AD 744508

INSTITUTE FOR AEROSPACE STUDIES

UNIVERSITY OF TORONTO

AFOSR - TR - 72 - 1278

NEAR-FREE-MOLECULE FLOW CALCULATIONS-MONTE CARLO STUDY OF SKIMMER INTERACTIONS

by

N.A. Derzko

October, 1971.

UTIAS Technical Note No. 169

UNCLASSIFIED
Security Classification

PAT

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) UNIVERSITY OF TORONTO INSTITUTE FOR AEROSPACE STUDIES TORONTO 5, ONTARIO, CANADA	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP

3. REPORT TITLE

NEAR-FREE-MOLECULE FLOW CALCULATIONS-MONTE CARLO STUDY OF SKIMMER INTERACTIONS

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)
Scientific Interim

5. AUTHOR(S) (First name, middle initial, last name)

N A DERZKO

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
October 1971	11	8

8a. CONTRACT OR GRANT NO. AFOSR-1481-68 b. PROJECT NO. 9783-01 c. 61102F d. 681307	9a. ORIGINATOR'S REPORT NUMBER(S) UTIAS T.N. NO. 169 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) AFOSR - TR - 72 - 1278

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES TECH, OTHER	12. SPONSORING MILITARY ACTIVITY AF Office of Scientific Research (NAM) 1400 Wilson Boulevard Arlington, Virginia 22209

13. ABSTRACT

The Monte Carlo simulation method of G. A. Bird was applied to the study of flow through a skimmer using various geometries, speed ratios, and Knudsen numbers. The computer program followed a sample of 3500 hard sphere molecules, accumulating distribution function profiles in each of 25 regions of the flowfield, the beam profile at the downstream end of the flowfield, and the mass flow through the skimmer orifice. The results show beam intensities of the order of .3 of the no skimmer interaction value at Knudsen numbers of 01 and demonstrate clearly that most of the interaction occurs from 0 to 4 mean free paths upstream of the orifice.

DD FORM 1 NOV 65 1473

UNCLASSIFIED
Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
SKIMMER INTERACTION						
MONTE CARLO SIMULATION						
BOLTZMANN EQUATION						
MOLECULAR BEAM						

NEAR-FREE-MOLECULE FLOW CALCULATIONS-
MONTE CARLO STUDY OF SKIMMER INTERACTIONS

by

N. A. Derzko

Submitted March, 1971.

October, 1971. UTIAS Technical Note No. 169

1C

ACKNOWLEDGEMENTS

I am indebted to Dr. G. N. Patterson and to Dr. J. H. deLecuw for the part they have played in suggesting and assisting this project, and to J. Kesten for programming assistance in its early stages. This work was supported by the U. S. Air Force Office of Scientific Research under Grant No. AF-AFOSR-68-1481A.

SUMMARY

The Monte Carlo simulation method of G. A. Bird was applied to the study of flow through a skimmer using various geometries, speed ratios, and Knudsen numbers. The computer program followed a sample of 3500 hard sphere molecules, accumulating distribution function profiles in each of 25 regions of the flowfield, the beam profile at the downstream end of the flowfield, and the mass flow through the skimmer orifice.

The results show beam intensities of the order of .3 of the no skimmer interaction value at Knudsen numbers of 01 and demonstrate clearly that most of the interaction occurs from 0 to 4 mean free paths upstream of the orifice.

TABLE OF CONTENTS

INTRODUCTION

This report deals with part of the work we have been doing during the past three summers on the development and testing of a number of computer programs designed to simulate molecular flow problems in the transition regime. The simulation is done with hard sphere model using the Monte Carlo method developed by G. A. Bird (2).

We describe here its application to the study of the flow through a skimmer under conditions similar to those involved in the production of molecular beams of high intensity and speed ratio. We simulate the flow which occurs in a cylindrical region reaching upstream and downstream of the skimmer but not including the nozzle.

Since its proposal by Kantrowitz and Grey in 1951, the free jet-skimmer combination has proved far superior to the older methods using the effusive flow from an orifice in an oven source. Nevertheless, in practice the intensity of a beam produced in this way is substantially lower than that one would obtain if the skimmer just removed the portion of the free jet around the beam region - that is, did a perfect job. Considerable experimental work has gone into studying the nature and causes of this effect. Recently Govers, LeRoy and Deckers (6) have studied the effects of background scattering on skimmer interaction; Bossel, Hurlbut, and Sherman (3) have done experiments to study the magnitude of the skimmer interaction for various geometries and jets.

The effect of the skimmer interaction, whatever its nature, is quite clear If one plots beam intensity in particles per sec. per steradian vs nozzle skimmer separation, the graph starts at a low value for small separation, increases to a maximum and then drops rapidly as the separation exceeds the distance to the Mach disc (Fig. 4, p. 994, 6). More or less successful attempts have been made to correlate the large amount of experimental data available to enable one to describe a given situation with a fairly small number of parameters With skimmers of similar geometry, the ratio of mean free path to skimmer orifice diameter appears to be significant Roger Campargue (4) has discussed a number of dimensionless quantities relevant to the nozzle beam apparatus.

The theory associated with a nozzle beam divides naturally into three areas:

(a) Description of the free jet - that is the density, velocity, and speed ratio at every point downstream of the nozzle.

(b) Skimmer interaction.

(c) Development of the beam downstream of the skimmer (assuming that any molecules involved in skimmer interaction leave the beam).

A satisfactory theory for (a) was published by Ashkenas and Sherman in 1966 (1). A number of authors have also adequately studied (c), e.g. Zapata et al in 1960 (8). A summary appears in Hagena and Morton (7). Thus far (b) has eluded a satisfactory theoretical discussion. It is the purpose of this work to shed some light on this dark area in the theory of nozzle beams.

DESCRIPTION OF THE PROGRAM

§1. METHOD USED

The Monte Carlo method used in this study of skimmer interaction is basically the method of G. A. Bird which has been described by him in a number of publications, for example (2). We include a short description here for sake of completeness and to enable us to explain the changes we have introduced to accommodate the peculiarities of this problem.

Bird's method is a Monte Carlo Algorithm for constructing a statistical sample of given size N of the the molecular paths occurring in some finite regions of an actual flow. The construction process is an imitation of what occurs in nature, at least insofar as probability distributions are concerned. The particle paths are functions of time, so that at any instant only the N coordinates in phase space (position, velocity-space) appear in computer memory. A time counter T is also kept in memory.

During each iterate of the calculations T is advanced by a preset quantity DTM (actually a number DT approximately equal to DTM) and the coordinates in the molecule list are updated.

§2. INITIAL CONDITIONS

The initial sample is from a distribution based on what we would have if the skimmer absorbed every molecule which hit it. Actually, the initial distribution is of secondary importance since experience has shown that the molecule list approaches the correct steady state distribution in relatively few iterations.

As soon as T reaches a preset value TEQ (time at which steady flow is assumed to have been achieved) data collection for subsequent output is begun and continues until T reaches TEND. Following output the program automatically stops or does the next case.

§3. ONE ITERATION

Each iteration has two components:

(a) Changes to molecule list due to collision.

(b) Changes to molecule list due to convection.

The computation of (a) requires that the flow region be divided into cells (Fig. 2) in such a way that the distribution function remains fairly constant over each cell. At any moment the molecules residing in a given cell constitute a sample from the distribution for that cell. If these molecules are allowed to collide among themselves then these collisions are representatives of the collisions actually taking place in the given cell. The theory backing this procedure is stronger than the above suggests. We refer to Bird's original papers on the subject.

The details of the collision process are as follows: A pair of molecules is chosen in such a way that the probability of being chosen is proportional to the relative velocity. If $\bar{v}_a$ and $\bar{v}_b$ are the velocities of the pair chosen, they are replaced according to the hard sphere collision law with $\bar{v}_a'$, $\bar{v}_b'$, where

$$\bar{V}_a = 1/2[\bar{v}_a + \bar{v}_b - |\bar{v}_b - \bar{v}_a| \bar{n}]$$
$$\bar{V}_b = 1/2[\bar{v}_a + \bar{v}_b + |\bar{v}_a - \bar{v}_b| \bar{n}],$$

where $\bar{n}$ is chosen from the uniform distribution on the unit sphere.

The position coordinates remain unchanged and a time parameter DT is advanced by

$$DDT = \frac{TCONS}{n^2|\bar{v}_b - v_a|}.$$

The number TCONS is fixed for each cell and is explained in §6. Collisions are computed for a given cell until

$$DT + DDT > DTM.$$

The last collision is kept with probability

$$\frac{DTM - DT}{DDT}$$

and ignored otherwise. This last refinement is essential here because some cells have such low collision frequencies that no collisions at all should be computed during some intervals DTM.

