must be assembled globally. The bookkeeping involved in evaluating the two scattering integrals is simplified. For this reason, the quadratic fit using derivatives as finite element nodes discussed in chapter 2, found to be nonexistent in right these terms numerically, and with analytical approximations, and describes the results of these efforts. The cubic interpolant of chapter 3 was used as the finite element approximation for flux.

B. Mesh Arrangement

Integration of equation (4-3) is over both angular variables u and u'. This proves to be cumbersome. Consider integration from x=a to x=b of Figure 4-1

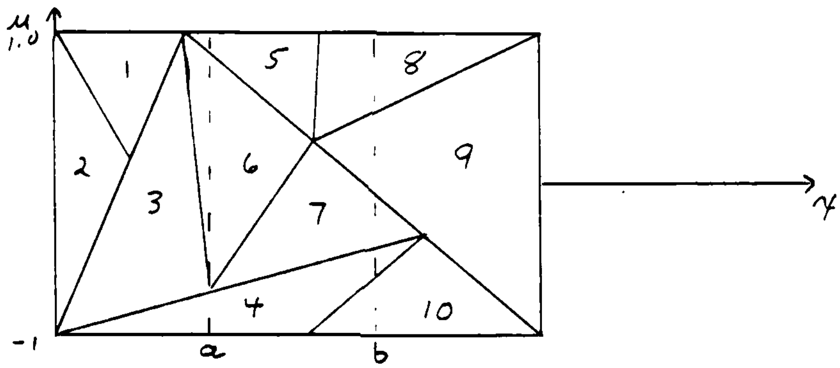


Figure 4-1
An Unrestricted Finite Element Mesh
Over the Benchmark Solution Domain of
Figure 3-1

$$\int_a^b \int_{-1}^{1} \int_{-1}^{1} du' \, du \, dx \tag{4-4}$$

Φ and Φ_x are given by differing cubic interpolatation functions in each of the 7 triangles composing this area. Proper bookkeeping becomes a serious challenge.

To avoid this, triangles are restricted to columns. While this makes successive mesh refinement cumbersome, and will probably be awkward if the method is ever extended to another dimension, it is certainly appropriate for an early cut at the

4. The Case of Isotropic Scatter

A) The Non Local Terms

When Scattering in allowed to occur (1-16) must be evaluated in its entirety. This equation can be rewritten as

$$\frac{1}{2}\iint dx\,du \left\{ \mu^2\left(\frac{\partial\phi}{\partial x}\right)^2 + \Sigma_t^2\phi^2 + 2\mu\Sigma_t\frac{\partial\phi}{\partial x}\phi \right.$$

$$\left. + \left(-\mu\Sigma_s\frac{\partial\phi}{\partial x} - \Sigma_t\Sigma_s\phi\right)\int_{-1}^{1}du'\,\phi' + \frac{\Sigma_s}{2}\int_{-1}^{1}du'\,\phi'\frac{\Sigma_s}{2}\int_{-1}^{1}du''\,\phi''\right\} \quad (4\text{-}1)$$

If one neglects the three local integrals the non local, or scattering integrals are left,

$$\frac{1}{2}\iiint dx\,du\,du'\,\frac{\Sigma_s^2}{4}\phi'\int_{-1}^{1}\phi''\,du''$$

$$+\frac{1}{2}\iiint dx\,du\,du'\left(-\mu\Sigma_s\frac{\partial\phi}{\partial x} - \Sigma_t\Sigma_s\phi\right)\phi' \quad (4\text{-}2)$$

Since their is no u dependence in the first integral, it may be integrated out, then a change of variables from u" to u may occur, allowing both integrals to be combined, resulting in

$$\frac{1}{2}\iiint dx\,du\,du'\left(\frac{\Sigma_s^2}{2} - \Sigma_t\Sigma_s\right)\phi\phi' + \frac{1}{2}\iiint dx\,du\,du'\left(-\mu\Sigma_s\right)\frac{\partial\phi}{\partial x}\phi' \quad (4\text{-}3)$$

These are the two scattering integrals. Since they involve integration over u and u', they result in non-local terms. Only one of them ($\phi\phi'$) results in a naturally symmetric non- local matrix after global assembelage. These integrals are referred to as A and B in this report and the code of appendix A. This section describes the preparation for digitization of mesh 4 is worse than meshes 2 and 3. Scrutiny of element penalties reveals that this is due to elements of mesh 4 being refined at the area where the largest derivative discontinuity occurs (x=0, elements 4 and 5). This can be considered as

Mesh	Total Penalty linear	quadratic	cubic
1	.02556	.00337	.000398
2	.01074	.000559	.0000474
3	.01433	.000637	.0000526
4	.00716	.000373	.0000502
5	.00136	.000067	.0000158
6	.00101		

Table 3-3
Mesh Penalties as Convergence Indicators

further evidence of the cubic fits' power, it is flagging to the programmers attention the nonphysical boundary condition; the C^0 fits are not sophisticated enough to display the anomaly.

E. Summary

The three terms which comprise the transport functional in the case of no scatter are all local. Their derivations are straightforward and nearly trivial in the linear case. For higher order interpolants the derivation is still easy to follow, but very long. When assembled globally these terms represent the transport functional. Setting the variation of this functional to zero leaves a positive definite set of simultaneous linear equations that is solved to find nodal values. The cubic interpolant (with C^1 continuity) is more powerful, and may be faster than the C^0 interpolants, which require excessive mesh refinement before converging. Penalties are powerful indicators of convergence.

Two pieces of data appear as anomalies in Table 3-2. The first of these is that the cubic fit for mesh 1 appears better than mesh 2 or 3. Table 3-3 lists total penalties of the meshes, and indicates that since mesh 2 and mesh 3 have lower penalties, the finite element fit is actually better in these meshes. Mesh 1 was "lucky" for the cubic in that nodal values came out so close to the analytical.

Mesh	For u>0 # of Triangles / Mean Free Path	Avg Perc Diff of Analytic to Numeric Linear		Quadratic		Cubic	
1	1.33	46.6	(1.5)	13.73	(4.1)	1.37	(9.2)
2	2.00	28.4	(2.2)	5.55	(5.5)	2.83	(12.1)
3	2.00	29.5	(2.3)	5.64	(6.7)	2.76	(12.1)
4	3.67	27.2	(2.9)	4.24	(8.5)	.84	(18.8)
5	8.50	8.7	(5.0)	5.45	(17.4)	.55	(35.2)
6	21.33	3.3	(14.3)	...	...	...	...

Table 3-2
Comparison of Mesh Refinement Required For Convergence of Local Terms. Average Percent Difference is From Analytic to Numerical Solution for u > 0. Values in Parenthesis are CPU Seconds of Runtime on a Vax 11-780, unix Berkely 1.2, During Periods of Moderate to Almost Heavy Use.

Likewise mesh 5 for the quadratic fit appears worse than less refined meshes. The total penalty bears out that mesh 5 is a better fit. These and other similar experiences emphasis two points that should not be neglected with the finite element method. The first of these is that element penalties can be as good a measure of convergence, or better, than comparing nodal values to some "exact" solution. Secondly, the meshes used in this study are not necessarily successively refined. Without this type of refinement, steady convergence of [illegible] values to the exact solution may not be observed (3:79).

One anomaly appears in table 3-3. That is the cubic fit for refinement. The difference is clearly caused because flux

u=1 x	Analytic	Linear	Quadratic	Cubic
.375	.6873	.6360	.6904	.6865
1.000	.4724	.3834	.4282	.4707
1.500	.2231	.1405	.1707	.2230
3.000	.0498	.0270	.0302	.0512
CPU units (sec's)	...	2.9	8.5	18.8

Table 3-1
Comparison for Mesh 4 of Accuracy
and Run Times for 3 Interpolation Fits on a Vax 11-780
(unix, Berkely 1-2) for u=1

spatial derivatives are held continuous in the C^1 fit. The C^0 quadratic fit is only slightly better than the linear. Consider the expansion of 1-17 and 1-18 in the no scattering case.

$$-2\mu^2 \frac{\partial^2 \phi}{\partial x^2} + 2\Sigma_t^2 \phi = 0 \qquad (1\text{-}17a)$$

$$2\mu \left[\mu \frac{\partial \phi}{\partial x} + \Sigma_t \phi \right] = 0 \qquad (1\text{-}18a)$$

These are the differential equations being satisfied when the variational functional is minimized, and the natural boundary condition. Three quantities in these equations must be approximated by finite elements, ϕ, $\frac{\partial \phi}{\partial x}$ and $\frac{\partial^2 \phi}{\partial x^2}$. C^0 continuity holds only one of these continuous across elements boundaries. The C^1 fit holds two of the three continuous and as a result converges faster. This analysis further indicates that a C^2 fit would converge even faster, and a C^3 fit would be no better than a C^2, since no higher order terms are needed to satisfy these equations.

$$\phi = \mu e^{-\frac{\Sigma_t x}{\mu}} \qquad \mu > 0 \tag{3-19}$$

and

$$\phi = 0 \qquad \mu \leq 0 \tag{3-20}$$

Since most streaming is occuring near u = 1, meshes were refined more in that area. Also, notice that the boundary conditions are not physical, there is a discontinuity of derivatives along the u=0 line. For now one must realize that this discontinuity is a source of error that becomes apparent for u near zero.

C. Results

The area of the Benchmark problem was discretized with 6 separate meshes for the computer runs. All are drawn and listed in appendix F. Meshes 1 through 5 are those used by Goff (2:114) while mesh 6 is a very well refined mesh of 80 triangles in 3 mean free paths. Mesh 2 and 3 consist of the same number of triangles, but with a different pattern, to test sensitivity to element orientation. Table 3-1 shows a comparision of interpolation fit accuracy with the analytical solution for mesh 4.

Table 3-2 Lists the average nodal percent difference from the analytical solution to allow a comparison of the degree of mesh refinement required for convergence. It is seen from these results that the cubic fit is extremely powerful. Run times should not be taken as absolute, but it appears that the price paid in terms of extra calculations for the more accurate fit is not extreme. The linear fit converges as finite element theory says it will, but only with an excessive amount of mesh performed for a triangle, the resultant quadratic form can be written as

$$I = \frac{1}{2}\,\underline{\hat{\varphi}}\;\underline{\underline{\widehat{GT}}}\left[\underline{\underline{MA}} + \underline{\underline{MB}} + \underline{\underline{MS}}\right]\underline{\underline{GT}}\;\underline{\varphi} = \frac{1}{2}\,\underline{\hat{\varphi}}\;\underline{\underline{\widehat{GT}}}\;\underline{\underline{ML}}\;\underline{\underline{GT}}\;\underline{\varphi} \qquad (3\text{-}17)$$

where $\underline{\underline{ML}}$ is the local matrix. The global assembelage of these local matrices results in a quadratic form for the variational integral over the entire area under scrutiny. When minimized, the remaining set of simultaneous linear equations has a coefficient matrix that is always positive definite, and is solved by cholesky decomposition in this study.

B. The Test Case

The problem chosen to test the digitization of the local terms was a monoenergetic lambertian (flux = cosine of the angle it is traveling) source of particles incident upon an absorbing only slab 3 mean free paths thick. The slab is surrounded on both sides by a vacuum so there is no returning flux from the right boundary.

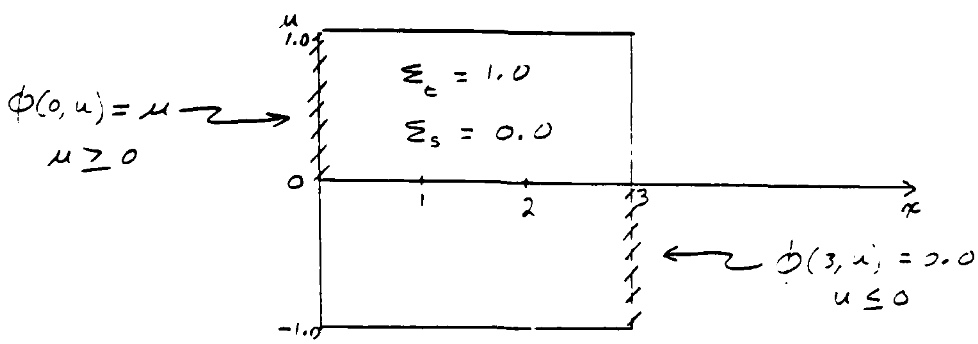


Figure 3-1
Benchmark Problem Description

In this case the transport equation is

$$\mu \frac{\partial \phi}{\partial x} + \Sigma_t \phi = 0 \qquad (3\text{-}18)$$

The solution is

interpolant evaluation are in appendix C .

3. The Streaming term.

Expansion of μ yields six integrals which must be evaluated.

$$\mu^2 = \mu_1^2 l_1^2 + \mu_2^2 l_2^2 + \mu_3^2 l_3^2 + 2\mu_1\mu_2 l_1 l_2 + 2\mu_1 \mu_3 l_1 l_3 + 2\mu_2\mu_3 l_2 l_3 \tag{3-11}$$

$$\frac{1}{2}\int dA\, \mu^2\left(\frac{\partial\phi}{\partial x}\right)^2 = \frac{1}{2}\,\hat{\underline{\varphi}}\,\underline{\underline{GT}}\Big[\int dA\,\mu_1^2 l_1^2\,\underline{m}_x\,\hat{\underline{m}}_x + \int dA\,\mu_2^2 l_2^2\,\underline{m}_x\,\hat{\underline{m}}_x + \int dA\,\mu_3^2 l_3^2\,\underline{m}_x\,\hat{\underline{m}}_x + \int dA\,2\mu_1\mu_2 l_1 l_2\,\underline{m}_x\,\hat{\underline{m}}_x + \int dA\,2\mu_1\mu_3 l_1 l_3\,\underline{m}_x\,\hat{\underline{m}}_x + \int dA\,2\mu_2\mu_3 l_2 l_3\,\underline{m}_x\,\hat{\underline{m}}_x\Big]\underline{\underline{GT}}\,\underline{\varphi} \tag{3-12}$$

$$= \frac{1}{2}\,\hat{\underline{\varphi}}\,\hat{\underline{\underline{GT}}}\left[\underline{\underline{MS1}} + \underline{\underline{MS2}} + \underline{\underline{MS3}} + \underline{\underline{MS4}} + \underline{\underline{MS5}} + \underline{\underline{MS6}}\right]\underline{\underline{GT}}\,\underline{\varphi} \tag{3-13}$$

$$= \frac{1}{2}\,\hat{\underline{\varphi}}\,\hat{\underline{\underline{GT}}}\;\underline{\underline{MS}}\;\underline{\underline{GT}}\,\underline{\varphi} \tag{3-14}$$

When linear interpolants are substituted the streaming term is

$$\underline{\underline{MS}} = \frac{F}{24A}\begin{bmatrix} g_1^2 & g_1 g_2 & g_1 g_3 \\ g_1 g_2 & g_2^2 & g_2 g_3 \\ g_1 g_3 & g_2 g_3 & g_3^2 \end{bmatrix} \tag{3-15}$$

where

$$F = \mu_1(\mu_1+\mu_2) + \mu_2(\mu_2+\mu_3) + \mu_3(\mu_3+\mu_1) \tag{3-16}$$

Derivation of this term for the quadratic and cubic fits are the most lengthy of all; results of this effort are in appendix D.

4. The Local Matrix. The sum of these three terms, evaluated over a triangle in the mesh, is the value of the variational integral over that element. After these calculations are

The Boundary term can be written as

$$\frac{1}{2}\int dA\,\Sigma_t\,2\mu\,\frac{\partial\phi}{\partial x}\,\phi = \Sigma_t\int dA\,\mu\,\underline{\hat{\varphi}}\,\underline{\underline{\widetilde{GT}}}\,\underline{m}_x\,\underline{\hat{m}}\,\underline{\underline{GT}}\,\underline{\varphi} \tag{3-6}$$

using (2-17) and since

$$\frac{\partial\phi}{\partial x} = \frac{\partial}{\partial x}\left[\underline{\hat{\varphi}}\,\underline{\underline{\widetilde{GT}}}\,\underline{m}\right] = \underline{\hat{\varphi}}\,\underline{\underline{\widetilde{GT}}}\,\underline{m}_x \tag{3-7}$$

the Boundary term can be expressed as the sum of three integrals

$$= \sum_{i=1}^{3}\Sigma_t\int dA\,\mu_i\,l_i\,\underline{\hat{\varphi}}\,\underline{\underline{\widetilde{GT}}}\,\underline{m}_x\,\underline{\hat{m}}\,\underline{\underline{GT}}\,\underline{\varphi} \tag{3-8}$$

in the linear case, evaluation of the integral yields

$$= \frac{1}{2}\underline{\hat{\varphi}}\,\underline{\underline{\widetilde{GT}}}\left[\mu_1\Sigma_t 2A\begin{bmatrix}2g_1 & g_2 & g_3\\ 2g_1 & g_2 & g_3\\ 2g_1 & g_2 & g_3\end{bmatrix} + \mu_2\Sigma_t 2A\begin{bmatrix}g_1 & 2g_2 & g_3\\ g_1 & 2g_2 & g_3\\ g_1 & 2g_2 & g_3\end{bmatrix} + \mu_3\Sigma_t 2A\begin{bmatrix}g_1 & g_2 & 2g_3\\ g_1 & g_2 & 2g_3\\ g_1 & g_2 & 2g_3\end{bmatrix}\right]\underline{\underline{GT}}\,\underline{\varphi}$$

$$= \frac{1}{2}\underline{\hat{\varphi}}\,\underline{\underline{\widetilde{GT}}}\left[\underline{\underline{MB1}} + \underline{\underline{MB2}} + \underline{\underline{MB3}}\right]\underline{\underline{GT}}\,\underline{\varphi} = \frac{1}{2}\underline{\hat{\varphi}}\,\underline{\underline{GT}}\,\underline{\underline{MB}}\,\underline{\underline{GT}}\,\underline{\varphi} \tag{3-9}$$

The resultant matrix will not be symmetric, but since for each quadratic form, only one symmetric matrix exists (4:342), the boundary term can be symmetrized. That is

$$\frac{1}{2}\int dA\,\Sigma_t\,\mu\,\frac{\partial\phi}{\partial x}\,\phi = \frac{1}{2}\underline{\hat{\varphi}}\,\underline{\underline{\widetilde{GT}}}\left[\left(\underline{\underline{MB}} + \underline{\underline{\widehat{MB}}}\right)/2.0\right]\underline{\underline{GT}}\,\underline{\varphi} \tag{3-10}$$

This term is much more complicated to encode than the absorbing term, since it involves derivatives of natural coordinates with respect to x, which are different for each separate geometry. The results of quadratic and cubic

The absorbing term can be written as

$$\frac{1}{2}\int dx\,dy\;\Sigma_t^2\,\phi = \frac{\Sigma_t^2}{2}\int dA\;\hat{\underline{\varphi}}\;\widetilde{\underline{\underline{GT}}}\;\underline{m}\;\hat{\underline{m}}\;\underline{\underline{GT}}\;\underline{\varphi}$$

$$= \frac{\Sigma_t^2}{2}\;\hat{\underline{\varphi}}\;\widetilde{\underline{\underline{GT}}}\left[\int dA\;\underline{m}\;\hat{\underline{m}}\right]\underline{\underline{GT}}\;\underline{\varphi} \tag{3-2}$$

Since $\underline{\varphi}$ and $\underline{\underline{GT}}$ are constant within an element, they can be removed from the integral. Evaluation of the term then reduces to taking the product of $\underline{m}\,\hat{\underline{m}}$ and analytically performing the integral.

In the linear case $\underline{\underline{GT}} = \underline{\underline{I}}$ and

$$\underline{m}\,\hat{\underline{m}} = \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix}\begin{bmatrix} l_1 & l_2 & l_3 \end{bmatrix} = \begin{bmatrix} l_1^2 & l_1 l_2 & l_1 l_3 \\ l_1 l_2 & l_2^2 & l_2 l_3 \\ l_1 l_3 & l_2 l_3 & l_3^2 \end{bmatrix} \tag{3-3}$$

which, using (2-7) is

$$\Sigma_t^2\int dA\;\underline{m}\;\hat{\underline{m}} = \frac{2A\,\Sigma_t^2}{4!}\begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} = \underline{\underline{MA}} \tag{3-4}$$

where A is the area of the triangle.

The absorbing term can now be written as

$$\frac{1}{2}\int dA\;\Sigma_t^2\,\phi^2 = \frac{1}{2}\;\hat{\underline{\varphi}}\left[\widetilde{\underline{\underline{GT}}}\;\underline{\underline{MA}}\;\underline{\underline{GT}}\right]\underline{\varphi} \tag{3-5}$$

It is precisely the above expression in brackets which is digitized and evaluated for each element.

With the C^0 quadratic fit, and C^1 cubic previously described in chapter 2, the term derivation is analagous, expect that $\underline{\underline{MA}}$ in these cases is of order 6 and 10 respectively. Appendix B has results of these computations.

2. The Boundary term

of variables included in the appendix.

Note that LI and NLI are actually the same integral, just over different triangles.

$$\int d\mu\, \phi = \underset{\sim}{\hat{\varphi}}\ \underline{LI}$$

$$\int d\mu'\, \phi' = \underline{NLI}\ \underset{\sim}{\varphi} = \underline{\hat{LI}}\ \underset{\sim}{\varphi}$$

Subroutine ANING takes advantage of the fact that $\underline{LI} = \underline{\widetilde{NLI}}$ and calculates the $\int d\mu\, \phi$ over every triangle in the mesh, storing it in memory to be recalled when needed. Simultaneously it calculates $\int d\mu\ \mu\ \phi_x$, the only other angle integral needed, storing these in ILFD.

D. Cubic Analytical Approximation

Since ϕ and ϕ_x are approximated with cubic functions, the scattering integrals of eqn (4-3), which integrate the products of ϕ and ϕ_x are integrating a hexadic. Explained in this section is an attempt to substitute another cubic for the "exact" sixth order fit required by $\phi\phi'$ and $\phi_x \phi'$.

In three dimensions, the local and non local triangles map out tetrahedrons. Four possible cases can occur, depending upon the orientation of the local and non local triangles as depicted in Figure 4-4. Case 2 and case 4 result in pyramids, which can split along their center into two four node tetrahedra each, and integration can then be performed separately over each tetrahedron, and summed.

Consider the first scattering integral

$$\int dx\, d\mu\, d\mu'\, \alpha\, \phi\, \phi' \qquad (4\text{-}14)$$

$$= \alpha \int dx\ d\mu\ d\mu'\ F \qquad (4\text{-}15)$$

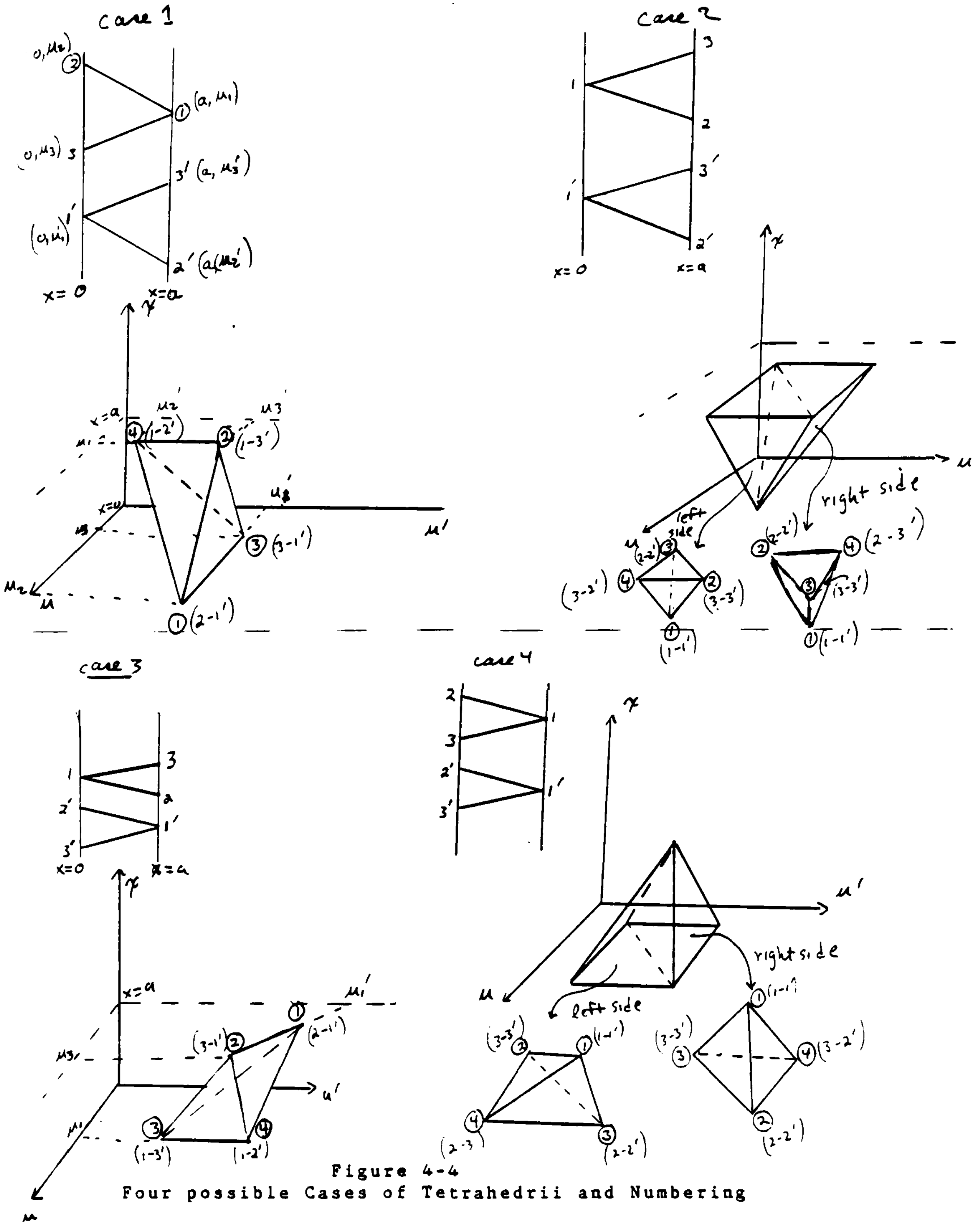


Figure 4-4
Four possible Cases of Tetrahedrii and Numbering

F, the product can be approximated in three dimensions with a cubic polynomical of complete basis as

$$F = \underline{\hat{m}} \; \underline{\underline{GT}} \; \underline{f} \tag{4-16}$$

where $\underline{m}$ and GT are as given in (2-41) and (2-44) respectively. F can be considered to be a column matrix of twenty 10 X 10 matrices, one representing F at each of the twenty nodes of figure 2-6. For instance, case 1, node 1 is the intersection of nodes 2 (local) and node 1 (primed). f is then

$$f_1 = \phi_2 \phi_1' = \underline{\hat{\varphi}} \; \underline{\underline{\widetilde{GT}}} \; \underline{m} \; \underline{\hat{m}}' \; \underline{\underline{GT}}' \; \underline{\varphi}' \tag{4-17}$$

evaluation of $\underline{m}$ at (0,1,0) and $\underline{m}'$ at (1,0,0), and carrying out the multiplication is guaranteed to yield

$$f_1 = \underline{\hat{\varphi}} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4-18}$$

since both ϕ_2 and ϕ_1' are finite element interpolation nodes.

With this cubic approximation, the second scattering integral is only slightly more complicated.

$$-\Sigma_s \int dx\, du\, du' \, \mu \phi_x \phi' = -\Sigma_s \int dx\, du\, du' \left(\mu_1 L_1 + \mu_2 L_2 + \mu_3 L_3 + \mu_4 L_4 \right) G \tag{4-19}$$

if the same cubic approximation is made for $G = \phi_x \phi'$

$$G = \underline{\tilde{m}} \; \underline{\underline{GT}} \; \underline{g} \tag{4-20}$$

then four integrals, caused by the expansion of u, must be evaluated.

The f_i's and g_i's are in most instances trivial, and can be written down by inspection for each of the four cases. This is done in appendix E.

The integrations are simple compared to those done for the streaming case since there are no cross products of $\underline{m}$ and $\underline{m}_x$.

E. The Test Case

Chosen to test the numerical and analytical evaluations of the integrals was the same domain as in the no scattering case, with the region depth under scrutiny varying from one to five mean free paths. Graciously provided by Dr. Shankland was a spherical harmonics solution of the problem using up to 46 legendre polynomials. Results of these calculations are listed in appendix G for scattering cross sections corresponding to c of .5 and .9 where $c\Sigma_t = \Sigma_s$. Dr. Shankland used as a lower right boundary condition no return flux at infinity. Therefore, the lower right boundary used in the finite element solution is the Pn angular flux for u < 0 at 1,2,3,4 or 5 mean free paths, depending upon the depth of investigation desired. In this case, there are no sources in the region under scrutiny, and the coupled Pn equations are solved with a Green's function.

The lambertian source, depicted in figure 3-1 is non physical. The derivative discontinuity at x=0 is very difficult to approximate with a finite polynomial series. The expected solution for the lambertian at this spatial point for the cases of c=.5 and c=.9 would be similar to figure 4-5, with more backscatter in the c=.9 case than the c=.5 . As a result, the approximating function generated by the legendre polynomials changes less rapidly about u=0 for c=.9 than for c=.5, and can be constructed with less polynomials.

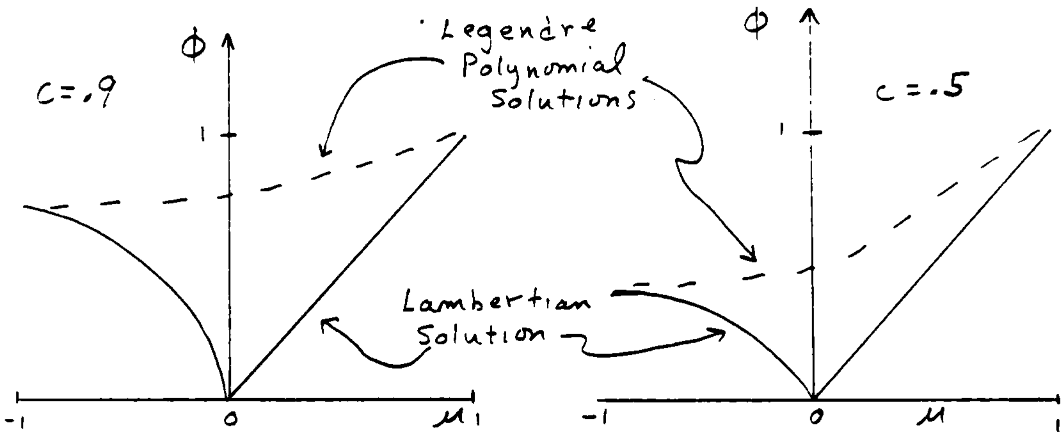


Figure 4-5
Expected Solution at x=0 for the Lambertian Source and Expected Legendre Polynomial Approximations

Used as finite element left hand boundary conditions are the legendre polynomial values of angular flux in appendix H for X=0 and u>0 . With these boundary conditions the finite element solution was tested, and its results compared to the spherical harmonics solution throughout the regions, of varying depth, and with varying cross sections, under scrutiny.

D. Results

Penalty functions are guaranteed to be positive with this method since the value of the functional is the integral of a quantity squared.

$$I = \frac{1}{2} \int_D dD \left(\mathcal{L}\phi - s\right)\left(\mathcal{L}\phi - s\right) \qquad (1\text{-}11)$$

If a negative penalty occurs, it is an indicator of error. In the case of no scatter, every penalty was greater than zero because flux was approximated as a cubic, and the integration was exact. In the scattering case, numerical evaluation of the scattering integrals is hardly exact, nor is approximating a hexadic with a cubic, and then analytically performing the integral. Negative penalties could occur, and they would indicate error in evaluating the scattering integral.

The global matrix is guaranted to be positive definite. In the streaming case it always was positive definite, again due to the exactness of the integration. In the scattering case, a non positive definite global matrix is another indicator of error in scattering integral evaluation. This type of matrix could be solved, and the solution might be fairly accurate, but a desirable charateristic of the finite element method is that the resulting set of linear equations has a positive definite coefficient matrix, since it can be solved quicky and accurately by direct means. Other than positive definite matrices solution techniques must rely upon iterative solution methods, or very long direct schemes, therefore this study insists that the means used to evaluate scattering results in positive definite global coefficient matrices.

Numerical Results

Both negative penalties and non positive definite matrices were common with the three numerical techniques used. Simpsons rule was used first. For simple meshes (meshes 1-4) positive definite global matrices occured. For any further refinement, error in the integration grew, and the matrix became non positive definite. Since Weddle's rule fits a cubic, it was tried next, and more refinement could occur until the same effect happened. Weddle's rule for n=6 was the last numerical technique tried, and its error causes non positive definite matrices with around 15 triangles per mean free path at c=.5 . In table 4-1 are results for Weddle's N=6 rule in mesh D with a depth of 3 mean free paths and c=.5. It shows that the method does not provide acceptable accuracy. It seems odd that mesh refinement would increase error. What is occuring is that as the number of triangles is increased, the numerical integration must be performed more often. The error accumulates until it destroys the global matrices positive definiteness. This is even more apparent if c=.9 is used; the scattering integral's contributions are greater, and even less refined meshes produce negative definite matrices. Numerical integration of the scattering integrals holds no potential. Orthogonal relations could be tried, they are used successfully in the Sn method, but they would probably not be the solution. In the Sn method, iteration throughout the mesh must occur to reach the proper solution. The finite element technique, as formulated in this study, is not adaptive to iterative, or "marching" methods.

Cubic Approximation

Comparison of finite element and PN solution for c=.5 and c=.9 were conducted. Two types of boundary conditions were tried. First, only fluxes were specified and secondly fluxes and its derivatives with respect to u were specified. Derivatives

```
ED WCOUT
LI,1,5
1   ===> CO MSHE3.5C MESH
2   ===> XE
3   IER IS ...          0
4   NTRIA      N      SIGMAS
5     46      151       0.500
LI,276,50
```

276	COORDINATES		CURRENTS		FIN ELEM	Legendre	
277	X	U	X	U	FLUX	PN FLUX	% DIFF
278	0.000	1.000	-0.849	0.988	1.014	1.014	0.000
279	0.000	0.750	-0.806	0.976	0.767	0.767	0.000
280	0.000	0.500	-0.731	0.983	0.526	0.526	0.000
281	0.000	0.250	-0.441	0.810	0.275	0.275	0.000
282	0.000	0.000	0.064	0.617	0.121	0.121	0.000
283	0.000	-0.250	-0.176	0.352	0.135	0.103	31.817
284	0.000	-0.500	-0.110	0.074	0.106	0.101	5.345
285	0.000	-0.750	-0.088	0.045	0.091	0.074	23.119
286	0.000	-1.000	-0.083	0.219	0.072	0.052	38.357
287	0.750	1.000	-0.457	0.859	0.542	0.533	1.802
288	0.750	0.750	-0.369	0.656	0.349	0.343	1.757
289	0.750	0.250	-0.085	0.191	0.107	0.095	12.497
290	0.750	0.000	-0.033	0.167	0.093	0.064	46.769
291	0.750	-0.250	-0.046	0.104	0.068	0.052	30.086
292	0.750	-0.750	-0.031	0.006	0.048	0.037	26.993
293	0.750	-1.000	-0.043	0.306	0.025	0.033	25.721
294	1.500	1.000	-0.256	0.694	0.290	0.273	6.264
295	1.500	0.500	-0.100	0.252	0.083	0.074	11.697
296	1.500	0.000	0.018	0.091	0.035	0.030	15.972
297	1.500	-0.500	0.001	0.126	0.023	0.020	15.874
298	1.500	-1.000	-0.004	0.373	-0.001	0.016	103.227
299	2.250	1.000	-0.134	0.456	0.150	0.139	7.686
300	2.250	0.750	-0.074	0.192	0.072	0.069	4.287
301	2.250	0.250	-0.015	0.042	0.019	0.019	4.260
302	2.250	0.000	0.011	0.112	0.010	0.014	30.249
303	2.250	-0.250	-0.005	0.049	0.014	0.011	28.081
304	2.250	-0.750	-0.012	0.016	0.013	0.008	58.603
305	2.250	-1.000	0.006	0.214	-0.001	0.007	119.351
306	3.000	1.000	-0.068	0.251	0.075	0.071	6.296
307	3.000	0.500	-0.014	0.044	0.016	0.014	11.580
308	3.000	0.000	-0.013	0.005	0.007	0.007	0.000
309	3.000	-0.250	-0.009	0.004	0.005	0.005	0.000
310	3.000	-0.500	-0.010	0.003	0.004	0.004	0.000
311	3.000	-0.750	-0.010	0.002	0.004	0.004	0.000
312	3.000	-1.000	-0.002	0.002	0.003	0.003	0.000

$\underbrace{\text{rows 278–282}}_{\text{Boundary Conditions}}$ $\underbrace{\text{rows 308–312}}_{\text{Boundary conditions}}$

```
313  AVERAGE % DIFFERENCE IS ..      19.07682837
314  FOR AN AVERSGE OF      15.333      TRIANGLES PER MEAN FREE PATH
EOT..
UP
```

26.69% For points not specified by boundary cond's

Table 4-1
Weddle's Rule n=6 Results for Mesh D, with Range=3.0,
u Derivatives and Fluxes Specified on the Boundary, c=.5

were found by the use of difference equations on the Pn data. A total of four meshes, A through D were used (appendix F). A,B, and C meshes all have a depth of 3 mean free paths. Mesh D depth was varied from 1 to 5 mean free paths. Since data from these meshes is extensive, selected output is displayed in appendix A, and results are summarized in table 4-2.

The results of table 4-2 indicate that convergence is occuring for c=.9 data only after excessive mesh refinement. The data for c=.5 indicates convergence of the finite element code is occuring, but not to the Pn solution. Specifying u derivatives speeds up convergence. Penalties, expounded as being so important in chapter 3, appear to carry no useful information.

Both boundary conditions are appropriately specified. It is customary in the widely used Pn and Sn transport codes, to specify only fluxes. In general these codes do not directly use the u derivatives on boundary. However if the flux is known as a function of u at a specific spatial location, then certainly the flux derivative with respect to u is known at that point. Since the finite element code uses ϕ_u as an interpolation node, specifying its value on the boundary is appropriate, and can only speed convergence to the same solution.

