LA-6627-MS

Informal Report

UC-32

Issued: December 1976

ESECT/EMAP:
Mapping Algorithm for Computing Intersection
Volumes of Overlaid Meshes in Cylindrical Geometry

by

B. R. Wienke
R. D. O'Dell

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87545

An Affirmative Action/Equal Opportunity Employer

MASTER

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161
Price: Printed Copy \$4.00 Microfiche \$3.00

The report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research and Development Administration nor any of their employees or agents of their contractors, subcontractors, or their employees make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.

**ESECT/EMAP:
MAPPING ALGORITHM FOR COMPUTING INTERSECTION VOLUMES OF
OVERLAID MESHES IN CYLINDRICAL GEOMETRY**

by

D. H. Wienke and K. D. O'Dell

ABSTRACT

The authors have developed an algorithm for computing intersection volumes of two overlapping meshes in cylindrical geometry. The algorithm is based on the principle of inclusion-exclusion and is applicable to arbitrary meshes. The algorithm is implemented in a computer program which is available to the public. The program is written in Fortran and runs on the CDC 7600 computer. The program is capable of computing intersection volumes for meshes with up to 1000 nodes and 1000 elements. The program is also capable of computing the intersection volumes for meshes with up to 1000 nodes and 1000 elements. The program is also capable of computing the intersection volumes for meshes with up to 1000 nodes and 1000 elements.

1. Subroutine Identification: ESECT/EMAP.
2. Computer: CDC 7600, CDC 6600.
3. Description: ESECT and EMAP are subroutines which provide a computer algorithm for mapping arbitrary meshes onto rectangular meshes in cylindrical (r,z) geometry. Intersection volumes between two meshes are computed and stored along with two indices tagging the intersection zones on each mesh. Input consists of the lines defining the rectangular mesh and the coordinates of the arbitrary mesh, which are assumed to be joined by straight lines. Output consists of the intersection volumes with designation of common mesh zones. The ESECT and EMAP routines do not comprise a "free-standing" code but, instead, are intended for inclusion in existing codes for which one mesh structure (typically Lagrangian) needs to be mapped onto an Eulerian mesh.
4. Method of Solution: Exact expressions for the volumes of rotation (about z-axis) generated by the planar mesh intersection areas are used. Intersection points of the two meshes are computed and mapped onto corresponding regions on the rectangular mesh. Intersection points with the same regional indices are reordered into multilaterals and the multilaterals triangulated to facilitate computation of the intersection volumes.
5. Restrictions: Dimension statements within ESECT/EMAP presently allow for rectangular and arbitrary meshes of 10k and 3.6k grid points. The intersection volumes, cell volumes and indices are stored in large core memory (LCM) and all other arrays in small core memory (SCM). Scaling of all arrays to suit individual applications is easily effected.
6. Running time: Computations of intersection volumes generated by overlapping 10k rectangular and 2.3k radial meshes require an average of 18 s computer time, while computation times for the same meshes scaled by a factor of 1/4 in number of grid points average 3 s on the CDC 7600. The arbitrary meshes employed in the calculations are symmetrical center converging webs of 5° and 20° segments, respectively, with the largest number of cells near the center of convergence. The rectangular meshes are uniform. Generally, cases of small cell rectangular meshes overlaid on large cell arbitrary meshes require the longer running times.

7. Unusual Features: None.
8. Machine Requirements: CDC 7600, CROS operating system, with standard input/output units is required. Large bulk memory is desirable for interfacing.
9. Related Programs: GUANO-S_N
10. Material Available: Source deck (560 cards), test problems and this report available from author (S. R. Wienke). Listings of the subroutines appear in this report.

I. INTRODUCTION

The routines ESECT and EMAP are a general purpose package which maps an arbitrary grid onto a rectangular mesh and computes intersection volumes in cylindrical geometry. Such mappings are of interest in coupled hydrodynamic and neutronic calculations, with the hydrodynamics done on a Lagrangian grid and the neutronics performed on a fixed rectangular Eulerian grid, and also in other hydrodynamic applications requiring smearing of material properties from various configurations onto an Eulerian grid. The mapping algorithms and calculations in ESECT/EMAP are, however, independent of physical systems and constraints and can be employed generally in applications.

Input consists of two sets of meshes, subject to the constraint that the rectangular mesh enclose the other grid. Adapting common nomenclature, the rectangular mesh is also referred to as the Eulerian mesh while the arbitrary grid is also called the Lagrangian mesh. The Lagrangian mesh may be rectangular and the Lagrangian zones may be three or four sided. Output consists of a set of intersection volumes with designation of common, parent Lagrangian and Eulerian cells. Computed intersection volumes are exact, no approximations are employed. Execution diagnostics flag overrun arrays, coordinates of Lagrangian cells not amenable to the mapping algorithms, troublesome cell shapes such as two-sided figures, inversions, and improperly ordered multilaterals. Execution terminates at possible infractions and a listing of Lagrangian zone coordinates, intersections, and mapping information is printed for clarification. No attempts are made internally to rectify Lagrangian cell shapes which pose the above incompatibilities.

The mapping algorithm consists of two segments: ESECT, the first main subroutine, which receives the Eulerian grid lines and Lagrangian grid points, performs the sweep, the multilateral reordering, and the computations; and EMAP, the second subroutine (called from ESECT) which maps all computed intersections and Lagrangian cell vertices onto Eulerian zones prior to processing by ESECT.

Users must supply the requisite mesh inputs to ESECT through COMMON or DATA statements. There is no provision in ESECT for reading "card" input.

For test purposes only a driver test program with card input is available from the authors for the sample problems given in Appendix A.

The intersection volumes computed can be used to volume weight (smear) arbitrary quantities such as cross sections, sources, fluxes, scalar densities, material properties, masses, energies, etc., in transforming from Lagrangian to Eulerian representations, and vice versa in transport, hydrodynamic, and other related application areas.

II. INTERSECTION VOLUMES AND GEOMETRY

Some types of overlapping cylindrical grids processed by ESECT/EMAP are shown in Figs. 1, 2, 3, and 4. Rotation through angle 2π is assumed about the z-axis.

Fig. 1. Uniform rectangular Eulerian, nonuniform rectangular Lagrangian.

Fig. 2. Nonuniform rectangular Eulerian, nonuniform quadrilateral Lagrangian.

Fig. 3. Uniform rectangular Eulerian, symmetrical center-converging web Lagrangian.

Fig. 4. Nonuniform rectangular Eulerian, nonsymmetrical center-converging web Lagrangian.

The rectangular Eulerian meshes may be uniform or unevenly spaced. The Lagrangian meshes are arbitrary and the cells may be three or four sided. The r -values of the coordinates range $0 \leq r \leq \infty$ while the z -values of the coordinates range $-\infty \leq z \leq \infty$. As mentioned before, the Eulerian mesh should completely enclose the Lagrangian mesh. The Eulerian meshes are drawn in solid lines, while the Lagrangian meshes are drawn in dotted lines in the following figures.

Figure 5 shows a possible intersection configuration. The quadrilateral Lagrangian cell (dotted lines) intersects the Eulerian mesh in eight points. The Eulerian mesh also intersects itself at two points in the Lagrangian cell interior. Along with the 4 vertices of the Lagrangian cell, these internal and boundary intersections provide a 14-point multilateral intersection domain. From these 14 points, it is not hard to see that 7 multilateral figures emerge, each in the same Lagrangian cell but a different Eulerian cell. Each of these seven multilaterals is triangulated, as indicated in Fig. 6, after appropriate vertices of the multilateral are sequentially reordered. The triangular subregions of the multilateral are then rotated about the z -axis and summed to give the total intersection volume of the particular intersection multilateral. Indices tagging the parent Lagrangian and Eulerian cells containing the multilaterals are also computed and stored along with the intersection volumes.

The ordering sequence for these Lagrangian and Eulerian zone indices is shown in Figs. 7 and 8. The indices on the Eulerian mesh start with the smallest z -values and increase continuously; first, with increasing r -values and then with increasing z -values. The Lagrangian indices, however, start with largest z -values and increase continuously, first, with increasing r -values and then with decreasing z -values. For the case of a center converging web as shown in Fig. 8 one imagines that lines of equal z -values are now passed through the origin so that the previously indicated numbering scheme is transformed as shown.