The computation of (b) is basically simple. The position coordinates are translated on the basis of the time interval DTM and velocities. The computation involves some work because we use cylindrical coordinates in the program in which the translation equations are nonlinear. We have had to introduce two additional complications which do not appear necessary at first sight - referencing the molecule list §9 - and stock splitting §5. At the end of the collision convection cycle T is incremented appropriately and the whole process is repeated.

§4. GEOMETRY

Since the skimmer problem has cylindrical symmetry about the common axis of the free jet and the skimmer, two coordinates suffice to specify position - the axial and radial. They are denoted by (x,y) in the program (rather than (z,r) as is customary because essentially the same program with a few changed subroutines can be used in Cartesian coordinates). The velocity coordinates (u,v,w) are with respect to the Cartesian frame $\bar{i},\bar{j},\bar{k}$ where $\bar{i}$ is in the axial direction, $\bar{j}$ is in the radical direction and $\bar{k}$ is in the direction of increasing θ. The actual region of the flow under study is the solid of revolution generated by the shape in Fig. 1.

The flow region is subdivided into cells as required in the collision calculation. The 25 cells and the parameters necessary to specify the geometry are marked in Fig. 1. These parameters are kept in common area (GEOM) (see Appendix 1). Some of them must be specified initially (see Block Data subprogram) and the rest are computed in subroutine SETUP.

§5. STOCK SPLITTING

If the coordinates in computer memory simply constituted a sample of the molecular coordinates in an actual flow, we should find that their density in the x,y-plane increased linearly with x. This would be unsatisfactory because then most of the computation would be done for the region far from the axis while the statistical scatter for the beam region containing only a small number of molecules would render these results almost useless.

The solution we have adopted to this difficulty involves subdividing the flow region into a number of subregions, using the streamlines of the flow to construct the boundary surfaces whenever possible. The subregions are then assigned weights in such a way that a simulator molecule in a subregion far from the axis "represents" several times as many "real" molecules as one in an axial region. For programming reasons we have found it convenient to make the weights inversely proportional to the representation multiples, so that the subregions farthest from the axis have the lowest weights.

Such an arrangement calls for a special strategy when a molecule happens to move between subregions of different weights. The strategy is this: When a molecule moves from a weight w_1 region to a weight w_2 region, it is replaced with

$$\mathrm{INTR}(w_2/w_1)$$

identical molecules. The random variable INTR is defined as follows:

$$\mathrm{INTR}(\alpha) = \begin{cases} [\alpha] & \text{with probability } 1-\alpha+[\alpha] \\ [\alpha]+1 & \text{with probability } \alpha-[\alpha] \end{cases}$$

where $[\alpha]$ denotes the greatest integer less than or equal to α as is customary. We have called this process "stock splitting".

§6. TCONS CALCULATION

The original formula of G. A. Bird for the time advance per collision is $DDT = 2\Omega_k/(a_k n_k^2 v)$ where Ω_k, n_k, a_k are respectively the volume, total number of simulator molecules, and cross-section per simulator molecule for cell k, and v is the relative velocity of the collision pair. When the cells are weighted then a_k must be inversely proportional to the weight. That is if the weights are w_k, then

$$a_k\, w_k = \text{const} = \lambda.$$

The additional requirements on a_k in an actual calculation are

(1) The cross-section density in a specified region of the flow is given (see §10).

(2) The total number of molecules must be in the vicinity of some N_0 smaller than the capacity of the molecule list.

Let us say N_0 is the number of molecules obtained in the flowfield if the skimmer caused no disturbance, i.e. absorbed everything that hit it. Let A_k be the total cross-section ((cross-section/molecule) x number of molecules) in cell k. A_k can be computed from the free jet structure and the cross section density A_0 at the skimmer orifice (subroutine VPOL). Then

$$N_k = \frac{A_k}{a_k} = \frac{A_k w_k}{\lambda}$$

and

$$N_0 = \frac{1}{\lambda} \sum_{\kappa} A_\kappa w_\kappa ,$$

so that

$$\lambda = \frac{1}{N_0} \Sigma A_\kappa w_\kappa$$

and

$$TCONS(\kappa) = 2\Omega_\kappa / a_\kappa = 2\Omega_\kappa w_\kappa N_0 / \Sigma A w \quad , \; DDT = \frac{TCONS(\kappa)}{n_\kappa^2 v} .$$

§7. THE INCOMING DISTRIBUTION

First the total influx rate CU to the flow region from the jet is computed from the jet cross-section density A_e at the entry point, its velocity v (taken = 1 in our units) and the weights of the cells receiving the molecules. The total entering cell κ (assuming all κ borders on the entry plane) is

$$\frac{A_e v}{a_\kappa} \text{ x area of entry surface element.}$$

CU is the sum of all such quantities.

The number entering the flow region during time DT is CU x DT. Each entering molecule has its position coordinates chosen at random in accordance with the Ashkenas-Sherman formula [1] for variation of jet density with distance from the axis

$$\frac{\rho(R,\theta)}{\rho(R,o)} = \cos^2 \left(\frac{T\,\theta}{2\phi}\right)$$

and the weight of the cell about to receive the molecule. The details of this calculation appear in subroutine INPUT (Appendix 1).

§8. BOUNDARY CONDITIONS

The boundaries in question are the skimmer walls and the artificial boundaries introduced to enclose the flow field. A molecule colliding with a skimmer has a choice of being reflected or reemitted from a cosine distribution at a specified temperature CR. The probability of reemission is a specified accommodation coefficient ACCOM.

The artificial boundaries are:

(1) The entry plane,
(2) The upstream outer boundary,
(3) The downstream boundary.

Molecules which collide with (1) and (3) are deleted. If a molecule collides with (2) its velocity vector is reflected and augmented by the vector 2pn where n is the outward normal to the wall the p is a specified constant. If the resulting velocity vector points out of the region the molecule is deleted; otherwise it is allowed to reenter. When p is roughly equal to the normal component of the average velocity of the molecules crossing the wall, its net effect is to simulate no discontinuity in the flow due to the boundary.

§9. REFERENCING THE MOLECULE LIST

We discuss here the actual arrangement of the molecule coordinates C_d d = 1,2,...in computer memory consistent with a fast method for referring to the coordinates corresponding to a given cell.

The list is arranged in such a way that

$$C_1 \quad \cdots \quad C_{N1}$$

are in cell 1,

$$C_{N1+1} \cdots C_{N2}$$

are in cell 2, etc. The numbers NI are stored in an auxiliary list called MAP. The basic operations which must be performed on this list are

(1) Removal of a molecule, and
(2) Addition of a molecule.

(1) To remove C_α where $N_{i-1} < \alpha \leq N_i$, store C_{N_i} in place of C_α, $C_{N_{i+1}}$ in place of C_{N_i}, etc. until the end of the list is reached.

(2) To insert coordinates C into cell i we make room for it by storing the first molecule of each cell in the location following the last molecule, starting from the end and going up to cell i + 1. This leaves location $N_i + 1$ for C.

Of course, in each case MAP is updated to reflect the new configuration.

It is clear that a simple transfer of molecule from one cell to another does not require the operations to go to the end of the coordinate list. A time saving procedure to accomplish this task is incorporated into the program.

§10. SIMULATING AN ACTUAL EXPERIMENT

The program takes the limiting mean centreline jet velocity to be 1. All other velocities are given in terms of it. The simulator geometry must be similar (in the geometric sense) to the real geometry, but the units are arbitrary. Once the velocity and the distance unit are fixed, the time unit is automatically determined. Since we are mostly concerned with the steady state we do not even need to compare computed times with real times except in the initial approach to the steady state.

The remaining parameter that must be assigned is the cross section density A at some reference point in the flow which we have chosen to be the skimmer orifice assuming perfect skimming. A has units of reciprocal length, in fact it is just a multiple of 1/mean free path. Thus Ad, where d is the skimmer orifice diameter is dimensionless, and we give it the same value in the simulator as it has in the experiment.

Though this prescription is simple, it has its problems, not the least being that the hard sphere cross section of real molecules depends on their relative velocity. The best we can do is to use some sort of average relative velocity. The realtive velocity of the beam and the molecules coming off the skimmer wall is a good candidate because this is when most of the collisions take place.

§11. OUTPUT FORMAT

The output from each run contains the following information (see Appendix 2).

(1) A printout of DT and MAP for each interval DTM for checking purposes.

(2) A flow density profile at the skimmer orifice as well as a total particle flow rate.

(3) A flow density profile at the far downstream end of the flowfield as well as the total particle flow rate through the geometrical beam area. This last figure is useful in estimating the centreline beam intensity

(4) Histograms for the integrated distribution function profiles ∫fdvdw, ∫fdwdu, ∫fdudv for each cell.