The lack of exactness in scattering integral evaluation has destroyed element, and total penalty usefulness. Consider the origin of a particular element's penalty, pen(i)

$$pen(i) = \underline{\hat{\varphi}}_i \, \underline{\underline{ML}} \, \underline{\varphi}_i + \underline{\tilde{\varphi}}_i \sum_{j=\text{top column } \Delta}^{\text{bottom } \Delta} \underline{\underline{NLM}}(i,j) \, \underline{\varphi}_j \qquad (4\text{-}21)$$

= streaming and absorbing contribution + scattering contribution

The streaming and absorbing component is always positive. It

c=.5

Mesh	Depth(mean free paths)	# triangles per mfp	Avg perc diff	Total penalty	Sum of Abs(pen)
D	1	46	7.81	.51E-5	.78E-2
D	2	23	7.37	-.53E-5	.87E-2
D	3	15.3	7.85(8.91)	-.36E-4	.87E-2
D	4	11.5	9.34	-.11E-3	.85E-2
D	5	9.2	13.5	-.25E-3	.82E-2
C	3	13.3	26.2 (7.38)	-.94E-4	.47E-2
B	3	4.0	* (22.53)		
A	3	1.3	21.4(64.51)	-.32E-2	.92E-2

C=.9

Mesh	Depth(mean free paths)	# triangles per mfp	Avg perc diff	Total penalty	Sum of Abs(pen)
D	1	46	1.11	.10E-5	.82E-2
D	2	23	1.27	-.11E-4	.11E-1
D	3	15.3	3.88(1.59)	-.47E-4	.11E-1
D	4	11.5	14.89	-.13E-3	.10E-1
D	5	9.2	45.79	-.31E-3	.86E-2
C	3	13.3	75.46(1.28)	.40E-4	.15E-2
B	3	4.0	* (6.69)		
A	3	1.3	58.0(18.91)	-.23E-2	.10E-1

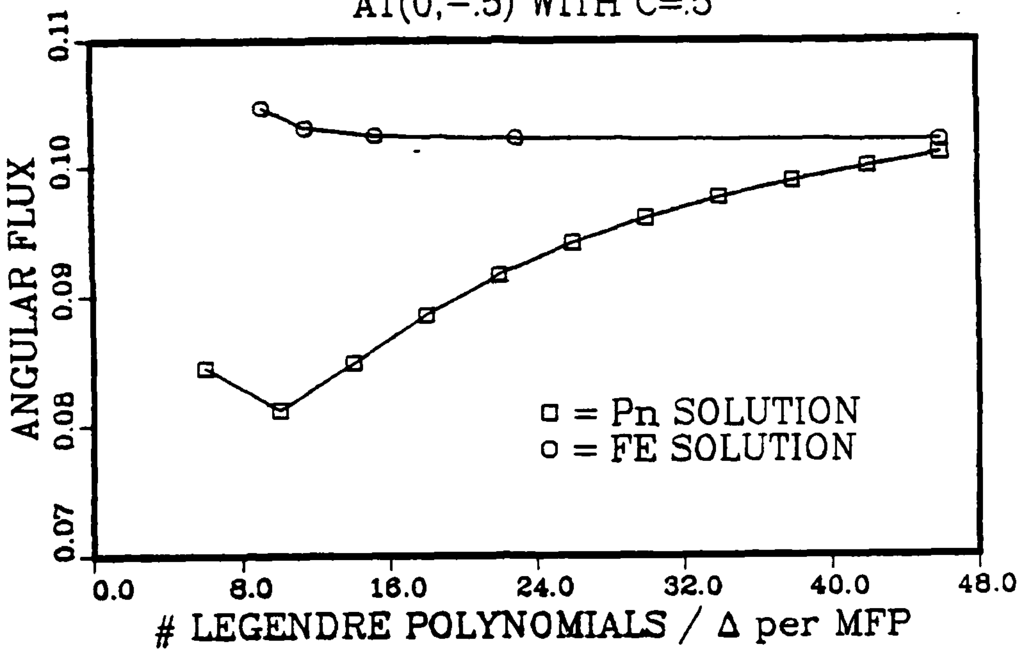
Table 4-2
Cubic Approximation of Scattering Integral Results Compared with Legendre Polynomial Solution for c=.5 and c=.9. Average Percent Difference of Nodal Values Other than Those Specified as Boundary Conditions, with flux only as a Boundary Condition. Values in Parenthesis are Same Meshes with Flux and u Derivative Specified. * is Non Positive Definite Global Matrix.

exactly this that constitutes penalties in the case of no scatter. The scattering component is expected to be negative, as can be seen by evaluating the signs of the integral coefficient constants of equation (4-3), $\left(\frac{\Sigma_s^2}{2} - \Sigma_s \Sigma_t\right)$ and $(-\Sigma_s)$ are both less than zero. The sum of the streaming and scattering penalty contributions should remain positive however, since they represent the square of a quantity. When scattering evaluation is with error, its penalty contribution can grow too large, and an element's penalty drops below zero. If this happens, summing of element penalties for a total mesh penalty leads to misleading information. It was thought that the magnitude of a penalty might carry the desired information on a fit's correctness, so the sum of penalty absolute values was computed. Comparison of this data, displayed in table 4-2 also lacks the desired information. Negative element penalties are in every instance associated with triangles where a large amount of scattering, and a small amount of streaming is occuring. Because the scattering integral evaluation is inexact, the penalty function has lost its value.

Graphs of figures 4-6 and 4-7 compare finite element solutions with the legendre polynomial solution, for various triangle densities and numbers of legendre polynomials being used. All finite element computations on the graphs were done with mesh D, varying the depth to change triangle densities. It appears from this data that with more legendre polynomials the finite element and Pn solutions would be exact. Convergence is faster in the c=.9 case because the larger backscatter creates a smoother flowing function, able to be approximated with fewer legendre polynomials than the more rapidly changing c=.5 solution. Boundary conditions, used in the finite element code as specified by the spherical harmonic solution, have not settled down yet either, as shown in table 4-3.

Based upon this information it appears that the finite

FIGURE 4–6
COMPARISON OF FE AND Pn SOLUTION
AT(0,–.5) WITH C=.5



AT(0,–.25) WITH C=.5

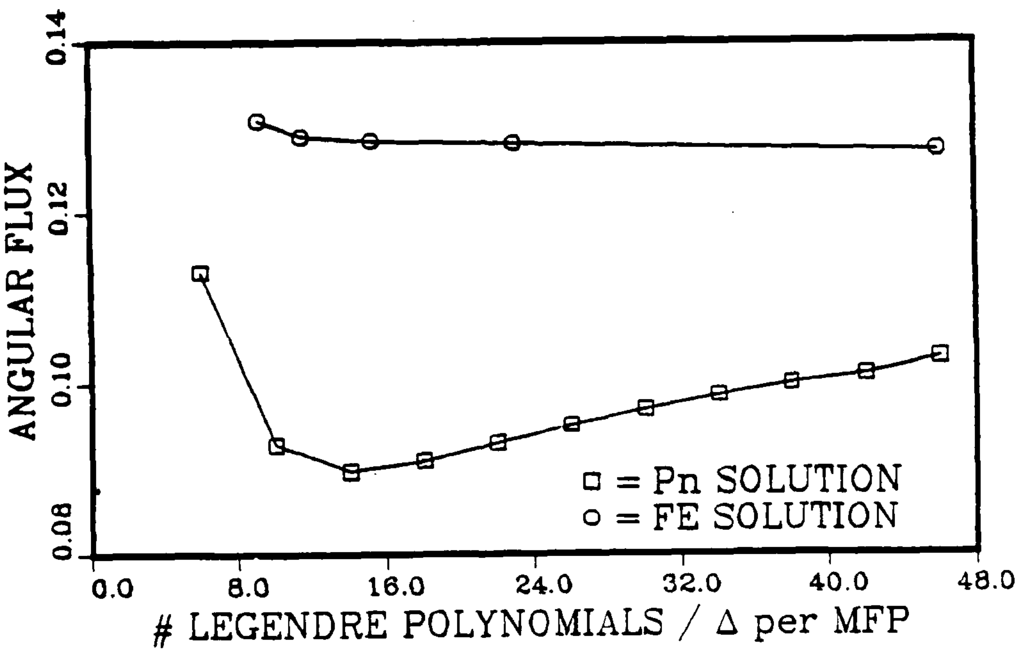


FIGURE 4-7
COMPARISON OF FE AND Pn SOLUTION
AT(0,-.25) WITH C=.9

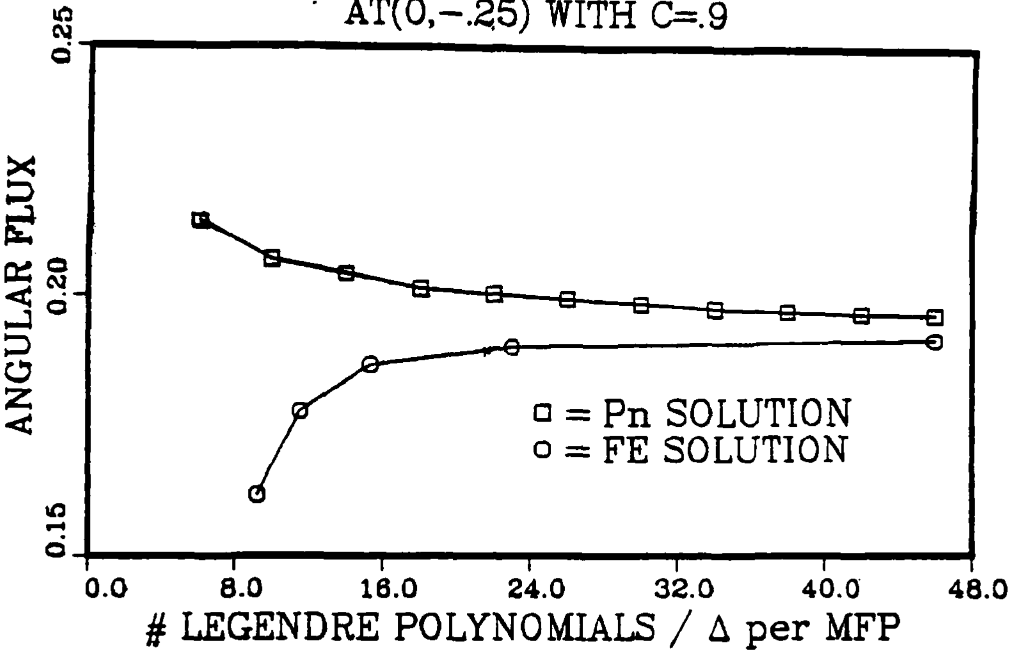
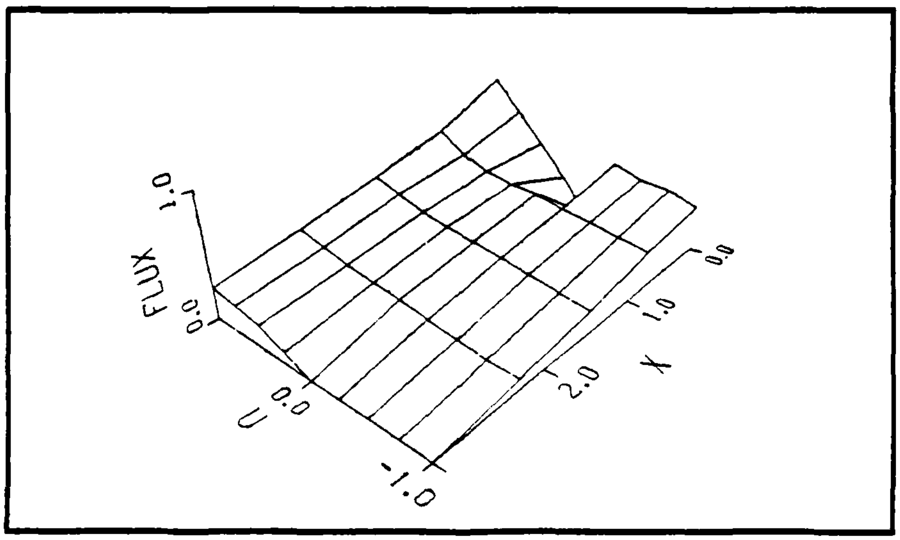


FIGURE 4-8
ANGULAR FLUX FROM A LAMBERTIAN
SCATTERING ONLY MEDIUM



(x,u)	c	change of flux	% change
(0,.5)	.5	4.6E-3	.87
0,.5	.9	1.1E-3	.33
0,.25	.5	7.6E-3	2.7
0,.25	.9	2.1E-3	.68
3,-1	.5	6.5E-5	1.9
3,-1	.9	1.0E-6	3.5E-3

Table 4-3
Pn Predicted Flux Rate of Change of Selected Boundary Points Over Polynomials 30-46

element calculations are succesfully predicting angular flux. Under these circumstances it is difficult to determine the amount of refinement required for convergence, but it appears that if fluxes only are specified on boundaries the method converges with 15 to 20 triangles per mean free path. If fluxes and derivatives are specified, convergence occurs with 10 - 15 elements per mean free path. Less angle refinement is also required if derivatives are specified on boundaries. This is approximately the same degree of refinement as an S4 calculation with two spatial nodes per mean free path. Positive definite matrices are not guaranteed (as in the case of mesh B.).

The only difference between mesh C and D is refinement over angle in the first and last columns. Close analysis of table 4-2 data shows that this angle refinement is more important than spatial refinement when fluxes only are used as boundary conditions. This is further indication of scattering term inexactness. The scattering calculations error can be estimated from c=.9 data of table 4-2. With 46 triangles per mean free path, the average nodal percent difference of 1.11% can be

Pen - element penalty

G1, G2, G3 - derivatives of triangular coordinates w.r.t. space

F1, F2, F3 - derivatives of triangular coordinates w.r.t. angle

X1,X2,X3,U1,U2,U3 - specific coordinates of the triangle under scrutinies geometric nodes

V - Array storing the integral of the twenty tetrahedral coordinate combinations which together form a complete basis for a cubic in three dimensions (2-41). Row two has the integral of times (2-41), row three times (2-41), rows 4 and 5 contain and times (2-41) integrated over tetrahedral volume respectively.

SGM - M5, M6, M7, and M8 of (2-42) and M18 of (2-43)

E, F, G, - arrays of dimension 4 storing the derivatives of the four tetrahedral coordinates w.r.t. space, incident angle and scattered angle respectively

H - the basis functions for each of the 5 scattering integrals (expansion of u requires that the second integral be done four separate times)

SA - the first scattering integral matrix

SB - the second scattering integral matrix

Integers Passed as Arguments

N - number of nodes

TRI - local triangle

TRIP - non local triangle

NTRIA - number of triangles

Appendix A - Program Listing

Glossary of Variables

Variables Passed as Common

MG - Global matrix

ML - local matrix

NLM - non local matrix

GT - matrix of interpolating function constants

Variables Passed as Double Precision Arguments

Cordnd - cartesian (x,u) coordinates of finite element nodes

Phi - angular flux

Areas - triangle areas

MA - absorbing matrix

SC1, SC2 , ... SC6 - coefficients of streaming matrices per appendix D

SR1, SR2, ... SR6 - per appendix D, row matrices to augment SC matrices

BC1, BC2, BC3, BR1, BR2 , BR3 - coefficients of boundary matrices per appendix C

MB - boundary matrix

MS - streaming matrix

DRVS - matrix of derivatives, overlayed on boundary term coeficients per appendix C

Range - depth, in mean free paths, of region under scrutiny

SIGMAT - Σ_t

SIGMAS - Σ_s

Bibliography

1. Duderstadt, James J. and William R. Martin, Transport Theory, New York, John Wiley & Sons, 1979

2. Goff, Alan D., A Finite Element Solution of a Self Adjoint Transport Equation in One Dimension, Masters Thesis, Air Force Institute of Technology, 1984. (AFIT/GNE/PH/84M-4)

3. Huebner, Kenneth H., and Earl A. Thornton, The Finite Element Method For Engineers, John Wiley & Sons 1982.

4. Kreysig, Erwin, Advanced Engineering Mathematics, John Wiley & Son, 1983.

5. Lewins, Jeffery. Importance, The Adjoint Function, Pergamon Press Ltd. Norwich, England 1965.

6. Strang, Gilbert, and George Fix, An Analysis of the Finite Element Method, Englewood Cliffs, N.J., 1973.

7. Westlake, Joan R., A Handbook of Numerical Matrix Inversion And Solution of Linear Equations, John Wiley & Sons, New York, 1968

Any interpolant that uses field variable derivatives as finite element interpolation nodes has basis functions that are geometry dependent, and increases calculations significantly. This type of interpolant was accepted for the local terms because the increase in accuracy made up for the extra calculations. It is not necessary to use a geometry dependent interpolating function for the nonlocal terms, and this type of approximation holds no accuracy benefits. Exact scattering term evaluation is possible with geometry independent interpolants, and is recommended as any subsequent study's first effort.

interpolants.

The streaming results clearly show the penalty function's usefulness. Not only is the global penalty a faultless indicator of accuracy, element penalties may be used to dictate where local refinement should occur.

Two methods were used to evaluate the scattering terms, and analysis of their results leads to a proprosal for a third integration technique, which should be both more efficient and accurate. Numerical techniques, of accuracy up to Weddle's for n=6 were unsuccessful. Cubic approximation of the hexadic scattering integral was accurate, required slightly more refinement than expected, and appears to be computationally excessive. Worst of all, the inexactness of the scattering integral evaluation destroys penalty value, and does not guarantee positive definite global matrices. Chapter 5 describes proposed exact hexadic integration, with geometry independent basis functions that should significantly reduce computations, return penalty usefulness, and insure positive definiteness. With exact scattering integral evaluation, accuracy equal to the streaming case should be achieved with comparable mesh refinement.

Extensions to other than isotropic scatter will be straightforward. If the scattering kernel is expanded in terms of a Legendre polynomial series as it ordinarily is, the scattering integrals would be slightly more complicated, but achievable. Integration with dx, du and du' over a four node tetrahedron would still result, only the form of the function being integrated would be changed.

6. Conclusion

The finite element method has been very successful in a variety of fields. It was felt that since the self adjoint reformulation of the transport operator could be expressed as a quadratic functional, finite elements could be applied succesfully to transport problems. Concisely stated, the result of this study is that the method works, and that it appears to hold potential for very accurate solutions with moderately refined meshes. The present digitization of the method, described in this document, and written in appendix A, bears improvement, both in accuracy and computational efficiency.

It was found that with linear interpolants the method converged in the case of no scatter, with around 25 triangles per mean free path for $u>0$. Linear interpolants were not tested in the scattering case, but straightforward extension of the streaming results suggests that at least 50 triangles per mean free path will be required to reach an accurate solution. This is an enormous amount of refinement. The C^0 quadratic fit was only slightly better. Unfortunately columnar mesh restriction destroyed a semi C^1 fit, and cubic interpolants were used. These are very powerful in the streaming case, achieving accuracy of greater than 99% with around 4 triangles per mean free path. Codes used in this study where not written with the intention of comparing speeds. Run time comparisons are therefore not absolute, but they do indicate that the more accurate fit is not computationally excessive, and may even require less cpu time to converge than either of the C^0

and

$$\Phi_\gamma = \underline{\tilde{h}}_\gamma \, \underline{\varphi}$$

(5-2)

where $\underline{h}$ and $\underline{h}_\gamma$ in this instance represent the dimension (10) distinct basis functions and their spatial derivatives of the cubic fits over a triangle, found while calculating the local terms.

Polynomial	Quantity	Polynomial	Quantity
L_i^6	4	$L_i^3 L_j^2 L_k$	24
$L_i^5 L_j$	12	$L_i^3 L_j L_k L_l$	4
$L_i^4 L_j^2$	12	$L_i^2 L_j^2 L_k^2$	4
$L_i^4 L_j L_k$	12	$L_i^2 L_j^2 L_k L_l$	6
$L_i^3 L_j^3$	6		

Table 5-1
84 Polynomials for Three Dimensional Hexadic

C. Summary

Results of chapter 4 dictate the need for exact scattering integral evaluation. The hexadic using flux only as degrees of freedom will integrate exactly, and probably reduce computations. Time precluded digitization of this fit, and it is recommended as the first effort of any subsequent study.

tetrahedron of figure 5-1 into six layers of equal height. Natural coordinates of these nodes can be computed using 2-38 and the matrix $\underline{\underline{GT}}$ can be found by the method described in chapter 2.

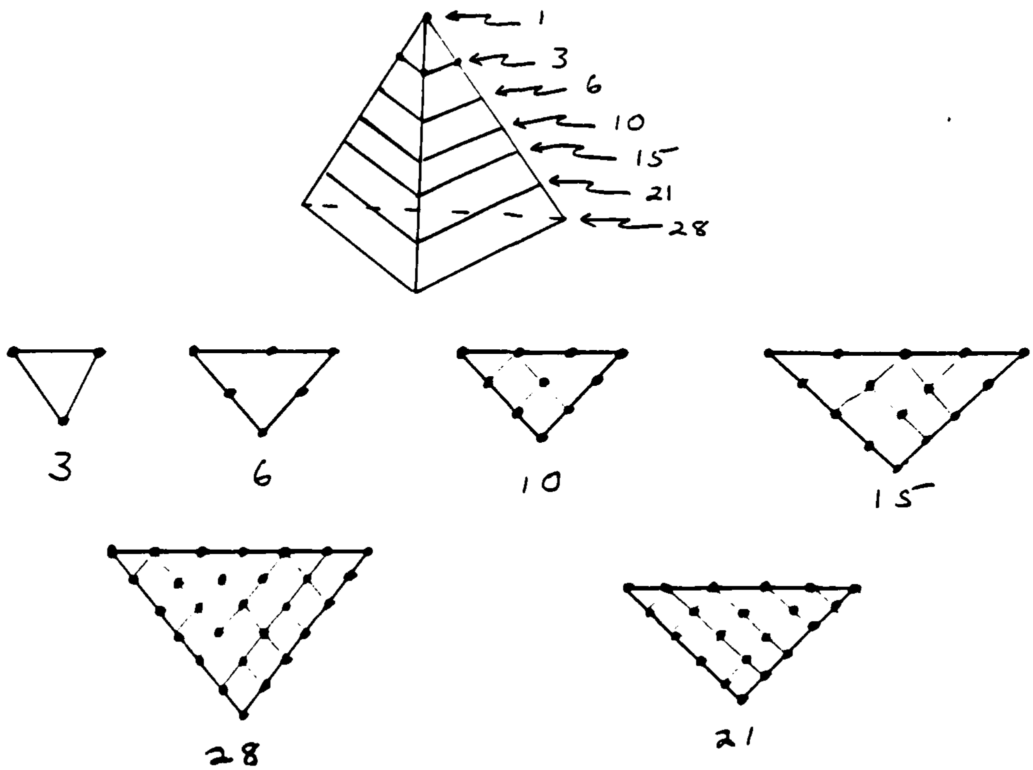


Figure 5-1
84 Flux Interpolation Nodes of Hexadic Three Dimensional Fit

The 84 polynomials, which together constitute a complete basis for the hexadic are given in table 5-1, with quantities indicated.

With this information the basis functions are specified. The degrees of freedom $F = \Phi \Phi'$ and $G = \Phi_x \Phi'$ are distinct for each tetrahedron. Φ and Φ_x need to be calculated only once for each triangle with

$$\Phi = \underline{\tilde{h}} \; \underline{\varphi} \tag{5-1}$$

5. Exact Scattering Integral Evaluation

Chapter 4 results show that the finite element method, in the case of isotropic scatter works, but one would like to see it converge with less mesh refinement, and with a smaller number of computations. A method of exactly integrating the scattering terms is explained in this chapter that should meet this objective, as well as restore the penalty function's usefulness and guarantee positive definite matrices.

A. Hexadic Interpolation With Flux

A hexadic function in three dimensions requires 84 degrees of freedom to be completely specified. If all are flux, then they can be described in terms of the twenty nodal two dimensional cubic interpolants (ten from each triangle), independent of tetrahedral geometry. That is to say, the basis functions would be constant, since they no longer involve derivatives of natural coordinates. There are five distinct integrals to be performed, because of the u expansion in the second scattering integral. Basis functions can be calculated separately, and stored in a single matrix of dimension (5,84). This significantly reduces calculations, and eliminates the requirement for the finite element transport code to find three dimensional interpolants entirely.

B. Interpolation Nodes and Basis Functions

The following nodes are evenly volume distributed and should provide a good hexadic fit. Consider slicing the derivatives of the flux on boundaries.

The penalty function's usefulness has been ruined by inexact scattering integral evaluation.

element mesh D results in nearly 800 tetrahedra over which the scattering integrals must be evaluated. Each of these tetrahedra, treated in the code as geometrically distinct, require separate interpolation functions, and this is the probable cause of calculational excesses.

Summary

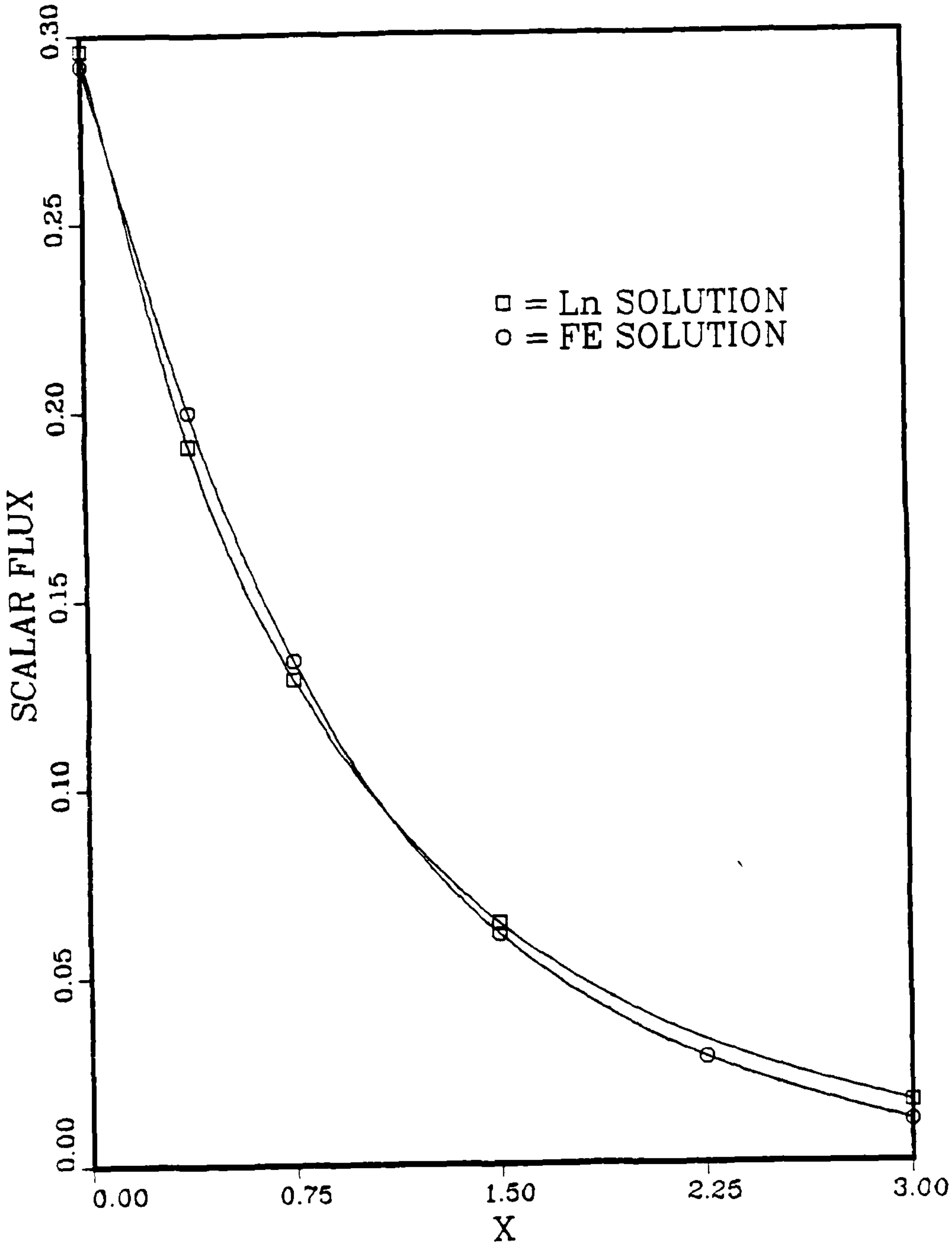
When scattering occurs, the variational integral is evaluated by integrating over space, angle, and scattered angle. To simplify this calculation triangles are constrained in this study to columns. Since cubic interpolating functions are used for flux, the scattering integrals involve the product of two cubics, or hexadics. Mapped in three dimensions, the local and nonlocal triangles create tetrahedra.

Two methods were tried to evaluate these integrals. Strict numerical evaluation with relations of accuracy up to Weddle's for n=6 did not obtain acceptable accuracy. Error with these techniques was cumulative, and refinement resulted in a loss of the global matrices positive definiteness with around 15 triangles per mean free path, prior to convergence. Numerical evaluation of the scattering integrals appears to hold no potential with the method's present formulation. Approximating the hexadic with another cubic gave better results, but did not guarantee positive definiteness. Solution accuracy was sufficiently verified against a Pn benchmark over the test domain. Convergence appears to occur with a reasonable but larger amount of mesh refinement than in the no scattering case. The number of computations needing to be performed may be excessive. Convergence is speeded by specifying angle

FIGURE 4–9

COMPARISON OF Ln AND FE SCALAR FLUX

LAMBERTIAN SOURCE, WITH C=.5



finite element angular flux was integrated over angle with the assistance of IMSL routine ICSCCU, cubic spline interpolation and

$$\text{Scalar Flux} = \frac{1}{2}\int_{-1}^{1} du\ \phi(x,u) \tag{4-22}$$

A comparison of the Ln and FE results for mesh D, with a depth of three mean free paths, is displayed in table 4-4, and graphed in figure 4-9.

x	0	.375	.75	1.5	2.25	3.0
Ln	.296	.191	.129	.064	¢	.016
FE	.292	.200	.134	.061	.028	.011

Table 4-4
Ln and Finite Element Comparison of Scalar Fluxes
Lambertian Source, c=.5, Mesh D, Fluxes and Derivatives Specified as Boundary Conditions

Agreement between the two codes is good. Differences are of the same order magnitude as the scattering error estimation previously done for this mesh. At x=3, the percentage difference is large, but the magnitude of the variation is small. The graph of figure 4-9 displays the close correlation between the two separate calculational results.

Computationally the method can be considered excessive. Mesh E, composed of 46 elements, requires over 4 minutes of CPU time on a Harris 800 computer. The correlation between Harris times and the Vax times of chapter 3 is unknown, but clearly the number of calculations has greatly increased. The 4 column 46 considered as entirely due to truncation of the polynomial series. Mesh D calculations, with a depth of three mean free paths, and no scattering show an average of 0.12% difference at u=1 from analytically computed angular flux. This represents the error from cubic approximation of flux, for the mesh, and degree of refinement under scrutiny. Comparing this to the scattering mesh D case of three mean free paths leaves a remainder of 2.65%, an approximate estimate of scattering integral evaluation error in this case.

Table 4-2 contains 3 instances where less refined meshes appear to give more accurate answers than a denser mesh. If penalties were exact they should indicate, as in the no scattering case, that better finite element fits can occur without necessarily observing steady convergence of nodal values to an "exact" answer.

Further indication of the finite element methods success comes from investigating the lambertian flux incident on the left boundary with c=1.0 . The angular flux in this scattering only medium of depth equal to three mean free paths reflects the hump predicted at x=0 of figure 4-5. The surface of angular flux, plotted in figure 4-8 shows that particles leak out both ends, and that angular flux is approaching isotropy as the region is penetrated.

Cited in Goff's thesis (2:67), were the benchmark case results of a transport code known as Ln. This a program recently developed as a P.H.D. dissertation by LCDR. Kirk A. Mathews (AFIT/GNE/85D). The output of this code is scalar flux, so the

CASE - integer reflecting the orientation of local and non local triangles w.r.t. each other

TIME - integer reflecting which half of case 2 and case 4 is being currently calculated

PTNODE - array storing the global numbering of a triangles finite element interpolation nodes

COLUMN - array storing the column each triangle belongs to, the top element of that column, and the number of elements the column posseses

Variables, Not Passed, by Subroutine, Requiring Definition

Subroutine SINFCN

SGT - matrix of interpolating function constants for the terahedral cubic (2-43)

M - array storing the 4x4 partitioned matrices of (2-42) and (2-43)

Subroutine SCATA and SCATB

W1,W2,W3,W4,W5,W6 - dimension 10 vectors storing local and non local flux, and its derivatives, at locations that are not triangular cubic interpolation nodes