The actual intersection volume v computed over small region v common to the l^{th} Lagrangian cell and e^{th} Eulerian cell is formally written

$$v_{le} = \int_v r dr dz d\phi, \quad (1)$$

where, if v_l denotes the volume of the l^{th} Lagrangian cell and v_e denotes the volume of the e^{th} Eulerian cell,

$$v_l = \sum_e v_{le}, \quad v_e = \sum_l v_{le}. \quad (2)$$

Rotating about the z -axis through angle 2π , one has

$$v_{le} = 2\pi \int_v r dr dz = 2\pi \int_v r dA, \quad (3)$$

with area element $dA = r dr dz$. Except for rectangular regions, evaluation of Eq. (3) over v directly is tedious. For this reason, the multilateral regions designated by v is first triangulated. Centroid arms of triangles of revolution terminate at the intersections of the three medians. By computing these arms, multiplying them by corresponding triangular areas and summing these products

Fig. 5. Intersection domain and multi-lateral formation for overlapping meshes.

Fig. 6. Triangulation of intersection multilateral.

Fig. 7. Eulerian and Lagrangian cell numbering sequence 1.

Fig. 8. Eulerian and Lagrangian cell numbering sequence 2.

over the triangulated multilateral region, v , exact intersection volumes can be computed for any multilateral.

Accordingly, the intersection volume common to the l^{th} Lagrangian and e^{th} Eulerian cell is finally written

$$v_{le} = 2\pi \sum_{i=1}^{m-2} \bar{r}_i \Delta A_i \quad (4)$$

with m designating the number of sides of the multilateral, ΔA_i the area of the i^{th} triangular element

$$\Delta A_i = \frac{1}{2} |(r_{i3} - r_{i2})z_{i1} + (r_{i1} - r_{i3})z_{i2} + (r_{i2} - r_{i1})z_{i3}|. \quad (5)$$

$\bar{r}_i$ the centroid arm of the i^{th} triangular element

$$\bar{r}_i = \frac{1}{3} (r_{i1} + r_{i2} + r_{i3}) \quad (6)$$

and (r_{i1}, z_{i1}) , (r_{i2}, z_{i2}) , (r_{i3}, z_{i3}) the vertices of the i^{th} triangle. From Fig. 6, it is obvious that a multilateral with m sides can be triangulated into $(m-2)$ distinct, nonoverlapping elements. The individual triangular elements ΔA_i sum to the total multilateral area

$$\sum_{i=1}^{m-2} \Delta A_i = \int_v dA. \quad (7)$$

For purposes of mapping or averaging quantities between meshes, the standard volume weighting fractions are defined

$$f_e = \frac{v_{le}}{v_l} \quad (8)$$

$$f_l = \frac{v_{le}}{v_e},$$

with f_e , f_l the volume fractions normalized consistently with Eq. (2)

$$\sum_e f_e = 1, \quad \sum_l f_l = 1. \quad (9)$$

III. PROGRAM FLOW AND ANALYSIS

A schematic flow chart for ESECT/EMAP is shown in Fig. 9. A listing of the routines appears in Sec. IV. Briefly, the program logic and analysis proceeds in the following manner.

The Lagrangian grid point ZI,RI and Eulerian lines ZE,RE are passed to ESECT through a subroutine call statement. A preliminary edit and test is made on the mesh data and all other arrays are zeroed. A sweep is made through the Lagrangian cell vertices and the Eulerian line intersections with the Lagrangian cell boundaries and the self-intersections of the Eulerian lines in the interior of the Lagrangian cell are computed and stored in arrays ZI,RI. To all vertices of the Lagrangian cell and to all boundary and internal intersections are also assigned a set of two common line slopes (tagging the lines which intersect at the points) and a set of four integer Eulerian indices (denoting the possible Eulerian cells containing the points). The slopes are stored in arrays SI,SL while the mapping indices are contained in the arrays MLZ,MLR. Sweeps over the Eulerian mesh are controlled and determined by the vertices of the Lagrangian cell. The vertices of the Lagrangian cell define a maximum domain of intersection and subsequent sweeps to compute intersections and ranges of mapping searches are controlled by the parameters MXZ, MIZ, MXR, and M'R. Coarse region mapping parameters for the arrays ZI and RI are contained in arrays LZ,LR and serve as pivots for computing MLZ,MLR. The arrays ZI,RI contain all possible coordinates for construction of intersection multilaterals formed by the overlay of the Lagrangian cell on the Eulerian mesh. A sweep is then made through the intersection domain and all points possessing the same mapping indices MLZ,MLR are placed into the subarrays ZZI,RRI, while the corresponding line slopes are placed into the subarrays SLI and SOI. The subarrays contain the actual vertices and line slopes of the individual multilaterals. A reordering algorithm first sequentially joins the multilateral vertices through common line slopes checks to make sure that connected points lie on the same segments and then triangulates the figure. The intersection volume of the triangulated multilateral is finally computed using Eqs. (4), (5), and (6) and the results are given by VF and stored in the array VV. The indices tagging the Lagrangian and Eulerian cells common to the intersection multilateral, LM and IE respectively, are stored in the arrays LLM and IIE. The rotation volumes of the full parent Lagrangian and Eulerian cells tagged by LM and IE are computed exactly and stored in arrays GV and EV. Division of the intersection volumes VV by the corresponding parent cell volumes GV or EV produces a volume weighing fraction for mapping or averaging between meshes, as indicated by Eqs. (8) and (9). All computed volumes are normalized to 2π , that is, the assumed rotation angle about the z-axis is 2π .

The listing which appears in Appendix B annotates the above program variable.

In the event of default in the calculation due to overrun arrays, ordering or mapping failures, etc., a listing of the large intersection arrays RI, ZI, SI, SO, LR, LZ, MLR, and MLZ is first printed, followed by the smaller multilateral arrays RRI, ZZI, SLI, and SOI, the ordering indices KE, KD, and the zone parameters LM,IE.

The output from three sample test problems is shown in Appendix A. All three test problems print out the rotation angle and radial segments of a center converging Lagrangian web and the z and r lines of the Eulerian mesh. The first two problems merely list the execution times and numbers of intersection volumes computed. Test Problem 3 additionally lists the computed intersection volumes, Lagrangian and Eulerian indices and parent Lagrangian and Eulerian cell volumes for the simple grid shown in Fig. 10. The dotted lines and numbers indicate

Fig. 9. Flow chart.

Fig. 10. Test Grid.

the corresponding Lagrangian and Eulerian zones and indices. The driver test program (ETEST) and input listing for the sample problems appear in Appendices C and D.

IV. APPLICATIONS AND RESULTS

Timing runs for various mesh configurations are given in Tables I and II. The total number of intersection volumes computed are also listed alongside run times. In Table I, Eulerian mesh spacings are uniform and radial and angular segments of the center-converging Lagrangian mesh are equal. Geometrically, the meshes compared in Table I are pictured as semicircular webs fully inscribed in rectangles with the boundaries of the rectangles tangent at angles of 0° , 90° , and 180° . The meshes compared in Table II are both rectangular with coincident boundaries and equal spacing within each mesh. Run times are rounded off to next highest integer seconds and intersection volumes are rounded off to next highest hundred intersection number. It can be seen from these Tables that processing times vary from a minimum of approximately 1 s/1k intersection volumes for simple mesh configurations to a rough maximum of 3 s /1k intersection volumes for large Eulerian and small Lagrangian grids. Existing algorithms will process two orthogonal, rectangular grid patterns the fastest since line slopes and intercepts need not be computed, compared, and ordered.

TABLE I
COMPARATIVE RUN TIMES AND NUMBERS OF INTERSECTION
VOLUMES FOR CENTER-CONVERGING LAGRANGIAN/RECTANGULAR EULERIAN MESHES

	Eulerian 100 x 100 <u>(Rectangular)</u>	Eulerian 50 x 50 <u>(Rectangular)</u>	Eulerian 25 x 25 <u>(Rectangular)</u>
Lagrangian 36 x 100 <u>(5° Center Converging)</u>	27 s 25.9k int. vol.	10 s 14.5k int. vol.	5 s 8.5k int. vol.
Lagrangian 18 x 50 <u>(10° Center Converging)</u>	22 s 17.7k int. vol.	6 s 7.3k int. vol.	2 s 3.6k int. vol.
Lagrangian 9 x 25 <u>(20° Center Converging)</u>	37 s 12.3k int. vol.	6 s 4.3k int. vol.	1 s 1.8k int. vol.