The distribution function is broken down into the distribution function of the molecules which have not undergone collision in the flowfield and the remainder. The mean velocity, mean square velocity, and mean number of molecules are also given along with a theoretical comparison number for no skimmer interaction. The histograms are obtained by time averaging regular observations of the flowfield for the duration of the computation after equilibrium is reached

RESULTS

A simulator program such as this one should be viewed rather like an experimental facility suitable for investigating a range of problems connected with the skimmer. We describe here the results of a study of skimmer interaction.

A preliminary study was done to check a model proposed by French (5) suggesting a significant interaction downstream of the skimmer orifice. This was done in a program which preceeded the present one by making the downstream skimmer wall absorbing. No downstream effect was observed. A check of the distribution functions in the present study supports this conclusion.

In fact this study shows that the skimmer interaction is caused by a screen of molecules originating from the wall and from collisions extending approximately one mean free path upstream of the skimmer orifice The supporting material for this conclusion is contained in the following graphs and tables

Table 1 shows the cases computed in this study.

Fig. 1 defines the geometric variables in the program and Figs 2 - 4 give the specific geometries used in the calculations.

Table 2 shows condition near the skimmer wall for an assortment of cases. It is most significant that for all but the sharpest skimmer (angles 50°, 40°) the flow is subsonic. We suspect there is always a subsonic region close to the skimmer wall which is very thin for small angles but grows rapidly as the angle increases. Also the surface interaction with the skimmer plays a significant role, since case 23 (ACCOM = 0) gives a high Mach number even for a blunt skimmer.

Fig. 5 shows the mass flow rate through the skimmer orifice vs A in terms of the theoretical no interaction value. The mass flow is less than one in all cases computed but shows a tendency to go to 1 in the cold skimmer and no lip cases.

Fig. 6 shows beam intensity vs A. The importance of the skimmer lip sharpness is here clearly indicated. The skimmer angle seems to have less effect than the skimmer temperature. Though the general shape of the experimental beam intensity vs Knudsen number curves is evident, our results do not show decreases in intensity as strong as those in [3] for instance.

Figs. 7, 8 show mass flow and beam intensity vs various parameters. It is most interesting that an increase in PRES which leads to a decrease in the number of molecules entering the flowfield through the outer upstream boundary leads to a decreased mass flow but no significant change in beam intensity. Furthermore, a low accommodation coefficient gives high values in both graphs.

Figs. 9, 10 were obtained by smoothing the histograms produced by the program. It is clear that a Knudsen number of 1/2 (A = 4) leads to more or less complete destruction of the beam by skimmer interaction.

Figs. 11, 12 show clearly a sharp increase of molecules having a broad distribution function and infiltrating the beam region between Knudsen numbers 1/2 and 1.

Finally a note on the standard deviation of the results. The beam intensity and mass flow curves have a relative standard deviation of between 3% and 5%. It follows from the theory of Monte Carlo simulators that the probability distribution of each value obtained is normal. Accuracy is expensive in these calculations. A fourfold increase in computing time would cut the standard deviation in half.

TABLE 1

CASES COMPUTED

	CHARACTERISING PARAMETERS							COMMENTS
	A	SIA	CR	PRES	ACCOM	THLIP	GEOM	
1	0.5	0.1	1.	1.	0.5	0.1	3	Sharp skimmer
2	1.	0.1	1.	1.	0.5	0.1	3	
3	2.	0.1	1.	1.	0.5	0.1	3	
4	4.	0.1	1.	1.	0.5	0.1	3	
5	2.	0.1	1.	2.	0.5	0.1	1	Effect of PRES
6	2.	0.1	1.	1.5	0.5	0.1	1	
7	2.	0.1	1.	0.75	0.5	0.1	1	
8	2.	0.1	1.	0.5	0.5	0.1	1	
9	2.	0.0546	1.	1.	0.5	0.1	1	High mach number
10	1.	0.1	1.	2.	0.5	0.1	1	Absorbing walls
11	0.5	0.1	0.5	1.	0.5	0.1	1	Cold skimmer series
12	1.	0.1	0.5	1.	0.5	0.1	1	
13	2.	0.1	0.5	1.	0.5	0.1	1	
14	4.	0.1	0.5	1.	0.5	0.1	1	
15	0.5	0.1	1.	1.	0.5	0.1	1	Standard series blunt skimmer
16	1.0	0.1	1.	1.	0.5	0.1	1	
17	2.0	0.1	1.	1.	0.5	0.1	1	
18	4.0	0.1	1.	1.	0.5	0.1	1	
19	0.5	0.1	1.	1.	0.5	0.0	1	Perfect edge
20	1.0	0.1	1.	1.	0.5	0.0	1	
21	2.0	0.1	1.	1.	0.5	0.0	1	
22	4.0	0.1	1.	1.	0.5	0.0	1	
23	2.	0.1	1.	1.	0.	0.1	1	Effects of surface interaction
24	2.	0.1	1.	1.	1.	0.1	1	
25	0.5	0.1	1.	1.	0.5	0.1	2	Intermediate angle
26	1.	0.1	1.	1.	0.5	0.1	2	
27	2.	0.1	1.	1.	0.5	0.1	2	
28	4.	0.1	1.	1.	0.5	0.1	2	

Notes

A vs Knudsen number Kn

A	0.5	1	2	4
Kn	4	2	1	0.5

SIA, CR, PRES, ACCOM, THLIP are explained in Appendix 1.

GEOM 1, 2, 3 are shown in Figs. 1, 2, 3 respectively.

TABLE 2

CONDITIONS IN FRONT OF SKIMMER

	CELL 14		CELL 15		
Case	Mean Speed	Mach #	Mean Speed	Mach #	Inclination
15	0.55	0.64	0.51	0.55	0.83
18	0.52	0.79	0.34	0.44	0.69
5	0.62	0.88	0.48	0.57	0.52
8	0.33	0.44	0.28	0.34	0.73
1	0.80	1.4	0.72	1.2	0.28
4	0.86	2.1	0.71	1.	0.32
23	0.87	2.2	0.75	2.3	0.62
25	0.68	1.0	0.60	0.82	0.43
28	0.71	1.3	0.53	0.73	0.47

REFERENCES

1. H. Ashkenas and F. S. Sherman, The Structure and Utilization of Supersonic Free Jets in Low Density Wind Tunnels, p. 84, Supplement 3, Rarified Gas Dynamics, New York, 1969.

2. G. A. Bird, Direct Simulation Monte Carlo Method, p. 85, Vol. 2, Supplement 5, Rarifield Gas Dynamics, New York, 1969.

3. U. Bossel, F. C. Hurlbut, and F. S. Sherman, Extraction of Molecular Beams from Nearly Inviscid Free Jets, p. 945, Vol. 2, Supplement 5, Rarified Gas Dynamics, New York, 1969.

4. R. Campargue, Dimensionless Number Linked to Background and Skimmer Jet Interaction in Nozzle Beam Generation, p. 1003, Vol. 2, Supplement 5, Rarified Gas Dynamics, New York, 1969.