F - array storing the twenty 10x10 matrices used as interpolation nodes for the three dimensional cubic

```
LI,1,2500
      PROGRAM FECUBE
* FINITE ELEMENT SOLUTION OF ONE SPEED TRANSPORT EQUATION IN
* SLAB GEOMETRY, ISOTROPIC SCATTER. CUBIC APPROXIMATION OF
* FLUX, CUBIC APPROXIMATION OF HEXADIC SCATTERING INTEGRAL

      PARAMETER (MNODE=151 , MNTRIA=50)

      DOUBLE PRECISION CORDND(MNODE,2),PHI(MNODE)
      DOUBLE PRECISION  AREAS(MNTRIA),MA(10,10)
       DOUBLE PRECISION  SC1(10,33),SR1(18),SC2(10,33),SR2(18)
       DOUBLE PRECISION  SC3(10,33),SR3(18),SC4(10,33),SR4(18)
       DOUBLE PRECISION  SC5(10,33),SR5(18),SC6(10,33),SR6(18)
       DOUBLE PRECISION  BC1(10,20),BR1(10)
       DOUBLE PRECISION  BC2(10,20),BR2(10),D1,D2
       DOUBLE PRECISION BC3(10,20),BR3(10),AS(MNODE*(MNODE-1)/2)
       DOUBLE PRECISION ML(MNTRIA,10,10)
       DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
       DOUBLE PRECISION MG(MNODE,MNODE)
       DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
       DOUBLE PRECISION  MB(10,10),MS(10,10),DRVS(10,2)
       DOUBLE PRECISION RANGE,SIGMAT,SIGMAS,PEN(MNTRIA)
       DOUBLE PRECISION G1,G2,G3,F1,F2,F3,A
       DOUBLE PRECISION  U1,U2,U3,X1,X2,X3
       DOUBLE PRECISION V(5,20),SGM(5,4,4)
       DOUBLE PRECISION E(4),F(4),G(4),H(5,20),SA(10,10),SB(10,10)
       INTEGER N,TRI,TRIP,NTRIA,CASE,TIME
       INTEGER PTNODE(MNTRIA,11),COLUMN(32,2)
       LOGICAL CHECK1
       COMMON MG,ML,NLM,NLI,LI,GT

       CHECK1 = .FALSE.

* READ INITIAL DATA
       CALL GDATA(NTRIA,N,PTNODE,COLUMN,CORDND,
      C   AREAS,RANGE,SIGMAT,SIGMAS,MA,BC1,BR1,BC2,BR2,BC3,BR3,
      C   SC1,SR1,SC2,SR2,SC3,SR3,SC4,SR4,SC5,SR5,SC6,SR6,
      C   V,SGM,PHI)


* RENUMBER THE MESH, GLOBALLY, AND LOCALLY PER FIGURE 2-4
       CALL CHGRID (PTNODE,CORDND,N,NTRIA)


* ZERO THE GLOBAL MATRIX
       DO 69 I=1,N
       DO 68 J=1,N
           MG(I,J)=0.0
 68      CONTINUE
 69     CONTINUE

* CALCULATE PARTICLE STREAMING,ABSORBING AND BOUNDARY TERMS
* ASSEMBLE INTO LOCAL MATRIX FOR A TRIANGLE, AND ASSEMBLE
* GLOBALLY
       DO 50 TRI=1,NTRIA
           U1=CORDND(PTNODE(TRI,1),2)
```

```
            U2=CORDND(PTNODE(TRI,4),2)
            U3=CORDND(PTNODE(TRI,7),2)
            X1=CORDND(PTNODE(TRI,1),1)
            X2=CORDND(PTNODE(TRI,4),1)
            X3=CORDND(PTNODE(TRI,7),1)
            A=AREAS(TRI)*2.0
            G1=(U2-U3)/A
            G2=(U3-U1)/A
            G3=(U1-U2)/A
            F1=(X3-X2)/A
            F2=(X1-X3)/A
            F3=(X2-X1)/A
            CALL INFCN(TRI,G1,G2,G3,F1,F2,F3)
            CALL BNDRY (U1,U2,U3,G1,G2,G3,BC1,BC2,BC3,BR1
     C           ,BR2,BR3,SIGMAT,MB,DRVS,AREAS,TRI)
            CALL STREAM(SC1,SR1,SC2,SR2,SC3,SR3,SC4,SR4,SC5,SR5,
     C                          SC6,SR6,MS,U1,U2,U3,G1,G2,G3,AREAS,TRI,
     C                             DRVS)
            CALL LMATRX(MA,MB,MS,AREAS,SIGMAT,TRI)
            CALL ASEMBL(PTNODE,TRI)
 50       CONTINUE

* CALCULATE SCATTERING CONTRIBUTION - FOR A TRIANGLE - FROM
* COLUMN TOP TO BOTTOM
        DO 150 TRI=1,NTRIA
            K=COLUMN(PTNODE(TRI,11),1)
            DO 125 TRIP=K,K-1+COLUMN(PTNODE(TRI,11),2)
                TIME=1
 130            CALL CASEDT(TRI,TRIP,CORDND,PTNODE,TIME,E,F,G,
     C                V6,CASE,U1,U2,U3,X1,X2,X3)
                CALL SINFCN(E,F,G,V,SGM,H)
                CALL SCATA(H,TRI,TRIP,CASE,TIME,SA,CORDND,PTNODE)
                CALL SCATB(U1,U2,U3,X1,X2,X3,TRI,TRIP,AREAS,H,
     C                            CASE,TIME,SB,CORDND,PTNODE)
                CALL NLMTRX(TRI,TRIP,SIGMAS,SIGMAT,
     C                             TIME,V6,SA,SB)
                IF (CASE.EQ.2.OR.CASE.EQ.4) THEN
                    IF (TIME.EQ.1) THEN
                        TIME=TIME+1
                        GO TO 130
                      ENDIF
                  ENDIF
                CALL SASMBL(PTNODE,TRI,TRIP)
 125          CONTINUE
 150      CONTINUE

* PUT GLOBAL MATRIX IN ITS QUADRATIC FORM
        DO 250 I=1,N
            DO 200 J=1,I
                MG(I,J)=(MG(I,J)+MG(J,I))/2.0
                MG(J,I)=MG(I,J)
 200          CONTINUE
 250      CONTINUE

* IF DESIRED, DIAGNOSTIC DATA CAN BE TURNED ON IN 'OUTPUT' HERE
        CALL OUTPUT(PHI,N,PTNODE,CORDND,NTRIA,CHECK1
```

```
     C                ,PEN,SIGMAS,RANGE,SIGMAT)
      CHECK1 = .TRUE.
* APPLY THE BOUNDARY CONDITIONS
      CALL BNDCND(CORDND,PHI,N,NTRIA,RANGE)
* PLACE GLOBAL MATRIX IN BAND STORAGE FOR IMSL
      K=1
      DO 350 I=1,N
          DO 300 J=1,I
              AS(K)=MG(I,J)
              K=K+1
300     CONTINUE
350     CONTINUE
* SOLVE THE SET OF LINEAR EQUATIONS
      CALL LEQT1P(AS,1,N,PHI,MNODE,IDGT,D1,D2,IER)
      PRINT*,'IER IS ...',IER
* CALCULATE PENALTIES, AND PRINT OUT RESULTS
      CALL PENLTY(PHI,PTNODE,PEN,NTRIA,COLUMN)
      CALL OUTPUT(PHI,N,PTNODE,CORDND,NTRIA,CHECK1
     C                ,PEN,SIGMAS,RANGE,SIGMAT)
      END
************************************************************
* GATHER INITIAL DATA - READS THREE DATA FILES
* MESH - GRID DATA (APPENDIX F)
* CODATA - COEFFICIENTS OF LOCAL MATRICES (APPENDICES B,C,D)
* SDATA - CONSTANTS. FIVE OF THE PARTITIONED MATRICES OF 2-42
*         AND 2-43, AS WELL AS THE INTEGRALS OF BASIS POLYNOMIALS
      SUBROUTINE GDATA(NTRIA,N,PTNODE,COLUMN,CORDND,
     C   AREAS,RANGE,SIGMAT,SIGMAS,MA,BC1,BR1,BC2,BR2,BC3,BR3,
     C   SC1,SR1,SC2,SR2,SC3,SR3,SC4,SR4,SC5,SR5,SC6,SR6,
     C    V,SGM,PHI)
      PARAMETER (MNODE=151 , MNTRIA=50)
      DOUBLE PRECISION  CORDND(MNODE,2)
      DOUBLE PRECISION  AREAS(MNTRIA),MA(10,10)
      DOUBLE PRECISION  SC1(10,33),SR1(18),SC2(10,33),SR2(18)
      DOUBLE PRECISION  SC3(10,33),SR3(18),SC4(10,33),SR4(18)
      DOUBLE PRECISION  SC5(10,33),SR5(18),SC6(10,33),SR6(18)
      DOUBLE PRECISION  BC1(10,20),BR1(10)
      DOUBLE PRECISION  BC2(10,20),BR2(10)
      DOUBLE PRECISION  BC3(10,20),BR3(10)
      DOUBLE PRECISION V(5,20),SGM(5,4,4)
      DOUBLE PRECISION  RANGE,SIGMAT,SIGMAS,PHI(MNODE)
      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
```

```
      INTEGER N,NTRIA,TRI,NB
      INTEGER PTNODE(MNTRIA,11),COLUMN(32,2)
      CHARACTER TRASH*21
      COMMON MG,ML,NLM,NLI,LI,GT

      OPEN(15,FILE='MESH',STATUS='OLD')
      REWIND 15

      READ(15,'(A16)') TRASH
      READ(15,'(3(1X,I7))') NTRIA,N,NCOL
      READ(15,'(1X)')


      READ(15,'(A16)') TRASH
      READ(15,'(3(1X,F7.3))') RANGE,SIGMAT,SIGMAS
      RANGE=RANGE*SIGMAT
      READ(15,'(1X)')


      READ(15,'(A16)') TRASH
      DO 60 I=1,NTRIA
          READ(15,'(1X,I7,8X,4(1X,I7))') TRI,(PTNODE(I,J),J=1,4)
60      CONTINUE
      READ(15,'(1X)')

      READ(15,'(A16)') TRASH
      DO 70 I=1,NCOL
          READ(15,'(3(1X,I7,8X))') TRI,(COLUMN(I,J),J=1,2)
70      CONTINUE
      READ(15,'(1X)')

      READ(15,'(A16)') TRASH
      DO 80 I=1,N
       READ(15,'(1X,I7,8X,2(2X,F7.3))') NODE,(CORDND(I,J),J=1,2)
          CORDND(I,1)=CORDND(I,1)*RANGE
80      CONTINUE
      READ(15,'(1X)')

       DO 82 I=1,3*N+NTRIA
          PHI(I)=0.0
82      CONTINUE
      READ(15,'(A16)') TRASH
      READ(15,'(I7)') NB
      DO 83 I=1,NB
          READ(15,'(1X,I7,8X,E11.5)') J,PHI(J)
83      CONTINUE

      CLOSE (15)

      DO 90 TRI=1,NTRIA
      U3=CORDND(PTNODE(TRI,3),2)
      U2=CORDND(PTNODE(TRI,2),2)
      X2=CORDND(PTNODE(TRI,2),1)
      X1=CORDND(PTNODE(TRI,1),1)
      AREAS(TRI)=ABS(.5*(U3-U2)*(X2-X1))
      IF (AREAS(TRI).LT.1.0E-15) THEN
```

```
            PRINT*,'AREA OF ZERO IN ELEMENT',TRI
          ENDIF
 90       CONTINUE

        OPEN(16,FILE='CODATA',STATUS='OLD')
        REWIND 16
        DO 100 I=1,10
            READ (16,'(10(1X,F5.1))') (MA(I,J),J=1,10)
 100      CONTINUE



        DO 110 I=1,10
            READ (16,'(10(1X,F5.1))') (BC1(I,J),J=1,10)
            READ (16,'(10(1X,F5.1))') (BC1(I,J),J=11,20)
 110      CONTINUE
        READ (16,'(10(1X,F5.1))') (BR1(I),I=1,10)

        DO 120 I=1,10
            READ (16,'(10(1X,F5.1))') (BC2(I,J),J=1,10)

            READ (16,'(10(1X,F5.1))') (BC2(I,J),J=11,20)
 120      CONTINUE
        READ (16,'(10(1X,F5.1))') (BR2(I),I=1,10)

        DO 130 I=1,10
            READ(16,'(10(1X,F5.1))') (BC3(I,J),J=1,10)
            READ(16,'(10(1X,F5.1))') (BC3(I,J),J=11,20)
 130      CONTINUE
        READ(16,'(10(1X,F5.1))') (BR3(I),I=1,10)

        DO 140 I=1,10
            READ (16,4200) (SC1(I,J),J=1,10)
            READ (16,4200) (SC1(I,J),J=11,20)
            READ (16,4200) (SC1(I,J),J=21,30)
            READ (16,4100) (SC1(I,J),J=31,33)
 140      CONTINUE
        READ (16,4200) (SR1(I),I=1,10)
        READ (16,4000) (SR1(I),I=11,18)

        DO 150 I=1,10
            READ (16,4200) (SC2(I,J),J=1,10)
            READ (16,4200) (SC2(I,J),J=11,20)
            READ (16,4200) (SC2(I,J),J=21,30)
            READ (16,4100) (SC2(I,J),J=31,33)
 150      CONTINUE
        READ (16,4200) (SR2(I),I=1,10)
        READ (16,4000) (SR2(I),I=11,18)

        DO 160 I=1,10
            READ (16,4200) (SC3(I,J),J=1,10)
            READ (16,4200) (SC3(I,J),J=11,20)
            READ (16,4200) (SC3(I,J),J=21,30)
            READ (16,4100) (SC3(I,J),J=31,33)
 160      CONTINUE
        READ (16,4200) (SR3(I),I=1,10)
```

```
      READ (16,4000) (SR3(I),I=11,18)

      DO 180 I=1,10
          READ (16,4200) (SC4(I,J),J=1,10)
          READ (16,4200) (SC4(I,J),J=11,20)
          READ (16,4200) (SC4(I,J),J=21,30)
          READ (16,4100) (SC4(I,J),J=31,33)
180     CONTINUE
      READ (16,4200) (SR4(I),I=1,10)
      READ (16,4000) (SR4(I),I=11,18)

      DO 190 I=1,10
          READ (16,4200) (SC5(I,J),J=1,10)
          READ (16,4200) (SC5(I,J),J=11,20)
          READ (16,4200) (SC5(I,J),J=21,30)
          READ (16,4100) (SC5(I,J),J=31,33)
190     CONTINUE
      READ (16,4200) (SR5(I),I=1,10)
      READ (16,4000) (SR5(I),I=11,18)

      DO 200 I=1,10
          READ (16,4200) (SC6(I,J),J=1,10)
          READ (16,4200) (SC6(I,J),J=11,20)
          READ (16,4200) (SC6(I,J),J=21,30)
          READ (16,4100) (SC6(I,J),J=31,33)
200     CONTINUE
      READ (16,4200) (SR6(I),I=1,10)
      READ (16,4000) (SR6(I),I=11,18)
      CLOSE (16)

      OPEN(17,FILE='SDATA',STATUS='OLD')
      REWIND 17
      DO 210 I=1,5
          READ(17,4200) (V(I,J),J=1,10)
          READ(17,4200) (V(I,J),J=11,20)
210     CONTINUE
      DO 230 K=1,5
          DO 220 I=1,4
              READ(17,4300) (SGM(K,I,J),J=1,4)
220       CONTINUE
230     CONTINUE
      DO 260 K=1,4
          DO 250 I=1,4
              DO 240 J=1,4
                  SGM(K,I,J)=SGM(K,I,J)/27.0
240           CONTINUE
250       CONTINUE
260     CONTINUE
      CLOSE (17)


4000  FORMAT (8(1X,F6.1))
4100  FORMAT (3(1X,F6.1))
4200  FORMAT (10(1X,F6.1))
4300  FORMAT (4(1X,F6.1))

```

```
      END


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

* RENUMBERS THE GRID - SINCE MESH IS NUMBERED DIFFERENTLY
* FOR EACH FIT (LINEAR,QUADRATIC,AND CUBIC) ALLOWS THE DATA
* FILE 'MESH' TO REMAIN SIMPLE, AND BE USED BY ALL THREE
* CODES - NUMBERING IS AS PER FIGURE 2-4

      SUBROUTINE CHGRID (PTNODE,CORDND,N,NTRIA)

      PARAMETER (MNODE=151 , MNTRIA=50)

      DOUBLE PRECISION  CORDND(MNODE,2),D,E
      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      INTEGER N,NTRIA,TRI,A,B,C
      INTEGER PTNODE(MNTRIA,11)
      COMMON MG,ML,NLM,NLI,LI,GT


      DO 100 I=1,NTRIA
        K=(3*N)+I
       CORDND(K,1)=(1.0/3.0)*(CORDND(PTNODE(I,1),1) + CORDND(
     C          PTNODE(I,2),1) + CORDND(PTNODE(I,3),1))
       CORDND(K,2)=(1.0/3.0)*(CORDND(PTNODE(I,1),2) + CORDND(
     C              PTNODE(I,2),2) + CORDND(PTNODE(I,3),2))
100    CONTINUE



      DO 110 TRI=1,NTRIA
      A=PTNODE(TRI,1)
      B=PTNODE(TRI,2)
      C=PTNODE(TRI,3)
      PTNODE(TRI,11)=PTNODE(TRI,4)
      PTNODE(TRI,1)=3*A-2
      PTNODE(TRI,2)=3*A-1
      PTNODE(TRI,3)=3*A
      PTNODE(TRI,4)=3*B-2
      PTNODE(TRI,5)=3*B-1
      PTNODE(TRI,6)=3*B
      PTNODE(TRI,7)=3*C-2
      PTNODE(TRI,8)=3*C-1
      PTNODE(TRI,9)=3*C
      PTNODE(TRI,10)=3*N+TRI
110   CONTINUE


      DO 120 I=N,1,-1
      D=CORDND(I,1)
      E=CORDND(I,2)
      K=3*I-2
```

```
      CORDND(K,1)=D
      CORDND(K,2)=E
      CORDND(K+1,1)=D
      CORDND(K+1,2)=E
      CORDND(K+2,1)=D
      CORDND(K+2,2)=E
120     CONTINUE

      N=3*N + NTRIA



      END




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

* FIND THE MATRIX GT, OF (2-34)

      SUBROUTINE INFCN(TRI,G1,G2,G3,F1,F2,F3)

      PARAMETER (MNODE=151 , MNTRIA=50)
      DOUBLE PRECISION  GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      DOUBLE PRECISION  G1,G2,G3,F1,F2,F3,F
      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      INTEGER TRI
      COMMON MG,ML,NLM,NLI,LI,GT

      DO 550 I=1,10
          DO 500 J=1,10
              GT(TRI,I,J)=0.0
500         CONTINUE
550     CONTINUE

      F=G2*F3-G3*F2
      GT(TRI,1,1)=1.0
      GT(TRI,2,1)=3*(G3*F1-G1*F3)/F
      GT(TRI,2,2)=F3/F
      GT(TRI,2,3)=-G3/F
      GT(TRI,3,1)=3*(G1*F2-G2*F1)/F
      GT(TRI,3,2)=-F2/F
      GT(TRI,3,3)=G2/F

      F=G3*F1-G1*F3
      GT(TRI,4,4)=1.0
      GT(TRI,5,4)=3*(G1*F2-G2*F1)/F
      GT(TRI,5,5)=F1/F
      GT(TRI,5,6)=-G1/F
      GT(TRI,6,4)=3*(G2*F3-G3*F2)/F
      GT(TRI,6,5)=-F3/F
      GT(TRI,6,6)=G3/F

```

```
      F=G1*F2-G2*F1
      GT(TRI,7,7)=1.0
      GT(TRI,8,7)=3*(G2*F3-G3*F2)/F
      GT(TRI,8,8)=F2/F
      GT(TRI,8,9)=-G2/F
      GT(TRI,9,7)=3*(G3*F1-G1*F3)/F
      GT(TRI,9,8)=-F1/F
      GT(TRI,9,9)=G1/F

      GT(TRI,10,10)=27.0

      DO 130 I=1,7,3
          DO 120 J=I,I+2
              GT(TRI,10,I)=GT(TRI,10,I)-GT(TRI,J,I)
              GT(TRI,10,I+1)=GT(TRI,10,I+1) -GT(TRI,J,I+1)
              GT(TRI,10,I+2)=GT(TRI,10,I+2) -GT(TRI,J,I+2)
120         CONTINUE
130     CONTINUE

      END


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

* BOUNDARY MATRIX - ASSEMBLAGE EXPLAINED IN APPENDIX C

      SUBROUTINE BNDRY (U1,U2,U3,G1,G2,G3,BC1,BC2,BC3,BR1
     C    ,BR2,BR3,SIGMAT,MB,D,AREAS,TRI)

      PARAMETER (MNODE=151 , MNTRIA=50)
      DOUBLE PRECISION  U1,U2,U3,G1,G2,G3,F
      DOUBLE PRECISION  BC1(10,20),BC2(10,20),BC3(10,20)
      DOUBLE PRECISION BC(10,20)
      DOUBLE PRECISION  BR1(10),BR2(10),BR3(10),BR(10)
      DOUBLE PRECISION  MB(10,10),D(10,2),SIGMAT,AREAS(MNTRIA)
      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      INTEGER TRI
      COMMON MG,ML,NLM,NLI,LI,GT

* ASSEMBLE THE DERIVATIVE MATRIX, TO BE OVERLAYED
* NOTE - PASSED AS DRVS
      D(1,1)=G1
      D(2,1)=G1
      D(3,1)=G1
      D(4,1)=G2
      D(5,1)=G2
      D(6,1)=G2
      D(7,1)=G3
      D(8,1)=G3
      D(9,1)=G3
      D(10,1)=G1
      D(1,2)=0.0
      D(2,2)=G2
```

```
      D(3,2)=G3
      D(4,2)=0.0
      D(5,2)=G3
      D(6,2)=G1
      D(7,2)=0.0
      D(8,2)=G1
      D(9,2)=G2
      D(10,2)=G2

* MULTIPLY THE COEFICIENT MATRICES AND ROWS BY APPROPRIATE
* U VALUE - THEN SUM
      F=SIGMAT*4.0*AREAS(TRI)/40320.0
      DO 100 I=1,10
          BR(I)=(U1*BR1(I)+U2*BR2(I)+U3*BR3(I))*F
          DO 50 J=1,20
              BC(I,J)=(U1*BC1(I,J)+U2*BC2(I,J)+U3*BC3(I,J))*F
50            CONTINUE
100       CONTINUE

* OVERLAY THE DERIVATIVE MATRIX TO FORM MB
      DO 250 I=1,10
          DO 200 J=1,10
              MB(I,J)=BC(I,(2*J)-1)*D(I,1)+BC(I,2*J)*D(I,2)
200           CONTINUE
250       CONTINUE

* AUGMENT THE LAST ROW
      DO 300 I=1,10
          MB(10,I)=MB(10,I)+G3*BR(I)
300       CONTINUE

* PLACE IN ITS QUADRATIC FORM
      DO 400 I=1,10
          DO 350 J=1,I
              MB(I,J)=(MB(I,J)+MB(J,I))/2.0
              MB(J,I)=MB(I,J)
350           CONTINUE
400       CONTINUE

      END


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

* STREAMING MATRIX - ASSEMBLAGE EXPLAINED IN APPENDIX D

      SUBROUTINE STREAM (SC1,SR1,SC2,SR2,SC3,SR3,SC4,SR4,SC5,SR5,
     C                   SC6,SR6,MS,U1,U2,U3,G1,G2,G3,AREAS,TRI,
     C                     DRVS)

      PARAMETER (MNODE=151 , MNTRIA=50)

      DOUBLE PRECISION  AREAS(MNTRIA)
      DOUBLE PRECISION  SC1(10,33),SR1(18),SC2(10,33),SR2(18)
      DOUBLE PRECISION  SC3(10,33),SR3(18),SC4(10,33),SR4(18)
      DOUBLE PRECISION  SC5(10,33),SR5(18),SC6(10,33),SR6(18)
```

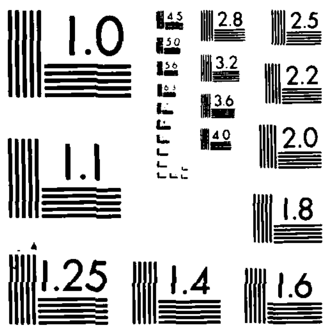
```
      DOUBLE PRECISION  GG(3),DS(10,33),SC(10,33),A,B,C,D,E,F,G
      DOUBLE PRECISION  SR(18),DR(18)
      DOUBLE PRECISION  MS(10,10),DRVS(10,2)
      DOUBLE PRECISION  G1,G2,G3,U1,U2,U3
      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      INTEGER TRI
      COMMON MG,ML,NLM,NLI,LI,GT

* ASSEMBLE THE MATRIX OF DERIVATIVES
      GG(1)=G1
      GG(2)=G2
      GG(3)=G3

* FILL IN COLUMNS 1,2,3,12,13,14,23,24,25 OF DS
      DO 110 I=1,7,3
          L=1+(I-1)/3
          K=1+((I-1)*11/3)
          DO 100 J=1,10
              DS(J,K)=DRVS(J,1)*GG(L)
              DS(J,K+1)=DRVS(J,2)*GG(L)
              DS(J,K+2)=0.0
100         CONTINUE
110     CONTINUE

      DS(10,3)=G1*G3
      DS(10,14)=G2*G3
      DS(10,25)=G3*G3

* FILL IN REMAINING COLUMNS
      DO 120 J=1,10
          DS(J,4)=DRVS(J,1)*G1
          DS(J,5)=DRVS(J,2)*G1
          DS(J,6)=DRVS(J,1)*G2
          DS(J,7)=DRVS(J,2)*G2
          DS(J,8)=DS(J,4)
          DS(J,9)=DS(J,5)
          DS(J,10)=DRVS(J,1)*G3
          DS(J,11)=DRVS(J,2)*G3
          DS(J,15)=DS(J,6)
          DS(J,16)=DS(J,7)
          DS(J,17)=DS(J,10)
          DS(J,18)=DS(J,11)
          DS(J,19)=DS(J,6)
          DS(J,20)=DS(J,7)
          DS(J,21)=DS(J,4)
          DS(J,22)=DS(J,5)
          DS(J,26)=DS(J,10)
          DS(J,27)=DS(J,11)
          DS(J,28)=DS(J,4)
          DS(J,29)=DS(J,5)
          DS(J,30)=DS(J,10)
          DS(J,31)=DS(J,11)
          DS(J,32)=DS(J,6)
```

```
              DS(J,33)=DS(J,7)
 120       CONTINUE

        DS(10,6)=G1*G3
        DS(10,7)=0.0
        DS(10,10)=G1*G3
        DS(10,11)=0.0
        DS(10,17)=G2*G3
        DS(10,18)=0.0
        DS(10,21)=G2*G3
        DS(10,22)=0.0
        DS(10,28)=G3*G3
        DS(10,29)=0.0
        DS(10,32)=G3*G3
        DS(10,33)=0.0

        DR(1)=G2*G2
        DR(2)=G2*G3
        DR(3)=DR(2)
        DR(4)=G3*G3
        DR(5)=G1*G3
        DR(6)=DR(4)
        DR(7)=G1*G1
        DR(8)=DR(5)
        DR(9)=DR(7)
        DR(10)=G1*G2
        DR(11)=DR(10)
        DR(12)=DR(1)
        DR(13)=DR(7)
        DR(14)=DR(10)
        DR(15)=DR(5)
        DR(16)=DR(1)
        DR(17)=DR(2)
        DR(18)=DR(4)

* MULTIPLY THE COEFICIENT MATRICES AND ROWS BY THE
* APPROPRIATE U'S - THEN SUM
        A=U1*U1
        B=U2*U2
        C=U3*U3
        D=U1*U2*2.0
        E=U2*U3*2.0
        F=U1*U3*2.0
        G=2.0*AREAS(TRI)/40320.0
        DO 140 I=1,10
            DO 130 J=1,33
                SC(I,J)=(SC1(I,J)*A + SC2(I,J)*B + SC3(I,J)*C +
      C                      SC4(I,J)*D + SC5(I,J)*E + SC6(I,J)*F)*G
 130         CONTINUE
 140       CONTINUE
        DO 150 I=1,18
            SR(I)=(SR1(I)*A + SR2(I)*B + SR3(I)*C +
      C                 SR4(I)*D + SR5(I)*E + SR6(I)*F)*G
 150       CONTINUE

* COMPUTE COLUMNS 1,4,7 OF STREAMING MATRIX
```

```
      DO 170 I=1,10
          DO 160 J=1,7,3
              K=1 + ((J-1)*11/3)
              MS(I,J)=SC(I,K)*DS(I,K) + SC(I,K+1)*DS(I,K+1)
     C                  + SC(I,K+2)*DS(I,K+2)
160           CONTINUE
170       CONTINUE

* COLUMNS 2,5, AND 8
      DO 190 I=1,10
          DO 180 J=2,8,3
              K=4+((J-2)*11/3)
              MS(I,J)=SC(I,K)*DS(I,K) + SC(I,K+1)*DS(I,K+1)
     C             + SC(I,K+2)*DS(I,K+2) + SC(I,K+3)*DS(I,K+3)
              K=K+4
              MS(I,J+1)=SC(I,K)*DS(I,K) + SC(I,K+1)*DS(I,K+1)
     C             + SC(I,K+2)*DS(I,K+2) + SC(I,K+3)*DS(I,K+3)
180           CONTINUE
190       CONTINUE
* AUGMENT THE LAST ROW
      DO 200 I=2,8,3
          K=1 + 4*(I-2)/3
          MS(10,I)=MS(10,I)+SR(K)*DR(K)+SR(K+1)*DR(K+1)
          MS(10,I+1)=MS(10,I+1)+SR(K+2)*DR(K+2)+SR(K+3)*DR(K+3)
200       CONTINUE

      MS(10,10)=0.0
      DO 210 I=13,18
          MS(10,10)=MS(10,10)+SR(I)*DR(I)
210       CONTINUE

* FORM COLUMN 10 WITH SYMMETRY
      DO 220 I=1,9
          MS(I,10)=MS(10,I)
220       CONTINUE

      END


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

* LOCAL MATRIX - MULTIPLY ABSORBING,BOUNDARY AND STREAMING
* BY APPROPRIATE CONSTANTS - AND SUM
* PRE AND POST MULTIPLY BY GT TO COMPLETELY PREPARE FOR
* GLOBAL ASSEMBLAGE

      SUBROUTINE LMATRX(MA,MB,MS,AREAS,SIGMAT,TRI)

      PARAMETER (MNODE=151 , MNTRIA=50)
      DOUBLE PRECISION  AREAS(MNTRIA),MA(10,10)
      DOUBLE PRECISION MB(10,10),MS(10,10),MT(10,10)
      DOUBLE PRECISION SIGMAT,F
      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
```

AD-A159 245 A FINITE ELEMENT SOLUTION OF THE TRANSPORT EQUATION(U) 2/3
AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL
OF ENGINEERING F A TARANTINO MAR 85 AFIT/GNE/PH/85M-19

UNCLASSIFIED F/G 12/1 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

```
      INTEGER TRI
      COMMON MG,ML,NLM,NLI,LI,GT

      F=SIGMAT*SIGMAT*2.0*AREAS(TRI)/40320.0


      DO 830 I=1,10
          DO 820 J=1,10
              ML(TRI,J,I)=MB(J,I) + MA(J,I)*F + MS(J,I)
820         CONTINUE
830     CONTINUE

      DO 860 I=1,10
          DO 850 J=1,10
              MT(I,J)=0.0
              DO 840 K=1,10
                  MT(I,J)=MT(I,J) + GT(TRI,K,I)*ML(TRI,K,J)
840             CONTINUE
850         CONTINUE
860     CONTINUE
      DO 890 I=1,10
          DO 880 J=1,10
              ML(TRI,I,J)=0.0
              DO 870 K=1,10
                  ML(TRI,I,J)=ML(TRI,I,J) + MT(I,K)*GT(TRI,K,J)
870             CONTINUE
880         CONTINUE
890     CONTINUE

      END


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

* ASSEMBLE LOCAL TERMS GLOBALLY

      SUBROUTINE ASEMBL(PTNODE,TRI)

      PARAMETER (MNODE=151 , MNTRIA=50)

      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      INTEGER PTNODE(MNTRIA,11),TRI,R(10)
      COMMON MG,ML,NLM,NLI,LI,GT

      DO 900 I=1,10
          R(I)=PTNODE(TRI,I)
900    CONTINUE

      DO 920 I=1,10
          DO 910 J=1,10
              MG(R(I),R(J))=MG(R(I),R(J)) + ML(TRI,I,J)
910        CONTINUE
920    CONTINUE
```

```
      END


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

* INSURE FLUXES (AND IN THIS CASE U-CURRENTS) ARE AS SPECIFIED
* ON BOUNDARIES - IF FLUXES ONLY ARE TO BE SPECIFIED DELETE
* THE I+2 TERMS MODIFICATION

      SUBROUTINE BNDCND(CORDND,PHI,N,NTRIA,RANGE)

      PARAMETER (MNODE=151 , MNTRIA=50)

      DOUBLE PRECISION  CORDND(MNODE,2),PHI(MNODE)
      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      DOUBLE PRECISION RANGE
      INTEGER N,NTRIA
      COMMON MG,ML,NLM,NLI,LI,GT

      DO 120 I=1,N-NTRIA,3
          IF (CORDND(I,1).EQ.0.0.AND.CORDND(I,2).GE.0.0) THEN
              MG(I,I)=MG(I,I)*1.0E+20
              PHI(I)=PHI(I)*MG(I,I)
              MG(I+2,I+2)=MG(I+2,I+2)*1.0E+20
              PHI(I+2)=PHI(I+2)*MG(I+2,I+2)
           ELSE
           IF (CORDND(I,1).EQ.RANGE.AND.CORDND(I,2).LE.0.0) THEN
                  MG(I,I)=MG(I,I)*1.0E+20
                  PHI(I)=PHI(I)*MG(I,I)
                  MG(I+2,I+2)=MG(I+2,I+2)*1.0E+20
                  PHI(I+2)=PHI(I+2)*MG(I+2,I+2)
                ENDIF
            ENDIF
120     CONTINUE

      END


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

* CALCULATE VALUE OF VARIATIONAL INTEGRAL OVER AN ELEMENT

      SUBROUTINE PENLTY(PHI,PTNODE,PEN,NTRIA,COLUMN)

      PARAMETER (MNODE=151 , MNTRIA=50)
      DOUBLE PRECISION  ML(MNTRIA,10,10),PHI(MNODE),PEN(MNTRIA)
      DOUBLE PRECISION P(10),L(10)
      DOUBLE PRECISION S(10),F
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      INTEGER TRI,NTRIA,PTNODE(MNTRIA,11)
```

```
      INTEGER TRIP,COLUMN(32,2)
      COMMON MG,ML,NLM,NLI,LI,GT

      DO 100 TRI=1,NTRIA
* LOCAL MATRIX CONTRIBUTION
      DO 20 I=1,10
          P(I)=PHI(PTNODE(TRI,I))
20          CONTINUE
      DO 40 I=1,10
          L(I)=0.0
          DO 30 J=1,10
              L(I)=L(I) + ML(TRI,I,J)*P(J)
30              CONTINUE
40          CONTINUE
      PEN(TRI)=0.0
      DO 50 I=1,10
          PEN(TRI)=PEN(TRI) + L(I)*P(I)
50          CONTINUE
* SUM OF NON LOCAL MATRICES CONTRIBUTIONS
      K=COLUMN(PTNODE(TRI,11),1)
      DO 70 TRIP=K,K-1+COLUMN(PTNODE(TRI,11),2)
          DO 55 I=1,10
          S(I)=PHI(PTNODE(TRIP,I))
55        CONTINUE
          DO 65 I=1,10
              L(I)=0.0
              DO 60 J=1,10
              L(I)=L(I) + NLM(TRI,TRIP,I,J)*S(J)
60              CONTINUE
65          CONTINUE
          DO 68 I=1,10
            PEN(TRI)=PEN(TRI)+L(I)*P(I)
68          CONTINUE
70        CONTINUE
      PEN(TRI)=.5*PEN(TRI)
100     CONTINUE

      END

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

* PRINT OUTPUT - COMPARE FE SOLUTION WITH Pn OR ANALYTICAL
* IF NO SCATTER

      SUBROUTINE OUTPUT(PHI,N,PTNODE,CORDND,NTRIA,
     C                         CHECK1,PEN,SIGMAS,RANGE,SIGMAT)

      PARAMETER (MNODE=151 , MNTRIA=50)
      DOUBLE PRECISION  PHI(MNODE),CORDND(MNODE,2)
      DOUBLE PRECISION  PEN(MNTRIA),PENTOT
      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      DOUBLE PRECISION RANGE,SIGMAT,TPEN
      INTEGER PTNODE(MNTRIA,11),TRI,N,NTRIA
```

```
      LOGICAL CHECK1
      COMMON MG,ML,NLM,NLI,LI,GT

      IF (CHECK1) THEN

      PRINT*,'NTRIA     N     SIGMAS'
      WRITE (*,4999) NTRIA,N,SIGMAS
 4999 FORMAT (3X,I3,5X,I3,5X,F6.3)
      PRINT*,'RANGE IS....',RANGE

      PRINT*,'NODAL VALUES OF THE FLUX'
      J=N-NTRIA
      DO 100 I=1,J,6
          K=1+(I-1)/3
          WRITE(*,6010) K,PHI(I),K+1,PHI(I+3)
 6010     FORMAT(2(2X,I3,3X,F9.4))
 100    CONTINUE

      PRINT*,'ELEMENT PENALTY VALUES'
      PENTOT=0.0
      TPEN=0.0
      DO 110 I=1,NTRIA,2
          WRITE(*,6221) I,PEN(I),I+1,PEN(I+1)
          PENTOT=PENTOT + ABS(PEN(I)) + ABS(PEN(I+1))
          TPEN=TPEN + PEN(I) + PEN(I+1)
 110    CONTINUE
      PRINT*,'TOTAL PENALTY .... AND SUM OF ABS(PENALTY) ARE ..'
      WRITE(*,6222) TPEN,PENTOT
 6221 FORMAT (2(2X,I3,5X,E11.5))
 6222 FORMAT (2(10X,E11.5))

* COMPARE FE SOLUTION WITH APPROPRIATE BENCHMARK
      IF (SIGMAS.EQ.0.0) THEN
          CALL ANALY(PHI,CORDND,SIGMAT,N,NTRIA,RANGE)
        ELSE
          CALL PN(PHI,CORDND,SIGMAS,N,NTRIA,RANGE)
        ENDIF

      ELSE
* IF DESIRED TURN ON DIAGNOSTIC OUTPUT HERE
      GO TO 301
      PRINT*, 'MESH DEFINITION'
      PRINT*, '  TRIANGLE     GLOBAL NODES'
      DO 50 TRI=1,NTRIA
          WRITE(*,6050) TRI,(PTNODE(TRI,I),I=1,7,3)
 6050     FORMAT(4X,I3,8X,3(I3,3X))
 50     CONTINUE

      PRINT*
      PRINT*,'    NODE      COORDINATES (X,U)'
      DO 60 I=1,N-NTRIA,3
          K=(I+2)/3
          WRITE(*,6060) K,(CORDND(I,J),J=1,2)
 6060     FORMAT(4X,I3,7X,2(F7.3,3X))
 60     CONTINUE

```

```
      PRINT*
      PRINT*,'GLOBAL MATRIX'
      PRINT*
      DO 300 I=1,N
          WRITE(*,6210) (MG(I,J),J=1,N)
6210      FORMAT(1X,":",16(1X,F6.3))
          WRITE(*,6220)
6220      FORMAT(1X,":")
300     CONTINUE
301   ENDIF

350   END

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

* ASSEMBLE SCATTERING MATRICES (NON LOCAL) - A SEPARATE
* SUBROUTINE IS USED BECAUSE DIMENSIONS OF NLM ARE DIFFERENT
* THAN ML

      SUBROUTINE SASMBL(PTNODE,TRI,TRIP)

      PARAMETER (MNODE=151 , MNTRIA=50)
      DOUBLE PRECISION  ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      INTEGER PTNODE(MNTRIA,11),TRI,R(10),TRIP
      INTEGER L(10)
      COMMON MG,ML,NLM,NLI,LI,GT

      DO 900 I=1,10
          R(I)=PTNODE(TRI,I)
          L(I)=PTNODE(TRIP,I)
900    CONTINUE

      DO 920 I=1,10
          DO 910 J=1,10
              MG(R(I),L(J))=MG(R(I),L(J)) + NLM(TRI,TRIP,I,J)
910        CONTINUE
920    CONTINUE
      END



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

* COMPARE FE SOLUTION TO ANALYTICAL IN THE CASE OF NO SCATTER

      SUBROUTINE ANALY(PHI,CORDND,SIGMAT,N,NTRIA,RANGE)

      PARAMETER (MNODE=151 , MNTRIA=50)
      DOUBLE PRECISION CORDND(MNODE,2),PHI(MNODE),A,RANGE
      DOUBLE PRECISION PERC,TPERC
      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
```

```
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      INTEGER N,NTRIA
      COMMON MG,ML,NLM,NLI,LI,GT

      TPERC=0.0
      K=0
      PRINT*,'  COORDINATES       CURRENTS     FIN ELEM'
      PRINT*,'     X        U       X       U      FLUX      FLUX    % DIFF'
      DO 100 I=1,N-NTRIA,3
          IF (CORDND(I,2).GT.0.0) THEN
              A=CORDND(I,2)*EXP(-SIGMAT/CORDND(I,2)*CORDND(I,1))
              PERC=100*ABS(PHI(I)-A)/A
            WRITE(*,5002) CORDND(I,1),CORDND(I,2),PHI(I+1),PHI(I+2)
     C              ,PHI(I),A,PERC
              TPERC=TPERC+PERC
              IF (CORDND(I,1).NE.0.0.AND.CORDND(I,1).NE.RANGE) THEN
                  K=K+1
                ENDIF
            ENDIF
100     CONTINUE
      PRINT*,'AVERAGE % DIFFERENCE IS ..',TPERC/K
      D=NTRIA*.5/RANGE
      PRINT*,'FOR AN AVG. OF',D,'TRIANGLES PER MFP FOR U>0 '

5002  FORMAT(6(2X,F6.3),2X,F6.2)

      END
***************************************************************

* CALCULATE THE NON LOCAL MATRIX FOR TRIANGLE TRI
* INTO TRIANGLE TRIP

      SUBROUTINE NLMTRX(TRI,TRIP,SIGMAS,SIGMAT,
     C      TIME,V6,SA,SB)

      PARAMETER (MNODE=151 , MNTRIA=50)
      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      DOUBLE PRECISION SIGMAS,SIGMAT,SA(10,10),SB(10,10)
      DOUBLE PRECISION V6
      INTEGER TRI,TRIP,TIME
      COMMON MG,ML,NLM,NLI,LI,GT

* CALCULATE CONSTANTS
      A=.5*SIGMAS*SIGMAS - SIGMAS*SIGMAT
      B=-SIGMAS
      F=V6*A/720.0
      G=V6*B/5040.0

* ZERO THE NON LOCAL MATRIX
      IF (TIME.EQ.1) THEN
      DO 50 I=1,3
          DO 40 J=1,3
              NLM(TRI,TRIP,I,J)=0.0
```

```
40            CONTINUE
50        CONTINUE
        ENDIF

* CALCULATE THE NON LOCAL MATRIX
            DO 100 I=1,10
                DO 60 J=1,10
                    NLM(TRI,TRIP,I,J)=NLM(TRI,TRIP,I,J)+(F*SA(I,J)
     C            +G*SB(I,J))
60                CONTINUE
100           CONTINUE

        END
*************************************************************************
* FIND PHI OF (L1,L2,L3) FOR THE TRIANGLE IN QUESTION

        SUBROUTINE PHII(TRI,L1,L2,L3,D)

        PARAMETER (MNODE=151 , MNTRIA=50)
        DOUBLE PRECISION L1,L2,L3,D(10),W(10)
        DOUBLE PRECISION ML(MNTRIA,10,10)
        DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
        DOUBLE PRECISION MG(MNODE,MNODE)
        DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
        INTEGER TRI
        COMMON MG,ML,NLM,NLI,LI,GT

        W(1)=L1**3
        W(2)=L2*L1**2
        W(3)=L3*L1**2
        W(4)=L2**3
        W(5)=L3*L2**2
        W(6)=L1*L2**2
        W(7)=L3**3
        W(8)=L1*L3**2
        W(9)=L2*L3**2
        W(10)=L1*L2*L3
        DO 30 I=1,10
            D(I)=0.0
            DO 20 J=1,10
                D(I)=D(I)+W(J)*GT(TRI,J,I)
20            CONTINUE
30        CONTINUE

        END
*****************************************************************
* FIND D(PHI)/DX FOR THE TRIANGLE UNDER SCRUTINY

        SUBROUTINE PHIX(TRI,L1,L2,L3,G1,G2,G3,DX)

        PARAMETER (MNODE=151 , MNTRIA=50)
        DOUBLE PRECISION L1,L2,L3,G1,G2,G3,W(10),DX(10)
        DOUBLE PRECISION ML(MNTRIA,10,10)
        DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
        DOUBLE PRECISION MG(MNODE,MNODE)
```

```
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      INTEGER TRI
      COMMON MG,ML,NLM,NLI,LI,GT



* FIND D(PHI)/DX
      W(1)=3.0*G1*L1**2
      W(2)=G2*L1**2+L2*G1*2.0*L1
      W(3)=G3*L1**2+L3*G1*2.0*L1
      W(4)=3.0*G2*L2**2
      W(5)=G3*L2**2+L3*G2*2.0*L2
      W(6)=G1*L2**2+L1*G2*2.0*L2
      W(7)=3.0*G3*L3**2
      W(8)=G1*L3**2+L1*G3*2.0*L3
      W(9)=G2*L3**2+L2*G3*2.0*L3
      W(10)=G1*L2*L3+G2*L1*L3+G3*L1*L2
      DO 50 I=1,10
          DX(I)=0.0
          DO 40 J=1,10
              DX(I)=DX(I)+W(J)*GT(TRI,J,I)
40          CONTINUE
50      CONTINUE

      END
*************************************************************
* FIND D(PHI)**2/DX**2

      SUBROUTINE PHIXX(TRI,L1,L2,L3,G1,G2,G3,DXX)

      PARAMETER (MNODE=151 , MNTRIA=50)
      DOUBLE PRECISION L1,L2,L3,G1,G2,G3,W(10),DXX(10)
      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      INTEGER TRI
      COMMON MG,ML,NLM,NLI,LI,GT

      W(1)=6.0*L1*G1**2
      W(2)=4.0*L1*G1*G2+2.0*L2*G1**2
      W(3)=4.0*L1*G1*G3+2.0*L3*G1**2
      W(4)=6.0*L2*G2**2
      W(5)=4.0*L2*G2*G3+2.0*L3*G2**2
      W(6)=4.0*L2*G1*G2+2.0*L1*G2**2
      W(7)=6.0*L3*G3**2
      W(8)=4.0*L3*G1*G3+2.0*L1*G3**2
      W(9)=4.0*L3*G2*G3+2.0*L2*G3**2
      W(10)=2.0*(L1*G2*G3+L2*G1*G3+L3*G1*G2)
      DO 70 I=1,10
          DXX(I)=0.0
          DO 60 J=1,10
              DXX(I)=DXX(I)+W(J)*GT(TRI,J,I)
60          CONTINUE
70      CONTINUE