TABLE II

COMPARATIVE RUN TIMES AND NUMBERS OF INTERSECTION
VOLUMES FOR RECTANGULAR LAGRANGIAN/RECTANGULAR EULERIAN MESHES

	Eulerian 100 x 100 (Rectangular)	Eulerian 50 x 50 (Rectangular)	Eulerian 25 x 25 (Rectangular)
Lagrangian 36 x 100 (Rectangular)	21 s 23.9k int. vol.	9 s 12.4k int. vol.	5 s 7.4k int. vol.
Lagrangian 18 x 50 (Rectangular)	12 s 11.6k int. vol.	3 s 3.3k int. vol.	1 s 2.1k int. vol.
Lagrangian 9 x 25 (Rectangular)	19 s 10.8k int. vol.	8 s 2.9k int. vol.	1 s 0.8k int. vol.

APPENDIX A
TEST PROBLEMS

TEST PROBLEM 1-FINDING AND COUNT FOR 10K EULERIAN AND 2.7K LAGRANGIAN

ROTATION ANGLE
5.0 X 36

LAGRANGIAN N-SEGMENTS

0.000	127	.255	.349	.345	.378	.411	.434	.458	.486
.513	.550	.605	.677	.771	.880	1.040	1.248	1.429	1.579
1.589	1.600	1.615	1.652	1.693	1.726	1.765	1.801	1.837	1.878
1.907	1.920	1.932	1.965	2.004	2.049	2.091	2.125	2.165	2.200
2.363	2.490	2.615	2.739	2.873	3.056	3.308	3.603	3.804	4.100
4.636	5.509	6.123	6.916	6.837	7.081	7.254	7.425	7.593	7.750
7.974	8.085	8.212							

EULERIAN Z-LINES

8.212	8.348	7.084	7.719	7.955	7.391	7.227	7.062	6.898	6.734
6.570	6.405	6.241	6.77	5.913	5.748	5.584	5.420	5.250	5.091
4.927	4.763	4.599	4.434	4.270	4.106	3.942	3.778	3.613	3.449
3.285	3.121	2.956	2.792	2.628	2.464	2.299	2.135	1.971	1.807
1.642	1.478	1.314	1.150	.985	.821	.657	.493	.328	.164
.000	-.164	-.328	-.493	-.657	-.821	-.985	-1.150	-1.314	-1.478
-1.642	-1.807	-1.971	-2.135	-2.299	-2.464	-2.628	-2.792	-2.956	-3.121
-3.285	-3.449	-3.613	-3.778	-3.942	-4.106	-4.270	-4.434	-4.599	-4.763
-4.927	-5.091	-5.256	-5.420	-5.584	-5.748	-5.913	-6.077	-6.241	-6.405
-6.570	-6.734	-6.898	-7.062	-7.227	-7.391	-7.555	-7.719	-7.884	-8.048
-8.212									

EULERIAN R-LINES

0.000	.382	.164	.246	.328	.411	.493	.575	.657	.739
.821	.903	.985	1. 68	1.153	1.232	1.314	1.396	1.478	1.560
1.642	1.725	1.807	1.889	1.971	2.053	2.135	2.217	2.299	2.381
2.464	2.546	2.628	2.710	2.792	2.874	2.956	3.038	3.121	3.203
3.285	3.367	3.449	3.531	3.613	3.695	3.778	3.860	3.942	4.024
4.106	4.188	4.270	4.352	4.434	4.517	4.599	4.681	4.763	4.845
4.927	5.009	5.091	5.174	5.256	5.338	5.421	5.502	5.584	5.666
5.748	5.831	5.913	5.995	6.077	6.159	6.241	6.323	6.405	6.487
6.570	6.652	6.734	6.816	6.898	6.980	7.062	7.144	7.227	7.309
7.391	7.473	7.555	7.637	7.719	7.801	7.884	7.966	8.048	8.130
8.212									

2-SEGMENTS 37 H-SEGMENTS 63 Z-LINES 101 H-LINES 101

TIME START 6.353

TIME END 26.773

TOTAL NUMBER OF INTERSECTION VOLUMES AND INDICES 19390

TEST PROBLEM 2-TIMING AND COUNT FOR 2.5K EULERIAN AND 2.2K LAGRANGIAN

ROTATION ANGLE

20.0 X 9

LAGRANGIAN H-SEGMENTS

0.000	.127	.255	.382	.509	.637	.765	.893	.920	.486
.513	.550	.605	.677	.771	.880	1.040	1.248	1.424	1.579
1.588	1.608	1.615	1.652	1.693	1.726	1.765	1.801	1.831	1.868
1.907	1.920	1.932	1.965	2.004	2.049	2.091	2.125	2.165	2.250
2.363	2.496	2.615	2.739	2.873	3.056	3.308	3.603	3.864	4.100
4.636	5.509	6.123	6.518	6.837	7.081	7.254	7.425	7.593	7.759
7.924	8.085	8.212							

EULERIAN Z-LINES

8.212	7.884	7.555	7.227	6.898	6.570	6.241	5.913	5.584	5.256
4.927	4.599	4.271	3.942	3.613	3.285	2.956	2.628	2.299	1.971
1.642	1.314	.985	.657	.328	.000	-.329	-.657	-.985	-1.314
-1.642	-1.971	-2.299	-2.628	-2.956	-3.285	-3.613	-3.942	-4.271	-4.599
-4.927	-5.256	-5.584	-5.913	-6.241	-6.570	-6.898	-7.227	-7.555	-7.884
-8.212									

EULERIAN R-LINES

0.000	.164	.328	.493	.657	.821	.985	1.150	1.314	1.478
1.642	1.807	1.971	2.135	2.299	2.464	2.628	2.792	2.956	3.121
3.285	3.449	3.613	3.778	3.942	4.106	4.270	4.434	4.599	4.763
4.927	5.091	5.256	5.420	5.584	5.748	5.913	6.077	6.241	6.405
6.570	6.734	6.898	7. 62	7.227	7.391	7.555	7.719	7.884	8.048
8.212									

Z-SEGMENTS 10 R-SEGMENTS 6 J Z-LINES 51 R-LINES 51

TIME START 26.377

TIME END 33.598

TOTAL NUMBER OF INTERSECTION VOLUMES AND INDICES 5856

••TEST PROBLEM 3-INTERSECTION VOLUME COMPIIATION FOR SIMPLE GRIDS•••

ROTATION ANGLE

90.0 X 2

LAGRANGIAN R-SEGMENTS

0.000 1.000 2.000

EULERIAN Z-LINES

2.000 1.000 0.000 -1.000 -2.000

EULERIAN R-LINES

0.000 1.000 2.000

Z-SEGMENTS 3 R-SEGMENTS 3 Z-LINES 5 R-LINES 3

TIME START 33.685

TIME END 33.692

TOTAL NUMBER OF INTERSECTION VOLUMES AND INDICES 8

INTERSECT VOL	LG	INDEX	EU	INDEX	LAGRANGIAN VOL	EULERIAN VOL
.10472000E+01		1		5	.10472000E+01	.31416000E+01
.10472000E+01		2		7	.73304000E+01	.31416000E+01
.20944000E+01		2		5	.73304000E+01	.31416000E+01
.41888000E+01		2		6	.73304000E+01	.94248000E+01
.10472000E+01		3		3	.10472000E+01	.31416000E+01
.20944000E+01		4		3	.73304000E+01	.31416000E+01
.41888000E+01		4		4	.73304000E+01	.94248000E+01
.10472000E+01		4		1	.73304000E+01	.31416000E+01