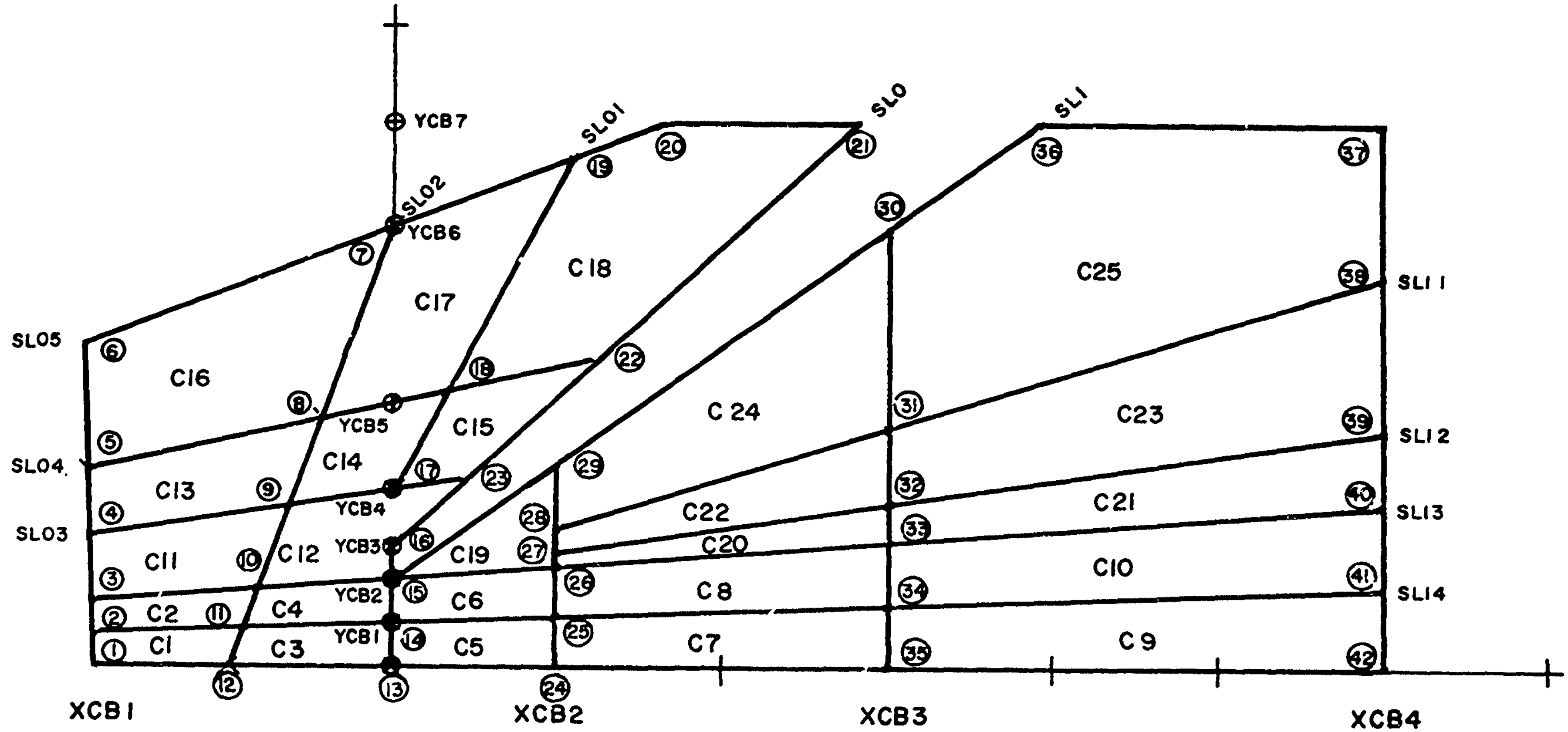
5. J. B. French, AGARD ograph 112 (N.A.T.O.) April 1966.

6. T. R. Govers, R. L. LeRoy, and J. M. Deckers, The Concurrent Effects of Skimmer Interaction and Background Scattering on the Intensity of a Supersonic Molecular Beam, p. 985, Vol. 2, Supplement 5, Rarified Gas Dynamics, New York, 1969.

7. Hagena, and Morton, Analysis of Intensity and Speed Distribution of a Molecular Beam from a Nozzle Source, p. 1370, Supplement 4, Rarified Gas Dynamics, New York, 1969.

8. I. N. Zapata, H. M. Parker, and J. H. Bodine, Performance of a Supersonic Molecular Beam, p. 67, Supplement 1, Rarified Gas Dynamics, New York, 1969.