```

```
      END
*************************************************************
* FIND D(PHI)**2/(DX*DU)

      SUBROUTINE PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,DXU)


      PARAMETER (MNODE=151 , MNTRIA=50)
      DOUBLE PRECISION F1,F2,F3
      DOUBLE PRECISION L1,L2,L3,G1,G2,G3,W(10),DXU(10)
      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      INTEGER TRI
      COMMON MG,ML,NLM,NLI,LI,GT

      W(1)=6.0*L1*F1*G1
      W(2)=2.0*L1*(F1*G2+F2*G1)+2.0*L2*F1*G1
      W(3)=2.0*L1*(F1*G3+F3*G1)+2.0*L3*F1*G1
      W(4)=6.0*L2*F2*G2
      W(5)=2.0*L2*(F3*G2+F2*G3)+2.0*L3*F2*G2
      W(6)=2.0*L2*(F1*G2+F2*G1)+2.0*L1*F2*G2
      W(7)=6.0*L3*F3*G3
      W(8)=2.0*L3*(F1*G3+F3*G1)+2.0*L1*F3*G3
      W(9)=2.0*L3*(F2*G3+F3*G2)+2.0*L2*F3*G3
      W(10)=L1*(F2*G3+F3*G2)+L2*(F1*G3+F3*G1)+L3*(F1*G2+F2*G1)
      DO 90 I=1,10
          DXU(I)=0.0
          DO 80 J=1,10
              DXU(I)=DXU(I)+W(J)*GT(TRI,J,I)
80            CONTINUE
90        CONTINUE

      END

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

* DETERMINE ELEMENT CASE, VOLUME, AND DERIV'S OF TETRAHEDRAL
* CO-O RESPECT TO X, U, AND U' COORDINATES

      SUBROUTINE CASEDT(TRI,TRIP,CORDND,PTNODE,TIME,E,F,G,V6,
     C    CASE,U1,U2,U3,X1,X2,X3)

      PARAMETER (MNODE=151 , MNTRIA=50)
      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      DOUBLE PRECISION E(4),F(4),G(4),V6
      DOUBLE PRECISION CORDND(MNODE,2),D1,D2
      DOUBLE PRECISION U1,U2,U3,B
      DOUBLE PRECISION X1,X2,X3,X2P,U1P,U2P,U3P
      DOUBLE PRECISION DE(4,4),WK(8),D(4,4)
      INTEGER PTNODE(MNTRIA,11),TRI,TRIP,CASE,TIME
      COMMON MG,ML,NLM,NLI,LI,GT
```

```

      IF (TIME.EQ.1) THEN
* CALCULATE COORDINATES
      X1=CORDND(PTNODE(TRI,1),1)
      X2=CORDND(PTNODE(TRI,4),1)
      X3=CORDND(PTNODE(TRI,7),1)
      X1P=CORDND(PTNODE(TRIP,1),1)
      U1=CORDND(PTNODE(TRI,1),2)
      U2=CORDND(PTNODE(TRI,4),2)
      U3=CORDND(PTNODE(TRI,7),2)
      U1P=CORDND(PTNODE(TRIP,1),2)
      U2P=CORDND(PTNODE(TRIP,4),2)
      U3P=CORDND(PTNODE(TRIP,7),2)

* DETERMINE THE CASE OF THE TRIANGLES
      CASE=2
      IF (X1.NE.X1P) THEN
          CASE=1
          IF (X1.LT.X1P) THEN
              CASE=3
            ENDIF
        ELSE
          IF (X1.GT.X2) THEN
              CASE=4
            ENDIF
        ENDIF
      ENDIF

* ASSEMBLE THE COORDINATE TRANSFORMATION MATRIX - DEPENDING
* ON CASE
      IF (CASE.EQ.1) THEN
          DE(2,1)=X2
          DE(2,2)=X1
          DE(2,3)=X2
          DE(2,4)=X1
          DE(3,1)=U2
          DE(3,2)=U1
          DE(3,3)=U3
          DE(3,4)=U1
          DE(4,1)=U1P
          DE(4,2)=U3P
          DE(4,3)=U1P
          DE(4,4)=U2P
        ENDIF

      IF (CASE.EQ.3) THEN
          DE(2,1)=X2
          DE(2,2)=X2
          DE(2,3)=X1
          DE(2,4)=X1
          DE(3,1)=U2
          DE(3,2)=U3
          DE(3,3)=U1
          DE(3,4)=U1
          DE(4,1)=U1P
          DE(4,2)=U1P
```

```
          DE(4,3)=U3P
          DE(4,4)=U2P
        ENDIF

      IF (CASE.EQ.2) THEN
          DE(2,1)=X1
          DE(2,2)=X2
          DE(2,3)=X2
          DE(2,4)=X2
          DE(3,1)=U1
          DE(3,2)=U3
          DE(3,3)=U2
          DE(3,4)=U3
          DE(4,1)=U1P
          DE(4,2)=U3P
          DE(4,3)=U2P
          DE(4,4)=U2P
          IF (TIME.EQ.2) THEN
              DE(3,2)=U2
              DE(3,3)=U3
              DE(3,4)=U2
              DE(4,2)=U2P
              DE(4,3)=U3P
              DE(4,4)=U3P
            ENDIF
        ENDIF

      IF (CASE.EQ.4) THEN
          DE(2,1)=X1
          DE(2,2)=X2
          DE(2,3)=X2
          DE(2,4)=X2
          DE(3,1)=U1
          DE(3,2)=U3
          DE(3,3)=U2
          DE(3,4)=U2
          DE(4,1)=U1P
          DE(4,2)=U3P
          DE(4,3)=U2P
          DE(4,4)=U3P
          IF (TIME.EQ.2) THEN
              DE(3,2)=U2
              DE(3,3)=U3
              DE(3,4)=U3
              DE(4,2)=U2P
              DE(4,3)=U3P
              DE(4,4)=U2P
            ENDIF
        ENDIF

      DO 10 I=1,4
          DE(1,I)=1.0
10      CONTINUE

* COPY MATRIX TO AVOID DECOMPOSITION BY IMSL
      DO 17 I=1,4
```

```
            DO 15 J=1,4
                D(I,J)=DE(I,J)
15            CONTINUE
17        CONTINUE
* FIND VOLUME FROM MATRIX DETERMINATE
        D1=0.0
        CALL LINV3F(DE,B,4,4,4,D1,D2,WK,IER)
        V6=D1*2**D2
* DERIVATIVES OF NATURAL COORDINATES
        D1=-1.0
        CALL LINV3F(D,B,1,4,4,D1,D2,WK,IER)
        DO 20 I=1,4
            E(I)=D(I,2)
            F(I)=D(I,3)
            G(I)=D(I,4)
20        CONTINUE
        END
************************************************************************
        SUBROUTINE SINFCN(E,F,G,V,SGM,H)
        PARAMETER (MNODE=151 , MNTRIA=50)
* FIND THE INTERPOLATING FUNCTION MATRIX (20 X 20 FOR A CUBIC
* IN 3THEN MULTIPLY BY THE Vi's TO GET THE Mi's
* (THESIS NOTATION)
* RESULT ARE THE BASIS FUNCTIONS FOR THE TETRAHEDRAL CUBIC
        DOUBLE PRECISION ML(MNTRIA,10,10)
        DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
        DOUBLE PRECISION MG(MNODE,MNODE)
        DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
        DOUBLE PRECISION M(17,4,4),SGT(20,20),WK(8),W(4,4)
        DOUBLE PRECISION SGM(5,4,4),E(4),F(4),G(4),V(5,20)
        DOUBLE PRECISION H(5,20),A,B,C,D1,D2,MT(4,4)
        COMMON MG,ML,NLM,NLI,LI,GT
* ZERO THE TETRAHEDRAL INTERPOLATING FUNCTION MATRIX
        DO 10 I=1,20
            DO 5 J=1,20
                SGT(I,J)=0.0
5            CONTINUE
10       CONTINUE
* ASSEMBLE THE PARTITIONED MATRICES ON THE DIAGONAL
        DO 30 K=1,4
            DO 20 J=1,4
                M(K,1,J)=1.0*(1/J)
                M(K,2,J)=3.0*E(J)-((J+1)/3)*2.0*E(J)
                M(K,3,J)=3.0*F(J)-((J+1)/3)*2.0*F(J)
                M(K,4,J)=3.0*G(J)-((J+1)/3)*2.0*G(J)
20            CONTINUE
```

```
            F(14,5,8)=1.0
            F(16,5,9)=1.0
            L1=0.0
            L2=1.0
            L3=0.0
            CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W1)
            CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W2)
            L1=1.0
            L2=0.0
            CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W3)
            CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W4)
            L1=0.0
            L3=1.0
            CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W5)
            CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W6)
            DO 150 I=1,10
                F(2,I,1)=W3(I)
                F(3,I,1)=W4(I)
                F(6,I,7)=W5(I)
                F(7,I,7)=W6(I)
                F(10,I,4)=W1(I)
                F(11,I,4)=W2(I)
                F(14,I,7)=W1(I)
                F(15,I,7)=W2(I)
150           CONTINUE
            L1=0.0
            L2=2.0/3.0
            L3=1.0/3.0
            CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W1)
            L2=1.0/3.0
            L3=2.0/3.0
            CALL PHII(TRIP,L1,L2,L3,W2)
            L1=1.0/3.0
            L2=2.0/3.0
            L3=0.0
            CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W3)
            L2=0.0
            L3=2.0/3.0
            CALL PHII(TRIP,L1,L2,L3,W4)
            L3=1.0/3.0
            L2=1.0/3.0
            CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W5)
            DO 170 I=1,10
                F(18,I,10)=W3(I)
                F(20,I,10)=W5(I)
                DO 160 J=1,10
                    F(17,I,J)=W1(I)*W2(J)
                    F(19,I,J)=W5(I)*W4(J)
160               CONTINUE
170           CONTINUE
          ELSE
            IF (CASE.EQ.4.AND.TIME.EQ.2) THEN
            UU(1)=U1
                UU(2)=U2
                UU(3)=U3
                UU(4)=U3
```

```
             L3=1.0
             CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W5)
             CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W6)
                 DO 120 I=1,10
                 F(2,I,1)=W3(I)
                 F(3,I,1)=W4(I)
                     F(6,I,4)=W1(I)
                     F(7,I,4)=W2(I)
                     F(10,I,7)=W5(I)
                     F(11,I,7)=W6(I)
                     F(14,I,7)=W1(I)
                     F(15,I,7)=W2(I)
 120               CONTINUE
                 L1=0.0
                 L2=2.0/3.0
                 L3=1.0/3.0
                 CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W1)
                 L2=1.0/3.0
                 L3=2.0/3.0
                 CALL PHII(TRIP,L1,L2,L3,W2)
                 L1=1.0/3.0
                 L2=0.0
                 L3=2.0/3.0
                 CALL PHII(TRIP,L1,L2,L3,W3)
                 L2=2.0/3.0
                 L3=0.0
                 CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W4)
                 L2=1.0/3.0
                 L3=1.0/3.0
                 CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W5)
                 DO 140 I=1,10
                     DO 130 J=1,10
                         F(17,I,J)=W1(I)*W2(J)
                         F(18,I,J)=W5(I)*W3(J)
 130                   CONTINUE
                     F(19,I,10)=W4(I)
                     F(20,I,10)=W5(I)
 140               CONTINUE
               ENDIF
           ENDIF

         IF (CASE.EQ.4.AND.TIME.EQ.1) THEN
             UU(1)=U1
             UU(2)=U3
             UU(3)=U2
             UU(4)=U2
             F(1,2,1)=1.0
             F(2,2,2)=1.0
             F(4,2,3)=1.0
             F(5,8,7)=1.0
             F(6,8,8)=1.0
             F(8,8,9)=1.0
             F(9,5,4)=1.0
             F(10,5,5)=1.0
             F(12,5,6)=1.0
             F(13,5,7)=1.0
```

```
                F(14,I,4)=W5(I)
                F(15,I,4)=W6(I)
 90           CONTINUE
            L1=0.0
            L2=1.0/3.0
            L3=2.0/3.0
            CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W1)
            L2=2.0/3.0
            L3=1.0/3.0
            CALL PHII(TRIP,L1,L2,L3,W2)
            L1=1.0/3.0
            L2=2.0/3.0
            L3=0.0
            CALL PHII(TRIP,L1,L2,L3,W3)
            L2=0.0
            L3=2.0/3.0
            CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W4)
            L3=1.0/3.0
            L2=1.0/3.0
            CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W5)
            DO 110 I=1,10
                F(19,I,10)=W4(I)
                F(20,I,10)=W5(I)
                DO 100 J=1,10
                    F(17,I,J)=W1(I)*W2(J)
                    F(18,I,J)=W5(I)*W3(J)
 100              CONTINUE
 110          CONTINUE
          ELSE
            IF (CASE.EQ.2.AND.TIME.EQ.2) THEN
            UU(1)=U1
                UU(2)=U2
                UU(3)=U3
                UU(4)=U2
            F(1,2,1)=1.0
            F(2,2,2)=1.0
            F(4,2,3)=1.0
                F(5,5,4)=1.0
                F(6,5,5)=1.0
                F(8,5,6)=1.0
                F(9,8,7)=1.0
                F(10,8,8)=1.0
                F(12,8,9)=1.0
                F(13,5,7)=1.0
                F(14,5,8)=1.0
                F(16,5,9)=1.0
                L1=0.0
                L2=1.0
                L3=0.0
                CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W1)
                CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W2)
                L1=1.0
                L2=0.0
                CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W3)
                CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W4)
            L1=0.0
```

```
            CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W1)
            CALL PHII(TRIP,L1,L2,L3,W6)
            L2=1.0/3.0
            L3=0.0
            CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W2)
            CALL PHII(TRIP,L1,L2,L3,W5)
            L1=L2
            L3=L1
            CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W3)
            DO 80 I=1,10
               F(17,I,10)=W1(I)
               F(18,I,10)=W2(I)
               DO 70 J=1,10
                  F(19,I,J)=W3(I)*W5(J)
                  F(20,I,J)=W3(I)*W6(J)
 70               CONTINUE
 80            CONTINUE
          ENDIF

        IF (CASE.EQ.2.AND.TIME.EQ.1) THEN
            UU(1)=U1
            UU(2)=U3
            UU(3)=U2
            UU(4)=U3
            F(1,2,1)=1.0
            F(2,2,2)=1.0
            F(4,2,3)=1.0
            F(5,8,7)=1.0
            F(6,8,8)=1.0
            F(8,8,9)=1.0
            F(9,5,4)=1.0
            F(10,5,5)=1.0
            F(12,5,6)=1.0
            F(13,8,4)=1.0
            F(14,8,5)=1.0
            F(16,8,6)=1.0
            L1=0.0
            L2=1.0
            L3=0.0
            CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W1)
            CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W2)
            L1=1.0
            L2=0.0
            CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W3)
            CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W4)
            L1=0.0
            L3=1.0
            CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W5)
            CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W6)
            DO 90 I=1,10
               F(2,I,1)=W3(I)
               F(3,I,1)=W4(I)
               F(6,I,7)=W5(I)
               F(7,I,7)=W6(I)
               F(10,I,4)=W1(I)
               F(11,I,4)=W2(I)
```

```
          L3=0.0
          CALL PHII(TRIP,L1,L2,L3,W2)
          L2=0.0
          L3=1.0/3.0
          CALL PHII(TRIP,L1,L2,L3,W3)
          DO 40 I=1,10
              DO 30 J=1,10
                  F(18,I,J)=W1(I)*W2(J)
                  F(20,I,J)=W1(I)*W3(J)
 30             CONTINUE
 40        CONTINUE
        ENDIF

      IF (CASE.EQ.3) THEN
          UU(1)=U2
          UU(2)=U3
          UU(3)=U1
          UU(4)=U1
          F(1,5,1)=1.0
          F(2,5,2)=1.0
          F(4,5,3)=1.0
          F(5,8,1)=1.0
          F(6,8,2)=1.0
          F(8,8,3)=1.0
          F(9,2,7)=1.0
          F(10,2,8)=1.0
          F(12,2,9)=1.0
          F(13,2,4)=1.0
          F(14,2,5)=1.0
          F(16,2,6)=1.0
          L1=0.0
          L2=1.0
          L3=0.0
          CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W1)
          CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W2)
          L1=1.0
          L2=0.0
          CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W3)
          CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W4)
          L1=0.0
          L3=1.0
          CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W5)
          CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W6)
          DO 60 I=1,10
              F(2,I,1)=W1(I)
              F(3,I,1)=W2(I)
              F(6,I,1)=W5(I)
              F(7,I,1)=W6(I)
              F(10,I,7)=W3(I)
              F(11,I,7)=W4(I)
              F(14,I,4)=W3(I)
              F(15,I,4)=W4(I)
 60        CONTINUE
          L1=2.0/3.0
          L2=0.0
          L3=1.0/3.0
```

```
* Fi'S BY CASE
       IF (CASE.EQ.1) THEN
           UU(1)=U2
           UU(2)=U1
           UU(3)=U3
           UU(4)=U1
           F(1,5,1)=1.0
           F(2,5,2)=1.0
           F(4,5,3)=1.0
           F(5,2,7)=1.0
           F(6,2,8)=1.0
           F(8,2,9)=1.0
           F(9,8,1)=1.0
           F(10,8,2)=1.0
           F(12,8,3)=1.0
           F(13,2,4)=1.0
           F(14,2,5)=1.0
           F(16,2,6)=1.0
           L1=0.0
           L2=1.0
           L3=0.0
           CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W1)
           CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W2)
           L1=1.0
           L2=0.0
           CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W3)
           CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W4)
           L1=0.0
           L3=1.0
           CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W5)
           CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W6)
           DO 10 I=1,10
               F(2,I,1)=W1(I)
               F(3,I,1)=W2(I)
               F(6,I,7)=W3(I)
               F(7,I,7)=W4(I)
               F(10,I,1)=W5(I)
               F(11,I,1)=W6(I)
               F(14,I,4)=W3(I)
               F(15,I,4)=W4(I)
 10          CONTINUE
           L1=2.0/3.0
           L2=0.0
           L3=1.0/3.0
           CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W1)
           L2=1.0/3.0
           L3=0.0
           CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W2)
           DO 20 I=1,10
               F(17,I,10)=W1(I)
               F(19,I,10)=W2(I)
 20          CONTINUE
           L1=1.0/3.0
           L3=1.0/3.0
           CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W1)
           L1=2.0/3.0
```

```
* MULTIPLY BY Hi'S AND SUM TO FIND THE SCATTERING CONTRIBUTION
* FROM THE (PHI)*(PHI') TERM
        DO 260 K=1,20
            DO 250 I=1,10
                DO 240 J=1,10
                    SA(I,J)=SA(I,J)+H(1,K)*F(K,I,J)
240             CONTINUE
250         CONTINUE
260     CONTINUE


        END
*************************************************************************

* SECOND SCATTERING INTEGRAL ( D(PHI)/DX * PHI' )

        SUBROUTINE SCATB(U1,U2,U3,X1,X2,X3,TRI,TRIP,AREAS,H
       C         ,CASE,TIME,SB,CORDND,PTNODE)


        PARAMETER (MNODE=151 , MNTRIA=50)

        DOUBLE PRECISION U1,U2,U3,X1,X2,X3,AREAS(MNTRIA)
        DOUBLE PRECISION SB(10,10),W6(10)
        DOUBLE PRECISION A,G1,G2,G3,F1,F2,F3
        DOUBLE PRECISION CORDND(MNODE,2)
        DOUBLE PRECISION W1(10),W2(10),W3(10),W4(10),W5(10)
        DOUBLE PRECISION F(20,10,10),L1,L2,L3,H(5,20),UU(4)
        DOUBLE PRECISION ML(MNTRIA,10,10)
        DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
        DOUBLE PRECISION MG(MNODE,MNODE)
        DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
        INTEGER CASE,TIME,TRI,TRIP
        INTEGER PTNODE(MNTRIA,11)
        COMMON MG,ML,NLM,NLI,LI,GT

* ZERO THE F MATRICES
        DO 7 K=1,20
            DO 6 I=1,10
                DO 5 J=1,10
                    F(K,I,J)=0.0
5               CONTINUE
6           CONTINUE
7       CONTINUE

* DERIVATIVES OF TRIANGULAR COORDINATES W.R.T. SPATIAL
* VARIABLES
        A=2.0*AREAS(TRI)
        G1=(U2-U3)/A
        G2=(U3-U1)/A
        G3=(U1-U2)/A
        F1=(X3-X2)/A
        F2=(X1-X3)/A
        F3=(X2-X1)/A

* ASSIGN THE U COORDS OF TETRAHEDRAL NODES AND ASSEMBLE THE
```

```
      ELSE
          IF (CASE.EQ.4.AND.TIME.EQ.2) THEN
          F(1,1,1)=1.0
          F(2,1,2)=1.0
          F(2,2,1)=1.0
          F(3,3,1)=1.0
          F(4,1,3)=1.0
              F(5,4,4)=1.0
              F(6,5,4)=1.0
              F(6,4,5)=1.0
              F(7,6,4)=1.0
              F(8,4,6)=1.0
              F(9,7,7)=1.0
              F(10,8,7)=1.0
              F(10,7,8)=1.0
              F(11,9,7)=1.0
              F(12,7,9)=1.0
              F(13,7,4)=1.0
              F(14,8,4)=1.0
              F(14,7,5)=1.0
              F(15,9,4)=1.0
              F(16,7,6)=1.0
              F(20,10,10)=1.0
              L1=0.0
              L2=1.0/3.0
              L3=2.0/3.0
              CALL PHII(TRI,L1,L2,L3,W1)
              L2=2.0/3.0
              L3=1.0/3.0
              CALL PHII(TRIP,L1,L2,L3,W2)
              DO 190 I=1,10
                  DO 180 J=1,10
                      F(17,I,J)=W1(I)*W2(J)
180                 CONTINUE
190             CONTINUE
              L1=1.0/3.0
              L2=0.0
              L3=2.0/3.0
              CALL PHII(TRI,L1,L2,L3,W2)
              L2=2.0/3.0
              L3=0.0
              CALL PHII(TRIP,L1,L2,L3,W1)
              DO 200 I=1,10
                  F(18,I,10)=W2(I)
                  F(19,10,I)=W1(I)
200             CONTINUE
            ENDIF
        ENDIF

* ZERO THE SCATTERING MATRIX
      DO 230 I=1,10
          DO 220 J=1,10
              SA(I,J)=0.0
220         CONTINUE
230     CONTINUE

```

```
          CALL PHII(TRIP,L1,L2,L3,W2)
          L2=2.0/3.0
          L3=0.0
          CALL PHII(TRI,L1,L2,L3,W1)
          DO 120 I=1,10
            F(18,10,I)=W2(I)
            F(19,I,10)=W1(I)
120        CONTINUE
         ENDIF
       ENDIF

      IF (CASE.EQ.4.AND.TIME.EQ.1)THEN
        F(1,1,1)=1.0
        F(2,1,2)=1.0
        F(2,2,1)=1.0
        F(3,3,1)=1.0
        F(4,1,3)=1.0
        F(5,7,7)=1.0
        F(6,7,8)=1.0
        F(6,8,7)=1.0
        F(7,9,7)=1.0
        F(8,7,9)=1.0
        F(9,4,4)=1.0
        F(10,4,5)=1.0
        F(10,5,4)=1.0
        F(11,6,4)=1.0
        F(12,4,6)=1.0
        F(20,10,10)=1.0
        F(13,4,7)=1.0
        F(14,5,7)=1.0
        F(14,4,8)=1.0
        F(15,6,7)=1.0
        F(16,4,9)=1.0
        L1=0.0
        L2=2.0/3.0
        L3=1.0/3.0
        CALL PHII(TRI,L1,L2,L3,W1)
        L2=1.0/3.0
        L3=2.0/3.0
        CALL PHII(TRIP,L1,L2,L3,W2)
        DO 160 I=1,10
          DO 150 J=1,10
            F(17,I,J)=W1(I)*W2(J)
150        CONTINUE
160      CONTINUE
        L1=1.0/3.0
        L2=2.0/3.0
        L3=0.0
        CALL PHII(TRI,L1,L2,L3,W2)
        L2=0.0
        L3=2.0/3.0
        CALL PHII(TRIP,L1,L2,L3,W1)
        DO 170 I=1,10
          F(18,I,10)=W2(I)
          F(19,10,I)=W1(I)
170      CONTINUE
```

```
           L3=1.0/3.0
           CALL PHII(TRIP,L1,L2,L3,W2)
           DO 80 I=1,10
               DO 70 J=1,10
                   F(17,I,J)=W1(I)*W2(J)
70               CONTINUE
80           CONTINUE
           L1=1.0/3.0
           L2=2.0/3.0
           L3=0.0
           CALL PHII(TRIP,L1,L2,L3,W2)
           L2=0.0
           L3=2.0/3.0
           CALL PHII(TRI,L1,L2,L3,W1)
           DO 90 I=1,10
               F(18,10,I)=W2(I)
               F(19,I,10)=W1(I)
90           CONTINUE
       ELSE
           IF (CASE.EQ.2.AND.TIME.EQ.2) THEN
           F(1,1,1)=1.0
           F(2,1,2)=1.0
           F(2,2,1)=1.0
           F(3,3,1)=1.0
           F(4,1,3)=1.0
               F(5,4,4)=1.0
               F(6,5,4)=1.0
               F(6,4,5)=1.0
               F(7,6,4)=1.0
               F(8,4,6)=1.0
               F(9,7,7)=1.0
               F(10,8,7)=1.0
               F(10,7,8)=1.0
               F(11,9,7)=1.0
               F(12,7,9)=1.0
               F(13,4,7)=1.0
               F(14,4,8)=1.0
               F(14,5,7)=1.0
               F(15,6,7)=1.0
               F(16,4,9)=1.0
           F(20,10,10)=1.0
               L1=0.0
               L2=2.0/3.0
               L3=1.0/3.0
               CALL PHII(TRI,L1,L2,L3,W1)
               L2=1.0/3.0
               L3=2.0/3.0
               CALL PHII(TRIP,L1,L2,L3,W2)
               DO 110 I=1,10
                   DO 100 J=1,10
                       F(17,I,J)=W1(I)*W2(J)
100                  CONTINUE
110              CONTINUE
               L1=1.0/3.0
               L2=0.0
               L3=2.0/3.0
```

```
              F(9,1,7)=1.0
              F(10,1,8)=1.0
              F(10,2,7)=1.0
              F(11,3,7)=1.0
              F(12,1,9)=1.0
              F(13,1,4)=1.0
              F(14,1,5)=1.0
              F(14,2,4)=1.0
              F(15,3,4)=1.0
              F(16,1,6)=1.0
              L1=2.0/3.0
              L2=0.0
              L3=1.0/3.0
              CALL PHII(TRI,L1,L2,L3,W1)
              CALL PHII(TRIP,L1,L2,L3,W2)
              DO 50 I=1,10
                  F(17,I,10)=W1(I)
                  F(20,10,I)=W2(I)
50              CONTINUE
              L2=1.0/3.0
              L3=0.0
              CALL PHII(TRI,L1,L2,L3,W1)
              CALL PHII(TRIP,L1,L2,L3,W2)
              DO 60 I=1,10
                  F(18,I,10)=W1(I)
                  F(19,10,I)=W2(I)
60              CONTINUE
            ENDIF

          IF (CASE.EQ.2.AND.TIME.EQ.1) THEN
              F(1,1,1)=1.0
              F(2,1,2)=1.0
              F(2,2,1)=1.0
              F(3,3,1)=1.0
              F(4,1,3)=1.0
              F(5,7,7)=1.0
              F(6,7,8)=1.0
              F(6,8,7)=1.0
              F(7,9,7)=1.0
              F(8,7,9)=1.0
              F(9,4,4)=1.0
              F(10,4,5)=1.0
              F(10,5,4)=1.0
              F(11,6,4)=1.0
              F(12,4,6)=1.0
              F(13,7,4)=1.0
              F(14,7,5)=1.0
              F(14,8,4)=1.0
              F(15,9,4)=1.0
              F(16,7,6)=1.0
              F(20,10,10)=1.0
              L1=0.0
              L2=1.0/3.0
              L3=2.0/3.0
              CALL PHII(TRI,L1,L2,L3,W1)
              L2=2.0/3.0
```

```
 20          CONTINUE
 30       CONTINUE

* INITIALIZE THE Fi'S DEPENDING UPON THE TETRAHEDRAL CASE
* AND TIME
         IF (CASE.EQ.1) THEN
            F(1,4,1)=1.0
            F(2,4,2)=1.0
            F(2,5,1)=1.0
            F(3,6,1)=1.0
            F(4,4,3)=1.0
            F(5,1,7)=1.0
            F(6,1,8)=1.0
            F(6,2,7)=1.0
            F(7,3,7)=1.0
            F(8,1,9)=1.0
            F(9,7,1)=1.0
            F(10,8,1)=1.0
            F(10,7,2)=1.0
            F(11,9,1)=1.0
            F(12,7,3)=1.0
            F(13,1,4)=1.0
            F(14,1,5)=1.0
            F(14,2,4)=1.0
            F(15,3,4)=1.0
            F(16,1,6)=1.0
            L1=2.0/3.0
            L2=0.0
            L3=1.0/3.0
            CALL PHII(TRI,L1,L2,L3,W1)
            CALL PHII(TRIP,L1,L2,L3,W2)
            DO 32 I=1,10
                F(17,I,10)=W1(I)
                F(20,10,I)=W2(I)
 32           CONTINUE
            L2=1.0/3.0
            L3=0.0
            CALL PHII(TRI,L1,L2,L3,W1)
            CALL PHII(TRIP,L1,L2,L3,W2)
            DO 34 I=1,10
                F(18,10,I)=W2(I)
                F(19,I,10)=W1(I)
 34           CONTINUE
          ENDIF

         IF (CASE.EQ.3) THEN
            F(1,4,1)=1.0
            F(2,4,2)=1.0
            F(2,5,1)=1.0
            F(3,6,1)=1.0
            F(4,4,3)=1.0
            F(5,7,1)=1.0
            F(6,8,1)=1.0
            F(6,7,2)=1.0
            F(7,9,1)=1.0
            F(8,7,3)=1.0
```

```
                SGT(I+12,J+12)=M(12,I,J)
160           CONTINUE
170        CONTINUE
         DO 190 I=17,20
            DO 180 J=1,16
                K=((J-1)/4)+13
                L=J-(K-13)*4
                SGT(I,J)=M(K,I-16,L)
180           CONTINUE
190        CONTINUE

         DO 197 I=17,20
            DO 195 J=17,20
                SGT(I,J)=SGM(5,I-16,J-16)
195           CONTINUE
197        CONTINUE

* FIND THE Hi'S
         DO 220 I=1,5
            DO 210 J=1,20
                H(I,J)=0.0
                DO 200 L=1,20
                    H(I,J)=H(I,J)+V(I,L)*SGT(L,J)
200              CONTINUE
210           CONTINUE
220        CONTINUE

         END
*********************************************************************

* CALCULATE THE FIRST ( PHI*PHI' ) SCATTERING INTEGRAL
* CUBIC FIT OVER A TETRAHEDRON WITH CORNER FLUXES, GRADIENTS
* AND FACE CENTERED FLUXES AS DEGREES OF FREEDOM

         SUBROUTINE SCATA(H,TRI,TRIP,CASE,TIME,SA,CORDND,PTNODE)

         PARAMETER (MNODE=151 , MNTRIA=50)

         DOUBLE PRECISION ML(MNTRIA,10,10)
         DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
         DOUBLE PRECISION MG(MNODE,MNODE)
         DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
         DOUBLE PRECISION X2,X2P,U2,U2P,CORDND(MNODE,2)
         DOUBLE PRECISION F(20,10,10),W1(10),W2(10),L1,L2,L3
         DOUBLE PRECISION SA(10,10)
         DOUBLE PRECISION H(5,20)
         INTEGER TRI,TRIP,CASE,TIME
         INTEGER PTNODE(MNTRIA,11)
         COMMON MG,ML,NLM,NLI,LI,GT

* ZERO THE Fi'S
         DO 30 I=1,20
            DO 20 J=1,10
                DO 10 K=1,10
                    F(I,J,K)=0.0
10                CONTINUE
```

```
            A=E(1)
            B=F(1)
            C=G(1)
            IF (K.LT.4) THEN
                E(1)=E(K+1)
                F(1)=F(K+1)
                G(1)=G(K+1)
                E(K+1)=A
                F(K+1)=B
                G(K+1)=C
              ENDIF
30        CONTINUE

* TAKE INVERSES OF MATRICES 1-4
        DO 80 K=1,4
            DO 50 I=1,4
                DO 40 J=1,4
                    W(I,J)=M(K,I,J)
40              CONTINUE
50            CONTINUE
            D1=-1.0
            CALL LINV3F(W,B,1,4,4,D1,D2,WK,IER)
            DO 70 I=1,4
                DO 60 J=1,4
                    M(K+8,I,J)=W(I,J)
60              CONTINUE
70            CONTINUE
80        CONTINUE

* FIND REMAINING SUB MATRICES
        DO 150 K=13,16
            DO 110 I=1,4
                DO 100 J=1,4
                    MT(I,J)=0.0
                    DO 90 L=1,4
                        MT(I,J)=MT(I,J)+SGM(K-12,I,L)*M(K-4,L,J)
90                    CONTINUE
100             CONTINUE
110           CONTINUE
            DO 140 I=1,4
                DO 130 J=1,4
                    M(K,I,J)=0.0
                    DO 120 L=1,4
                        M(K,I,J)=M(K,I,J)-SGM(5,I,L)*MT(L,J)
120                   CONTINUE
130             CONTINUE
140           CONTINUE
150       CONTINUE

* ASSEMBLE INTO THE TETRAHEDRAL (SCATTERING) INTERPOLATING
*  FUNCTION MATRIX OF CONSTANTS
        DO 170 I=1,4
            DO 160 J=1,4
                SGT(I,J)=M(9,I,J)
                SGT(I+4,J+4)=M(10,I,J)
                SGT(I+8,J+8)=M(11,I,J)
```

```
      F(1,2,1)=1.0
      F(2,2,2)=1.0
      F(4,2,3)=1.0
          F(5,5,4)=1.0
          F(6,5,5)=1.0
          F(8,5,6)=1.0
          F(9,8,7)=1.0
          F(10,8,8)=1.0
          F(12,8,9)=1.0
          F(13,8,4)=1.0
          F(14,8,5)=1.0
          F(16,8,6)=1.0
          L1=0.0
          L2=1.0
          L3=0.0
          CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W1)
          CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W2)
          L1=1.0
          L2=0.0
          CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W3)
          CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W4)
      L1=0.0
      L3=1.0
      CALL PHIXX(TRI,L1,L2,L3,G1,G2,G3,W5)
      CALL PHIXU(TRI,L1,L2,L3,F1,F2,F3,G1,G2,G3,W6)
          DO 180 I=1,10
          F(2,I,1)=W3(I)
          F(3,I,1)=W4(I)
              F(6,I,4)=W1(I)
              F(7,I,4)=W2(I)
              F(10,I,7)=W5(I)
              F(11,I,7)=W6(I)
              F(14,I,4)=W5(I)
              F(15,I,4)=W6(I)
180           CONTINUE
          L1=0.0
          L2=1.0/3.0
          L3=2.0/3.0
          CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W1)
          L2=2.0/3.0
          L3=1.0/3.0
          CALL PHII(TRIP,L1,L2,L3,W2)
          L1=1.0/3.0
          L2=0.0
          L3=2.0/3.0
          CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W3)
          L2=2.0/3.0
          L3=0.0
          CALL PHII(TRIP,L1,L2,L3,W4)
          L2=1.0/3.0
          L3=1.0/3.0
          CALL PHIX(TRI,L1,L2,L3,G1,G2,G3,W5)
          DO 200 I=1,10
              DO 190 J=1,10
                  F(17,I,J)=W1(I)*W2(J)
                  F(19,I,J)=W5(I)*W4(J)
```

```
190                 CONTINUE
                  F(18,I,10)=W3(I)
                  F(20,I,10)=W5(I)
200             CONTINUE
            ENDIF
        ENDIF

* ZERO THE SB MATRIX
      DO 220 I=1,10
          DO 210 J=1,10
              SB(I,J)=0.0
210         CONTINUE
220     CONTINUE

* ASSEMBLE SB
      DO 250 K=1,20
          DO 240 I=1,10
              DO 230 J=1,10
            SB(I,J)=SB(I,J)+H(2,K)*F(K,I,J)*UU(1)+H(3,K)*F(K,I,J)
     C        *UU(2)+H(4,K)*F(K,I,J)*UU(3)+H(5,K)*F(K,I,J)*UU(4)
230             CONTINUE
240         CONTINUE
250     CONTINUE

      END

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

      SUBROUTINE PN(PHI,CORDND,SIGMAS,N,NTRIA,RANGE)

      PARAMETER (MNODE=151 , MNTRIA=50)

      DOUBLE PRECISION CORDND(MNODE,2),PHI(MNODE),PSI(21,9)
      DOUBLE PRECISION RANGE,SIGMAS,PERC,TPERC,APERC
      DOUBLE PRECISION ML(MNTRIA,10,10)
      DOUBLE PRECISION NLM(MNTRIA,16,10,10),NLI(MNTRIA,4,10)
      DOUBLE PRECISION MG(MNODE,MNODE)
      DOUBLE PRECISION GT(MNTRIA,10,10),LI(MNTRIA,10,4)
      INTEGER N,NTRIA,X,U

      COMMON MG,ML,NLM,NLI,LI,GT
* READ IN ARRAY OF EXACT SOLUTION - THIS DATA FILE HAS C=.5
* RESULTS
      OPEN (18,FILE='PNDATA5',STATUS='OLD')
      REWIND (18)
      DO 100 I=1,21
          READ (18,5000) (PSI(I,U),U=1,9)
100     CONTINUE
      CLOSE (18)

* CALCULATE PERCENT DIFFERENCE
      K=0
      TPERC=0.0
      PRINT*,'  COORDINATES      CURRENTS     FIN ELEM'
      PRINT*,'     X        U        X       U      FLUX      FLUX    % DIFF'
      DO 200 I=1,N-NTRIA,3
```

```
           IF(INT(CORDND(I,1)/.25)*.25.EQ.CORDND(I,1)) THEN
           X=NINT(4*CORDND(I,1)+1)
           U=NINT(4*CORDND(I,2)+5)
           PERC=100.0*ABS(PSI(X,U)-PHI(I))/PSI(X,U)
       WRITE (*,5001) CORDND(I,1),CORDND(I,2),PHI(I+1),PHI(I+2),
      C                PHI(I),PSI(X,U),PERC
           IF (CORDND(I,1).EQ.0.0.AND.CORDND(I,2).GE.0.0)THEN
               GO TO 200
             ELSE
           IF (CORDND(I,1).EQ.RANGE.AND.CORDND(I,2).LE.0.0)THEN
                   GO TO 200
                 ENDIF
             ENDIF
           TPERC=TPERC+PERC
           K=K+1
           ENDIF
R,2399
200      CONTINUE
       APERC=TPERC/K
       PRINT*,'AVERAGE % DIFFERENCE IS ..',APERC
       D=NTRIA/RANGE
       PRINT*,'FOR AN AVERAGE OF',D,'TRIANGLES PER MEAN FREE PATH'

5000   FORMAT(2X,1P10E13.4)
5001   FORMAT(6(1X,F7.3),2X,F5.2)
       END
EOF..
EOT..
```

```
===> CO MESHE3.9 MESH
===> XEF9
IER IS ...           0
NTRIA       N      SIGMAS
   46      151        0.900
RANGE IS....     3.000000000
NODAL VALUES OF THE FLUX
    1        0.3534    2        0.3434
    3        0.3292    4        0.3044
    5        0.2500    6        0.1864
    7        0.1656    8        0.1481
    9        0.1413   10        0.2554
   11        0.2288   12        0.1614
   13        0.1381   14        0.1239
   15        0.0976   16        0.0904
   17        0.1745   18        0.1273
   19        0.0903   20        0.0713
   21        0.0591   22        0.1208
   23        0.0999   24        0.0691
   25        0.0606   26        0.0543
   27        0.0441   28        0.0398
   29        0.0826   30        0.0557
   31        0.0427   32        0.0377
   33        0.0337   34        0.0306
   35        0.0280   36        0.2539
 ELEMENT PENALTY VALUES
    1      0.57885E-03     2      0.65116E-03
    3      0.76555E-03     4      0.83027E-03
    5      0.45322E-03     6      0.10514E-03
    7      -.75873E-04     8      -.19484E-03
    9      -.30669E-03    10      -.41093E-03
   11      -.73112E-03    12      -.57047E-03
   13      -.64482E-03    14      -.48787E-03
   15      0.39001E-03    16      0.51147E-03
   17      0.41353E-03    18      0.53690E-04
   19      -.29261E-04    20      -.96223E-04
   21      -.21822E-03    22      -.39397E-03
   23      -.36922E-03    24      -.26522E-03
   25      0.15227E-03    26      0.29752E-03
   27      0.11491E-03    28      0.27448E-04
   29      -.16711E-04    30      -.33954E-04
   31      -.12193E-03    32      -.13218E-03
```

33	-.20394E-03	34	-.87472E-04
35	0.90719E-04	36	0.10032E-03
37	0.64651E-04	38	0.57640E-05
39	-.72660E-05	40	-.18721E-04
41	-.12605E-04	42	-.31310E-04
43	-.75572E-04	44	-.31328E-04
45	-.36920E-04	46	-.48821E-04

TOTAL PENALTY .... AND SUM OF ABS(PENALTY) ARE ..