APPENDIX B
ESECT/EMAP LISTING

```

SUBROUTINE ESECT(RE,ZE,RL,ZL,NIZ,NIR,IZR,ILR)
C
C   BRUCE R WIENKE 11/15/75
C   EULERIAN/LAGRANGIAN MESH MAPPING
C
C   SUBROUTINE ESECT PROVIDES THE ALGORITHMS FOR SWEEPING THROUGH
C   MESHES, COMPUTING INTERSECTION COORDINATES AND LINE SLOPES,
C   FORMING INTERSECTION MULTILATERALS, ORDERING VERTICES,
C   TRIANGULATING MULTILATERALS AND COMPUTING INTERSECTION VOLUMES.
C   SUBROUTINE EMAP IS CALLED FROM SUBROUTINE ESECT AND PASSES THE
C   MAPPING PARAMETERS LR, LZ, HLR AND MLZ THROUGH COMMON,
C
COMMON RI(350),ZI(350),RLN(101),ZLN(101)
COMMON LR(350),LZ(350),MLR(350,4),MLZ(350,4)
COMMON VV(25000),LLM(25000),IIE(25000),INDXL(37)
COMMON EV(10000),GV(3600)
DIMENSION RE(101),ZE(101),RL(37,101),ZL(37,101)
DIMENSION RRI(350),ZZI(350),MRI(350),MZ(350),MRR(350),MZZ(350)
DIMENSION SLI(350),SOI(350),NL(350),NO(350),KE(10)
DIMENSION SL(350),SO(350),KD(350)
C
C   LAGRANGIAN R AND Z INTERVAL NUMBERS, ILRM AND IZRM,
C   ILRM=ILR=1 $ IZRM=IZR=1
C
C   EULERIAN R AND Z INTERVAL NUMBERS,IRM AND IZM, R AND Z LINES,RLN
C   AND ZLN,
C   IZ=NIZ $ IR=NIR
C   IZM=IZ=1 $ IRM=IR=1
C   IC=0
C   DO 520 I=1,NIZ
C   IC=IC+1 $ II=I
520  ZLN(IC)=ZE(II)
C   IC=0
C   DO 521 J=1,NIR
C   IC=IC+1
521  RLN(IC)=RE(J)
C
C   EDIT MESHES, INITIALIZE SCRATCH ARRAYS RI,ZI,SL,SO,LR,LZ,KD,MLZ,MLR
C   ARRAY LIMIT PARAMETERS IA,JA AND INFINITE SLOPE FIN
C   KC=0 $ IA=350 $ JA=25000 $ FIN=1.00E+20
C   IDZ=IZ/3 $ IDZP=IDZ+2 $ IDR=IR/3 $ IDRP=IDR+2
C   DO 601 L=1,IZR
C   DO 601 M=1,ILR
C   NZ=1 $ IF (ZL(L,M).LT.ZLN(IDZ)) NZ=IDZ
C   IF (ZL(L,M).LT.ZLN(IDZP)) NZ=IDZP
C   DO 611 I=NZ,IZ
C   Z=ZLN(I)-ZL(L,M) $ Z=ABS(Z)
C   IF(Z.LE.1.00E-05) GO TO 613
C   IF (ZLN(I).LT.ZL(L,M)) GO TO 610
C   GO TO 611

```

```

613 ZL(L,M)=ZLN(I)
    GO TO 610
611 CONTINUE
610 CONTINUE
    NR=1 $ IF (RL(L,M).GT,RLN(IDR)) NR=IDR
    IF (RL(L,M).GT,RLN(IDRP)) NR=IDRP
    DO 616 J=NR,IR
    R=RLN(J)-RL(L,M) $ R=ABS(R)
    IF (R.LE,1.00E-05) GO TO 617
    IF (RLN(J).GT,RL(L,M)) GO TO 615
    GO TO 616
617 RL(L,M)=RLN(J)
    GO TO 613
616 CONTINUE
615 CONTINUE
601 CONTINUE
    DO 510 L=1,IA
    RI(L)=0.0 $ ZI(L)=0.0
    SL(L)=0.0 $ SO(L)=0.0
    LR(L)=0 $ LZ(L)=0
    KD(L)=0
    DO 511 J=1,4
    MLZ(L,J)=0
    MLR(L,J)=0
511 CONTINUE
510 CONTINUE

```

```

C
C   SHEEP LAGRANGIAN MESH, COMPUTE INTERSECTIONS WITH EULERIAN MESH,
C   STORE INTERSECTIONS IN RI AND ZI ARRAYS, MAP RI AND ZI ON EULERIAN
C   MESH AND STORE MAPPING PARAMETERS IN ARRAYS LR,LZ,MLR,AND MLZ,
C   MAPPING VIA SUBROUTINE EMAP, LARGEST POSSIBLE INTERSECTION ZONE
C   ON EULERIAN GRID SPECIFIED BY VALUES OF MIZM,MXZP,MIRM,MXRP,
C
C   STORE VERTICES OF LAGRANGIAN CELL IN ZI AND RI ARRAYS, COMPUTE
C   SLOPES OF LINES INTERSECTING AT VERTICES AND STORE IN SL AND SO
C   ARRAYS,
DO J2 L=1,IZRM
INDXL(L)=KC+1
DO J0 M=1,ILRM
LM=M+(L-1)*ILRM
RI(1)=RL(L,M) $ ZI(1)=ZL(L,M)
RI(2)=RL(L,M+1) $ ZI(2)=ZL(L,M+1)
RI(3)=RL(L+1,M+1) $ ZI(3)=ZL(L+1,M+1) $ K=3
GVV=(RI(1)+ZI(2)-ZI(3))+RI(2)+ZI(3)-ZI(1)+RI(3)+ZI(1)-ZI(2))
1,5
GVV=ABS(GVV)*(RI(1)+RI(2)+RI(3))/3.
GV(LM)=GVV
IF (RL(L+1,M).EQ,RT(1).AND,ZL(L+1,M).EQ,ZI(1)) GO TO 11
RI(4)=RL(L+1,M) $ ZI(4)=ZL(L+1,M) $ K=4
QVV=(RI(1)+ZI(3)-ZI(4))+RI(3)+ZI(4)-ZI(1)+RI(4)+ZI(1)-ZI(3))
1,5
QVV=ABS(QVV)*(RI(1)+RI(3)+RI(4))/3.
GV(LM)=GVV+QVV
11 CONTINUE
GV(LM)=GV(LM)*6.2032
N=K $ NB=N+1

```



```

      DO 20 KK=1,K
      KQ=KK+1 & IF(KK,EO,K) KQ=1
      RN=RI(KK)-RI(KQ) & ZO=ZI(KK)+ZI(KQ)
      IF(ZO,EO,.,.) GO TO 13
      SL(KK)=RN/ZO & SO(KQ)=RN/ZO
      GO TO 14
13  CONTINUE
      SL(KK)=FIN & SO(KQ)=FIN
14  CONTINUE
      CALL EMAP (KK,1,I2M,1,IRM)
20  CONTINUE
      IF (K,EO,3) GO TO 21
      MXR=MAXR(LR(1),LR(2),LR(3),LR(4))
      MJR=MJNR(LR(1),LR(2),LR(3),LR(4))
      MXZ=MAXZ(LZ(1),LZ(2),LZ(3),LZ(4))
      MJZ=MJNZ(LZ(1),LZ(2),LZ(3),LZ(4))
      GO TO 22
21  MXR=MAXR(LR(1),LR(2),LR(3))
      MJR=MJNR(LR(1),LR(2),LR(3))
      MXZ=MAXZ(LZ(1),LZ(2),LZ(3))
      MJZ=MJNZ(LZ(1),LZ(2),LZ(3))
22  CONTINUE
      MXRP=MXR & MJRM=MJR
      MXZP=MXZ & MJZM=MJZ
      IF (MXZ,NE,MJZ,OR,MXR,NE,MJR) GO TO 35
      NN=MXR & NZ=MXZ
50  KKF=K+2
      A=0,00
      DO 40 KK=1,KKF
      K2=KK+1 & K3=KK+2
      B=(RI(1)+RI(K2)+RI(K3))/3.
      AT=.5*(RI(1)+(ZI(K2)-ZI(K3))+RI(K2)+(ZI(K3)-ZI(1))+RI(K3)+(ZI(1)-Z
      I(K2)))
40  A=A+B*ABS(AT)
      IJ=LR+(NZ-1)*IRM
      IE=NR+(I2N-NZ)*IRM
      NC=KC+1 & IF(KC,GT,JA) GO TO 64
      EV(IJ)=(RLN(NR+1)+2*RLN(NR)+2)*(ZLN(NZ)+ZLN(NZ+1)).5
      EV(IE)=EV(IE)+6,2032 & A=A+6,2032
      VF=A
      VV(KC)=A & LLN(KC)=LN & IIE(KC)=IE
      GO TO 10
35  CONTINUE
      IT=0
C
C      FIND INTERSECTIONS OF SIDES OF LAGRANGIAN CELL WITH EULERIAN MESH,
C      STORE INTERSECTIONS IN ZI AND RI ARRAYS AND SLOPES IN SL AND SO
C      ARRAYS.
      DO 60 KK=1,K
      KQ=KK+1 & IF(KK,EO,K) KQ=1
      IF(SL(KK),EQ,0.) GO TO 62
      IF(SL(KK),EQ,FIN) GO TO 64
      RI=RI(KK)-SL(KK)*ZI(KK)
      DO 07 I=NB,N
      NL(I)=0
07  NO(I)=0
      DO 01 I=I2M,MXZP
      IF (ZI(KQ),LT,ZLN(I),AND,ZI(KK),GT,ZLN(I)) GO TO 02
      IF (ZI(KQ),GT,ZLN(I),AND,ZI(KK),LT,ZLN(I)) GO TO 02
      IF (ZI(KQ),GT,ZLN(I),AND,ZI(KK),GT,ZLN(I)) GO TO 70