FIG. 1 DEFINITION OF GEOMETRICAL PARAMETERS



KEY: SLI, ..., SL05 SLOPES
XCB1, ..., YCB4 X & Y INTERCEPTS RESP.
①, ..., ㊷ VERTEX NUMBERS
C1, ..., C25 CELL NUMBERS

FIG. 2 FLOWFIELD FOR GEOM = 1, SKIMMER ANGLES 96°, 70°

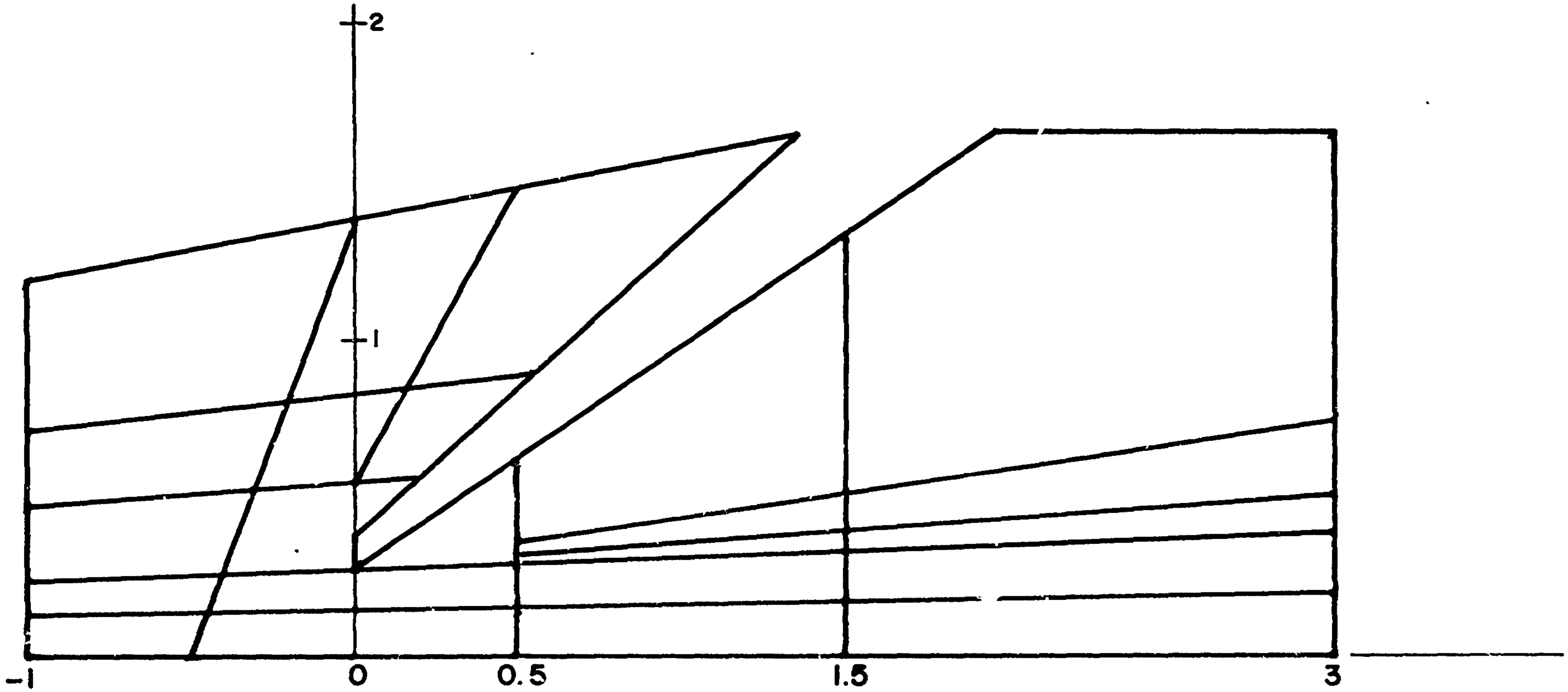


FIG. 3 FLOWFIELD FOR GEOM = 2, SKIMMER ANGLES 68°, 53°

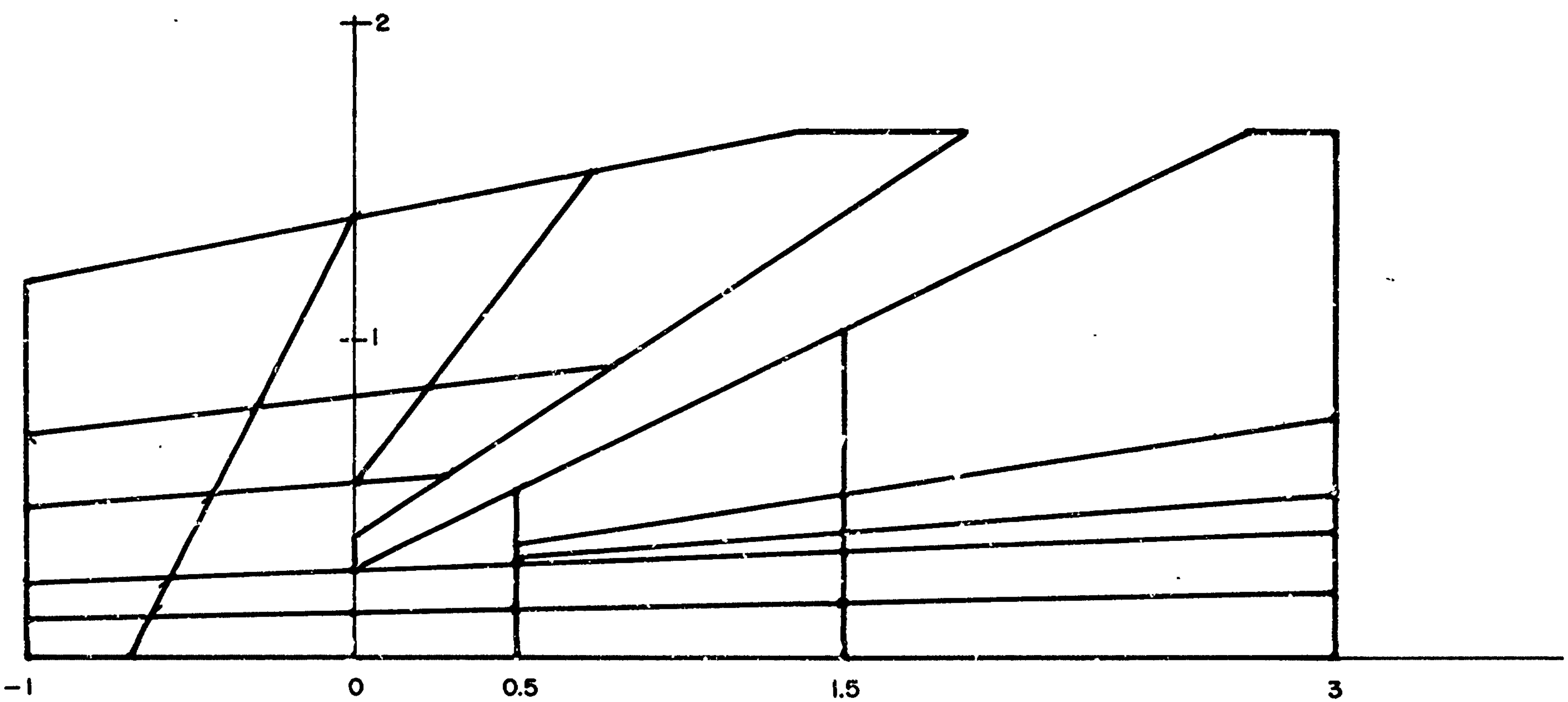