-.46949E-04 0.11260E-01

COORDINATES		CURRENTS		FIN ELEM		
X	U	X	U	FLUX	FLUX	% DIFF
0.000	1.000	-0.111	0.984	0.353	0.353	0.00
0.000	0.750	-0.164	-0.297	0.343	0.343	0.00
0.000	0.500	-0.212	0.115	0.329	0.329	0.00
0.000	0.250	-0.373	0.578	0.304	0.304	0.00
0.000	0.000	-0.529	0.403	0.250	0.250	0.00
0.000	-0.250	-0.238	0.076	0.186	0.196	4.67
0.000	-0.500	-0.156	0.041	0.166	0.171	3.06
0.000	-0.750	-0.118	0.020	0.148	0.157	5.42
0.000	-1.000	-0.099	-0.161	0.141	0.147	3.56
0.750	1.000	-0.119	0.552	0.255	0.262	2.61
0.750	0.750	-0.141	0.045	0.229	0.236	3.12
0.750	0.250	-0.129	0.148	0.161	0.169	4.49
0.750	0.000	-0.234	0.193	0.138	0.142	2.79
0.750	-0.250	-0.129	0.133	0.124	0.125	0.99
0.750	-0.750	-0.073	0.040	0.098	0.101	3.28
0.750	-1.000	-0.057	-0.064	0.090	0.091	1.23
1.500	1.000	-0.079	0.173	0.174	0.184	5.08
1.500	0.500	-0.082	0.158	0.127	0.132	3.38
1.500	0.000	-0.116	0.090	0.090	0.095	4.52
1.500	-0.500	-0.058	0.043	0.071	0.075	4.52
1.500	-1.000	-0.038	0.020	0.059	0.062	4.35
2.250	1.000	-0.060	0.122	0.121	0.127	4.74
2.250	0.750	-0.054	0.063	0.100	0.106	5.62
2.250	0.250	-0.047	0.044	0.069	0.073	5.56
2.250	0.000	-0.094	0.075	0.061	0.063	4.46
2.250	-0.250	-0.052	0.066	0.054	0.056	3.07
2.250	-0.750	-0.030	0.032	0.044	0.045	3.03
2.250	-1.000	-0.022	0.039	0.040	0.042	4.30
3.000	1.000	-0.040	0.076	0.083	0.087	4.85
3.000	0.500	-0.030	0.045	0.056	0.058	4.28
3.000	0.000	-0.035	0.012	0.043	0.043	0.00
3.000	-0.250	-0.033	0.291	0.038	0.038	0.00
3.000	-0.500	-0.018	-0.015	0.034	0.034	0.00
3.000	-0.750	-0.017	0.045	0.031	0.031	0.00
3.000	-1.000	-0.013	0.070	0.028	0.028	0.00

AVERAGE % DIFFERENCE IS .. 3.877686025

FOR AN AVERAGE OF 15.333 TRIANGLES PER MEAN FREE PATH

```
ED C9OUT
LI,1,300
===> CO MSHE3.9C MESH
===> XE
IER IS ...          0
NTRIA      N      SIGMAS
   46      151       0.900
RANGE IS....     3.000000000
NODAL VALUES OF THE FLUX
    1       0.3534     2       0.3434
    3       0.3292     4       0.3044
    5       0.2500     6       0.1892
    7       0.1691     8       0.1519
    9       0.1439    10       0.2580
   11       0.2325    12       0.1660
   13       0.1419    14       0.1269
   15       0.1004    16       0.0927
   17       0.1818    18       0.1293
   19       0.0930    20       0.0736
   21       0.0604    22       0.1253
   23       0.1035    24       0.0714
   25       0.0626    26       0.0562
   27       0.0450    28       0.0406
   29       0.0856    30       0.0571
   31       0.0427    32       0.0377
   33       0.0337    34       0.0306
   35       0.0280    36       0.2804
 ELEMENT PENALTY VALUES
    1       0.69119E-03     2       0.79918E-03
    3       0.70362E-03     4       0.79296E-03
    5       0.40508E-03     6       0.10532E-03
    7       -.84340E-04     8       -.19188E-03
    9       -.30460E-03    10       -.42177E-03
   11       -.75959E-03    12       -.58829E-03
   13       -.66508E-03    14       -.51004E-03
   15       0.44435E-03    16       0.54365E-03
   17       0.40379E-03    18       0.57488E-04
   19       -.32378E-04    20       -.10216E-03
   21       -.23084E-03    22       -.41572E-03
   23       -.39299E-03    24       -.28130E-03
   25       0.16749E-03    26       0.31947E-03
   27       0.11766E-03    28       0.30371E-04
   29       -.17476E-04    30       -.36011E-04
   31       -.12910E-03    32       -.14093E-03
   33       -.22000E-03    34       -.95110E-04
   35       0.99120E-04    36       0.10831E-03
   37       0.67810E-04    38       0.71457E-05
   39       -.77406E-05    40       -.20200E-04
   41       -.18113E-04    42       -.26107E-04
   43       -.80449E-04    44       -.35963E-04
   45       -.40512E-04    46       -.54356E-04
 TOTAL PENALTY .... AND SUM OF ABS(PENALTY) ARE ..
           -.39050E-04              0.11767E-01
   COORDINATES        CURRENTS       FIN ELEM
      X        U        X        U       FLUX     FLUX   % DIFF
   0.000    1.000   -0.134    0.040    0.353    0.353    0.00
```

55	0.000	0.750	-0.160	0.049	0.343	0.343	0.00
56	0.000	0.500	-0.227	0.078	0.329	0.329	0.00
57	0.000	0.250	-0.357	0.158	0.304	0.304	0.00
58	0.000	0.000	-0.489	0.218	0.250	0.250	0.00
59	0.000	-0.250	-0.213	0.043	0.189	0.196	3.26
60	0.000	-0.500	-0.148	0.045	0.169	0.171	1.04
61	0.000	-0.750	-0.113	0.027	0.152	0.157	3.03
62	0.000	-1.000	-0.097	-0.135	0.144	0.147	1.81
63	0.750	1.000	-0.115	0.111	0.258	0.262	1.62
64	0.750	0.750	-0.125	0.106	0.233	0.236	1.53
65	0.750	0.250	-0.127	0.152	0.166	0.169	1.77
66	0.750	0.000	-0.236	0.177	0.142	0.142	0.16
67	0.750	-0.250	-0.126	0.122	0.127	0.125	1.33
68	0.750	-0.750	-0.074	0.039	0.100	0.101	0.45
69	0.750	-1.000	-0.058	-0.052	0.093	0.091	1.35
70	1.500	1.000	-0.088	0.130	0.182	0.184	1.10
71	1.500	0.500	-0.079	0.128	0.129	0.132	1.90
72	1.500	0.000	-0.114	0.085	0.093	0.095	1.61
73	1.500	-0.500	-0.058	0.047	0.074	0.075	1.45
74	1.500	-1.000	-0.040	0.027	0.060	0.062	2.33
75	2.250	1.000	-0.063	0.105	0.125	0.127	1.16
76	2.250	0.750	-0.056	0.080	0.103	0.106	2.26
77	2.250	0.250	-0.049	0.048	0.071	0.073	2.43
78	2.250	0.000	-0.096	0.075	0.063	0.063	1.30
79	2.250	-0.250	-0.050	0.052	0.056	0.056	0.29
80	2.250	-0.750	-0.030	0.020	0.045	0.045	0.93
81	2.250	-1.000	-0.023	0.016	0.041	0.042	2.34
82	3.000	1.000	-0.043	0.075	0.086	0.087	1.40
83	3.000	0.500	-0.032	0.045	0.057	0.058	1.92
84	3.000	0.000	-0.042	0.020	0.043	0.043	0.00
85	3.000	-0.250	-0.030	0.018	0.038	0.038	0.00
86	3.000	-0.500	-0.024	0.014	0.034	0.034	0.00
87	3.000	-0.750	-0.017	0.012	0.031	0.031	0.00
88	3.000	-1.000	-0.018	0.010	0.028	0.028	0.00

89 AVERAGE % DIFFERENCE IS .. 1.590236152

90 FOR AN AVERAGE OF 15.333 TRIANGLES PER MEAN FREE PATH

EOT..

UP

Appendix B.

The Absorbing Term (quadratic, and cubic)

The absorbing term is

$$\frac{1}{2}\int dA\, \Sigma_t^2 \phi^2 \tag{3-2}$$

Which may be written as

$$= \frac{1}{2}\, \underline{\tilde{\varphi}}\ \underline{\underline{\widetilde{GT}}}\ \underline{\underline{MA}}\ \underline{\underline{GT}}\ \underline{\varphi} \tag{3-5}$$

The matrices on subsequent pages labeled as QCOABS and CCOABS (for Quadratic Coefficients Absorbing term and Cubic Coefficients) generate MA of equation (3-5) from

$$\underline{\underline{MA}} = \frac{2A\Sigma_t^2}{6!}\ \underline{\underline{QCOABS}} \tag{B-1}$$

and in the cubic case from

$$\underline{\underline{MA}} = \frac{2A\Sigma_t^2}{8!}\ \underline{\underline{CCOABS}} \tag{B-2}$$

where A is triangle area.

cat coabs

QCOABS

24.0	4.0	4.0	6.0	2.0	6.0
4.0	24.0	4.0	6.0	6.0	2.0
4.0	4.0	24.0	2.0	6.0	6.0
6.0	6.0	2.0	4.0	2.0	2.0
2.0	6.0	6.0	2.0	4.0	2.0
6.0	2.0	6.0	2.0	2.0	4.0

cat ccoabs

CCOABS

720.0	120.0	120.0	36.0	12.0	48.0	36.0	48.0	12.0	24.0
120.0	48.0	24.0	48.0	12.0	36.0	12.0	12.0	8.0	12.0
120.0	24.0	48.0	12.0	8.0	12.0	48.0	36.0	12.0	12.0
36.0	48.0	12.0	720.0	120.0	120.0	36.0	12.0	48.0	24.0
12.0	12.0	8.0	120.0	48.0	24.0	48.0	12.0	36.0	12.0
48.0	36.0	12.0	120.0	24.0	48.0	12.0	8.0	12.0	12.0
36.0	12.0	48.0	36.0	48.0	12.0	720.0	120.0	120.0	24.0
48.0	12.0	36.0	12.0	12.0	8.0	120.0	48.0	24.0	12.0
12.0	8.0	12.0	48.0	36.0	12.0	120.0	24.0	48.0	12.0
24.0	12.0	12.0	24.0	12.0	12.0	24.0	12.0	12.0	8.0

Appendix C - Boundary Term (quadratic and cubic)

The boundary term (3-6) is

$$\int dA\, \varepsilon_t u \frac{\partial \phi}{\partial x} \phi = \varepsilon_t \int dA\, u\, \underline{\hat{\tilde{\varphi}}}\, \underline{\underline{\hat{\tilde{G}}}}{}^T\, \underline{m}_x \underline{\tilde{m}}\, \underline{\underline{G}}{}^T \underline{\varphi} \qquad (3\text{-}6)$$

When u is expanded as in equation 2-3, the boundary matrix can be written as the sum of 3 matrices.

$$= \frac{1}{2}\, \underline{\hat{\tilde{\varphi}}}\, \underline{\underline{\hat{\tilde{G}}}}{}^T \left[\underline{\underline{MB}}1 + \underline{\underline{MB}}2 + \underline{\underline{MB}}3 \right] \underline{\underline{G}}{}^T \underline{\varphi} \qquad (3\text{-}10)$$

These matrices are complicated by the fact that the basis functions contain derivatives of natural co-ordinates, which are distinct for every separate triangle geometry. In the quadratic case $\underline{m}_x$ is given by

$$\underline{m}_x = \begin{bmatrix} 2\ell_1 g_1 \\ 2\ell_2 g_2 \\ 2\ell_3 g_3 \\ \ell_2 g_1 + \ell_1 g_2 \\ \ell_3 g_2 + \ell_2 g_3 \\ \ell_1 g_3 + \ell_3 g_1 \end{bmatrix} \qquad (2\text{-}19)$$

the product $\underline{m}_x \hat{\underline{m}}$ for any of the three matrices results in what can be thought of as a matrix of constants, multiplied by g_i's as appropriate. If a matrix $\underline{\underline{D}}$ is formed of the g_i's .

$$\underline{\underline{D}} = \begin{bmatrix} g_1 & 0 \\ g_2 & 0 \\ g_3 & 0 \\ g_1 & g_2 \\ g_2 & g_3 \\ g_3 & g_1 \end{bmatrix} \qquad (C\text{-}1)$$

then this can be computed for each triangle and " overlayed " in a sense on each column of constants to produce the boundary matrix. As an example consider the matrix referred to on the next page as QCOBNDI (for Quadratic COeffients Bndry term #1) It is a 6 X 12 matrix, that when D is overlayed on, and when multiplied by $\frac{\mu_1 \varepsilon_t 2A}{6!}$ produces MBI

$$\underline{\underline{MB1}} = \frac{\mu_1 \varepsilon_t 2A}{6!} \begin{bmatrix} 48g_1 + 0 \\ 12g_2 + 0 \\ 12g_3 + 0 \\ 6g_1 + 24g_2 \\ 6g_2 + 6g_3 \\ 24g_3 + 6g_1 \end{bmatrix} \quad \text{etc.} \tag{C-2}$$

MB2 and MB3 must be formed of course with the appropriate constants (μ_2 and μ_3) from the matrices labeled QCOBND2 and QCOBND3.

Boundary matrices for the cubic fit are found in an analagous manner, with 3 (10 x 20) matrices, CCOBND1, CCOBND2, and CCOBND3. In the cubic case, the derivative matrix D to be overlayed is

$$\underline{\underline{D}} = \begin{bmatrix} g_1 & 0 \\ g_1 & g_2 \\ g_1 & g_3 \\ g_2 & 0 \\ g_2 & g_3 \\ g_2 & g_1 \\ g_3 & 0 \\ g_3 & g_1 \\ g_3 & g_2 \\ g_1 & g_2 \end{bmatrix} \tag{C-3}$$

```
2070          CALL FMIXG(TR1,L1,L2,L3,F1,F2,F3,G1,G2,G3,W4)
2071          L1=0.0
```


In this case row 10 must be augmented by another term. The last row of CCOBND1, CCOBND2, and CCOBND3 represent 3 each dimension (10) matrices (BR1, BR2, BR3). After is formed from

$$\underline{\underline{MB}} = \underline{\underline{MB1}} + \underline{\underline{MB2}} + \underline{\underline{MB3}} \tag{C-4}$$

if

For i=1,10

$$BR(i) = BR1(i) * u_1 + BR2(i) * u_2 + BR3(i) * u_3$$

Then

For i=1,10

$$MB(10,i) = MB(10,i) + BR(i) * ZA * \varepsilon_t / 6! \tag{C-5}$$

and the boundary matrix is now completely formed.

cat cobnd1

QCOBND1

48.0	.0	8.0	.0	8.0	.0	12.0	.0	4.0	.0	12.0	.0
12.0	.0	12.0	.0	4.0	.0	8.0	.0	4.0	.0	4.0	.0
12.0	.0	4.0	.0	12.0	.0	4.0	.0	4.0	.0	8.0	.0
6.0	24.0	6.0	4.0	2.0	4.0	4.0	6.0	2.0	2.0	2.0	6.0
6.0	6.0	2.0	6.0	6.0	2.0	2.0	4.0	2.0	2.0	4.0	2.0
24.0	6.0	4.0	2.0	4.0	6.0	6.0	2.0	2.0	2.0	6.0	4.0

% cat cobnd2

QCOBND2

12.0	.0	12.0	.0	4.0	.0	8.0	.0	4.0	.0	4.0	.0
8.0	.0	48.0	.0	8.0	.0	12.0	.0	12.0	.0	4.0	.0
4.0	.0	12.0	.0	12.0	.0	4.0	.0	8.0	.0	4.0	.0
4.0	6.0	24.0	6.0	4.0	2.0	6.0	4.0	6.0	2.0	2.0	2.0
2.0	4.0	6.0	24.0	6.0	4.0	2.0	6.0	4.0	6.0	2.0	2.0
6.0	2.0	6.0	6.0	2.0	6.0	4.0	2.0	2.0	4.0	2.0	2.0

% cat cobnd3

QCOBND3

12.0	.0	4.0	.0	12.0	.0	4.0	.0	4.0	.0	8.0	.0
4.0	.0	12.0	.0	12.0	.0	4.0	.0	8.0	.0	4.0	.0
8.0	.0	8.0	.0	48.0	.0	4.0	.0	12.0	.0	12.0	.0
2.0	6.0	6.0	2.0	6.0	6.0	2.0	2.0	4.0	2.0	2.0	4.0
4.0	2.0	4.0	6.0	24.0	6.0	2.0	2.0	6.0	4.0	6.0	2.0
6.0	4.0	2.0	4.0	6.0	24.0	2.0	2.0	2.0	6.0	4.0	6.0

cat ccobnd1

CCOBND1

2160.	0.	360.	0.	360.	0.	108.	0.	36.	0.
144.	0.	108.	0.	144.	0.	36.	0.	72.	0.
240.	720.	96.	120.	48.	120.	96.	36.	24.	12.
72.	48.	24.	36.	24.	48.	16.	12.	24.	24.
240.	720.	48.	120.	96.	120.	24.	36.	16.	12.
24.	48.	96.	36.	72.	48.	24.	12.	24.	24.
144.	0.	108.	0.	36.	0.	360.	0.	72.	0.
144.	0.	36.	0.	24.	0.	36.	0.	36.	0.
48.	48.	24.	36.	24.	12.	48.	120.	24.	24.
24.	48.	48.	12.	24.	8.	24.	12.	16.	12.
240.	48.	96.	36.	48.	12.	96.	120.	24.	24.
72.	48.	24.	12.	24.	8.	16.	12.	24.	12.
144.	0.	36.	0.	108.	0.	36.	0.	36.	0.
24.	0.	360.	0.	144.	0.	72.	0.	36.	0.
240.	48.	48.	12.	96.	36.	24.	12.	16.	12.
24.	8.	96.	120.	72.	48.	24.	24.	24.	12.
48.	48.	24.	12.	24.	36.	48.	12.	24.	12.
24.	8.	48.	120.	24.	48.	24.	24.	16.	12.
24.	120.	12.	24.	12.	48.	24.	12.	12.	8.
12.	12.	24.	48.	12.	36.	12.	12.	8.	12.
120.	48.	24.	48.	12.	36.	12.	12.	8.	12.

% cat ccobnd2

CCBND2

360.	.	144.	.	72.	.	144.	.	36.	.
108.	.	36.	.	36.	.	24.	.	36.	.
96.	120.	72.	48.	24.	24.	240.	48.	48.	12.
96.	36.	24.	12.	16.	12.	24.	8.	24.	12.
48.	120.	24.	48.	24.	24.	48.	48.	24.	12.
24.	36.	48.	12.	24.	12.	24.	8.	16.	12.
108.	.	144.	.	36.	.	2160.	.	360.	.
360.	.	108.	.	36.	.	144.	.	72.	.
24.	36.	24.	48.	16.	12.	240.	720.	96.	120.
48.	120.	96.	36.	24.	12.	72.	48.	24.	24.
96.	36.	72.	48.	24.	12.	240.	720.	48.	120.
96.	120.	24.	36.	16.	12.	24.	48.	24.	24.
36.	.	24.	.	36.	.	144.	.	108.	.
36.	.	360.	.	72.	.	144.	.	36.	.
48.	12.	24.	8.	24.	12.	48.	48.	24.	36.
24.	12.	48.	120.	24.	24.	24.	48.	16.	12.
24.	12.	24.	8.	16.	12.	240.	48.	96.	36.
48.	12.	96.	120.	24.	24.	72.	48.	24.	12.
12.	24.	12.	12.	8.	12.	120.	24.	48.	12.
24.	12.	48.	24.	12.	12.	36.	12.	12.	8.
48.	36.	12.	120.	24.	48.	12.	8.	12.	12.

% cat ccostr3

```
                          CCOSTR3

  432.0      .0      .0    72.0      .0   144.0      .0   216.0      .0   144.0
     .0    72.0      .0      .0    72.0      .0    24.0      .0    72.0      .0
   24.0      .0   432.0      .0      .0   216.0      .0   144.0      .0    72.0
     .0   144.0      .0
   72.0   144.0      .0    32.0    24.0    24.0    48.0    48.0    72.0    24.0
   48.0    72.0    24.0      .0    48.0    24.0    24.0     8.0    32.0 [illegible]
   24.0     8.0   144.0   144.0      .0    48.0    72.0    48.0    48.0    48.0
   24.0    48.0    48.0
  216.0   144.0      .0    48.0    24.0    72.0    48.0   192.0    72.0    72.0
   48.0    72.0    24.0      .0    96.0    24.0    24.0     8.0    48.0    24.0
   24.0     8.0   720.0   144.0      .0   192.0    72.0   240.0    48.0    96.0
   24.0   240.0    48.0
   72.0      .0      .0    72.0      .0    24.0      .0    72.0      .0    24.0
     .0   432.0      .0      .0   216.0      .0   144.0      .0    72.0      .0
  144.0      .0   432.0      .0      .0    72.0      .0   144.0      .0   216.0
     .0   144.0      .0
   72.0    24.0      .0    48.0    24.0    24.0     8.0    96.0    24.0    24.0
    8.0   216.0   144.0      .0   192.0    72.0    72.0    48.0    48.0    24.0
   72.0    48.0   720.0   144.0      .0    96.0    24.0   240.0    48.0   192.0
   72.0   240.0    48.0
   72.0    24.0      .0    32.0    24.0    24.0     8.0    48.0    24.0    24.0
    8.0    72.0   144.0      .0    48.0    72.0    24.0    48.0    32.0    24.0
   24.0    48.0   144.0   144.0      .0    48.0    24.0    48.0    48.0    48.0
   72.0    48.0    48.0
  432.0      .0      .0   144.0      .0   144.0      .0   720.0      .0   144.0
     .0   432.0      .0      .0   720.0      .0   144.0      .0   144.0      .0
  144.0      .0  6480.0      .0      .0   720.0      .0  2160.0      .0   720.0
     .0  2160.0      .0
  216.0   144.0      .0    48.0    48.0    72.0    48.0   192.0   240.0    72.0
   48.0    72.0   144.0      .0    96.0   240.0    24.0    48.0    48.0    48.0
   24.0    48.0   720.0  2160.0      .0   192.0   240.0   240.0   720.0    96.0
  240.0   240.0   720.0
   72.0   144.0      .0    48.0    48.0    24.0    48.0    96.0   240.0    24.0
   48.0   216.0   144.0      .0   192.0   240.0    72.0    48.0    48.0    48.0
   72.0    48.0   720.0  2160.0      .0    96.0   240.0   240.0   720.0   192.0
  240.0   240.0   720.0
   36.0   108.0    36.0    24.0    36.0    16.0      .0    48.0    96.0    36.0
     .0   108.0    36.0    36.0    96.0    48.0    36.0      .0    36.0    24.0
   16.0      .0   360.0   360.0    72.0    72.0    96.0    24.0      .0    96.0
   72.0    24.0      .0
   36.0    12.0    36.0    12.0    36.0    12.0    36.0    12.0   120.0   120.0
  120.0   120.0    48.0    48.0    24.0    48.0    24.0     8.0
```

% cat ccostr2

```
                 CCOSTR2
  432.0      .0      .0   216.0      .0   144.0      .0    72.0      .0   144.0
     .0   432.0      .0      .0    72.0      .0   144.0      .0   216.0      .0
  144.0      .0    72.0      .0      .0    72.0      .0    24.0      .0    72.0
     .0    24.0      .0
  216.0   144.0      .0   192.0    72.0    72.0    48.0    48.0    24.0    72.0
   48.0   720.0   144.0      .0    96.0    24.0   240.0    48.0   192.0    72.0
  240.0    48.0    72.0    24.0      .0    48.0    24.0    24.0     8.0    96.0
   24.0    24.0     8.0
   72.0   144.0      .0    48.0    72.0    24.0    48.0    32.0    24.0    24.0
   48.0   144.0   144.0      .0    48.0    24.0    48.0    48.0    48.0    72.0
   48.0    48.0    72.0    24.0      .0    32.0    24.0    24.0     8.0    48.0
   24.0    24.0     8.0
  432.0      .0      .0   720.0      .0   144.0      .0   144.0      .0   144.0
     .0  6480.0      .0      .0   720.0      .0  2160.0      .0   720.0      .0
 2160.0      .0   432.0      .0      .0   144.0      .0   144.0      .0   720.0
     .0   144.0      .0
   72.0   144.0      .0    96.0   240.0    24.0    48.0    48.0    48.0    24.0
   48.0   720.0  2160.0      .0   192.0   240.0   240.0   720.0    96.0   240.0
  240.0   720.0   216.0   144.0      .0    48.0    48.0    72.0    48.0   192.0
  240.0    72.0    48.0
  216.0   144.0      .0   192.0   240.0    72.0    48.0    48.0    48.0    72.0
   48.0   720.0  2160.0      .0    96.0   240.0   240.0   720.0   192.0   240.0
  240.0   720.0    72.0   144.0      .0    48.0    48.0    24.0    48.0    96.0
  240.0    24.0    48.0
   72.0      .0      .0    72.0      .0    24.0      .0    72.0      .0    24.0
     .0   432.0      .0      .0   216.0      .0   144.0      .0    72.0      .0
  144.0      .0   432.0      .0      .0    72.0      .0   144.0      .0   216.0
     .0   144.0      .0
   72.0    24.0      .0    48.0    24.0    24.0     8.0    32.0    24.0    24.0
    8.0   144.0   144.0      .0    48.0    72.0    48.0    48.0    48.0    24.0
   48.0    48.0    72.0   144.0      .0    32.0    24.0    24.0    48.0    48.0
   72.0    24.0    48.0
   72.0    24.0      .0    96.0    24.0    24.0     8.0    48.0    24.0    24.0
    8.0   720.0   144.0      .0   192.0    72.0   240.0    48.0    96.0    24.0
  240.0    48.0   216.0   144.0      .0    48.0    24.0    72.0    48.0   192.0
   72.0    72.0    48.0
   36.0    36.0   108.0    48.0    36.0    96.0      .0    24.0    16.0    36.0
     .0   360.0    72.0   360.0    96.0    24.0    72.0      .0    72.0    24.0
   96.0      .0   108.0    36.0    36.0    36.0    16.0    24.0      .0    96.0
   36.0    48.0      .0
   12.0    36.0    12.0    36.0   120.0   120.0   120.0   120.0    36.0    12.0
   36.0    12.0    48.0    24.0    48.0     8.0    24.0    48.0
```

```
% cat ccostr1


                  CCOSTR1

  6480.0      .0      .0   720.0      .0  2160.0      .0   720.0      .0  2160.0
      .0   432.0      .0      .0   144.0      .0   144.0      .0   720.0      .0
   144.0      .0   432.0      .0      .0   720.0      .0   144.0      .0   144.0
      .0   144.0      .0
   720.0  2160.0      .0   192.0   240.0   240.0   720.0    96.0   240.0   240.0
   720.0   216.0   144.0      .0    48.0    48.0    72.0    48.0   192.0   240.0
    72.0    48.0    72.0   144.0      .0    96.0   240.0    24.0    48.0    48.0
    48.0    24.0    48.0
   720.0  2160.0      .0    96.0   240.0   240.0   720.0   192.0   240.0   240.0
   720.0    72.0   144.0      .0    48.0    48.0    24.0    48.0    96.0   240.0
    24.0    48.0   216.0   144.0      .0   192.0   240.0    72.0    48.0    48.0
    48.0    72.0    48.0
   432.0      .0      .0   216.0      .0   144.0      .0    72.0      .0   144.0
      .0   432.0      .0      .0    72.0      .0   144.0      .0   216.0      .0
   144.0      .0    72.0      .0      .0    72.0      .0    24.0      .0    72.0
      .0    24.0      .0
   144.0   144.0      .0    48.0    72.0    48.0    48.0    48.0    24.0    48.0
    48.0    72.0   144.0      .0    32.0    24.0    24.0    48.0    48.0    72.0
    24.0    48.0    72.0    24.0      .0    48.0    24.0    24.0     8.0    32.0
    24.0    24.0     8.0
   720.0   144.0      .0   192.0    72.0   240.0    48.0    96.0    24.0   240.0
    48.0   216.0   144.0      .0    48.0    24.0    72.0    48.0   192.0    72.0
    72.0    48.0    72.0    24.0      .0    96.0    24.0    24.0     8.0    48.0
    24.0    24.0     8.0
   432.0      .0      .0    72.0      .0   144.0      .0   216.0      .0   144.0
      .0    72.0      .0      .0    72.0      .0    24.0      .0    72.0      .0
    24.0      .0   432.0      .0      .0   216.0      .0   144.0      .0    72.0
      .0   144.0      .0
   720.0   144.0      .0    96.0    24.0   240.0    48.0   192.0    72.0   240.0
    48.0    72.0    24.0      .0    48.0    24.0    24.0     8.0    96.0    24.0
    24.0     8.0   216.0   144.0      .0   192.0    72.0    72.0    48.0    48.0
    24.0    72.0    48.0
   144.0   144.0      .0    48.0    24.0    48.0    48.0    48.0    72.0    48.0
    48.0    72.0    24.0      .0    32.0    24.0    24.0     8.0    48.0    24.0
    24.0     8.0    72.0   144.0      .0    48.0    72.0    24.0    48.0    32.0
    24.0    24.0    48.0
    72.0   360.0   360.0    24.0    72.0    96.0      .0    24.0    96.0    72.0
      .0    36.0    36.0   108.0    16.0    24.0    36.0      .0    36.0    48.0
    96.0      .0    36.0   108.0    36.0    36.0    96.0    48.0      .0    16.0
    36.0    24.0      .0
   120.0   120.0   120.0   120.0    12.0    36.0    12.0    36.0    12.0    36.0
    12.0    36.0     8.0    24.0    24.0    48.0    48.0    48.0
```

% cat costr5

QCOSTR5

8.0	.0	8.0	.0	8.0	.0	4.0	.0	4.0
.0	4.0	.0	4.0	.0	4.0	.0	4.0	.0
8.0	.0	24.0	.0	16.0	.0	12.0	.0	4.0
.0	8.0	.0	12.0	.0	4.0	.0	8.0	.0
8.0	.0	16.0	.0	24.0	.0	8.0	.0	4.0
.0	12.0	.0	8.0	.0	4.0	.0	12.0	.0
4.0	4.0	12.0	4.0	8.0	4.0	6.0	2.0	2.0
2.0	4.0	2.0	6.0	2.0	2.0	2.0	4.0	2.0
4.0	4.0	8.0	12.0	12.0	8.0	4.0	6.0	2.0
2.0	6.0	4.0	4.0	6.0	2.0	2.0	6.0	4.0
4.0	4.0	4.0	8.0	4.0	12.0	2.0	4.0	2.0
2.0	2.0	6.0	2.0	4.0	2.0	2.0	2.0	6.0

% cat costr6

QCOSTR6

24.0	.0	8.0	.0	16.0	.0	4.0	.0	12.0
.0	8.0	.0	4.0	.0	12.0	.0	8.0	.0
8.0	.0	8.0	.0	8.0	.0	4.0	.0	4.0
.0	4.0	.0	4.0	.0	4.0	.0	4.0	.0
16.0	.0	8.0	.0	24.0	.0	4.0	.0	8.0
.0	12.0	.0	4.0	.0	8.0	.0	12.0	.0
4.0	12.0	4.0	4.0	4.0	8.0	2.0	2.0	2.0
6.0	2.0	4.0	2.0	2.0	2.0	6.0	2.0	4.0
8.0	4.0	4.0	4.0	12.0	4.0	2.0	2.0	4.0
2.0	6.0	2.0	2.0	2.0	4.0	2.0	6.0	2.0
12.0	8.0	4.0	4.0	8.0	12.0	2.0	2.0	6.0
4.0	4.0	6.0	2.0	2.0	6.0	4.0	4.0	6.0

```
% cat costr3



                QCOSTR3

  16.0     .0    8.0     .0   24.0     .0    4.0     .0    8.0
    .0   12.0     .0    4.0     .0    8.0     .0   12.0     .0
   8.0     .0   16.0     .0   24.0     .0    8.0     .0    4.0
    .0   12.0     .0    8.0     .0    4.0     .0   12.0     .0
  24.0     .0   24.0     .0   96.0     .0   12.0     .0   12.0
    .0   48.0     .0   12.0     .0   12.0     .0   48.0     .0
   4.0    8.0    8.0    4.0   12.0   12.0    4.0    2.0    2.0
   4.0    6.0    6.0    4.0    2.0    2.0    4.0    6.0    6.0
  12.0    4.0   12.0    8.0   48.0   12.0    6.0    4.0    6.0
   2.0   24.0    6.0    6.0    4.0    6.0    2.0   24.0    6.0
   8.0   12.0    4.0   12.0   12.0   48.0    2.0    6.0    4.0
   6.0    6.0   24.0    2.0    6.0    4.0    6.0    6.0   24.0
% cat costr4



                QCOSTR4

  24.0     .0   16.0     .0    8.0     .0    8.0     .0   12.0
    .0    4.0     .0    8.0     .0   12.0     .0    4.0     .0
  16.0     .0   24.0     .0    8.0     .0   12.0     .0    8.0
    .0    4.0     .0   12.0     .0    8.0     .0    4.0     .0
   8.0     .0    8.0     .0    8.0     .0    4.0     .0    4.0
    .0    4.0     .0    4.0     .0    4.0     .0    4.0     .0
   8.0   12.0   12.0    8.0    4.0    4.0    6.0    4.0    4.0
   6.0    2.0    2.0    6.0    4.0    4.0    6.0    2.0    2.0
   4.0    8.0    4.0   12.0    4.0    4.0    2.0    6.0    2.0
   4.0    2.0    2.0    2.0    6.0    2.0    4.0    2.0    2.0
  12.0    4.0    8.0    4.0    4.0    4.0    4.0    2.0    6.0
   2.0    2.0    2.0    4.0    2.0    6.0    2.0    2.0    2.0
```

```
% cat costr1



            QCOSTR1

   96.0     .0   24.0     .0   24.0     .0   12.0     .0   48.0
     .0   12.0     .0   12.0     .0   48.0     .0   12.0     .0
   24.0     .0   16.0     .0    8.0     .0    8.0     .0   12.0
     .0    4.0     .0    8.0     .0   12.0     .0    4.0     .0
   24.0     .0    8.0     .0   16.0     .0    4.0     .0   12.0
     .0    8.0     .0    4.0     .0   12.0     .0    8.0     .0
   12.0   48.0    8.0   12.0    4.0   12.0    4.0    6.0    6.0
   24.0    2.0    6.0    4.0    6.0    6.0   24.0    2.0    6.0
   12.0   12.0    4.0    8.0    8.0    4.0    2.0    4.0    6.0
    6.0    4.0    2.0    2.0    4.0    6.0    6.0    4.0    2.0
   48.0   12.0   12.0    4.0   12.0    8.0    6.0    2.0   24.0
    6.0    6.0    4.0    6.0    2.0   24.0    6.0    6.0    4.0
% cat costr2
```

```
                  QCOSTR2

   16.0     .0   24.0     .0    8.0     .0   12.0     .0    8.0
     .0    4.0     .0   12.0     .0    8.0     .0    4.0     .0
   24.0     .0   96.0     .0   24.0     .0   48.0     .0   12.0
     .0   12.0     .0   48.0     .0   12.0     .0   12.0     .0
    8.0     .0   24.0     .0   16.0     .0   12.0     .0    4.0
     .0    8.0     .0   12.0     .0    4.0     .0    8.0     .0
   12.0    8.0   48.0   12.0   12.0    4.0   24.0    6.0    6.0
    4.0    6.0    2.0   24.0    6.0    6.0    4.0    6.0    2.0
    4.0   12.0   12.0   48.0    8.0   12.0    6.0   24.0    2.0
    6.0    4.0    6.0    6.0   24.0    2.0    6.0    4.0    6.0
    8.0    4.0   12.0   12.0    4.0    8.0    6.0    6.0    4.0
    2.0    2.0    4.0    6.0    6.0    4.0    2.0    2.0    4.0
```

column 9

$$
\begin{bmatrix}
g_1g_3 & 0 & g_1g_2 & 0 \\
g_1g_3 & g_2g_3 & g_1g_2 & g_2^2 \\
g_1g_3 & g_3^2 & g_1g_2 & g_2g_3 \\
g_3g_2 & 0 & g_2^2 & 0 \\
g_3g_2 & g_3^2 & g_2^2 & g_3g_2 \\
g_3g_2 & g_3g_1 & g_2^2 & g_1g_2 \\
g_3^2 & 0 & g_2g_3 & 0 \\
g_3^2 & g_3g_1 & g_2g_3 & g_1g_2 \\
g_3^2 & g_2g_3 & g_2g_3 & g_2^2 \\
g_1g_3 & g_2g_3 & g_3^2 & 0
\end{bmatrix}
$$