```

```

      GO TO A1
A2 DO A3 J=1RM,MRMP
   IF (R1(KQ),GT,RLN(J),AND,R1(KK),LT,RLN(J)) GO TO B4
   IF (R1(KQ),LT,RLN(J),AND,R1(KK),GT,RLN(J)) GO TO B4
   IF (R1(KQ),LT,RLN(J),AND,R1(KK),LT,RLN(J)) GO TO 79
   GO TO A3
B4 RC=RLN(J)
   ZC=ZLN(I)
   GC=RC-ZC*SL(KK)=B1
   GC=ABS(GC)
   IF (GC,LE,1,QUE=BS) GO TO B6
   GO TO B3
B6 N=N+1
   R1(N)=RLN(J) & Z1(N)=ZLN(I)
   SL(N)=SL(KK) & SO(N)=B,0
   NL(N)=I & NO(N)=J
   IT=IT+1
   IF (IT,EQ,2,OR,IT,FO,4,OR,IT,EQ,6,OR,IT,EQ,8) SO(N)=FIN
   CALL EMAP (N,MIZM,MXZP,MIRM,MXRP)
B3 CONTINUE
B1 CONTINUE
79 CONTINUE
   DO 75 J=1RM,MRMP
   IF (R1(KQ),GT,RLN(J),AND,R1(KK),LT,RLN(J))GO TO B8
   IF (R1(KQ),LT,RLN(J),AND,R1(KK),GT,RLN(J))GO TO B8
   IF (RLN(J),GT,R1(KQ),AND,RLN(J),GT,R1(KK))GO TO 75
   GO TO 75
B8 RC=RLN(J)
   ZC=(RC-B1)/SL(KK)
   DO 94 J=NB,N
   IF (NO(JN),EQ,J) GO TO 75
   IF (RC,EQ,R1(JN),AND,ZC,EQ,Z1(JN)) GO TO 75
   N=N+1
   R1(N)=RLN(J)
   Z1(N)=(R1(N)-B1)/SL(KK)
   SL(N)=SL(KK) & SO(N)=B,0
   CALL EMAP (N,MIZM,MXZP,MIRM,MXRP)
75 CONTINUE
B5 CONTINUE
   DO 95 I=MIZM,MXZP
   IF (Z1(KQ),LT,ZLN(I),AND,Z1(KK),GT,ZLN(I))GO TO 98
   IF (Z1(KQ),GT,ZLN(I),AND,Z1(KK),LT,ZLN(I))GO TO 98
   IF (ZLN(I),LT,Z1(KQ),AND,ZLN(I),LT,Z1(KK))GO TO 68
   GO TO 95
98 RC=SL(KK)*ZLN(I)+B1
   ZC=ZLN(I)
   DO 92 J=NB,N
   IF (NL(JN),EQ,I) GO TO 95
92 IF (RC,EQ,R1(JN),AND,ZC,EQ,Z1(JN)) GO TO 95
   N=N+1
   Z1(N)=ZLN(I)
   R1(N)=SL(KK)*Z1(N)+B1
   SL(N)=SL(KK) & SO(N)=FIN
   CALL EMAP (N,MIZM,MXZP,MIRM,MXRP)
95 CONTINUE
   GO TO 60
62 DO 100 I=MIZM,MXZP
   IF (Z1(KQ),LT,ZLN(I),AND,Z1(KK),GT,ZLN(I))GO TO 105
   IF (Z1(KQ),GT,ZLN(I),AND,Z1(KK),LT,ZLN(I))GO TO 105
   IF (ZLN(I),LT,Z1(KQ),AND,ZLN(I),LT,Z1(KK))GO TO 68

```

```

      GO TO 140
105 N=N+1
      ZI(N)=ZLN(I)
      RI(N)=RI(KK)
      SL(N)=P,N $ SO(N)=FIN
      CALL EMAP (N,MIZM,MXZP,MIRM,MXRP)
110 CONTINUE
      GO TO 67
115 DO 120 J=MIRM,MXRP
      IF (RI(KQ),GT,RLN(J),AND,RI(KK),LT,RLN(J))GO TO 125
      IF (RI(KQ),LT,RLN(J),AND,RI(KK),GT,RLN(J))GO TO 125
      IF (RLN(J),GT,RI(KQ),AND,RLN(J),GT,RI(KK))GO TO 68
      GO TO 120
125 N=N+1
      RI(N)=RLN(J)
      ZI(N)=ZI(KK)
      SL(N)=SL(KK) $ SO(N)=0,0
      CALL EMAP (N,MIZM,MXZP,MIRM,MXRP)
120 CONTINUE
130 CONTINUE
      IF (N,GT,IA) GO TO 66
      GO TO 67
C
C      FAULT= ARRAY OVERFLOW, PRINT ARRAY DIMENSIONS,LAGRANGIAN AND
C      INTERSECTION COORDINATES, TERMINATE.
140 WRITE (6,204)
      WRITE (6,205)
      WRITE (6,217)
      WRITE (6,215)
      WRITE (6,201) NNR,NNZ
      WRITE (6,201) L,M,N
      WRITE (6,202) ((ZI(LA),RI(LA),SL(LA),SO(LA),LZ(LA),MLZ(LA,1
1),MLR(LA,1),MLZ(LA,2),MLR(LA,2),MLZ(LA,3),MLR(LA,3),MLZ(LA,4),MLR(
2LA,4) ),LA=1,N)
      STOP
147 CONTINUE
      IF (MXR,EQ,MIR) GO TO 340
      IF (MXZ,EQ,MIZ) GO TO 340
C
C      FIND INTERSECTIONS OF ORTHOGONAL EULERIAN LINES INSIDE LAGRANGIAN
C      CELL, STORE INTERSECTIONS IN ZI AND RI ARRAYS AND SLOPES OF ZERO
C      OR FIN IN SL AND SO ARRAYS.
      NB=K+1 $ NE=N $ NNR=0 $ NNZ=0 $ NQ=N-1
      DO 300 NN=1,NQ
      LB=NN+1 $ IF (NN,LE,K) LB=K+1
      DO 301 LN=LB,NE
      IF (RI(NN),EQ,RI(LN)) GO TO 305
1506 IF (ZI(NN),EQ,ZI(LN)) GO TO 307
      GO TO 301
305 NNR=NNR+1
      RI(NNR)=RI(LN) $ MRR(NNR)=NN $ MR(NNR)=LN
      GO TO 306
307 NNZ=NNZ+1
      ZI(NNZ)=ZI(LN) $ MZZ(NNZ)=NN $ MZ(NNZ)=LN
301 CONTINUE
300 CONTINUE
      IF (NNR,GT,IA,OR,NNZ,GT,IA) GO TO 66
      IF (NNR,EQ,0,OR,NNZ,EQ,0) GO TO 340
      DO 320 I=1,NNZ
      IMZ=MZ(I) $ IMZZ=MZZ(I)