FIG. 4 FLOWFIELD FOR GEOM = 3, SKIMMER ANGLES 50°, 40°

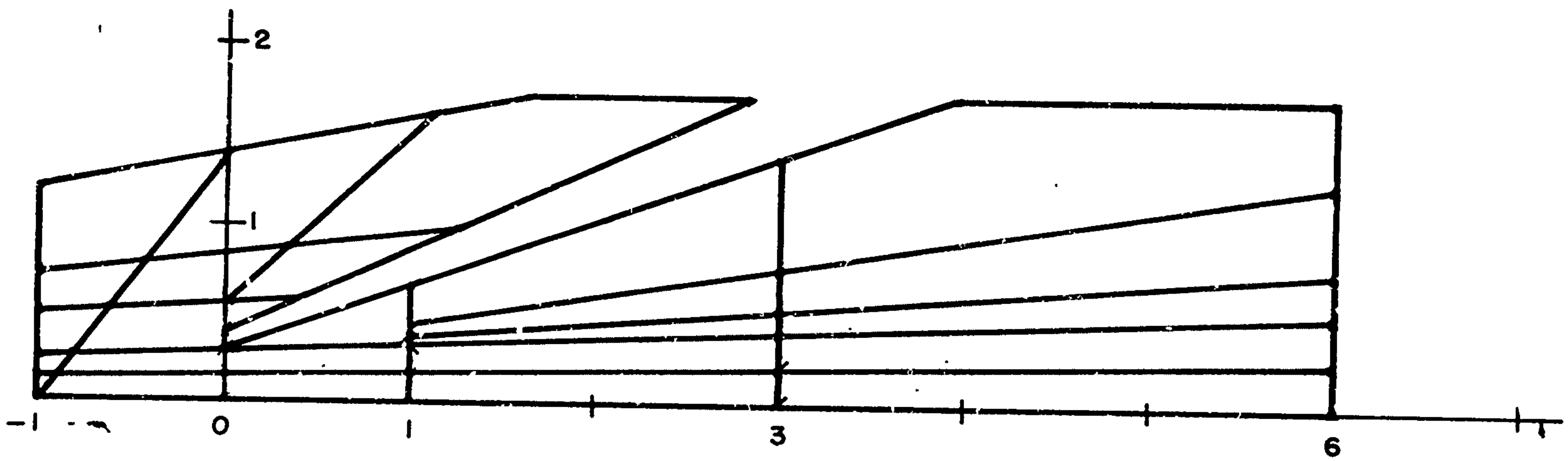


FIG. 5 MASS FLOW THROUGH SKIMMER UNDER VARIOUS CONDITIONS

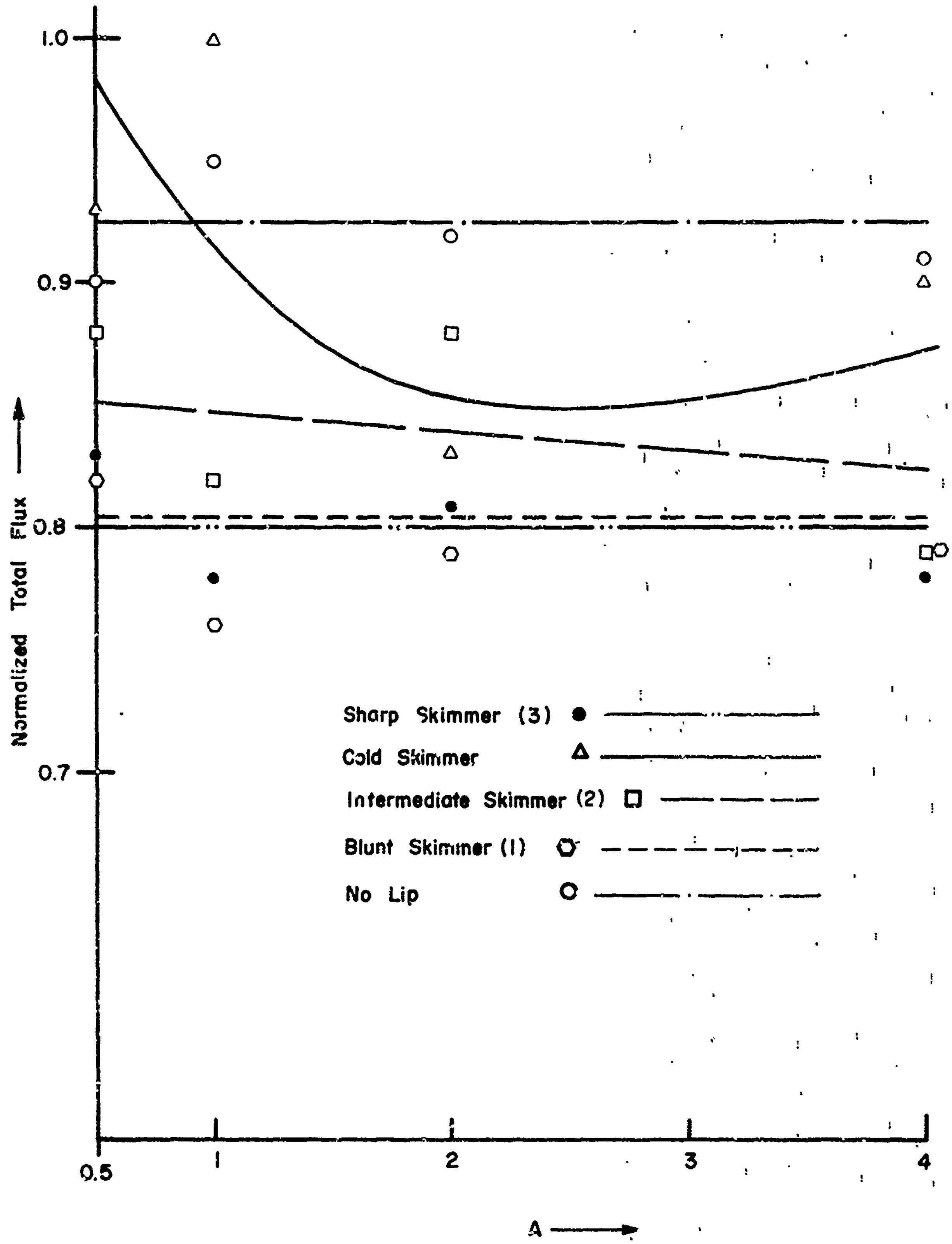


FIG. 6 BEAM INTENSITY UNDER VARIOUS CONDITIONS

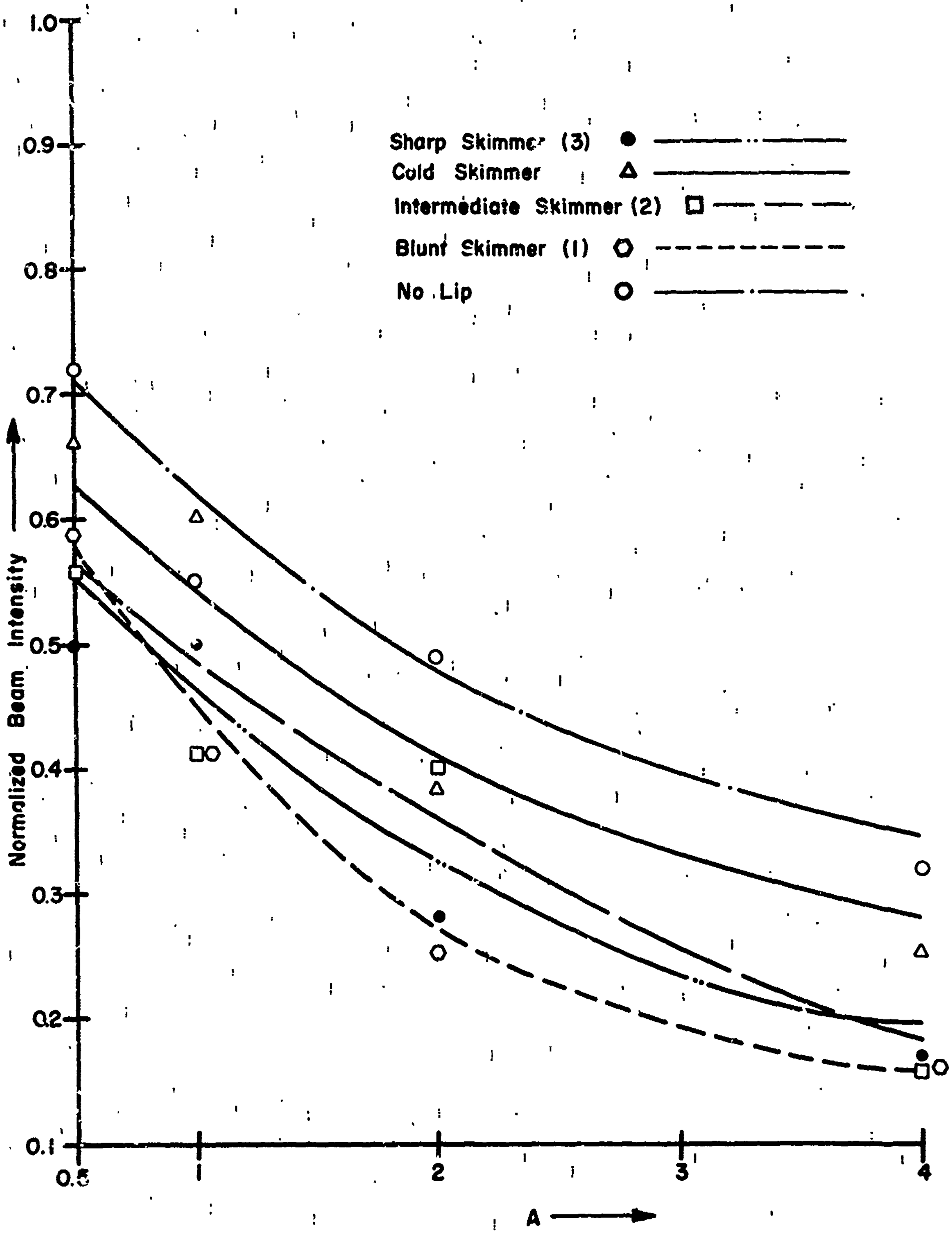


FIG. 7 MASS FLOW THROUGH SKIMMER VS OTHER PARAMETERS