Figure D-2
(continued)

column 5				column 6			
g_1g_2	0	g_1g_3	0	g_1g_2	0	g_1^2	0
g_1g_2	g_2^2	g_1g_3	g_2g_3	g_1g_2	g_2^2	g_1^2	g_1g_2
g_1g_2	g_2g_3	g_1g_3	g_3^2	g_1g_2	g_2g_3	g_1^2	g_1g_3
g_2^2	0	g_2g_3	0	g_2^2	0	g_1g_2	0
g_2^2	g_2g_3	g_2g_3	g_3^2	g_2^2	g_2g_3	g_1g_2	g_1g_3
g_2^2	g_1g_2	g_2g_3	g_1g_3	g_2^2	g_1g_2	g_1g_2	g_1^2
g_2g_3	0	g_3^2	0	g_2g_3	0	g_1g_3	0
g_2g_3	g_1g_2	g_3^2	g_1g_3	g_2g_3	g_1g_2	g_1g_3	g_1^2
g_2g_3	g_2^2	g_3^2	g_2g_3	g_2g_3	g_2^2	g_1g_3	g_1g_2
g_1g_2	g_2^2	g_2g_3	0	g_1g_2	g_2^2	g_2g_3	0

column 7			column 8			
g_1g_3	0	0	g_1g_3	0	g_1^2	0
g_1g_3	g_2g_3	0	g_1g_3	g_2g_3	g_1^2	g_1g_2
g_1g_3	g_3^2	0	g_1g_3	g_3^2	g_1^2	g_1g_3
g_2g_3	0	0	g_2g_3	0	g_1g_2	0
g_2g_3	g_3^2	0	g_2g_3	g_3^2	g_1g_2	g_1g_3
g_2g_3	g_1g_3	0	g_2g_3	g_1g_3	g_1g_2	g_1^2
g_3^2	0	0	g_3^2	0	g_1g_3	0
g_3^2	g_1g_3	0	g_3^2	g_1g_3	g_1g_3	g_1^2
g_3^2	g_2g_3	0	g_3^2	g_2g_3	g_1g_3	g_1g_2
g_1g_3	g_2g_3	g_3^2	g_1g_3	g_2g_3	g_3^2	0

Figure D-2
(continued)

column 1 column 2

$$\underline{\underline{DS}} = \left[\begin{array}{ccc|cccc}
g_1^2 & 0 & 0 & g_1^2 & g_1 g_2 & & \\
g_1^2 & g_1 g_2 & 0 & g_1^2 & g_1 g_2 & g_1 g_2 & g_2^2 \\
g_1^2 & g_1 g_3 & 0 & g_1^2 & g_1 g_3 & g_1 g_2 & g_2 g_3 \\
g_1 g_2 & 0 & 0 & g_1 g_2 & 0 & g_2^2 & 0 \\
g_1 g_2 & g_1 g_3 & 0 & g_1 g_2 & g_1 g_3 & g_2^2 & g_2 g_3 \\
g_1 g_2 & g_1^2 & 0 & g_1 g_2 & g_1^2 & g_2^2 & g_1 g_2 \\
g_1 g_3 & 0 & 0 & g_1 g_3 & 0 & g_2 g_3 & 0 \\
g_1 g_3 & g_1^2 & 0 & g_1 e_3 & g_1^2 & g_2 g_3 & g_1 g_2 \\
g_1 g_3 & g_1 g_2 & 0 & g_1 g_3 & g_1 g_2 & g_2 g_3 & g_2^2 \\
g_1^2 & g_1 g_2 & g_1 g_3 & g_1^2 & g_1 g_2 & g_1 g_3 & 0
\end{array}\right.$$

column 3 column 4

$$\left.\begin{array}{cccc|ccc}
g_1^2 & g_1 g_3 & 0 & 0 & g_1 g_2 & 0 & 0 \\
g_1^2 & g_1 g_2 & g_1 g_3 & g_2 g_3 & g_1 g_2 & g_2^2 & 0 \\
g_1^2 & g_1 g_3 & g_1 g_3 & g_3^2 & g_1 g_2 & g_2 g_3 & 0 \\
g_1 g_2 & g_2 g_3 & 0 & 0 & g_2^2 & 0 & 0 \\
g_1 g_2 & g_1 g_3 & g_2 g_3 & g_3^2 & g_2^2 & g_2 g_3 & 0 \\
g_1 g_2 & g_1^2 & g_2 g_3 & g_1 g_3 & g_2^2 & g_1 g_2 & 0 \\
g_1 g_3 & g_3^2 & 0 & 0 & g_2 g_3 & 0 & 0 \\
g_1 g_3 & g_1^2 & g_3^2 & g_1 g_3 & g_2 g_3 & g_1 g_2 & 0 \\
g_1 g_3 & g_1 g_2 & g_3^2 & g_2 g_3 & g_2 g_3 & g_2^2 & 0 \\
g_1^2 & g_1 g_2 & g_1 g_3 & 0 & g_1 g_2 & g_2^2 & g_2 g_3
\end{array}\right|$$

Figure D-2
Overlayed Matrix of Derivatives For Streaming Matrix with Cubic Fit

$$
\underline{\underline{DS}} =
\begin{array}{cc|cc|cc|cccc}
\multicolumn{2}{c}{\text{column 1}} & \multicolumn{2}{c}{\text{column 2}} & \multicolumn{2}{c}{\text{column 3}} & \multicolumn{4}{c}{\text{column 4}} \\
g_1 & 0 & g_1 & 0 & g_1 & 0 & g_1^2 & 0 & g_1 g_2 & 0 \\
g_2 & 0 & g_2 & 0 & g_2 & 0 & g_2 g_1 & 0 & g_2^2 & 0 \\
g_3 & 0 & g_3 & 0 & g_3 & 0 & g_3 g_1 & 0 & g_3 g_2 & 0 \\
g_1 & g_1 g_2 & g_1 & g_2^2 & g_1 & g_2 g_3 & g_1^2 & g_2^2 & g_1^2 & g_2^2 \\
g_2 & g_1 g_3 & g_2 & g_2 g_3 & g_2 & g_3^2 & g_2 g_1 & g_3 g_3 & g_2^2 & g_2 g_3 \\
g_3 & g_1^2 & g_3 & g_1 g_2 & g_3 & g_1 g_3 & g_3 g_1 & g_1^2 & g_2 g_3 & g_1 g_2
\end{array}
$$

$$
\begin{array}{cccc|cccc}
\multicolumn{4}{c}{\text{column 5}} & \multicolumn{4}{c}{\text{column 6}} \\
g_1 g_2 & 0 & g_1 g_3 & 0 & g_1 g_3 & 0 & g_1^2 & 0 \\
g_2^2 & 0 & g_2 g_3 & 0 & g_2 g_3 & 0 & g_1 g_2 & 0 \\
g_2 g_3 & 0 & g_3^2 & 0 & g_3^2 & 0 & g_1 g_3 & 0 \\
g_1 g_2 & g_2^2 & g_1 g_3 & g_2 g_3 & g_1 g_3 & g_2 g_3 & g_1^2 & g_1 g_2 \\
g_2^2 & g_2 g_3 & g_2 g_3 & g_3^2 & g_2 g_3 & g_3^2 & g_1 g_2 & g_1 g_3 \\
g_2 g_3 & g_1 g_2 & g_3^2 & g_1 g_3 & g_3^2 & g_1 g_3 & g_1 g_3 & g_1^2
\end{array}
$$

Figure D-1
Overlayed Matrix of Derivatives For Streaming
Matrix with Quadratic Fit

Column 10 of the streaming matrix is found by noting that it is the transpose of row ten.

with groups of 2 or 3 columns at a time, as divided by dotted lines on fig C-1, until all 10 columns of the streaming matrix are assembled.

In the cubic case, the arrays listed as CCOSTRI, CCOSTR2, .., CCOSTR6 represent 10 x 33 matrices of constants, for rows 1 through 9 of the 6 streaming matrices with a (1 X 18) row matrix below to augment row 10 terms. DS is a 10 x 33 matrix of , which when overlayed on the sum of CCOSTRI thru 6, multiplied by the appropriate U 'S and ( ) from the integration, forms the streaming matrix.

Column 1-3 of DS overlayed on columns 1-3 of the COSTR sum produce the first column of MS. Subsequent columns of the streaming matrix are founed by the next 3 or 4 columns of DS, overlayed on the corresponding CCOSTR columns, as separated by the dotted lines in DS of figure c-2.

Row 10 of the streaming matrices must be augmented by the dimension (18) matrix (SRI,SR2,..,SR6) on the last two lines of COSTR, such that

For $i=1,18$

$$SR(i)=\left[u_1^2 SR1(i)+u_2^2 SR2(i)+u_3^2 SR3(i)+2u_1u_2 SR4(i)+2u_2u_3 SR5(i)+2u_1u_3 SR6(i)\right] * 2A/9!$$

AND

$$MS(10,2)=MS(10,2)+SR(1)g_2^2+SR(2)g_2g_3$$
$$MS(10,3)=MS(10,3)+SR(3)g_2g_3+SR(4)g_3^2$$
$$MS(10,5)=MS(10,5)+SR(5)g_1g_3+SR(6)g_3^2$$
$$MS(10,6)=MS(10,6)+SR(7)g_1^2+SR(8)g_1g_3$$
$$MS(10,8)=MS(10,8)+SR(9)g_1^2+SR(10)g_1g_2$$
$$MS(10,9)=MS(10,9)+SR(11)g_1g_2+SR(12)g_2^2$$
$$MS(10,10)=SR(13)g_1^2+SR(14)g_1g_2+SR(15)g_1g_3+SR(16)g_2^2+SR(17)g_2g_3+SR(18)g_3^2$$

Appendix D - The Streaming Term (quadratic and cubic)

The Streaming term is

$$\frac{1}{2}\int dA \left(\mu \frac{\partial \phi}{\partial x}\right)^2 \tag{D-1}$$

which can be written as the sum of 6 distinct matrices

$$= \frac{1}{2}\ \underline{\tilde{\varphi}}\ \underline{\underline{\tilde{G}^T}}\left[\underline{\underline{MS1}} + \underline{\underline{MS2}} + \underline{\underline{MS3}} + \underline{\underline{MS4}} + \underline{\underline{MS5}} + \underline{\underline{MS6}}\right]\underline{\underline{G^T}}\ \underline{\varphi}$$

$$= \frac{1}{2}\ \underline{\hat{\varphi}}\ \underline{\underline{\tilde{G}^T}}\ \underline{\underline{MS}}\ \underline{\underline{G^T}}\ \underline{\varphi} \tag{3-13}$$

Evaluating these mtrices involves taking the product $\underline{m}_\gamma \tilde{\underline{m}}_\gamma$ which results in cross products of the natural coordinates derivatives with respect to cartesian coordinates. The six streaming matrices, as in the boundary case (Appendix C) can be thought of as distinct matrices of constants, which after being multiplied respectively by μ_1^2, μ_2^2, μ_3^2, $2\mu_1\mu_2$, $2\mu_2\mu_3$ and $2\mu_1\mu_3$ can be summed, and then "overlayed" by a matrix of g_i's. Due to the cross products, this matrix of derivatives ($\underline{\underline{DS}}$) is complicated. It is generated in Subroutine Stream, (appendix A) for the cubic case, and written out below for both the quadratic and cubic cases.

DS for the quadratic case is listed in figure d-1. After multiplying the 6 QCOSTR matrices by the appropriate u values; and a factor of ($2A/6!$) from the integration, they may be summed to form a single 6x18 matrix. The first two columns of this matrix overlay on the first two columns of DS to form the first column of the streaming matrix. The process continues

% cat ccobnd3

CCOBND3

360.	.	72.	.	144.	.	36.	.	24.	.
36.	.	144.	.	108.	.	36.	.	36.	.
48.	120.	24.	24.	24.	48.	48.	12.	24.	8.
24.	12.	48	48.	24.	36.	24.	12.	16.	12.
96.	120.	24.	24.	72.	48.	24.	12.	24.	8.
16.	12.	240.	48.	96.	36.	48.	12.	24.	12.
36.	.	36.	.	24.	.	360.	.	144.	.
72.	.	144.	.	36.	.	108.	.	36.	.
24.	12.	16.	12.	24.	8.	96.	120.	72.	48.
24.	24.	240.	48.	48.	12.	96.	36.	24.	12.
48.	12.	24.	12.	24.	8.	48.	120.	24.	48.
24.	24.	48.	48.	24.	12.	24.	36.	16.	12.
108.	.	36.	.	144.	.	108.	.	144.	.
36.	.	2160.	.	360.	.	360.	.	72.	.
96.	36.	24.	12.	72.	48.	24.	36.	24.	48.
16.	12.	240.	720.	96.	120.	48.	120.	24.	24.
24.	36.	16.	12.	24.	48.	96.	36.	72.	48.
24.	12.	240.	720.	48.	120.	96.	120.	24.	24.
12.	48.	8.	12.	12.	36.	48.	12.	36.	12.
12.	8.	120.	120.	24.	48.	48.	24.	12.	12.
24.	12.	12.	24.	12.	12.	24.	12.	12.	8.

0.

```
% cat ccostr4


                            CCOSTR4
    1080.0        .0        .0     288.0        .0     360.0        .0     144.0        .0     360.0
        .0     324.0        .0        .0      72.0        .0     108.0        .0     288.0        .0
     108.0        .0     108.0        .0        .0     144.0        .0      36.0        .0      72.0
        .0      36.0        .0
     288.0     360.0        .0     144.0      96.0      96.0     120.0      48.0      48.0      96.0
     120.0     288.0     108.0        .0      48.0      24.0      96.0      36.0     144.0      96.0
      96.0      36.0      48.0      36.0        .0      48.0      48.0      16.0      12.0      48.0
      24.0      16.0      12.0
     144.0     360.0        .0      48.0      96.0      48.0     120.0      48.0      48.0      48.0
     120.0      72.0     108.0        .0      32.0      24.0      24.0      36.0      48.0      96.0
      24.0      36.0      72.0      36.0        .0      48.0      48.0      24.0      12.0      32.0
      24.0      24.0      12.0
     324.0        .0        .0     288.0        .0     108.0        .0      72.0        .0     108.0
        .0    1080.0        .0        .0     144.0        .0     360.0        .0     288.0        .0
     360.0        .0     108.0        .0        .0      72.0        .0      36.0        .0     144.0
        .0      36.0        .0
      72.0     108.0        .0      48.0      96.0      24.0      36.0      32.0      24.0      24.0
      36.0     144.0     360.0        .0      48.0      48.0      48.0     120.0      48.0      96.0
      48.0     120.0      72.0      36.0        .0      32.0      24.0      24.0      12.0      48.0
      48.0      24.0      12.0
     288.0     108.0        .0     144.0      96.0      96.0      36.0      48.0      24.0      96.0
      36.0     288.0     360.0        .0      48.0      48.0      96.0     120.0     144.0      96.0
      96.0     120.0      48.0      36.0        .0      48.0      24.0      16.0      12.0      48.0
      48.0      16.0      12.0
     108.0        .0        .0      48.0        .0      36.0        .0      72.0        .0      36.0
        .0     108.0        .0        .0      72.0        .0      36.0        .0      48.0        .0
      36.0        .0     216.0        .0        .0      72.0        .0      72.0        .0      72.0
        .0      72.0        .0
     144.0      36.0        .0      48.0      16.0      48.0      12.0      48.0      24.0      48.0
      12.0      72.0      36.0        .0      32.0      24.0      24.0      12.0      48.0      16.0
      24.0      12.0      72.0      72.0        .0      48.0      24.0      24.0      24.0      32.0
      24.0      24.0      24.0
      72.0      36.0        .0      48.0      16.0      24.0      12.0      32.0      24.0      24.0
      12.0     144.0      36.0        .0      48.0      24.0      48.0      12.0      48.0      16.0
      48.0      12.0      72.0      72.0        .0      32.0      24.0      24.0      24.0      48.0
      24.0      24.0      24.0
      36.0      72.0     144.0      24.0      36.0      72.0        .0      16.0      24.0      36.0
        .0      72.0      36.0     144.0      24.0      16.0      36.0        .0      36.0      24.0
      72.0        .0      36.0      36.0      24.0      24.0      24.0      24.0        .0      24.0
      24.0      24.0        .0
      24.0      48.0      24.0      48.0      24.0      48.0      24.0      48.0      12.0      12.0
      12.0      12.0      12.0      16.0      24.0      12.0      24.0      36.0
```

```
% cat ccostr5
```

CCOSTR5

```
    216.0        .0        .0      72.0        .0      72.0        .0      72.0        .0      72.0
       .0     108.0        .0        .0      48.0        .0      36.0        .0      72.0        .0
     36.0        .0     108.0        .0        .0      72.0        .0      36.0        .0      48.0
       .0      36.0        .0
     72.0      72.0        .0      48.0      24.0      24.0      24.0      32.0      24.0      24.0
     24.0     144.0      36.0        .0      48.0      16.0      48.0      12.0      48.0      24.0
     48.0      12.0      72.0      36.0        .0      32.0      24.0      24.0      12.0      48.0
     16.0      24.0      12.0
     72.0      72.0        .0      32.0      24.0      24.0      24.0      48.0      24.0      24.0
     24.0      72.0      36.0        .0      48.0      16.0      24.0      12.0      32.0      24.0
     24.0      12.0     144.0      36.0        .0      48.0      24.0      48.0      12.0      48.0
     16.0      48.0      12.0
    108.0        .0        .0     144.0        .0      36.0        .0      72.0        .0      36.0
       .0    1080.0        .0        .0     288.0        .0     360.0        .0     144.0        .0
    360.0        .0     324.0        .0        .0      72.0        .0     108.0        .0     288.0
       .0     108.0        .0
     48.0      36.0        .0      48.0      48.0      16.0      12.0      48.0      24.0      16.0
     12.0     288.0     360.0        .0     144.0      96.0      96.0     120.0      48.0      48.0
     96.0     120.0     288.0     108.0        .0      48.0      24.0      96.0      36.0     144.0
     96.0      96.0      36.0
     72.0      36.0        .0      48.0      48.0      24.0      12.0      32.0      24.0      24.0
     12.0     144.0     360.0        .0      48.0      96.0      48.0     120.0      48.0      48.0
     48.0     120.0      72.0     108.0        .0      32.0      24.0      24.0      36.0      48.0
     96.0      24.0      36.0
    108.0        .0        .0      72.0        .0      36.0        .0     144.0        .0      36.0
       .0     324.0        .0        .0     288.0        .0     108.0        .0      72.0        .0
    108.0        .0    1080.0        .0        .0     144.0        .0     360.0        .0     288.0
       .0     360.0        .0
     72.0      36.0        .0      32.0      24.0      24.0      12.0      48.0      48.0      24.0
     12.0      72.0     108.0        .0      48.0      96.0      24.0      36.0      32.0      24.0
     24.0      36.0     144.0     360.0        .0      48.0      48.0      48.0     120.0      48.0
     96.0      48.0     120.0
     48.0      36.0        .0      48.0      24.0      16.0      12.0      48.0      48.0      16.0
     12.0     288.0     108.0        .0     144.0      96.0      96.0      36.0      48.0      24.0
     96.0      36.0     288.0     360.0        .0      48.0      48.0      96.0     120.0     144.0
     96.0      96.0     120.0
     24.0      36.0      36.0      24.0      24.0      24.0        .0      24.0      24.0      24.0
       .0     144.0      36.0      72.0      72.0      24.0      36.0        .0      36.0      16.0
     24.0        .0     144.0      72.0      36.0      36.0      24.0      16.0        .0      72.0
     36.0      24.0        .0
     12.0      12.0      12.0      12.0      48.0      24.0      48.0      24.0      48.0      24.0
     48.0      24.0      36.0      24.0      24.0      12.0      16.0      12.0
```

```
% cat ccostr6
```

CCOSTR6

```
  1080.0     .0      .0   144.0      .0   360.0      .0   288.0      .0   360.0
      .0  108.0      .0      .0    72.0      .0    36.0      .0   144.0      .0
    36.0     .0   324.0      .0      .0   288.0      .0   108.0      .0    72.0
      .0  108.0      .0
   144.0  360.0      .0    48.0    48.0    48.0   120.0    48.0    96.0    48.0
   120.0   72.0    36.0      .0    32.0    24.0    24.0    12.0    48.0    48.0
    24.0   12.0    72.0   108.0      .0    48.0    96.0    24.0    36.0    32.0
    24.0   24.0    36.0
   288.0  360.0      .0    48.0    48.0    96.0   120.0   144.0    96.0    96.0
   120.0   48.0    36.0      .0    48.0    24.0    16.0    12.0    48.0    48.0
    16.0   12.0   288.0   108.0      .0   144.0    96.0    96.0    36.0    48.0
    24.0   96.0    36.0
   108.0     .0      .0    72.0      .0    36.0      .0    48.0      .0    36.0
      .0  216.0      .0      .0    72.0      .0    72.0      .0    72.0      .0
    72.0     .0   108.0      .0      .0    48.0      .0    36.0      .0    72.0
      .0   36.0      .0
    72.0   36.0      .0    32.0    24.0    24.0    12.0    48.0    16.0    24.0
    12.0   72.0    72.0      .0    48.0    24.0    24.0    24.0    32.0    24.0
    24.0   24.0   144.0    36.0      .0    48.0    16.0    48.0    12.0    48.0
    24.0   48.0    12.0
   144.0   36.0      .0    48.0    24.0    48.0    12.0    48.0    16.0    48.0
    12.0   72.0    72.0      .0    32.0    24.0    24.0    24.0    48.0    24.0
    24.0   24.0    72.0    36.0      .0    48.0    16.0    24.0    12.0    32.0
    24.0   24.0    12.0
   324.0     .0      .0    72.0      .0   108.0      .0   288.0      .0   108.0
      .0  108.0      .0      .0   144.0      .0    36.0      .0    72.0      .0
    36.0     .0  1080.0      .0      .0   288.0      .0   360.0      .0   144.0
      .0  360.0      .0
   288.0  108.0      .0    48.0    24.0    96.0    36.0   144.0    96.0    96.0
    36.0   48.0    36.0      .0    48.0    48.0    16.0    12.0    48.0    24.0
    16.0   12.0   288.0   360.0      .0   144.0    96.0    96.0   120.0    48.0
    48.0   96.0   120.0
    72.0  108.0      .0    32.0    24.0    24.0    36.0    48.0    96.0    24.0
    36.0   72.0    36.0      .0    48.0    48.0    24.0    12.0    32.0    24.0
    24.0   12.0   144.0   360.0      .0    48.0    96.0    48.0   120.0    48.0
    48.0   48.0   120.0
    36.0  144.0    72.0    16.0    36.0    24.0      .0    24.0    72.0    36.0
      .0   36.0    24.0    36.0    24.0    24.0    24.0      .0    24.0    24.0
    24.0     .0    72.0   144.0    36.0    36.0    72.0    24.0      .0    24.0
    36.0   16.0      .0
    48.0   24.0    48.0    24.0    12.0    12.0    12.0    12.0    24.0    48.0
    24.0   48.0    12.0    24.0    16.0    36.0    24.0    12.0
```

Appendix E - The Scattering Integrals

The first scattering integral is

$$\iiint dx\, du\, du'\, \alpha\, \phi \phi' \tag{4-3}$$

where $\alpha = \dfrac{\Sigma_s^2}{2} - \Sigma_t \Sigma_s$. If the product $\phi\phi' = F$, then a cubic can be specified over the tetrahedron using the twenty degrees of freedom specified in figure 2-6. Consider case 1 as depicted in figure 4-4.

$$\underline{F} = \begin{bmatrix} \varphi_2\varphi_1' & \varphi_{2x}\varphi_1' + \varphi_2\varphi_{1x}' & \varphi_{2u}\varphi_1' & \varphi_2\varphi_{1u}' \\ \varphi_1\varphi_3' & \varphi_{1x}\varphi_3' + \varphi_1\varphi_{3x}' & \varphi_{1u}\varphi_3' & \varphi_1\varphi_{3u}' \\ \varphi_3\varphi_1' & \varphi_{3x}\varphi_1' + \varphi_3\varphi_{1x}' & \varphi_{3u}\varphi_1' & \varphi_3\varphi_{1u}' \\ \varphi_1\varphi_2' & \varphi_{1x}\varphi_2' + \varphi_1\varphi_{2x}' & \varphi_{1u}\varphi_2' & \varphi_1\varphi_{2u}' \\ \varphi_{12}\varphi_{10}' & \varphi_{10}\varphi_{14} & \varphi_{11}\varphi_{10}' & \varphi_{10}\varphi_{13} \end{bmatrix} \tag{E-1}$$

points 11 and 12 are on the local triangle at $(\frac{2}{3},\frac{1}{3},0)$ and $(\frac{2}{3},0,\frac{1}{3})$ respectively and 13 and 14 are on the non local triangle at $(\frac{2}{3}\ 0,\frac{1}{3})$ and $(\frac{2}{3},\frac{1}{3},0)$ respectively. It should be noted that these are not finite element interpolation nodes, but that they can be written in terms of these nodes using (2-13). The second scattering integral is

$$\iiint dx\, du\, du' \left(-\Sigma_s\right) u \frac{\partial \phi}{\partial x} \phi' \tag{4-3}$$

If $G = \frac{\partial \phi}{\partial x} \phi'$, then the twenty degrees of freedom for case 1 are

$$\underline{G} = \begin{bmatrix} \varphi_{2x}\varphi_1' & \varphi_{2xx}\varphi_1' + \varphi_{2x}\varphi_{1x}' & \varphi_{2ux}\varphi_1' & \varphi_{2x}\varphi_{1u}' \\ \varphi_{1x}\varphi_3' & \varphi_{1xx}\varphi_3' + \varphi_{1x}\varphi_{3x}' & \varphi_{1ux}\varphi_3' & \varphi_{1x}\varphi_{3u}' \\ \varphi_{3x}\varphi_1' & \varphi_{3xx}\varphi_1' + \varphi_{3x}\varphi_{1x}' & \varphi_{3ux}\varphi_1' & \varphi_{3x}\varphi_{1u}' \\ \varphi_{1x}\varphi_2' & \varphi_{1xx}\varphi_2' + \varphi_{1x}\varphi_{2x}' & \varphi_{1ux}\varphi_2' & \varphi_{1x}\varphi_{2u}' \\ \varphi_{12x}\varphi_{10}' & \varphi_{10x}\varphi_{14} & \varphi_{11x}\varphi_{10}' & \varphi_{10x}\varphi_{13} \end{bmatrix} \quad \text{(E-2)}$$

For case 3

$$\underline{F} = \begin{bmatrix} \varphi_2\varphi_1' & \varphi_{2x}\varphi_1' + \varphi_2\varphi_{1x}' & \varphi_{2u}\varphi_1' & \varphi_2\varphi_{1u}' \\ \varphi_3\varphi_1' & \varphi_{3x}\varphi_1' + \varphi_3\varphi_{1x}' & \varphi_{3u}\varphi_1' & \varphi_3\varphi_{1u}' \\ \varphi_1\varphi_3' & \varphi_{1x}\varphi_3' + \varphi_1\varphi_{3x}' & \varphi_{1u}\varphi_3' & \varphi_{1x}\varphi_{3u}' \\ \varphi_1\varphi_2' & \varphi_{1x}\varphi_2' + \varphi_1\varphi_{2x}' & \varphi_{1u}\varphi_2' & \varphi_1\varphi_{2u}' \\ \varphi_{11}\varphi_{10}' & \varphi_{12}\varphi_{10}' & \varphi_{10}\varphi_{13} & \varphi_{10}\varphi_{14} \end{bmatrix} \quad \text{(E-3)}$$

and

$$\underline{G} = \begin{bmatrix} \varphi_{2x}\varphi_1' & \varphi_{2xx}\varphi_1' + \varphi_{2x}\varphi_{1x}' & \varphi_{2ux}\varphi_1' & \varphi_{2x}\varphi_{1u}' \\ \varphi_{3x}\varphi_1' & \varphi_{3xx}\varphi_1' + \varphi_{3x}\varphi_{1x}' & \varphi_{3ux}\varphi_1' & \varphi_{3x}\varphi_{1u}' \\ \varphi_{1x}\varphi_3' & \varphi_{1xx}\varphi_3' + \varphi_{1x}\varphi_{3x}' & \varphi_{1ux}\varphi_3' & \varphi_{1x}\varphi_{3u}' \\ \varphi_{1x}\varphi_2' & \varphi_{1xx}\varphi_2' + \varphi_{1x}\varphi_{2x}' & \varphi_{1ux}\varphi_2' & \varphi_{1x}\varphi_{2u}' \\ \varphi_{11x}\varphi_{10}' & \varphi_{12x}\varphi_{10}' & \varphi_{10x}\varphi_{13} & \varphi_{10x}\varphi_{14} \end{bmatrix} \quad \text{(E-4)}$$

where φ_{11} and φ_{12} are on the local triangle at $(\frac{2}{3}\ 0\ \frac{1}{3})$ and $(\frac{2}{3}\ \frac{1}{3}\ 0)$ respectively. Points 13 and 14 are non local at $(\frac{2}{3}\ \frac{1}{3}\ 0)$ and $(\frac{2}{3}\ 0\ \frac{1}{3})$.

Continuing to number as in figure 4-4, the integrals for case 2 and case 4 must be done separately over the two halves and summed. Case 2, the left half is

$$\underline{F} = \begin{bmatrix} \varphi_1\varphi_1' & \varphi_{1x}\varphi_1' + \varphi_1\varphi_{1x}' & \varphi_{1u}\varphi_1' & \varphi_1\varphi_{1u}' \\ \varphi_3\varphi_3' & \varphi_{3x}\varphi_3' + \varphi_3\varphi_{3x}' & \varphi_{3u}\varphi_3' & \varphi_3\varphi_{3u}' \\ \varphi_2\varphi_2' & \varphi_{2x}\varphi_2' + \varphi_2\varphi_{2x}' & \varphi_{2u}\varphi_2' & \varphi_2\varphi_{2u}' \\ \varphi_3\varphi_2' & \varphi_{3x}\varphi_2' + \varphi_3\varphi_{2x}' & \varphi_{3u}\varphi_2' & \varphi_3\varphi_{2u}' \\ \varphi_{15}\text{-}\varphi_{18} & \varphi_{10}\varphi_{14} & \varphi_{11}\varphi_{10}' & \varphi_{10}\varphi_{10}' \end{bmatrix} \tag{E-5}$$

and

$$\underline{G} = \begin{bmatrix} \varphi_{1x}\varphi_1' & \varphi_{1xx}\varphi_1' + \varphi_{1x}\varphi_{1x}' & \varphi_{1ux}\varphi_1' & \varphi_{1x}\varphi_{1u}' \\ \varphi_{3x}\varphi_3' & \varphi_{3xx}\varphi_3' + \varphi_{3x}\varphi_{3x}' & \varphi_{3ux}\varphi_3' & \varphi_{3x}\varphi_{3u}' \\ \varphi_{2x}\varphi_2' & \varphi_{2xx}\varphi_2' + \varphi_{2x}\varphi_{2x}' & \varphi_{2ux}\varphi_2' & \varphi_{2x}\varphi_{2u}' \\ \varphi_{3x}\varphi_2' & \varphi_{3xx}\varphi_2' + \varphi_{3x}\varphi_{2x}' & \varphi_{3ux}\varphi_2' & \varphi_{3x}\varphi_{2u}' \\ \varphi_{15x}\varphi_{18} & \varphi_{10x}\varphi_{14} & \varphi_{11x}\varphi_{10}' & \varphi_{10x}\varphi_{10}' \end{bmatrix} \tag{E-6}$$

The right half is

$$\underline{F} = \begin{bmatrix} \varphi_1\varphi_1' & \varphi_{1x}\varphi_1' + \varphi_1\varphi_{1x}' & \varphi_{1u}\varphi_1' & \varphi_1\varphi_{1u}' \\ \varphi_2\varphi_2' & \varphi_{2x}\varphi_2' + \varphi_2\varphi_{2x}' & \varphi_{2u}\varphi_2' & \varphi_2\varphi_{2u}' \\ \varphi_3\varphi_3' & \varphi_{3x}\varphi_3' + \varphi_3\varphi_{3x}' & \varphi_{3u}\varphi_3' & \varphi_3\varphi_{3u}' \\ \varphi_2\varphi_3' & \varphi_{2x}\varphi_3' + \varphi_2\varphi_{3x}' & \varphi_{2u}\varphi_3' & \varphi_2\varphi_{3u}' \\ \varphi_{16}\varphi_{17} & \varphi_{10}\varphi_{13} & \varphi_{12}\varphi_{10}' & \varphi_{10}\varphi_{10}' \end{bmatrix} \tag{E-7}$$

and

$$\underline{G} = \left[\begin{matrix} \varphi_{1x}\varphi_1' & \varphi_{1xx}\varphi_1' + \varphi_{1x}\varphi_{1x}' & \varphi_{1xu}\varphi_1' & \varphi_{1x}\varphi_{1u}' \\ \varphi_{2x}\varphi_2' & \varphi_{2xx}\varphi_2' + \varphi_{2x}\varphi_{2x}' & \varphi_{2xu}\varphi_2' & \varphi_{2x}\varphi_{2u}' \\ \varphi_{3x}\varphi_3' & \varphi_{3xx}\varphi_3' + \varphi_{3x}\varphi_{3x}' & \varphi_{3xu}\varphi_3' & \varphi_{3x}\varphi_{3u}' \\ \varphi_{2x}\varphi_3' & \varphi_{2xx}\varphi_3' + \varphi_{2x}\varphi_{3x}' & \varphi_{2xu}\varphi_3' & \varphi_{2x}\varphi_{3u}' \end{matrix} \right.$$

$$\varphi_{16x}\varphi_{17} \quad \varphi_{10x}\varphi_{13} \quad \varphi_{12x}\varphi_{10}' \quad \varphi_{10x}\varphi_{10}' \Big]$$

(E-8)

where points 11,12,15,and 16 are local at $(\frac{1}{3},0,\frac{2}{3})$, $(\frac{1}{3},\frac{2}{3},0)$, $(0\,\frac{1}{3}\,\frac{2}{3})$ and $(0,\frac{2}{3},\frac{1}{3})$ respectively. Points 12,13,17and 18 are on the non local triangle at $(\frac{1}{3},0,\frac{2}{3})$, $(\frac{1}{3},\frac{2}{3},0)$, $(0,\frac{1}{3},\frac{2}{3})$ and $(0,\frac{2}{3},\frac{1}{3})$.

Case 4 is similar. The left half is given by

$$\underline{E} = \begin{bmatrix} \varphi_1\varphi_1' & \varphi_{1x}\varphi_1' + \varphi_1\varphi_{1x}' & \varphi_{1u}\varphi_1' & \varphi_1\varphi_{1u}' \\ \varphi_3\varphi_3' & \varphi_{3x}\varphi_3' + \varphi_3\varphi_{3x}' & \varphi_{3u}\varphi_3' & \varphi_3\varphi_{3u}' \\ \varphi_2\varphi_2' & \varphi_{2x}\varphi_2' + \varphi_2\varphi_{2x}' & \varphi_{2u}\varphi_2' & \varphi_2\varphi_{2u}' \\ \varphi_2\varphi_3' & \varphi_{2x}\varphi_3' + \varphi_2\varphi_{3x}' & \varphi_{2u}\varphi_3' & \varphi_2\varphi_{3u}' \\ \varphi_{15}\varphi_{18} & \varphi_{11}\varphi_{10}' & \varphi_{10}\varphi_{14} & \varphi_{10}\varphi_{10}' \end{bmatrix}$$

(E-9)

$$\underline{G} = \begin{bmatrix} \varphi_{1x}\varphi_1' & \varphi_{1xx}\varphi_1' + \varphi_{1x}\varphi_{1x}' & \varphi_{1xu}\varphi_1' & \varphi_{1x}\varphi_{1u}' \\ \varphi_{3x}\varphi_3' & \varphi_{3xx}\varphi_3' + \varphi_{3x}\varphi_{3x}' & \varphi_{3xu}\varphi_3' & \varphi_{3x}\varphi_{3u}' \\ \varphi_{2x}\varphi_2' & \varphi_{2xx}\varphi_2' + \varphi_{2x}\varphi_{2x}' & \varphi_{2xu}\varphi_2' & \varphi_{2x}\varphi_{2u}' \\ \varphi_{2x}\varphi_3' & \varphi_{2xx}\varphi_3' + \varphi_{2x}\varphi_{3x}' & \varphi_{2xu}\varphi_3' & \varphi_{2x}\varphi_{3u}' \\ \varphi_{15}\varphi_{18} & \varphi_{11x}\varphi_{10}' & \varphi_{10x}\varphi_{14} & \varphi_{10x}\varphi_{10}' \end{bmatrix}$$

(E-10)

The right half is

$$\underline{E} = \Bigg[\begin{matrix} \varphi_1\varphi_1' & \varphi_{1x}\varphi_1' + \varphi_1\varphi_{1x}' & \varphi_{1u}\varphi_1' & \varphi_1\varphi_{1u}' \\ \varphi_2\varphi_2' & \varphi_{2x}\varphi_2' + \varphi_2\varphi_{2x}' & \varphi_{2u}\varphi_2' & \varphi_2\varphi_{2u}' \\ \varphi_3\varphi_3' & \varphi_{3x}\varphi_3' + \varphi_3\varphi_{3x}' & \varphi_{3u}\varphi_3' & \varphi_3\varphi_{3u}' \\ \varphi_3\varphi_2' & \varphi_{3x}\varphi_2' + \varphi_3\varphi_{2x}' & \varphi_{3u}\varphi_2' & \varphi_3\varphi_{2u}' \end{matrix}$$

$$
\varphi_{16}\varphi_{17} \qquad \varphi_{12}\varphi_{10}' \qquad \varphi_{10}\varphi_{13} \qquad \varphi_{10}\varphi_{10}' \Big]
$$

(E-11)

$$
\underline{G} = \begin{bmatrix}
\varphi_{1x}\varphi_1' & \varphi_{1xx}\varphi_1' + \varphi_{1x}\varphi_{1x}' & \varphi_{1xu}\varphi_1' & \varphi_{1x}\varphi_{1u}' \\
\varphi_{2x}\varphi_2' & \varphi_{2xx}\varphi_2' + \varphi_{2x}\varphi_{2x}' & \varphi_{2xu}\varphi_2' & \varphi_{2x}\varphi_{2u}' \\
\varphi_{3x}\varphi_3' & \varphi_{3xx}\varphi_3' + \varphi_{3x}\varphi_{3x}' & \varphi_{3xu}\varphi_3' & \varphi_{3x}\varphi_{3u}' \\
\varphi_{3x}\varphi_2' & \varphi_{3xx}\varphi_2' + \varphi_{3x}\varphi_{2x}' & \varphi_{3xu}\varphi_2' & \varphi_{3x}\varphi_{2u}' \\
\varphi_{16x}\varphi_{17} & \varphi_{12x}\varphi_{10}' & \varphi_{10x}\varphi_{13} & \varphi_{10x}\varphi_{10}'
\end{bmatrix}
$$

(E-12)

For case 4, points 11,12,15 and 16 are local at $(\frac{1}{3},\frac{2}{3},0)$ $(\frac{1}{3},0,\frac{2}{3})$ $(0,\frac{2}{3},\frac{1}{3})$ and $(0,\frac{1}{3},\frac{2}{3})$. Points 13,14,17 and 18 are at $(\frac{1}{3},\frac{2}{3},0)$ $(\frac{1}{3},0,\frac{2}{3})$, $(0,\frac{2}{3},\frac{1}{3})$ and $(0,\frac{1}{3},\frac{2}{3})$ on the non local triangle.