```

```

DO 321 JJ=1,NNR
JMR=MR(JJ) & JMRR=MR(JJ)
IF (ZI(JMR),GT,ZZI(11),AND,ZI(JMRR),LT,ZZI(11)) GO TO 325
IF (ZI(JMR),LT,ZZI(11),AND,ZI(JMRR),GT,ZZI(11)) GO TO 325
GO TO 321
325 IF (RI(IMZ),GT,RRJ(JJ),AND,RI(IMZZ),LT,RRJ(JJ)) GO TO 328
IF (RI(IMZ),LT,RRJ(JJ),AND,RI(IMZZ),GT,RRJ(JJ)) GO TO 328
GO TO 321
328 CONTINUE
DO 308 MC=MIRN,MXRP
IF (RLN(MC),EQ,RRJ(JJ)) GO TO 309
IF (RLN(MC),GT,RRJ(JJ)) GO TO 321
308 CONTINUE
GO TO 321
309 DO 310 MC=MIZM,MXZP
IF (ZLN(MC),EQ,ZZI(11)) GO TO 312
IF (ZLN(MC),LT,ZZI(11)) GO TO 321
310 CONTINUE
GO TO 321
312 CONTINUE
N=N+1
RI(N)=RRJ(JJ)
ZI(N)=ZZI(11)
SL(N)=0.0 & SO(N)=FIN
CALL EMAP (N,MIZM,MXZP,MIRN,MXRP)
321 CONTINUE
328 CONTINUE
IF (N,GT,1A) GO TO 66
348 CONTINUE
IF (N,NE,K) GO TO 9
IF (K,EQ,4) GO TO 138
LR(4)=0 & LZ(4)=0
138 CONTINUE
NR=MAX0(LR(1),LR(2),LR(3),LR(4))
NZ=MAX0(LZ(1),LZ(2),LZ(3),LZ(4))
GO TO 58
9 CONTINUE

```

```

C
C   CONSTRUCT INTERSECTION MULTILATERALS FROM COMMON MAPPING INDICES
C   MLR AND MLZ AND STORE COORDINATES IN SUB-ARRAYS RRI AND ZZI AND
C   SLOPES IN SUB-ARRAYS SLI AND SOI.
DO 400 II=MIZ,MXZ
DO 401 JJ=MIR,MXR
A=0.68 & NNN=0
DO 402 KK=1,N
DO 403 KKK=1,4
IF (MLZ(KK,KKK),EQ,II,AND,MLR(KK,KKK),EQ,JJ) GO TO 485
483 CONTINUE
GO TO 482
485 NNN=NNN+1
RRI(NNN)=RI(KK) & ZZI(NNN)=ZI(KK)
SLI(NNN)=SL(KK) & SOI(NNN)=SO(KK)
IF (NNN,GT,1A) GO TO 455
482 CONTINUE
IF (NNN,LE,2) GO TO 481
IF (NNN,EQ,3) GO TO 418
DO 475 KA=1,NNN
KE(KA)=0
475 KO(KA)=0

```

```

C      REORDER VERTICES OF INTERSECTION MULTILATERAL, TRIANGULATE FIGURE
C      AND COMPUTE INTERSECTION VOLUME BY SUMMING TRIANGULAR COMPONENTS.
C      INTERSECTION VOLUME IS VF, EULERIAN INDEX IS IE, LAGRANGIAN INDEX
C      IS LM, EULERIAN CELL VOLUME IS EV, LAGRANGIAN CELL VOLUME IS GV,
      AH=RI(1), AZ=ZZI(1) $ AL=SLI(1) $ KD(!)=1 $ AO=SOI(1)
      KE(!)=1
      DO 430 LLL=1, NNN
      DO 431 LLL=2, NNN
      IF (KD(LLL), NE, 0) GO TO 431
      IF (LL, EQ, NNN) GO TO 450
      IF (RRI(LLL), EQ, AH) GO TO 461
      IF (ZZI(LLL), EQ, AZ) GO TO 462
431  CONTINUE
      ITS=0
435  DO 432 LLL=1, NNN
      IF (KD(LLL), NE, 0) GO TO 432
      BR=AR-AZ*AL
      IF (SLI(LLL), EQ, AL) GO TO 451
      IF (SOI(LLL), EQ, AL) GO TO 452
      GO TO 432
451  TR=RRI(LLL)-ZZI(LLL)*SLI(LLL)
      ZR=TR-BR $ ZR=ABS(ZR)
      IF (ZR, LE, 1, 00E=05) GO TO 440
      GO TO 432
452  TR=RRI(LLL)-ZZI(LLL)*SOI(LLL)
      ZR=TR-BR $ ZR=ABS(ZR)
      IF (ZR, LE, 1, 00E=05) GO TO 441
432  CONTINUE
      ITS=ITS+1 $ IF (ITS, GT, 1) GO TO 455
      XAR=AL $ XAO=AO $ AL=XAO $ AO=XAR $ GO TO 435
461  IF (SLI(LLL), EQ, 0, 0) GO TO 440
      IF (SOI(LLL), EQ, 0, 0) GO TO 441
      GO TO 440
462  IF (SLI(LLL), EQ, FIN) GO TO 440
      IF (SOI(LLL), EQ, FIN) GO TO 441
440  AL=SOI(LLL)
      AO=SLI(LLL)
      GO TO 450
441  AL=SLI(LLL)
      AO=SOI(LLL)
450  LT=LLL
      KD(LLL)=LL
      KE(LL)=LLL
      GO TO 460
455  CONTINUE
C      FAULTY= MAPPING OR COMPUTATION FAILURE, PRINT ARRAY DIMENSIONS,
C      LAGRANGIAN AND INTERSECTION COORDINATES, MAPPING INDICES, ORDERING
C      OF VERTICES, TERMINATE.
      IJ=JJ+(II-1)*IRH
      WRITE (6, 204)
      WRITE (6, 206)
      WRITE (6, 217)
      WRITE (6, 215)
      WRITE (6, 201) L, M, N
      WRITE (6, 202) ((ZI(LA), RI(LA), SL(LA), SO(LA), LZ(LA), LR(LA), MLZ(LA, 1
1), MLR(LA, 1), MLZ(LA, 2), MLR(LA, 2), MLZ(LA, 3), MLR(LA, 3), MLZ(LA, 4), MLR(
2LA, 4) ), LA=1, N)
      WRITE (6, 203) LM, IJ, N
      WRITE (6, 217)