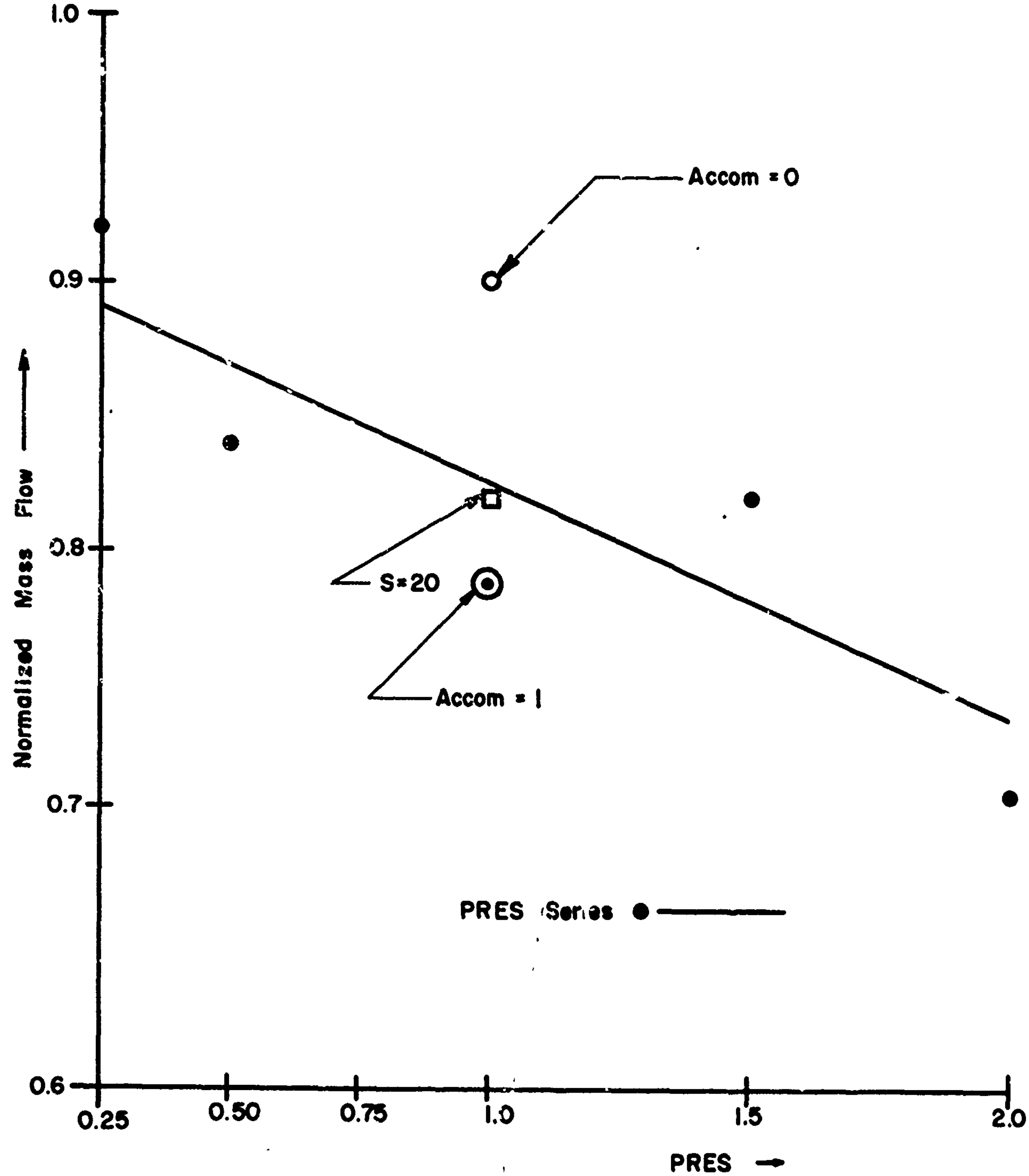


FIG. 8 BEAM INTENSITY VS OTHER PARAMETERS

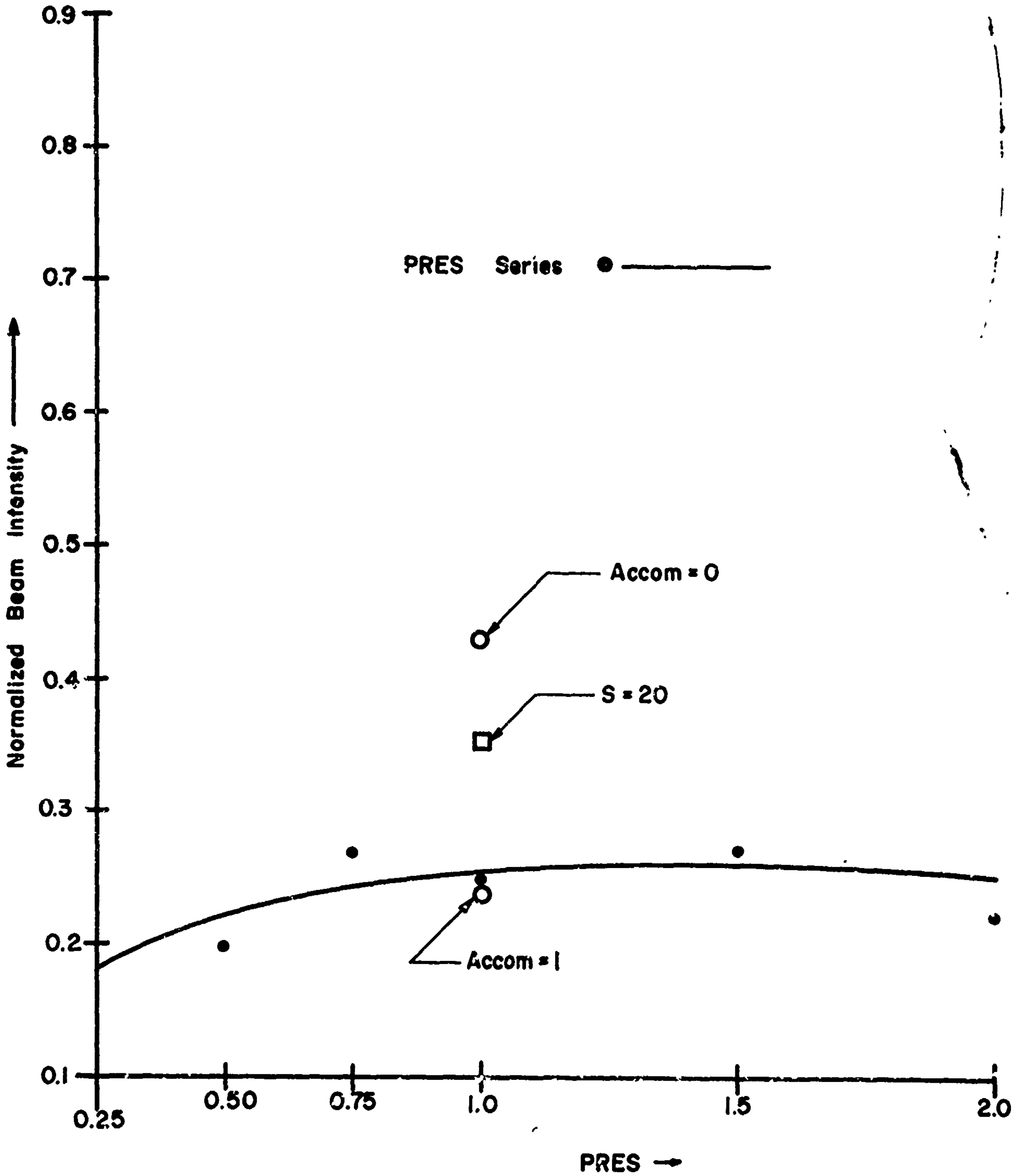


FIG. 9 FLUX DENSITY AT SKIMMER ORIFICE (GEOM=2)

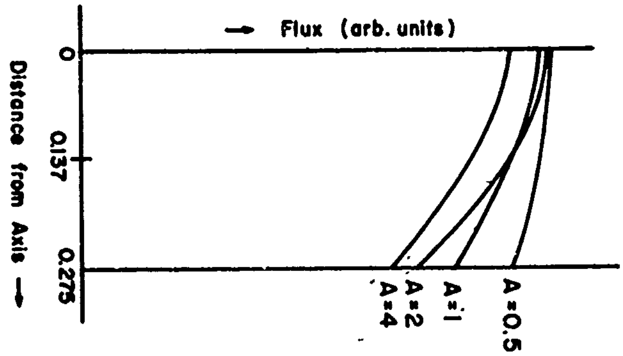


FIG. 10 BEAM PROFILES (GEOM=2)

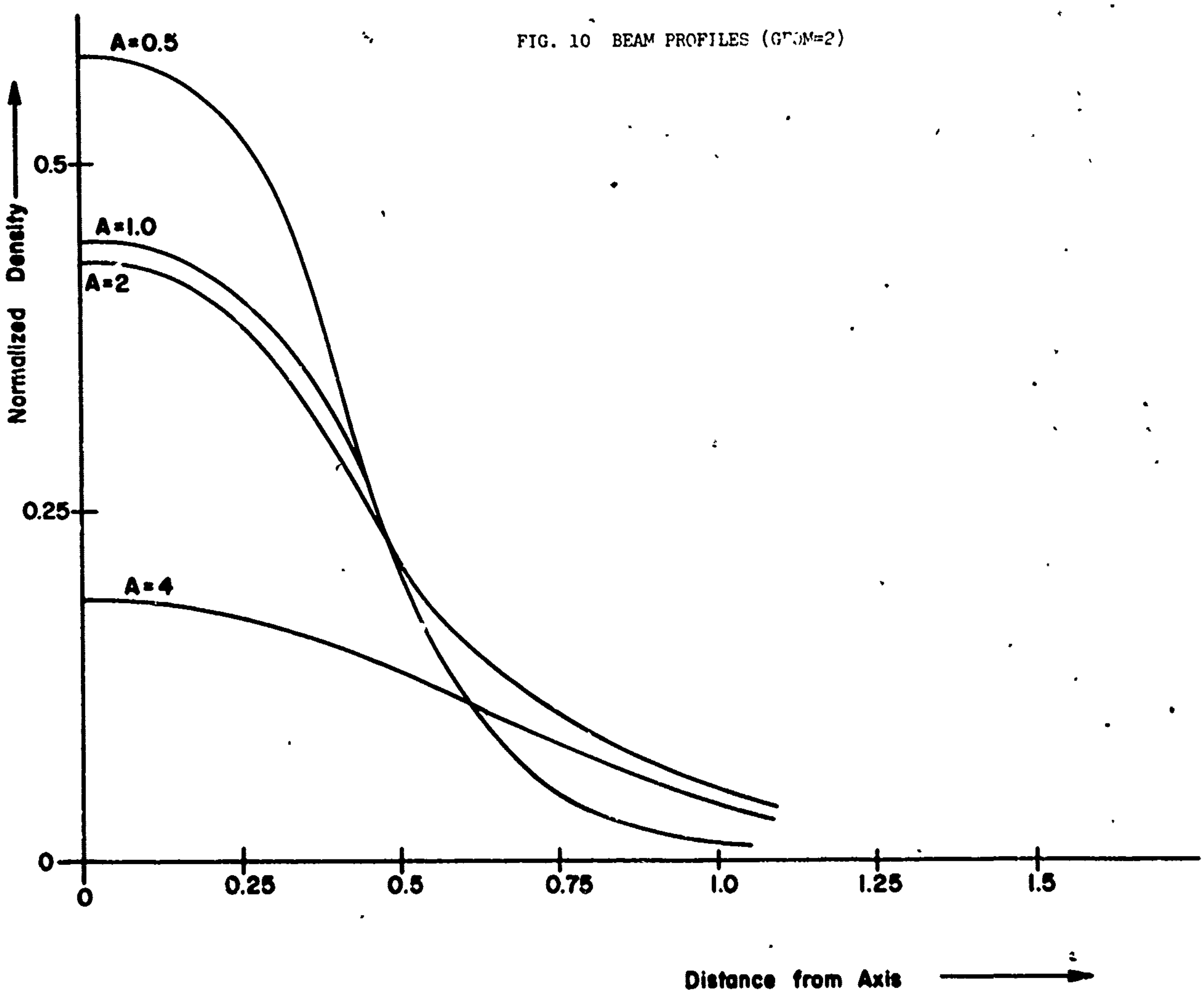


FIG. 11 DISTRIBUTION FUNCTION IN CELL 4 (GEOM=2)