Appendix F - Finite Element Meshes

1. Streaming meshes

    Mesh1

    Mesh2

    Mesh3

    Mesh4

    Mesh5

    Mesh6

2. Scattering meshes

    Mesha

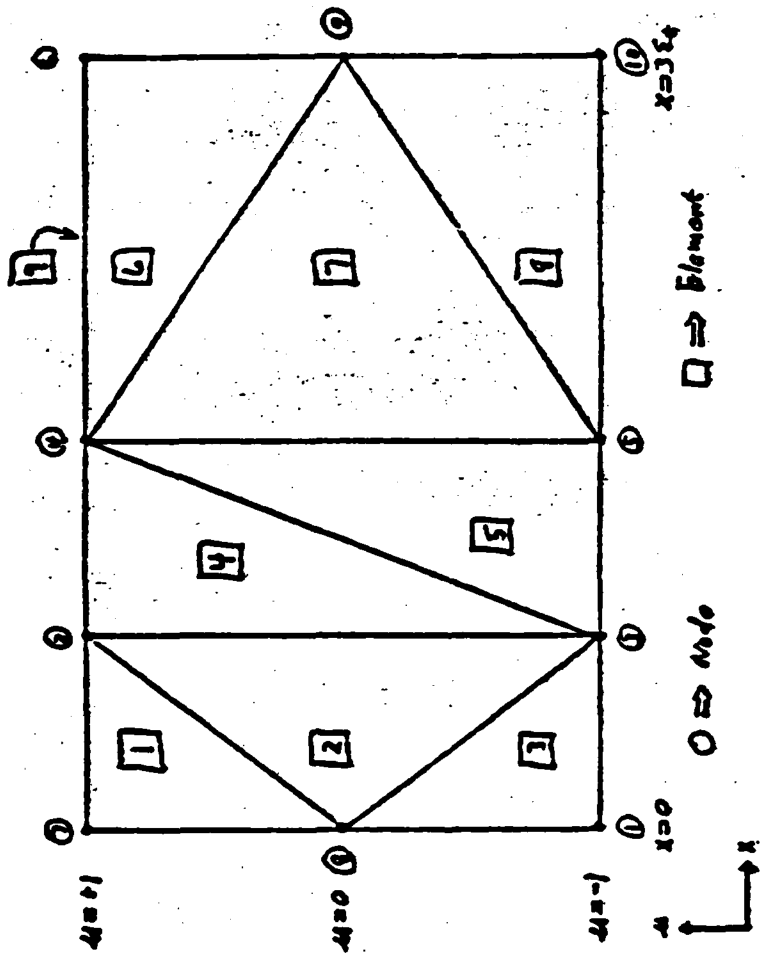
    Meshb

    Meshc

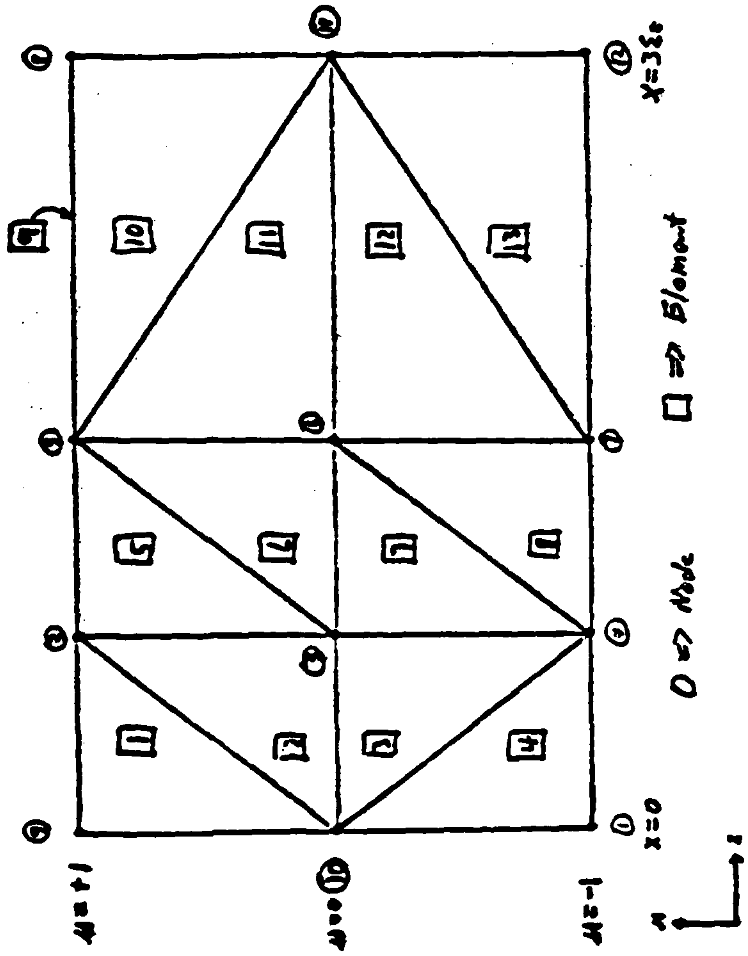
    Meshd

Meshes 1 through 4 are identical to those of reference (2).

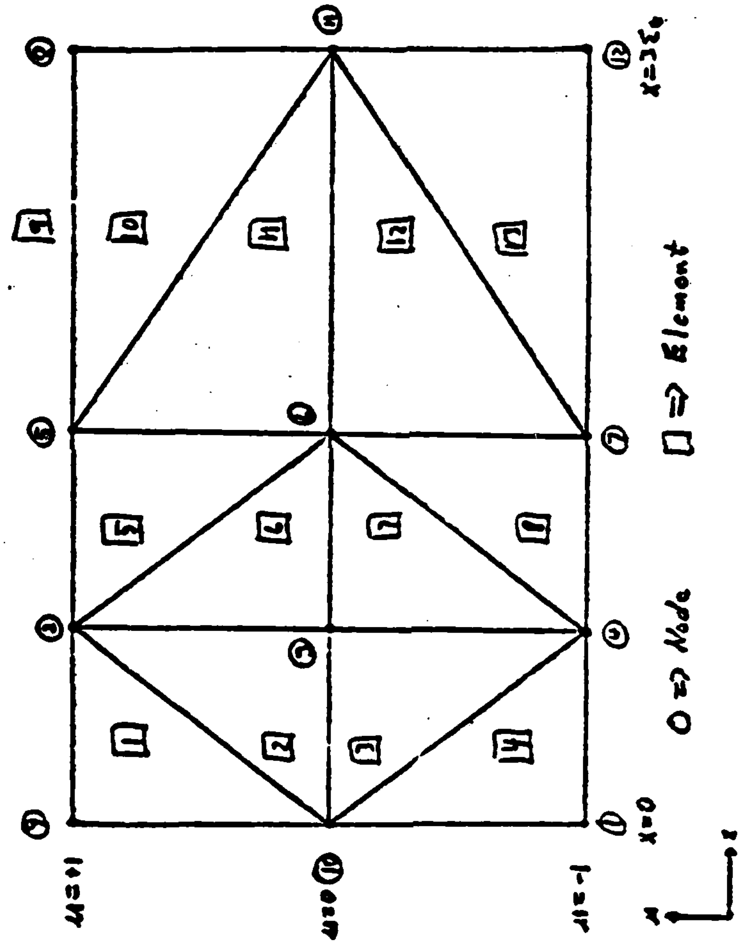
MESH 1



MESH 2

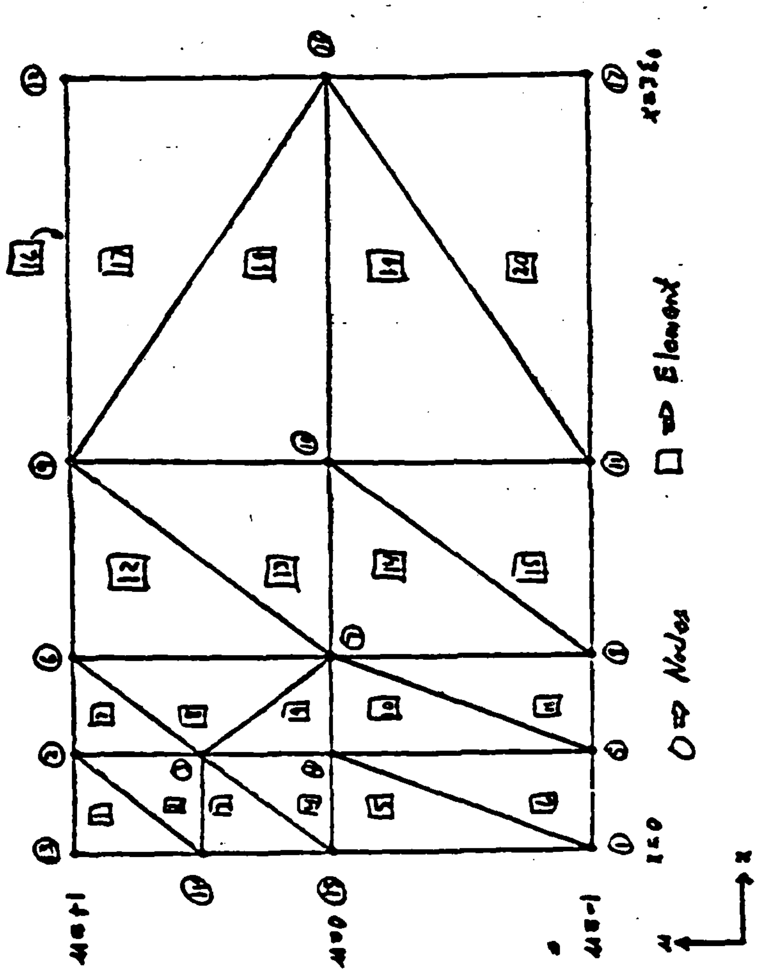


MESH 3



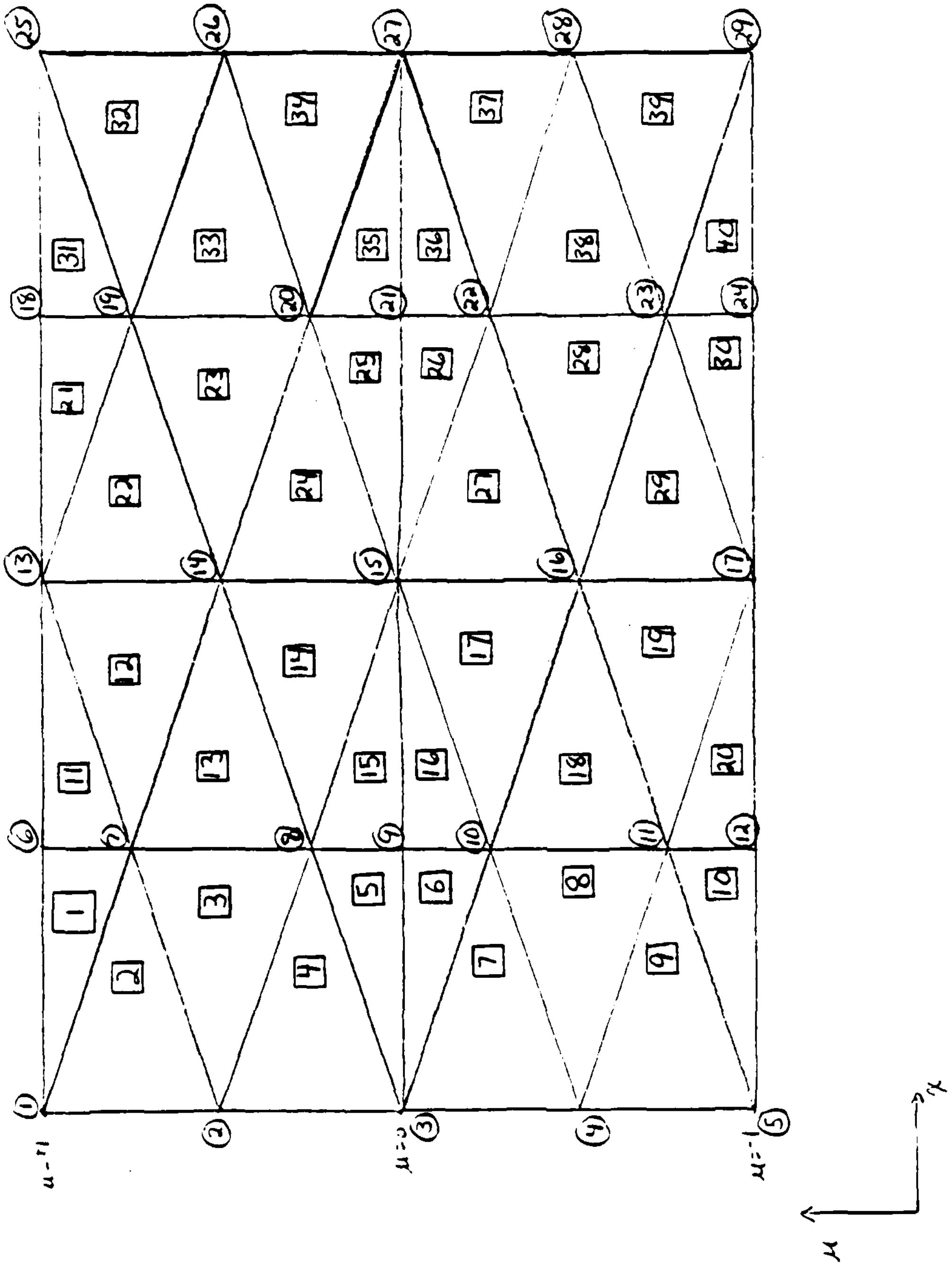
```
ED MESH3
LI
     NTRIAN   # NODE   NCOL
       12       12       3

       RANGE  SIGMAT  SIGMAS
        3.      1.      .5

     TRIANGLE          NODE1    NODE2    NODE3   COLUMN
         1               2        9       10       1
         2              10        3        2       1
         3              10        4        3       1
         4               4       10        1       1
         5               2        6        5       2
         6               6        2        3       2
         7               6        3        4       2
         8               4        7        6       2
         9               5       11        8       3
        10              11        5        6       3
        11              11        6        7       3
        12               7       12       11       3

       COLUMN      FIRST ELEMENT   NUMBER OF ELEMENTS
         1               1                4
         2               5                4
         3               9                4

       NODE           X-AXIS     U-AXIS
         1             .000      -1.000
         2             .250       1.000
         3             .250        .000
         4             .250      -1.000
         5             .500       1.000
         6             .500        .000
         7             .500      -1.000
         8            1.000       1.000
         9             .000       1.000
        10             .000        .000
        11            1.000        .000
        12            1.000      -1.000

EOF..
EOT..
UP
```

MESH 4



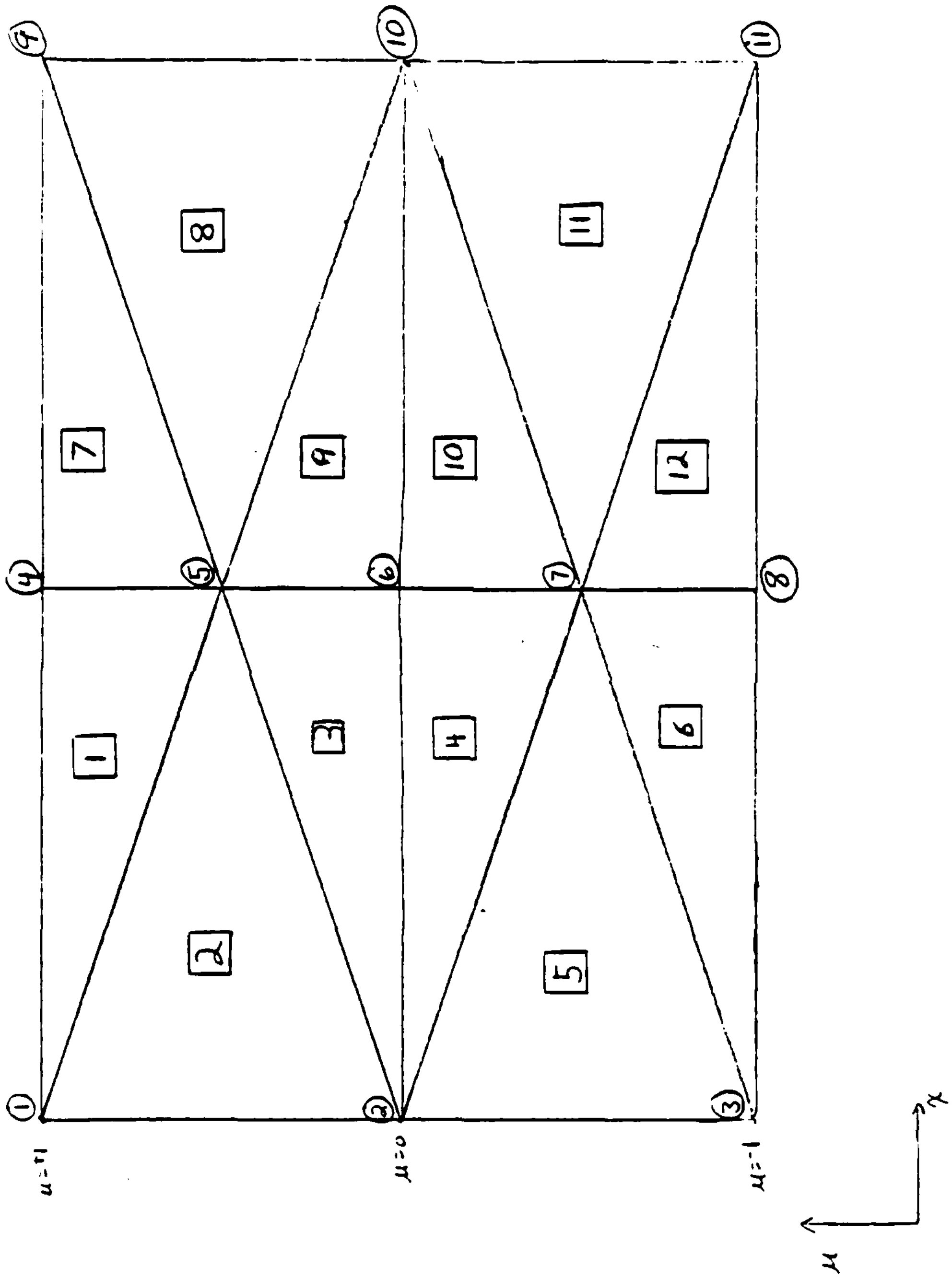
```
ED MSHC3.5C
LI,1,200
        NTRIAN   MNODE      NCOL
          40       29         4

         RANGE   SIGMAT   SIGMAS
          3.       1.        .5

        TRIANGLE             NODE1     NODE2     NODE3   COLUMN
           1                   1         7         6        1
           2                   7         1         2        1
           3                   2         8         7        1
           4                   8         2         3        1
           5                   3         9         8        1
           6                   3        10         9        1
           7                  10         3         4        1
           8                   4        11        10        1
           9                  11         4         5        1
          10                   5        12        11        1
          11                  13         6         7        2
          12                   7        14        13        2
          13                  14         7         8        2
          14                   8        15        14        2
          15                  15         8         9        2
          16                  15         9        10        2
          17                  10        16        15        2
          18                  16        10        11        2
          19                  11        17        16        2
          20                  17        11        12        2
          21                  13        19        18        3
          22                  19        13        14        3
          23                  14        20        19        3
          24                  20        14        15        3
          25                  15        21        20        3
          26                  15        22        21        3
          27                  22        15        16        3
          28                  16        23        22        3
          29                  23        16        17        3
          30                  17        24        23        3
          31                  25        18        19        4
          32                  19        26        25        4
          33                  26        19        20        4
          34                  20        27        26        4
          35                  27        20        21        4
          36                  27        21        22        4
          37                  22        28        27        4
          38                  28        22        23        4
          39                  23        29        28        4
          40                  29        23        24        4

          COLUMN       FIRST ELEMENT   NUMBER OF ELEMENTS
            1                1                10
            2               11                10
            3               21                10
            4               31                10

```

MESH C



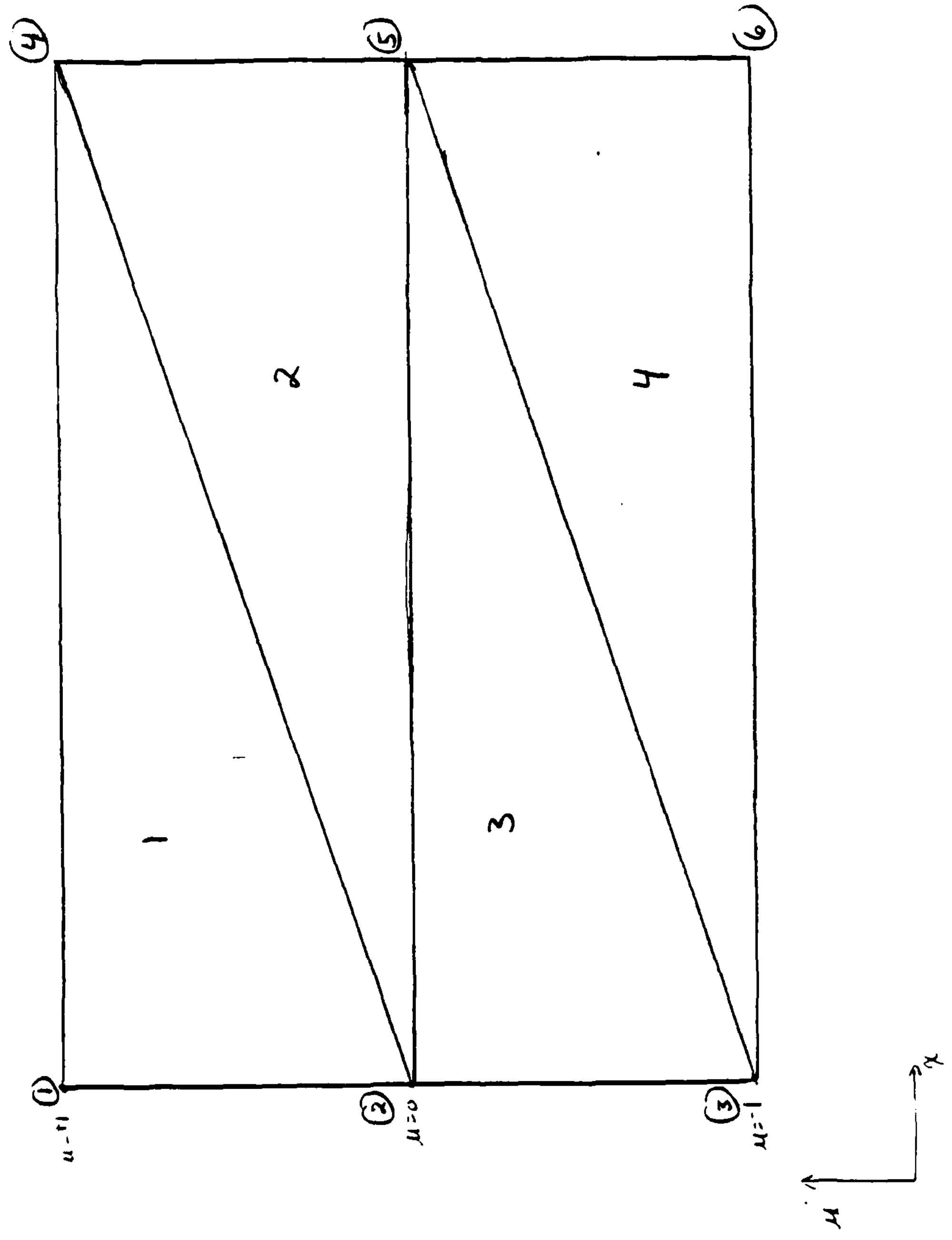
```
ED MSHB3.9C
LI,1,100
      NTRIAN   MNODE      NCOL
        12       11         2

        RANGE  SIGMAT   SIGMAS
          3.       1.       .9

      TRIANGLE             NODE1     NODE2     NODE3    COLUMN
          1                  1         5         4        1
          2                  5         1         2        1
          3                  2         6         5        1
          4                  2         7         6        1
          5                  7         2         3        1
          6                  3         8         7        1
          7                  9         4         5        2
          8                  5        10         9        2
          9                 10         5         6        2
         10                 10         6         7        2
         11                  7        11        10        2
         12                 11         7         8        2

        COLUMN        FIRST ELEMENT   NUMBER OF ELEMENTS
          1                 1                 6
          2                 7                 6

        NODE          X-AXIS      U-AXIS
          1             .000      1.000
          2             .000       .000
          3             .000     -1.000
          4             .500      1.000
          5             .500       .500
          6             .500       .000
          7             .500      -.500
          8             .500     -1.000
          9            1.000      1.000
         10            1.000       .000
         11            1.000     -1.000

     NODE (NB)          FLUX
       8
       1              3.5345E-01
       3              4.0360E-02
       4              2.5000E-01
       6              2.1768E-01
      28              4.2659E-02
      30              1.9896E-02
      31              2.7962E-02
      33              1.0452E-02
EOF..
EOT..
UP
```

MESH B



```
ED MSHA3.5C
LI
      NTRIAN  MNODE     NCOL
        4       6        1

       RANGE  SIGMAT  SIGMAS
        3.      1.      .5

      TRIANGLE           NODE1    NODE2    NODE3  COLUMN
         1                 4        1        2       1
         2                 2        5        4       1
          3                 5        2        3       1
          4                 3        6        5       1

        COLUMN      FIRST ELEMENT  NUMBER OF ELEMENTS
          1                 1                4

        NODE          X-AXIS    U-AXIS
          1             .000     1.000
          2             .000      .000
          3             .000    -1.000
          4            1.000     1.000
          5            1.000      .000
          6            1.000    -1.000

    NODE (NB)         FLUX
       8
       1             1.0142E+00
       3             9.8832E-01
       4             1.2126E-01
       6             6.1684E-01
      13             6.5917E-03
      15             5.3892E-03
      16             3.2862E-03
      18             1.9460E-03
EOF..
EOT..
UP
```

MESH A



111	12	.250	.000
112	13	.250	-1.000
113	14	.375	1.000
114	15	.375	.750
115	16	.375	.500
116	17	.375	.250
117	18	.375	.000
118	19	.375	-1.000
119	20	.500	1.000
120	21	.500	.750
121	22	.500	.500
122	23	.500	.250
123	24	.500	.000
124	25	.500	-1.000
125	26	.625	1.000
126	27	.625	.750
127	28	.625	.500
128	29	.625	.250
129	30	.625	.000
130	31	.625	-1.000
131	32	.750	1.000
132	33	.750	.750
133	34	.750	.500
134	35	.750	.250
135	36	.750	.000
136	37	.750	-1.000
137	38	.875	1.000
138	39	.875	.750
139	40	.875	.500
140	41	.875	.250
141	42	.875	.000
.42	43	.875	-1.000
143	44	1.000	1.000
144	45	1.000	.750
145	46	1.000	.500
146	47	1.000	.250
147	48	1.000	.000
148	49	1.000	-1.000
149	50	.000	1.000
150	51	.000	.750
151	52	.000	.500
152	53	.000	.250
153	54	.000	.000
154			

EOF..
EOT..

48	24	30	29	5
49	30	24	25	5
50	25	31	30	5
51	32	26	27	6
52	27	33	32	6
53	33	27	28	6
54	28	34	33	6
55	34	28	29	6
56	29	35	34	6
57	35	29	30	6
58	30	36	35	6
59	36	30	31	6
60	31	37	36	6
61	38	32	33	7
62	33	39	38	7
63	39	33	34	7
64	34	40	39	7
65	40	34	35	7
66	35	41	40	7
67	41	35	36	7
68	36	42	41	7
69	42	36	37	7
70	37	43	42	7
71	44	38	39	8
72	39	45	44	8
73	45	39	40	8
74	40	46	45	8
75	46	40	41	8
76	41	47	46	8
77	47	41	42	8
78	42	48	47	8
79	48	42	43	8
80	43	49	48	8

COLUMN	FIRST ELEMENT	NUMBER OF ELEMENTS
1	1	10
2	11	10
3	21	10
4	31	10
5	41	10
6	51	10
7	61	10
8	71	10

NODE	X-AXIS	U-AXIS
1	.000	-1.000
2	.125	1.000
3	.125	.750
4	.125	.500
5	.125	.250
6	.125	.000
7	.125	-1.000
8	.250	1.000
9	.250	.750
10	.250	.500
11	.250	.250

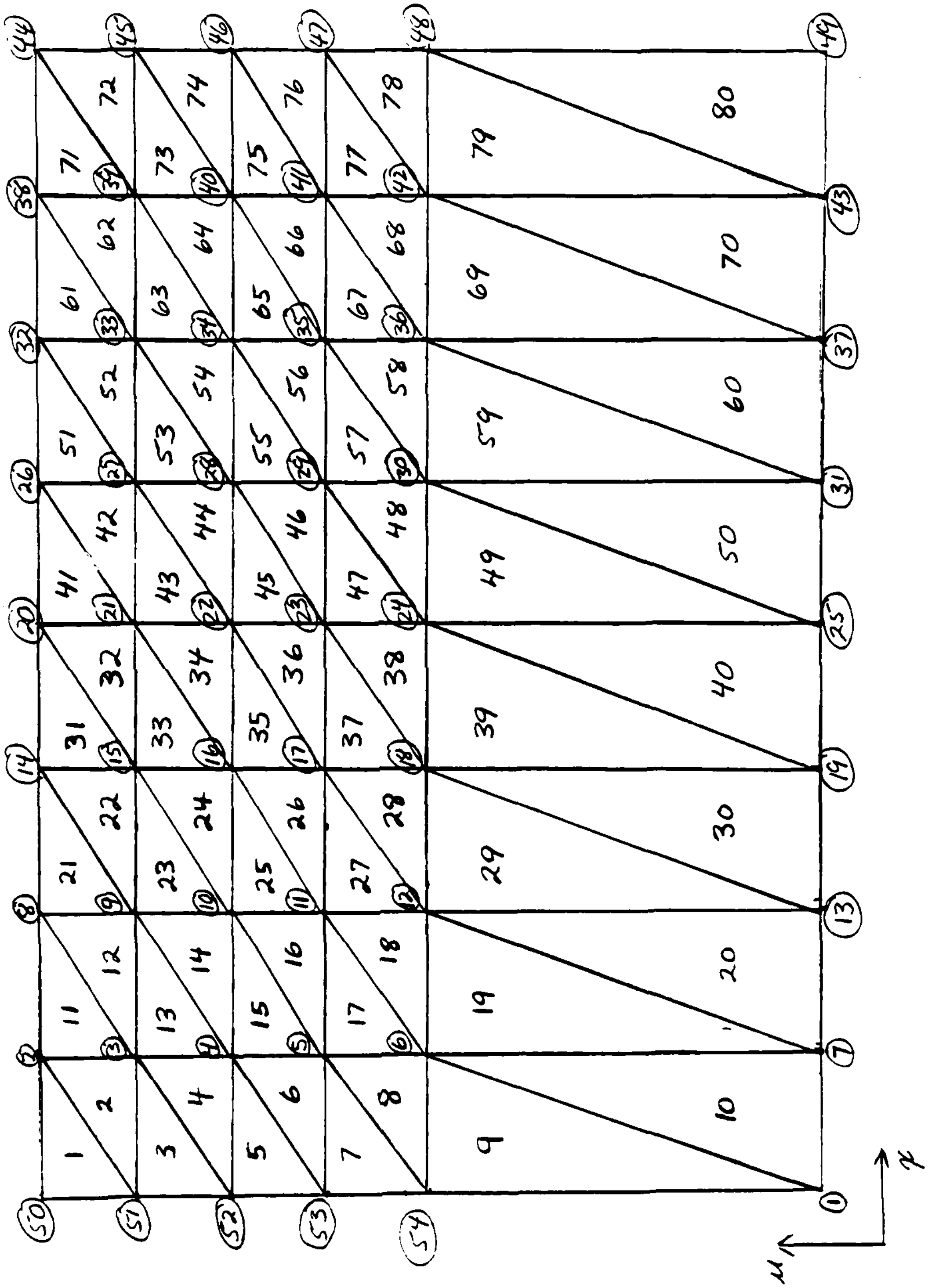
```
ED MESH6
LI,1,200
```

NTRIAN	# NODE	NCOL
80	54	8

RANGE	SIGMAT	SIGMAS
3.	1.	.5

TRIANGLE	NODE1	NODE2	NODE3	COLUMN
1	2	50	51	1
2	51	3	2	1
3	3	51	52	1
4	52	4	3	1
5	4	52	53	1
6	53	5	4	1
7	5	53	54	1
8	54	6	5	1
9	6	54	1	1
10	1	7	6	1
11	8	2	3	2
12	3	9	8	2
13	9	3	4	2
14	4	10	9	2
15	10	4	5	2
16	5	11	10	2
17	11	5	6	2
18	6	12	11	2
19	12	6	7	2
20	7	13	12	2
21	14	8	9	3
22	9	15	14	3
23	15	9	10	3
24	10	16	15	3
25	16	10	11	3
26	11	17	16	3
27	17	11	12	3
28	12	18	17	3
29	18	12	13	3
30	13	19	18	3
31	20	14	15	4
32	15	21	20	4
33	21	15	16	4
34	16	22	21	4
35	22	16	17	4
36	17	23	22	4
37	23	17	18	4
38	18	24	23	4
39	24	18	19	4
40	19	25	24	4
41	26	20	21	5
42	21	27	26	5
43	27	21	22	5
44	22	28	27	5
45	28	22	23	5
46	23	29	28	5
47	29	23	24	5

MESH 6



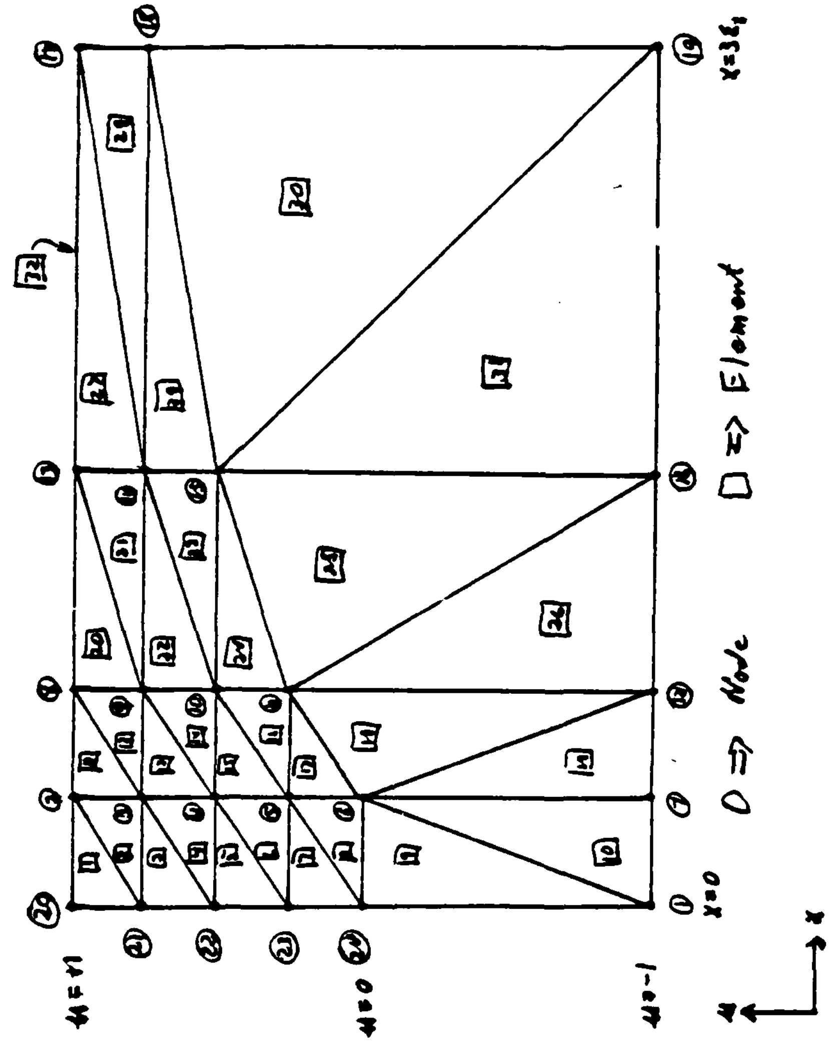
55	9	.250	.750
56	10	.250	.500
57	11	.250	.250
58	12	.250	-1.000
59	13	.500	1.000
60	14	.500	.750
61	15	.500	.500
62	16	.500	-1.000
63	17	1.000	1.000
64	18	1.000	.750
65	19	1.000	-1.000
66	20	.000	1.000
67	21	.000	.750
68	22	.000	.500
69	23	.000	.250
70	24	.000	.000
71			