```

```

WRITE(6,216)
WRITE (6,201) L,M,II,JJ,NNM
WRITE (6,200) ((ZZI(LC),RRI(LC),SLI(LC),SOI(LC)),LC=1,NNN)
WRITE (6,201) (KD(IJ),IJ=1,NNN)
WRITE (6,207) KC
STOP
460 CONTINUE
AR=RRI(LT) $ AZ=ZZI(LT)
430 CONTINUE
KKF=NNN-2
DO 420 KK=1,KKF
K2=KE(KK+1) $ K3=KF(KK+2)
B=(RRI(1)+RRI(K2)+RRI(K3))/3.
AT=.5*(RRI(1)*(ZZI(K2)-ZZI(K3))+RRI(K2)*(ZZI(K3)-ZZI(1))+RRI(K3)*(
ZZI(1)-ZZI(K2)))
420 A=A+B*ABS(AT)
GO TO 437
410 B=(RRI(1)+RRI(2)+RRI(3))/3.
AT=.5*(RRI(1)*(ZZI(2)-ZZI(3))+RRI(2)*(ZZI(3)-ZZI(1))+RRI(3)*(ZZI(1
)-ZZI(2)))
A=A+B*ABS(AT)
437 CONTINUE
IJ=JJ+(II-1)*IRM
IE=JJ+(I2M-II)*IRM
KC=KC+1 $ IF(KC,GT,JA) GO TO 455
EV(IE)=(RLN(JJ+1)**2-RLN(JJ)**2)*(ZLN(II)-ZLN(II+1))*5
EV(IE)=EV(IE)*6,2032 $ A=A*6,2032
VF=A
VV(KC)=A $ LLM(KC)=LM $ IIE(KC)=IE
401 CONTINUE
400 CONTINUE
10 CONTINUE
12 CONTINUE
INDXL(I2RM+1)=KC+1
NINSGT=INDXL(I2RM+1)-1
200 FORMAT(1X,4(1X,E14.8))
201 FORMAT (1X,10(1X,I4))
202 FORMAT (1X,4(1X,E14.8),10(1X,I3))
204 FORMAT(1X,30HARRAY OR MAPPING FAULT, JOB TERMINATION/)
205 FORMAT (1X,43H NNR,NNZ L,M,N ZI,RI,SL,SO,LZ,LR,MLZ,MLR KC)
206 FORMAT(1X,76H L,M,N ZI,RI,SL,SO,LZ,LR,MLZ,MLR LM,IJ,N L,M,II,JJ,NH
IN ZZI,RRI,SLI,SOI KD KC)
207 FORMAT (1X,IA)
208 FORMAT (10F8.3)
215 FORMAT(1X,30HINTERSECTION ARRAY, MAPPING INDICES/)
216 FORMAT(1X,30HMULTILATERAL VERTICES, ORDERING/)
217 FORMAT(/)
RETURN
END
SUBROUTINE EMAP (M,NZM,NZP,NRM,NRP)

```

```

C
C BRUCE R WIENKE 11/15/75
C EULERIAN/LAGRANGIAN MESH MAPPING
C

```

```

C SUBROUTINE EMAP IS CALLED FROM SUBROUTINE FSECT AND FURNISHES THE
C POINTWISE MAPPING ROUTINE FOR LAGRANGIAN CELL VERTICES AND ALL
C INTERSECTIONS OF THE SIDES OF THE LAGRANGIAN WITH THE EULERIAN
C GRID LINES. MAPPING INDICES ARE CONTINUOUS OVER THE FULL POSSIBLE
C RANGE OF EULERIAN CELLS AND BEGIN WITH THE FIRST EULERIAN CELL ON
C THE BOTTOM RIGHT HAND CORNER OF THE MESH AND WORK UPWARD, MAPPING

```

```

C      ARRAYS FOR THE INDICES ARE LR,LZ,MLR,MLZ,
C
      COMMON RI(350),ZI(350),RLN(101),ZLN(101)
      COMMON LR(350),LZ(350),MLR(350,4),MLZ(350,4)
      LCM VV(25000),LLM(25000),IIE(25000),INDXL(37)
      LCM EV(10000),GV(3600)
C
C      LR AND LZ REPRESENT COARSE POINT MAPPINGS,
      MZ=NZP+1 & MR=MRP+1
      DO 100 J=NRM,NRP
      IF(RI(M),LE,RLN(J+1),AND,RI(M),GE,RLN(J))GO TO 105
      GO TO 100
105  LR(M)=J
      GO TO 106
100  CONTINUE
106  CONTINUE
      DO 200 I=NZM,NZP
      IF(ZI(M),GE,ZLN(I+1),AND,ZI(M),LE,ZLN(I))GO TO 205
      GO TO 200
205  LZ(M)=I
      GO TO 206
200  CONTINUE
206  CONTINUE
C
C      MLR AND MLZ REPRESENT FINE POINTWISE MAPPINGS,
      MLZ(M,1)=LZ(M) & MLR(M,1)=LR(M)
      KZ=0
      DO 300 I=NZM,MZ
      IF (ZI(M),EQ,ZLN(1)) KZ=I
      IF (KZ,NE,0) GO TO 101
300  CONTINUE
101  CONTINUE
      KR=0
      DO 102 J=NRM,MR
      IF (RI(M),EQ,RLN(J)) KR=J
      IF (KR,NE,0) GO TO 103
102  CONTINUE
103  CONTINUE
      IF (KR,NE,0,AND,KZ,NE,0) GO TO 10
      IF (KR,NE,0) GO TO 11
      IF (KZ,NE,0) GO TO 12
15  DO 20 KK=2,4
      MLR(M,KK)=LR(M)
      MLZ(M,KK)=LZ(M)
20  CONTINUE
      GO TO 30
10  CONTINUE
      IF (KR,EQ,1,AND,KZ,EQ,1) GO TO 15
      IF (KR,EQ,1,AND,KZ,EQ,MZ) GO TO 15
      IF (KR,EQ,MR,AND,KZ,EQ,1) GO TO 15
      IF (KR,EQ,MR,AND,KZ,EQ,MZ) GO TO 15
      IF (KR,EQ,1,OR,KR,EQ,MR) GO TO 17
      IF (KZ,EQ,1,OR,KZ,EQ,MZ) GO TO 19
      MLR(M,2)=LR(M)+1 & MLZ(M,2)=LZ(M)
      MLR(M,3)=LR(M)+1 & MLZ(M,3)=LZ(M)+1
      MLR(M,4)=LR(M) & MLZ(M,4)=LZ(M)+1
      GO TO 30
17  CONTINUE
      MLR(M,2)=LR(M) & MLZ(M,2)=LZ(M)
      MLR(M,3)=LR(M) & MLZ(M,3)=LZ(M)+1

```

```

      MLR(M,4)=LR(M) & MLZ(M,4)=LZ(M)+1
      GO TO 10
10 CONTINUE
      MLR(M,2)=LR(M)+1 & MLZ(M,2)=LZ(M)
      MLR(M,3)=LR(M)+1 & MLZ(M,3)=LZ(M)
      MLR(M,4)=LR(M) & MLZ(M,4)=LZ(M)
      GO TO 30
11 CONTINUE
      IF (KR,EQ,1,OR,KR,EQ,MR) GO TO 15
      GO TO 19
12 CONTINUE
      IF (KZ,EQ,1,OR,KZ,EQ,MZ) GO TO 15
      GO TO 17
30 CONTINUE
      RETURN
      END

```

APPENDIX C TEST PROGRAM LISTING

```

PROGRAM ETEST (INP,OUT,FSET5=INP,FSET6=OUT)
C
C      BRUCE R WIENKE 11/15/75
C      EULERIAN/LAGRANGIAN MESH MAPPING
C
C      ETEST IS THE CALLING PROGRAM WHICH PROVIDES THE TEST PROBLEMS,SETS
C      UP THE EULERIAN AND LAGRANGIAN MESHES,TIMES THE MAPPING ROUTINE
C      AND PREPARES DATA. EULERIAN COARSE AND FINE MESH PARAMETERS ARE
C      READ AND THE EULERIAN R AND Z LINES ARE CONSTRUCTED AND STORED IN
C      THE ARRAYS REN AND ZEN. A CENTER CONVERGING LAGRANGIAN GRID IS
C      CONSTRUCTED FROM INPUT DATA AND STORED IN THE GRID ARRAYS RL AND
C      ZL. THE VARIABLES IR,IZ ARE THE NUMBERS OF EULERIAN R,Z LINES
C      AND IZR,ILR THE NUMBERS OF LAGRANGIAN R,Z SEGMENTS. THE ONLY
C      ESSENTIAL INFORMATION FOR THE MAPPING ROUTINES ESECT AND EMAP IS
C      PASSED THRU THE *CALL ESECT(REN,ZEN,RL,ZL,IZ,IR,IZR,ILR)*STATEMENT
C      USERS MAY EMPLOY THIS CALLING MODULE TO READ,CONSTRUCT OR SUPPLY
C      ARBITRARY EULERIAN AND LAGRANGIAN MESHES FOR THEIR USE, OR SIMPLY
C      EMPLOY THE FOLLOWING TWO SUBROUTINES (ESECT,EMAP) AS INTERFACES.
C      NOTE- ALL R COORDINATES POSITIVE AND ENTERED IN ASCENDING ORDER.
C      NOTE- ALL Z COORDINATES ENTERED IN DESCENDING ORDER.
C
      COMMON RI(350),ZI(350),RLN(101),ZLN(101)
      COMMON LR(350),LZ(350),MLR(350,4),MLZ(350,4)
      LCM VV(25000),LLM(25000),IIE(25000),INDXL(37)
      LCM EV(10000),GV(3600)
      DIMENSION REN(101),ZEN(101),RL(37,101),ZL(37,101)
      DIMENSION ZMESH(10),RMESH(10),OL(101),TIT(8)
C
      READ LABEL
2 READ (5,220)(TIT(K),K=1,8)
      IF(EOF,5) 1,3
3 WRITE(6,210)
      WRITE(6,220)(TIT(K),K=1,8)
      WRITE (6,203)
      WRITE (6,203)
C