EOF..
EOT..
UP

```
ED MESH5
LI
        NTRIAN   # NODE    NCOL
          31       24        4

         RANGE   SIGMAT   SIGMAS
          3.       1.       .5

         TRIANGLE             NODE1    NODE2    NODE3   COLUMN
            1                   2        20       21       1
            2                  21         3        2       1
            3                   3        21       22       1
            4                  22         4        3       1
            5                   4        22       23       1
            6                  23         5        4       1
            7                   5        23       24       1
            8                  24         6        5       1
            9                   6        24        1       1
           10                   1         7        6       1
           11                   8         2        3       2
           12                   3         9        8       2
           13                   9         3        4       2
           14                   4        10        9       2
           15                  10         4        5       2
           16                   5        11       10       2
           17                  11         5        6       2
           18                   6        12       11       2
           19                  12         6        7       2
           20                  13         8        9       3
           21                   9        14       13       3
           22                  14         9       10       3
           23                  10        15       14       3
           24                  15        10       11       3
           25                  11        16       15       3
           26                  16        11       12       3
           27                  17        13       14       4
           28                  14        18       17       4
           29                  18        14       15       4
           30                  15        19       18       4
           31                  19        15.      16       4

           COLUMN        FIRST ELEMENT   NUMBER OF ELEMENTS
              1                1                 10
              2               11                  9
              3               20                  7
              4               27                  5

           NODE            X-AXIS     U-AXIS
             1              .000      -1.000
             2              .125       1.000
             3              .125        .750
             4              .125        .500
             5              .125        .250
             6              .125        .000
             7              .125      -1.000
             8              .250       1.000
```

```
ED MESH4
LI
        NTRIAN   # NODE    NCOL
         19        17        4

         RANGE   SIGMAT   SIGMAS
          3.       1.       .5

        TRIANGLE              NODE1    NODE2    NODE3   COLUMN
           1                    2        13       14       1
           2                   14         3        2       1
           3                    3        14       15       1
           4                   15         4        3       1
           5                    4        15        1       1
           6                    1         5        4       1
           7                    6         2        3       2
           8                    3         7        6       2
           9                    7         3        4       2
          10                    7         4        5       2
          11                    5         8        7       2
          12                    9         6        7       3
          13                    7        10        9       3
          14                   10         7        8       3
          15                    8        11       10       3
          16                    9        16       12       4
          17                   16         9       10       4
          18                   16        10       11       4
          19                   11        17       16       4

          COLUMN      FIRST ELEMENT   NUMBER OF ELEMENTS
            1               1                6
            2               7                5
            3              12                4
            4              16                4

          NODE          X-AXIS     U-AXIS
            1            .000      -1.000
            2            .125       1.000
            3            .125        .500
            4            .125        .000
            5            .125      -1.000
            6            .250       1.000
            7            .250        .000
            8            .250      -1.000
            9            .500       1.000
           10            .500        .000
           11            .500      -1.000
           12           1.000       1.000
           13            .000       1.000
           14            .000        .500
           15            .000        .000
           16           1.000        .000
           17           1.000      -1.000

EOF..
EOT..
```

MESH 5

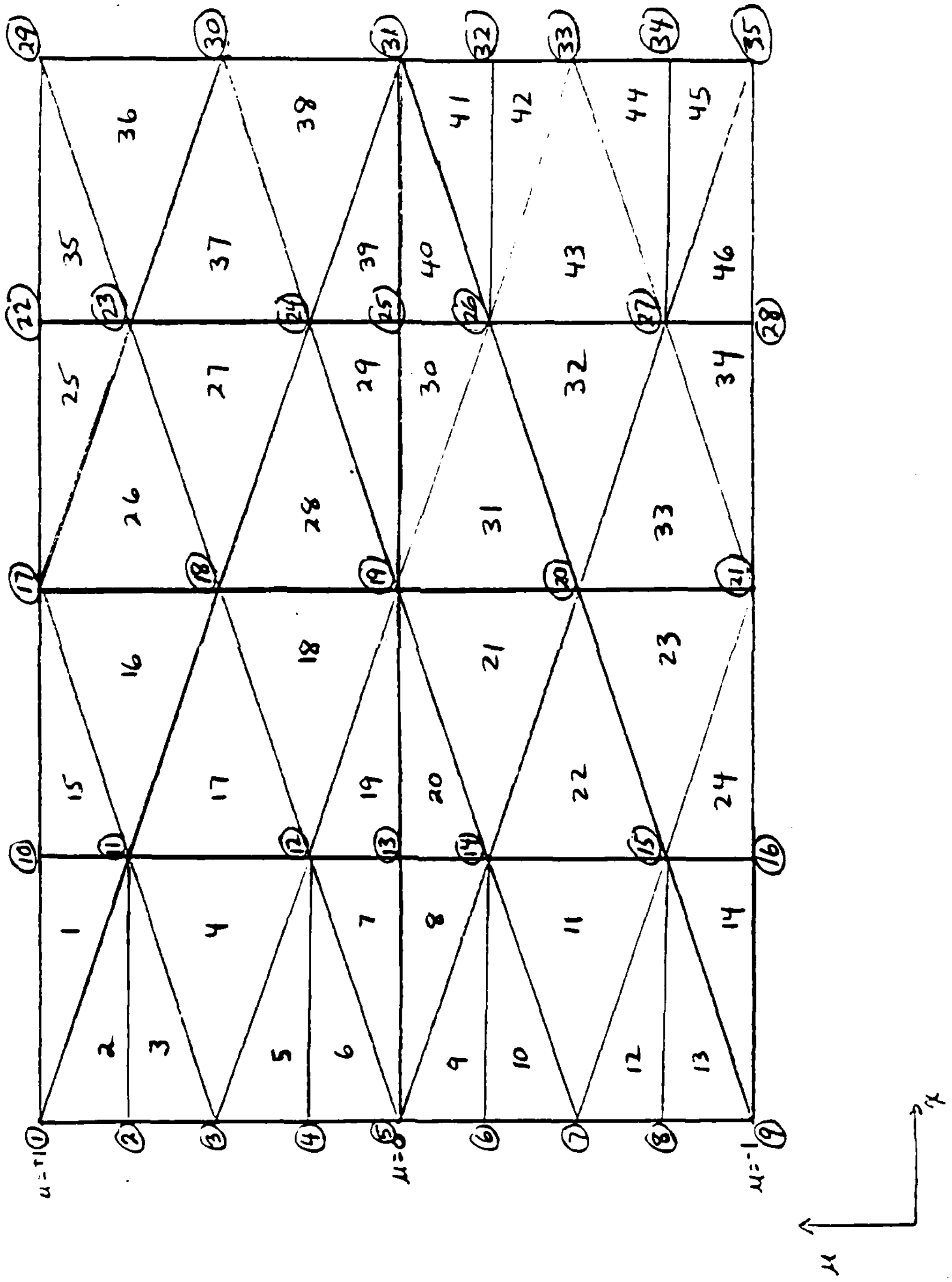


NODE	X-AXIS	U-AXIS
1	0.000	1.000
2	0.000	0.500
3	0.000	0.000
4	0.000	-0.500
5	0.000	-1.000
6	0.250	1.000
7	0.250	0.750
8	0.250	0.250
9	0.250	0.000
10	0.250	-0.250
11	0.250	-0.750
12	0.250	-1.000
13	0.500	1.000
14	0.500	0.500
15	0.500	0.000
16	0.500	-0.500
17	0.500	-1.000
18	0.750	1.000
19	0.750	0.750
20	0.750	0.250
21	0.750	0.000
22	0.750	-0.250
23	0.750	-0.750
24	0.750	-1.000
25	1.000	1.000
26	1.000	0.500
27	1.000	0.000
28	1.000	-0.500
29	1.000	-1.000

NODE (NB)	FLUX
12	
1	1.0142E+00
3	9.8832E-01
4	5.2627E-01
6	9.8330E-01
7	1.2126E-01
9	6.1684E-01
79	6.5917E-03
81	5.3892E-03
82	4.4268E-03
84	2.9434E-03
85	3.2862E-03
87	1.9460E-03

EDT..
UP

MESH D



```
ED MSHD3.5C
LI,1,200
      NTRIAN  MNODE     NCOL
        46      35        4

        RANGE  SIGMAT   SIGMAS
         3.      1.       .5

      TRIANGLE            NODE1    NODE2    NODE3   COLUMN
          1                 1       11       10        1
          2                11        1        2        1
          3                11        2        3        1
          4                 3       12       11        1
          5                12        3        4        1
          6                12        4        5        1
          7                 5       13       12        1
          8                 5       14       13        1
          9                14        5        6        1
         10                14        6        7        1
         11                 7       15       14        1
         12                15        7        8        1
         13                15        8        9        1
         14                 9       16       15        1
         15                17       10       11        2
         16                11       18       17        2
         17                18       11       12        2
         18                12       19       18        2
         19                19       12       13        2
         20                19       13       14        2
         21                14       20       19        2
         22                20       14       15        2
         23                15       21       20        2
         24                21       15       16        2
         25                17       23       22        3
         26                23       17       18        3
         27                18       24       23        3
         28                24       18       19        3
         29                19       25       24        3
         30                19       26       25        3
         31                26       19       20        3
         32                20       27       26        3
         33                27       20       21        3
         34                21       28       27        3
         35                29       22       23        4
         36                23       30       29        4
         37                30       23       24        4
         38                24       31       30        4
         39                31       24       25        4
         40                31       25       26        4
         41                26       32       31        4
         42                26       33       32        4
         43                33       26       27        4
         44                27       34       33        4
         45                27       35       34        4
         46                35       27       28        4
```

COLUMN	FIRST ELEMENT	NUMBER OF ELEMENTS
1	1	14
2	15	10
3	25	10
4	35	12

NODE	X-AXIS	U-AXIS
1	0.000	1.000
2	0.000	0.750
3	0.000	0.500
4	0.000	0.250
5	0.000	0.000
6	0.000	-0.250
7	0.000	-0.500
8	0.000	-0.750
9	0.000	-1.000
10	0.250	1.000
11	0.250	0.750
12	0.250	0.250
13	0.250	0.000
14	0.250	-0.250
15	0.250	-0.750
16	0.250	-1.000
17	0.500	1.000
18	0.500	0.500
19	0.500	0.000
20	0.500	-0.500
21	0.500	-1.000
22	0.750	1.000
23	0.750	0.750
24	0.750	0.250
25	0.750	0.000
26	0.750	-0.250
27	0.750	-0.750
28	0.750	-1.000
29	1.000	1.000
30	1.000	0.500
31	1.000	0.000
32	1.000	-0.250
33	1.000	-0.500
34	1.000	-0.750
35	1.000	-1.000

NODE (NB)	FLUX
20	
1	1.0142E+00
3	9.8832E-01
4	7.6712E-01
6	9.7586E-01
7	5.2627E-01
9	9.8330E-01
10	2.7547E-01
12	8.1002E-01
13	1.2126E-01
15	6.1684E-01
91	6.5917E-03

111	93	5.3892E-03
112	94	5.2444E-03
113	96	4.3298E-03
114	97	4.4268E-03
115	99	2.9434E-03
116	100	3.7727E-03
117	102	2.2812E-03
118	103	3.2862E-03
119	105	1.9460E-03

EOT..

Appendix G - Subroutines to numerically evaluate the scattering integral

Glossary of Variables

x1,x2,x3,u1,u2,u3 - triangle geometric coordinates

x(49),u(49) - coordinates of 49 integration points

MC - coefficient matrix of eqn 2-4

DET - deteminate of MC

DU(7) - delta u at each of the seven spatial integration points

LX(3) - derivatives of ℓ_1, ℓ_2 and ℓ_3 w.r.t. χ

DX(mntria) - column width

L (49,3) - array storing natural coordinates of integration points

M(49,10) - array storing m of 2-29 evaluated at the integration points

MX(49,10) - array of $\underline{m}_\chi$ of 2-30

NLM - non local matrix

LI - local integral sum of $\int u \frac{\partial \phi}{\partial x} du + \int \phi du$

NLI - non local integral $\int \phi' du'$

ILDF - integral local derivative of flux

FLUX(49,10) - flux at integration points

DFLUX(49,10) - derivative of flux

UI1,UI2,...,UI7 - u integrals at the seven points needed for spatial integration

```
LI,1,160
*************************************************************


      SUBROUTINE LCORD(AREAS,TRI,PTNODE,CORDND,M,MX,DU,DX,U,X)

      PARAMETER (MNODE=151 , MNTRIA=46)
      DOUBLE PRECISION X1,X2,X3,U1,U2,U3,XX(7),X(49),U(49)
      DOUBLE PRECISION MC(3,3),DET,DU(7),LX(3),DX(MNTRIA),L(49,3)
      DOUBLE PRECISION AREAS(MNTRIA),CORDND(MNODE,2),M(49,10),MX(49,
       DOUBLE PRECISION ML(MNTRIA,10,10),MG(MNODE,MNODE)
       DOUBLE PRECISION NLM(MNTRIA,14,10,10),LI(MNTRIA,10,7)
       DOUBLE PRECISION NLI(MNTRIA,7,10),AS(MNODE*(MNODE-1)/2),F,G
       INTEGER CASE,TRI,PTNODE(MNTRIA,11)
       COMMON MG,ML,NLM,LI,NLI,AS

* THIS SUBROUTINE FINDS THE NATURAL COORDINATES NEEDED FOR
* NUMERICAL INTEGRATION OF THE SCATTERING INTEGRAL
* 49 POINTS FOR WEDDLES N=6

* GET THE (X,U) COORDINATES OF THE TRIANGLE
       X1=CORDND(PTNODE(TRI,1),1)
       X2=CORDND(PTNODE(TRI,4),1)
       X3=CORDND(PTNODE(TRI,7),1)
       U1=CORDND(PTNODE(TRI,1),2)
       U2=CORDND(PTNODE(TRI,4),2)
       U3=CORDND(PTNODE(TRI,7),2)

* DETERMINE THE ORIENTATION OF THE ELEMENT
       CASE=2
       IF (X1.GT.X2) THEN
           CASE=1
         ENDIF

* GET X COORDS OF NUMERICAL INTEGRATION POINTS
       XX(1)=MIN(X1,X2,X3)
       XX(7)=MAX(X1,X2,X3)
       F=(XX(7)-XX(1))/6.0
       DO 50 I=2,6
           XX(I)=XX(1)+(I-1)*F
50       CONTINUE
       DO 70 I=1,7
           J=7*I-6
           DO 60 K=J,J+6
               X(K)=XX(I)
60           CONTINUE
70       CONTINUE


* GET U COORDS OF THE SAME POINTS
       IF (CASE.EQ.1) THEN
           U(1)=U3
           U(7)=U2
           F=(U1-U3)/6.0
           G=(U1-U2)/6.0
       DO 80 I=8,36,7
```

```
              J=(I-1)/7
              U(I)=U3+J*F
              U(I+6)=U2+J*G
   80       CONTINUE
              U(43)=U1
              U(49)=U1
            ELSE
              U(1)=U1
              U(7)=U1
              F=(U2-U1)/6.0
              G=(U3-U1)/6.0
              DO 95 I=8,36,7
                  J=(I-1)/7
                  U(I)=U1+J*F
                  U(I+6)=U1+J*G
   95       CONTINUE
              U(43)=U2
              U(49)=U3
            ENDIF
          DO 100 I=1,43,7
              F=(U(I+6)-U(I))/6.0
              DO 98 J=1,5
                  U(I+J)=U(I)+J*F
   98           CONTINUE
   100      CONTINUE

 * COMPUTE THE LOCAL NATURAL COORDINATES
 * INVERSE USING ADJOINT AND DETERMINANT
          DET=2.0*AREAS(TRI)
          MC(1,1)=(X2*U3-X3*U2)/DET
          MC(1,2)=(U2-U3)/DET
          MC(1,10)=(X3-X2)/DET
          MC(2,1)=(X3*U1-X1*U3)/DET
          MC(2,2)=(U3-U1)/DET
          MC(2,3)=(X1-X3)/DET
          MC(3,1)=(X1*U2-X2*U1)/DET
          MC(3,2)=(U1-U2)/DET
          MC(3,3)=(X2-X1)/DET

 * ASSEMBLE THE NATURAL COORDINATES INTO ARRAY L(49,3)
          DO 110 I=1,49
              DO 105 J=1,3
                  L(I,J)=MC(J,1) + MC(J,2)*X(I) + MC(J,3)*U(I)
   105          CONTINUE
   110      CONTINUE

 * FIND DELTA U AT THE SEVEN LOCATIONS WHERE INTEGRATION
 * OVER U IS NECESSARY
          DO 120 I=1,7
              J=(I-1)*7+1
              DU(I)=U(J+6)-U(J)
   120      CONTINUE

 * ASSEMBLE DERIVATIVES OF NATURAL COORDINATES INTO LX(3)
 * AND CALCULATE INTERVAL WIDTH FOR X INTEGRATION
          LX(1)=(U2-U3)/DET
```

```
      LX(2)=(U3-U1)/DET
      LX(3)=(U1-U2)/DET
      DX(TRI)=X(49)-X(1)
* EVALUATE M AND dM/dX AT THE 49 INTEGRATION POINTS
      DO 150 I=1,49
          M(I,1)=L(I,1)**3
          MX(I,1)=3.0*(L(I,1)**2)*LX(1)
          M(I,2)=L(I,1)**2 * L(I,2)
          MX(I,2)=L(I,1)*L(I,2)*2.0*LX(1) + L(I,1)**2 * LX(2)
          M(I,3)=L(I,1)**2 * L(I,3)
          MX(I,3)=L(I,1)*L(I,3)*2.0*LX(1) + L(I,1)**2 * LX(3)
          M(I,4)=L(I,2)**3
          MX(I,4)=L(I,2)**2 *3.0*LX(2)
          M(I,5)=L(I,2)**2 *L(I,3)
          MX(I,5)=L(I,2)*2.0*L(I,3)*LX(2) + L(I,2)**2 *LX(3)
          M(I,6)=L(I,2)**2 *L(I,1)
          MX(I,6)=L(I,2)*2.0*LX(2)*L(I,1) + L(I,2)**2 *LX(1)
          M(I,7)=L(I,3)**3
          MX(I,7)=3.0*L(I,3)**2 * LX(3)
          M(I,8)=L(I,3)**2 *L(I,1)
          MX(I,8)=2.0*LX(3)*L(I,3)*L(I,1) + L(I,3)**2 *LX(1)
          M(I,9)=L(I,3)**2 *L(I,2)
          MX(I,9)=2.0*L(I,3)*LX(3)*L(I,2) + L(I,3)**2 *LX(2)
          M(I,10)=L(I,1)*L(I,2)*L(I,3)
          MX(I,10)=LX(1)*L(I,2)*L(I,3) +LX(2)*L(I,1)*L(I,3)
          MX(I,10)=MX(I,10) + LX(3)*L(I,1)*L(I,2)
150     CONTINUE
      END
EOF..
EOT..
```

```
LI,1,100
*************************************************************

* THIS SUBROUTINE PERFORMS WEDDLES N=6 RULE INTEGRATION OVER
* A TRIANGLE, IN THE U DIRECTION, AT THE SEVEN
* X COORDINATES, FOR USE IN THE NUMERICAL INTEGRATION OVER
* SPACE IN SUBROUTINE ISPACE

      SUBROUTINE ANING(DU,M,GT,MX,U,TRI,SIGMAT,SIGMAS,X,CORDND)

       PARAMETER (MNODE=151 , MNTRIA=46)
       DOUBLE PRECISION DU(7),M(49,10),GT(10,10),ILDF(10,10)
       DOUBLE PRECISION U(49),MX(49,10),DFLUX(49,10),FLUX(49,10)
       DOUBLE PRECISION ML(MNTRIA,10,10),MG(MNODE,MNODE)
       DOUBLE PRECISION NLM(MNTRIA,14,10,10),LI(MNTRIA,10,7)
       DOUBLE PRECISION NLI(MNTRIA,7,10),AS(MNODE*(MNODE-1)/2)
       DOUBLE PRECISION SIGMAT,SIGMAS,A,B
       DOUBLE PRECISION X(49),CORDND(MNODE,2)
       INTEGER TRI
       COMMON MG,ML,NLM,LI,NLI,AS

* CALCULATE FLUX AT THE FORTY NINE INTEGRATION POINTS
* AND CALCULATE THE DERIVATIVE OF FLUX IN THE X DIRECTION
* AT THE INTEGRATION POINTS
       DO 100 I=1,49
           DO 50 J=1,10
               FLUX(I,J)=0.0
               DFLUX(I,J)=0.0
               DO 25 K=1,10
                   FLUX(I,J)=FLUX(I,J) + M(I,K)*GT(K,J)
                   DFLUX(I,J)=DFLUX(I,J) + MX(I,K)*GT(K,J)
25                 CONTINUE
50             CONTINUE
100        CONTINUE

* CALCULATE THE INTEGRAL OF FLUX OVER U
* AT THE SEVEN SPATIAL INTEGRAL POINTS
* PLACE IN ROWS, THIS IS NON-LOCAL INTEGRAL
       DO 200 I=1,7
           K=7*I-6
           DO 150 J=1,10
               NLI(TRI,I,J)=(DU(I)/20.0)*(FLUX(K,J)+5.0*FLUX(K+1,J)
      C   +FLUX(K+2,J)+6.0*FLUX(K+3,J)+FLUX(K+4,J)+5.0*FLUX(K+5,J)
      C   +FLUX(K+6,J))
150            CONTINUE
200        CONTINUE


* CALCULATE INTEGRAL U*DFLUX - ARANGE INTO COLUMNS, ADD
* NLI TO OBTAIN THE LOCAL INTEGRAL
       A=-SIGMAS
       B=.5*SIGMAS*SIGMAS - SIGMAS*SIGMAT

       DO 300 I=1,7
           K=7*I-6
           DO 250 J=1,10
```

```
              ILDF(I,J)=(DU(I)/20.0)*(DFLUX(K,J)*U(K)+5.0*DFLUX(K+1,
      C *U(K+1)+DFLUX(K+2,J)*U(K+2)+6.0*DFLUX(K+3,J)*U(K+3)+
      C DFLUX(K+4,J)*U(K+4)+5.0*DFLUX(K+5,J)*U(K+5)+
      C DFLUX(K+6,J)*U(K+6))*A
    250         CONTINUE
    300       CONTINUE
            DO 400 I=1,7
                DO 350 J=1,10
                   LI(TRI,J,I)=ILDF(I,J) + NLI(TRI,I,J)*B
    350         CONTINUE
    400       CONTINUE

            END

OF..
EOT..
UP
```

```
LI,1,50
 ****************************************************************

 * INTEGRATE OVER SPACE (X), ACROSS THE LOCAL TRIANGLE

       SUBROUTINE SPING(DX,TRI,TRIP)

       PARAMETER (MNODE=151 , MNTRIA=46)
       DOUBLE PRECISION NLI(MNTRIA,7,10),AS(MNODE*(MNODE-1)/2)
       DOUBLE PRECISION LI(MNTRIA,10,7)
        DOUBLE PRECISION UI1(10,10),UI2(10,10),UI3(10,10)
        DOUBLE PRECISION UI4(10,10),DX(MNTRIA),NLM(MNTRIA,14,10,10)
        DOUBLE PRECISION ML(MNTRIA,10,10),MG(MNODE,MNODE)
        DOUBLE PRECISION UI5(10,10),UI6(10,10),UI7(10,10)
        INTEGER TRI,TRIP
        COMMON MG,ML,NLM,LI,NLI,AS

 * TAKE PRODUCT OF LI, AND INLF - RECALL LI IS IN
 * COLUMNS, AND INLF IN ROWS
        DO 100 I=1,10
            DO 50 J=1,10
                UI1(I,J)=LI(TRI,I,1)*NLI(TRIP,1,J)
                UI2(I,J)=LI(TRI,I,2)*NLI(TRIP,2,J)
                UI3(I,J)=LI(TRI,I,3)*NLI(TRIP,3,J)
                UI4(I,J)=LI(TRI,I,4)*NLI(TRIP,4,J)
                UI5(I,J)=LI(TRI,I,5)*NLI(TRIP,5,J)
                UI6(I,J)=LI(TRI,I,6)*NLI(TRIP,6,J)
                UI7(I,J)=LI(TRI,I,7)*NLI(TRIP,7,J)
 50             CONTINUE
 100        CONTINUE


 * DO WEDDLES N=6 RULE INTEGRATION
        DO 200 I=1,10
            DO 150 J=1,10
                NLM(TRI,TRIP,I,J)=(DX(TRI)/20.0)*(UI1(I,J)+5.0*UI2(I,
     C  +UI3(I,J)+6.0*UI4(I,J)+UI5(I,J)+5.0*UI6(I,J)+UI7(I,J))
 150            CONTINUE
 200        CONTINUE

        END
EOF..
EOT..
```

Appendix H - Spherical Harmonic Angular Fluxes and Data File SDATA

This appendix contains the contents of three data files, PNDATA5, and PNDATA9 called to compare finite element angular fluxes in subroutine OUTPUT, and SDATA, a data file called by subroutine GDATA, which contains submatrices of the cubic three dimensional interpolate, as well as integrals of the 20 polynomials used in the three dimensional cubic fit.

The angular fluxes are those computed with 46 legendre polynomials. They are formatted differently in this appendix than in the manner the code of appendix A reads them.

XE
Pn BENCHMARK DATA WITH 46 LEGENDRE POLYNOMIALS
FOR c=.5

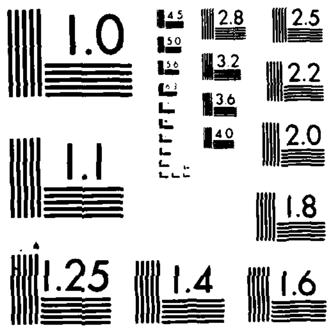
X	U=-1.0	-.75	-.50	-.25	0.0
0.00	0.5168E-01	0.7383E-01	0.1009E+00	0.1025E+00	0.1213E+00
0.25	0.4860E-01	0.6105E-01	0.7545E-01	0.8841E-01	0.1028E+00
0.50	0.4154E-01	0.4822E-01	0.5639E-01	0.6856E-01	0.8217E-01
0.75	0.3335E-01	0.3746E-01	0.4289E-01	0.5246E-01	0.6354E-01
1.00	0.2628E-01	0.2905E-01	0.3295E-01	0.4038E-01	0.4931E-01
1.25	0.2050E-01	0.2251E-01	0.2544E-01	0.3115E-01	0.3838E-01
1.50	0.1587E-01	0.1742E-01	0.1972E-01	0.2405E-01	0.2986E-01
1.75	0.1223E-01	0.1347E-01	0.1532E-01	0.1859E-01	0.2320E-01
2.00	0.9401E-02	0.1043E-01	0.1193E-01	0.1439E-01	0.1803E-01
2.25	0.7221E-02	0.8074E-02	0.9299E-02	0.1116E-01	0.1401E-01
2.50	0.5548E-02	0.6259E-02	0.7256E-02	0.8665E-02	0.1089E-01
2.75	0.4267E-02	0.4857E-02	0.5666E-02	0.6737E-02	0.8471E-02
3.00	0.3286E-02	0.3773E-02	0.4427E-02	0.5244E-02	0.6592E-02
3.25	0.2536E-02	0.2934E-02	0.3460E-02	0.4086E-02	0.5132E-02
3.50	0.1960E-02	0.2284E-02	0.2705E-02	0.3187E-02	0.3998E-02
3.75	0.1519E-02	0.1779E-02	0.2114E-02	0.2488E-02	0.3117E-02
4.00	0.1179E-02	0.1388E-02	0.1653E-02	0.1943E-02	0.2431E-02
4.25	0.9172E-03	0.1084E-02	0.1293E-02	0.1519E-02	0.1897E-02
4.50	0.7150E-03	0.8466E-03	0.1011E-02	0.1188E-02	0.1481E-02
4.75	0.5585E-03	0.6620E-03	0.7910E-03	0.9292E-03	0.1157E-02
5.00	0.4370E-03	0.5181E-03	0.6188E-03	0.7274E-03	0.9044E-03

X	U=.25	.50	.75	1.0
0.00	0.2755E+00	0.5263E+00	0.7671E+00	0.1014E+01
0.25	0.1821E+00	0.3626E+00	0.5886E+00	0.8265E+00
0.50	0.1300E+00	0.2580E+00	0.4498E+00	0.6647E+00
0.75	0.9471E-01	0.1863E+00	0.3433E+00	0.5328E+00
1.00	0.7029E-01	0.1359E+00	0.2619E+00	0.4266E+00
1.25	0.5301E-01	0.1001E+00	0.2000E+00	0.3413E+00
1.50	0.4042E-01	0.7435E-01	0.1528E+00	0.2730E+00
1.75	0.3104E-01	0.5561E-01	0.1169E+00	0.2182E+00
2.00	0.2396E-01	0.4185E-01	0.8950E-01	0.1744E+00
2.25	0.1856E-01	0.3166E-01	0.6861E-01	0.1393E+00
2.50	0.1441E-01	0.2406E-01	0.5265E-01	0.1112E+00
2.75	0.1122E-01	0.1835E-01	0.4045E-01	0.8878E-01
3.00	0.8743E-02	0.1405E-01	0.3111E-01	0.7084E-01
3.25	0.6821E-02	0.1079E-01	0.2395E-01	0.5649E-01
3.50	0.5326E-02	0.8310E-02	0.1845E-01	0.4503E-01
3.75	0.4160E-02	0.6415E-02	0.1423E-01	0.3588E-01
4.00	0.3252E-02	0.4962E-02	0.1099E-01	0.2857E-01
4.25	0.2542E-02	0.3846E-02	0.8489E-02	0.2275E-01
4.50	0.1988E-02	0.2985E-02	0.6564E-02	0.1810E-01
4.75	0.1555E-02	0.2321E-02	0.5080E-02	0.1440E-01
5.00	0.1217E-02	0.1806E-02	0.3934E-02	0.1144E-01

AD-A159 245 A FINITE ELEMENT SOLUTION OF THE TRANSPORT EQUATION(U) 3/3
AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL
OF ENGINEERING F A TARANTINO MAR 85 AFIT/GNE/PH/85M-19

UNCLASSIFIED F/G 12/1 NL

END
FILMED
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

XE
Pn BENCHMARK DATA WITH 46 LEGENDRE POLYNOMIALS
FOR c=.9

X	U=-1.0	-.75	-.50	-.25	0.0
0.00	0.1465E+00	0.1566E+00	0.1708E+00	0.1956E+00	0.2500E+00
0.25	0.1192E+00	0.1322E+00	0.1484E+00	0.1636E+00	0.1915E+00
0.50	0.1039E+00	0.1153E+00	0.1289E+00	0.1433E+00	0.1634E+00
0.75	0.9149E-01	0.1009E+00	0.1121E+00	0.1252E+00	0.1421E+00
1.00	0.8035E-01	0.8826E-01	0.9774E-01	0.1093E+00	0.1238E+00
1.25	0.7049E-01	0.7725E-01	0.8538E-01	0.9549E-01	0.1081E+00
1.50	0.6181E-01	0.6763E-01	0.7467E-01	0.8352E-01	0.9453E-01
1.75	0.5417E-01	0.5923E-01	0.6535E-01	0.7309E-01	0.8273E-01
2.00	0.4746E-01	0.5188E-01	0.5723E-01	0.6399E-01	0.7243E-01
2.25	0.4158E-01	0.4544E-01	0.5013E-01	0.5603E-01	0.6343E-01
2.50	0.3642E-01	0.3981E-01	0.4393E-01	0.4908E-01	0.5557E-01
2.75	0.3191E-01	0.3489E-01	0.3850E-01	0.4301E-01	0.4868E-01
3.00	0.2796E-01	0.3058E-01	0.3374E-01	0.3768E-01	0.4266E-01
3.25	0.2450E-01	0.2680E-01	0.2958E-01	0.3303E-01	0.3738E-01
3.50	0.2147E-01	0.2349E-01	0.2593E-01	0.2895E-01	0.3277E-01
3.75	0.1882E-01	0.2059E-01	0.2273E-01	0.2538E-01	0.2872E-01
4.00	0.1650E-01	0.1805E-01	0.1993E-01	0.2225E-01	0.2518E-01
4.25	0.1446E-01	0.1582E-01	0.1747E-01	0.1950E-01	0.2207E-01
4.50	0.1268E-01	0.1387E-01	0.1532E-01	0.1710E-01	0.1935E-01
4.75	0.1111E-01	0.1216E-01	0.1343E-01	0.1499E-01	0.1696E-01
5.00	0.9745E-02	0.1066E-01	0.1178E-01	0.1314E-01	0.1487E-01

X	U=.25	.50	.75	1.0
0.00	0.3044E+00	0.3292E+00	0.3434E+00	0.3534E+00
0.25	0.2396E+00	0.2817E+00	0.3049E+00	0.3228E+00
0.50	0.1983E+00	0.2387E+00	0.2690E+00	0.2926E+00
0.75	0.1690E+00	0.2041E+00	0.2362E+00	0.2623E+00
1.00	0.1455E+00	0.1757E+00	0.2068E+00	0.2336E+00
1.25	0.1261E+00	0.1519E+00	0.1809E+00	0.2074E+00
1.50	0.1098E+00	0.1318E+00	0.1582E+00	0.1838E+00
1.75	0.9575E-01	0.1146E+00	0.1384E+00	0.1626E+00
2.00	0.8366E-01	0.9986E-01	0.1210E+00	0.1437E+00
2.25	0.7318E-01	0.8713E-01	0.1059E+00	0.1268E+00
2.50	0.6405E-01	0.7611E-01	0.9261E-01	0.1118E+00
2.75	0.5609E-01	0.6654E-01	0.8105E-01	0.9856E-01
3.00	0.4914E-01	0.5821E-01	0.7095E-01	0.8681E-01
3.25	0.4305E-01	0.5094E-01	0.6212E-01	0.7642E-01
3.50	0.3773E-01	0.4460E-01	0.5440E-01	0.6723E-01
3.75	0.3307E-01	0.3907E-01	0.4765E-01	0.5913E-01
4.00	0.2899E-01	0.3422E-01	0.4174E-01	0.5199E-01
4.25	0.2541E-01	0.2999E-01	0.3657E-01	0.4569E-01
4.50	0.2228E-01	0.2628E-01	0.3204E-01	0.4015E-01
4.75	0.1953E-01	0.2303E-01	0.2808E-01	0.3527E-01
5.00	0.1712E-01	0.2019E-01	0.2461E-01	0.3097E-01

```
ED SDATA
LI,1,35
1  LINES 2-11 V(5,20)
2     6.0     2.0     2.0     2.0     6.0     2.0     2.0     2.0     6.0     2.0
3     2.0     2.0     6.0     2.0     2.0     2.0     1.0     1.0     1.0     1.0
4    24.0     6.0     6.0     6.0     6.0     4.0     2.0     2.0     6.0     4.0
5     2.0     2.0     6.0     4.0     2.0     2.0     1.0     2.0     2.0     2.0
6     6.0     4.0     2.0     2.0    24.0     6.0     6.0     6.0     6.0     2.0
7     4.0     2.0     6.0     2.0     4.0     2.0     2.0     1.0     2.0     2.0
8     6.0     2.0     4.0     2.0     6.0     2.0     4.0     2.0    24.0     6.0
9     6.0     6.0     6.0     2.0     2.0     4.0     2.0     2.0     1.0     2.0
10     6.0     2.0     2.0     4.0     6.0     2.0     2.0     4.0     6.0     2.0
11     2.0     4.0    24.0     6.0     6.0     6.0     2.0     2.0     2.0     1.0
12 SUB MATRICES M5 THROUGH M8 AND M18 OF TETRAHEDRAL CUBIC
13 INTERPOLANT COEFFICIENT MATRIX
14     0.0     0.0     0.0     0.0
15     1.0     0.0     1.0     1.0
16     1.0     1.0     0.0     1.0
17     1.0     1.0     1.0     0.0
18     1.0     0.0     1.0     1.0
19     0.0     0.0     0.0     0.0
20     1.0     1.0     0.0     1.0
21     1.0     1.0     1.0     0.0
22     1.0     0.0     1.0     1.0
23     1.0     1.0     0.0     1.0
24     0.0     0.0     0.0     0.0
25     1.0     1.0     1.0     0.0
26     1.0     0.0     1.0     1.0
27     1.0     1.0     0.0     1.0
28     1.0     1.0     1.0     0.0
29     0.0     0.0     0.0     0.0
30    27.0     0.0     0.0     0.0
31     0.0    27.0     0.0     0.0
32     0.0     0.0    27.0     0.0
33     0.0     0.0     0.0    27.0
EDT..
```

VITA

Captain Frederick Angelo Tarantino was born on 25 August 1955 in Hudson Falls New York. He graduated from St. Marys Academy of Glens Falls New York in June of 1973, and continued his studies at Rensellaer Polytechnic Institute, where he earned his B.S. in physics. A Distinguished Military Graduate of that institution's Army ROTC program, Tarantino received a RA commission in the U.S. Army Infantry. Military duties have included mechanized infantry assignments both overseas and CONUS, where he commanded Attack Company of the 2nd Bn(M) 34th Infantry, Fort Stewart Ga. While assigned to the Air Force Institute of Technology he pursued a Master of Science degree in Nuclear Science. Upon graduation, Tarantino will be assigned as a Military Research Associate at Lawrence Livermore Laboratories, Livermore California. He is a member of Tau Beta Pi. Married to the former Jazmine T. Herrera of Panama, Rep. of Pma., they have two children, Michael John and Monica Maria.

Permanent Address

7 Pender Street

Hudson Falls, N.Y.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

AD-A159245

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GNE/PH/85M-19	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION School of Engineering 6b. OFFICE SYMBOL (If applicable) AFIT/ENG	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS (City, State and ZIP Code) Air Force of Technology Wright-Patterson AFB, Ohio 45433	7b. ADDRESS (City, State and ZIP Code)
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State and ZIP Code)	10. SOURCE OF FUNDING NOS. PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT NO.
11. TITLE (Include Security Classification) See box 19	
12. PERSONAL AUTHOR(S) Frederick A. Tarantino, B.S., Cpt. IN, USA	

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Yr., Mo., Day)	15. PAGE COUNT
MS Thesis	FROM ____ TO ____	1985 March	195

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB. GR.	
18	11		Finite Elements; Variational Methods; Neutron Transport Theory; Numerical Methods of Transport; Numerical Analysis; Self-Adjoint Transport Operator

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Title : A FINITE ELEMENT SOLUTION OF THE TRANSPORT EQUATION

Thesis Chairman : Leslie L. Mckee
Maj(P) USAF, Asst. Professor of Nuclear Engineering

Approved for public release: IAW AFR 190-1.
LYNN E. WOLAVER 19 AUG 85
Dean for Research and Professional Development
Air Force Institute of Technology (AFC)
Wright-Patterson AFB OH 45433

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS ☐	UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE NUMBER (Include Area Code)	22c. OFFICE SYMBOL
Leslie L. Mckee Maj(P)	513-255-5533	AFIT/ENG/PH

DD FORM 1473, 83 APR EDITION OF 1 JAN 73 IS OBSOLETE.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

END

FILMED

10-85

DTIC