```



```

C   READ ANGULAR INTERVAL NUMBER LCZ,RADIAL INTERVAL NUMBER LCR, ANGLE
C   OF ROTATION AN,RADIAL SEGMENTS BL AND CONSTRUCT DEMONSTRATION
C   LAGRANGIAN CENTER CONVERGING TEST GRID,
  READ (5,200) LCZ,LCR,AN,ANCT
  NCT=ANCT
  ILR=LCR+1 $ IZR=LCZ+1
  READ (5,206) (BL(J),J=1,ILR)
  DO 8 J=1,ILR
    AN=J-1 $ AG=AN*AN*3.1416/180.
    DO 8 J=1,ILR
      RL(I,J)=BL(J)*SIN(AG)
      ZL(I,J)=BL(J)*COS(AG)
      IF (I,EQ,1,OR,I,EQ,IZR) RL(I,J)=0.
      IF (I,EQ,1) ZL(I,J)=BL(J) $ IF (I,EQ,IZR) ZL(I,J)=BL(J)
8  CONTINUE
  WRITE(6,212)
  WRITE(6,213) AN,LCZ
  WRITE(6,211)
  WRITE(6,206) (BL(J),J=1,ILR)

C   READ NUMBERS OF EULERIAN COARSE Z AND R INTERVALS,NCZ AND NCR,READ
C   NUMBERS OF FINE MESH Z AND R INTERVALS,NFZ AND NCR AND CONSTRUCT
C   DEMONSTRATION EULERIAN RECTANGULAR TEST GRID,
  READ (5,205) NCZ,NCR,NFZ,NFR
  NCZB=NCZ+1 $ NCRB=NCR+1
  READ (5,206) (ZMESH(I),I=1,NCZB)
  READ (5,206) (RMESH(J),J=1,NCRB)
  IZ=NCZ*NFZ+1 $ IR=NCR*NFR+1
  ANFZ=NFZ $ ANFR=NFR
  IC=0
  DO 20 I=1,NCZ
    ZME=(ZMESH(I)+1)/ANFZ
    DO 20,II=1,NFZ
      IC=IC+1 $ AII=II-1
20  ZEN(IC)=ZMESH(I)+AII*ZME
      IC=IC+1 $ ZEN(IC)=ZMESH(NCZB)
      IC=IC+1 $ ZLN(IC)=ZMESH(NCZB)
      IC=0
      DO 21 J=1,NCR
        RME=(RMESH(J+1)-RMESH(J))/ANFR
        DO 21 JJ=1,NFR
          IC=IC+1 $ AJJ=JJ-1
21  REN(IC)=RMESH(J)+AJJ*RME
          IC=IC+1 $ REN(IC)=RMESH(NCRB)
          WRITE(6,210)
          WRITE(6,206) (ZEN(KY),KY=1,IZ)
          WRITE(6,209)
          WRITE(6,206) (REN(KY),KY=1,IR)
          WRITE (6,203)
          WRITE (6,210) IZR,ILR,IZ,IR
          JA=25000 $ JAE=10000 $ JAL=3600
          DO 700 IFT=1,JA
            VV(IFT)=0.0
            LLM(IFT)=0 $ IIE(IFT)=0
            IF(IFT,LE,JAE)EV(IFT)=0.0
            IF(IFT,LE,JAL)GV(IFT)=0.0
700 CONTINUE

```

```

C
C CALL MAPPING AND COMPUTATION ALGORITHMS *ESECT,EMAP* WITH EULERIAN
C R AND 2 LINES,REN AND ZEN,LAGRANGIAN GRID COORDINATES,RL AND ZL,
C AND TIME MAPPING AND INTERSECTION VOLUME CALCULATIONS,
CALL SECOND (T)
WRITE(6,207)T
CALL ESECT(REN,ZEN,RL,ZL,IZ,IR,IZR,ILR)
CALL SECOND (T)
WRITE(6,208)T
WRITE (6,203)
NINSCT=INDX(IZR)=1
WRITE(6,218)NINSCT
IF(NCT,EQ,0) GO TO 2
WRITE(6,226)
DO 240 K=1,NINSCT
LM=LLM(K) $ IE=IIE(K)
240 WRITE (6,225) VV(K),LM,IE,GV(LM),EV(IE)
GO TO 2
1 CONTINUE
200 FORMAT (2I2,2F5,1)
201 FORMAT (1X,4(1X,I3))
203 FORMAT (1X,1H )
205 FORMAT (6I3)
206 FORMAT (10F8,3)
207 FORMAT(/1X,10HTIME START,3X,F8,3/)
208 FORMAT(1X,8HTIME END,5X,F8,3/)
209 FORMAT(/1X,14HEULERIAN R=LINES)
210 FORMAT(/1X,16HEULERIAN Z=LINES)
211 FORMAT(/1X,21HLAGRANGIAN R=SEGMENTS)
212 FORMAT(/1X,10HROTATION ANGLE)
213 FORMAT(1X,F5,1,1X,1HX,1X,I2)
219 FORMAT(1H1)
218 FORMAT(/1X,40HTOTAL NUMBER OF INTERSECTION VOLUMES AND INDICES,10
1/)
220 FORMAT(8A10)
225 FORMAT(1X,E14,8,1X,2(I8,1X),2(E14,8,1X))
226 FORMAT(/1X,14H INTERSECT VOL,1X,8H LG INDEX,1X,8HEU INDEX1X,14HLAGR
ANGIAN VOL,3X,12HEULERIAN VOL/)
230 FORMAT(1X,10HZ=SEGMENTS,I4,1X,10HR=SEGMENTS,I4,1X,7HZ=LINES,I4,1X,
17HR=LINES,I4,1X//)
STOP $ END

```

APPENDIX D

TEST PROBLEM CARD INPUT

TEST PROBLEM 1-TIMING AND COUNT FOR 10K EULERIAN AND 2,2K LAGRANGIAN

```

3662 5,0
,000 ,127 ,255 ,300 ,345 ,378 ,410 ,434 ,458 ,486
,513 ,550 ,605 ,677 ,771 ,880 1,040 1,248 1,429 1,539
1,588 1,688 1,615 1,652 1,693 1,726 1,765 1,801 1,837 1,898
1,907 1,920 1,932 1,965 2,004 2,049 2,091 2,125 2,185 2,250
2,363 2,496 2,615 2,739 2,873 3,056 3,308 3,603 3,864 4,100
4,636 5,509 6,123 6,518 6,837 7,081 7,254 7,425 7,593 7,759
7,924 8,085 8,212
1 1100100
8,212 8,212
8,000 8,212

```

TEST PROBLEM 2=TIMING AND COUNT FOR 2.5K EULERIAN AND 2.2K LAGRANGIAN

962 20.0

.000	.127	.255	.300	.345	.378	.410	.434	.458	.486
.513	.550	.605	.677	.771	.880	1.040	1.248	1.429	1.539
1.588	1.688	1.615	1.652	1.693	1.726	1.765	1.801	1.837	1.898
1.907	1.920	1.932	1.965	2.004	2.049	2.091	2.125	2.185	2.250
2.363	2.496	2.615	2.739	2.873	3.056	3.300	3.603	3.864	4.100
4.636	5.509	6.123	6.518	6.837	7.081	7.254	7.425	7.593	7.759

1 1 50 50

0.212 0.212

0.00 0.212

TEST PROBLEM 3=INTERSECTION VOLUME COMPUTATION FOR SIMPLE GRIDS

2 2 90.0 10.0

.000 1.000 2.000

1 1 4 2

2.000 2.000

0.00 2.000