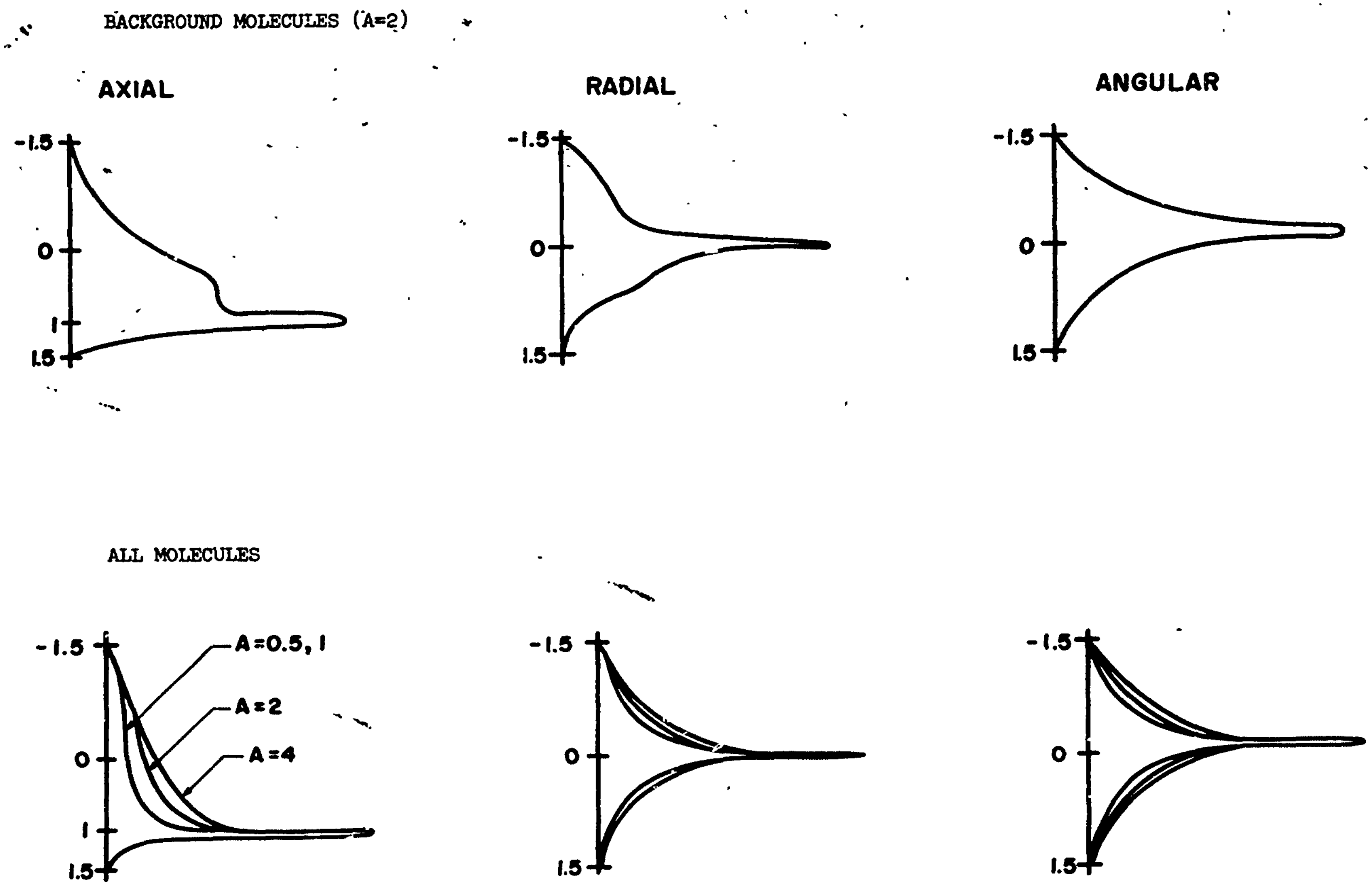
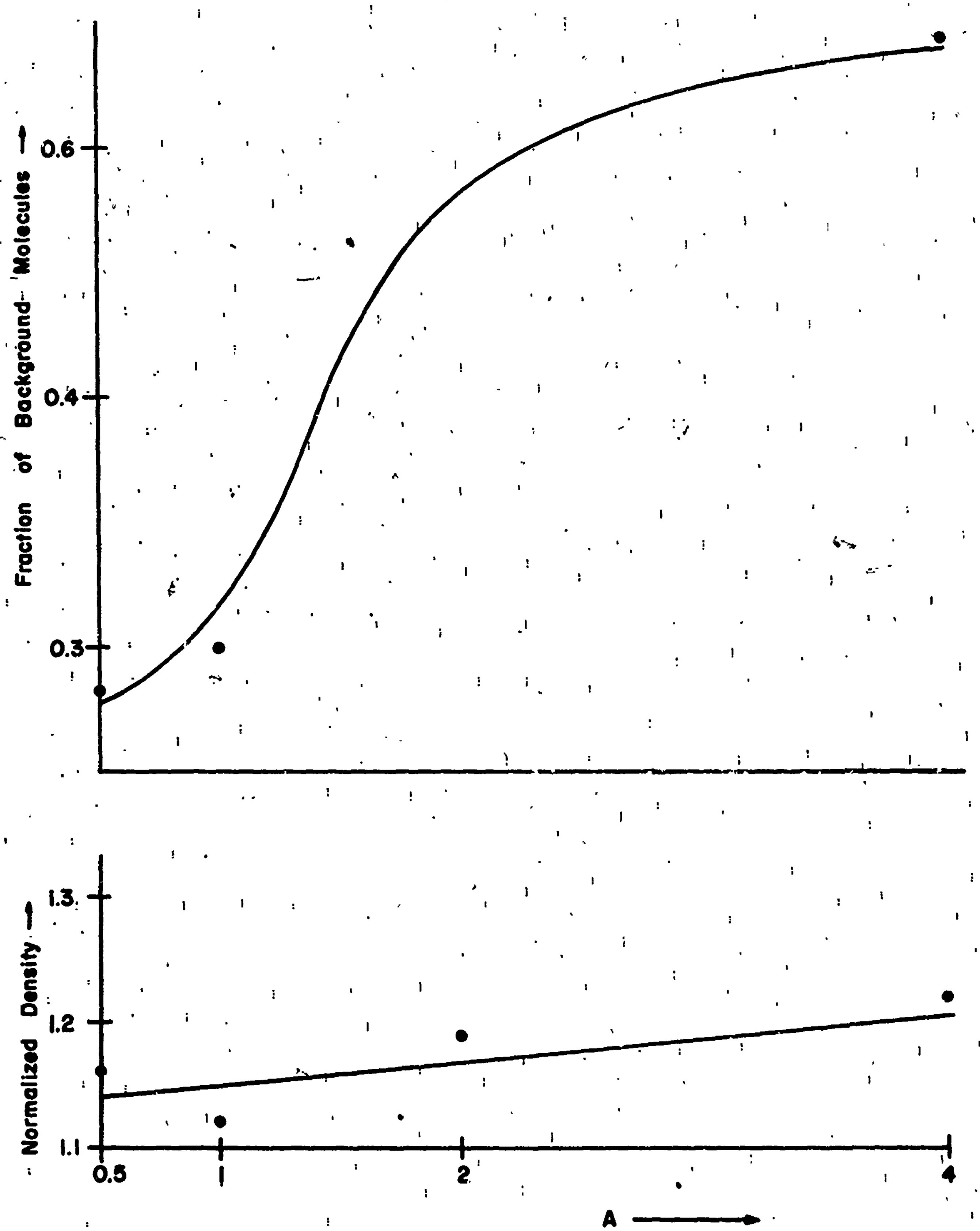


FIG. 12 CONDITIONS JUST UPSTREAM OF ORIFICE (GEOM=2)



COMMENTS

A. The Variables

Common Area	Purpose
/RCRD/	Storage area for beam and orifice profile histograms. BPRFL and HPRFL resp. DBPR and DHPR are the widths of the histogram cells in each case.
/GEOM/	Geometrical definition of the flow field - see Fig.
/NPRFL/	For internal use in the program to save computer time in the introduction of molecules through the input plane.
/CONS/	CU is the total number of molecules entering the flow field through the input plane per unit time, SIA is the speed ratio, in the jet, CR is a speed defining the temperature of the skimmer walls, ACCOM is the accommodation coefficient, PRES is related to the pressure on the upstream outer boundary, CLWT is the list containing the cell weights, TCONS is the list of time advance per collision constants for the cells. The remaining variables are for internal use of the program to save recomputing frequently used numbers. They are initially computed in SETUP.
/BEGIN/	BE and BG are initial values for the sequencing variables of the random number generators. A is the cross section density at the orifice, and NST is starting number of simulator molecules in the flow field.
/TIME/	T - Accumulated time variable for the program. DT - Weighted time per iteration (DT∼DTM). DTM - Preset time allowance per iteration. TEQ - Value of T for which equilibrium is assumed to have been achieved. TEND - Value of T at which computation ceases and output is begun. The remaining variables were not actually used in this program.
/STORE/	NCL is the number of cells, MCP = 5000, and MAP is for purposes of referencing the list of molecules. The remaining variables in this common area constitute the molecular list.
/HIST/	Storage area for the histograms FU etc; CFU etc; the mean velocity vectors for each cell - UB etc. and the mean square velocity vector for each cell UUB - CORG counts the number of molecules in each cell which have undergone more than one collision.
/SW/	Contains control parameters for the data collection operation.

/RAND/	Contains the sequencing parameters for the random number generators.
/VARS/	Temporary storage area for one molecule to save address computation time while the molecule is being processed.

B. Subroutines

MAIN	This program is used to set up the cases. Each call to PROG constitutes the computation of one case.
BLOCK DATA	Used to assign values to variables which are common to the computation of many cases.
PROG	Control program for the computation of one case.
SETUP	Computes variables not assigned in MAIN or BLOCK DATA and performs all the initialization necessary to begin computation of a case.
VPOL	Computes cell volumes and cell masses for the undisturbed free jet.
INPR	Sets up an array used to obtain the input profile by interpolation.
LOC	Function which computes the cell containing the coordinates (x,y). LOC works by computing a word IX which has 0 or 1 in the k-th bit depending on which side of the k-th boundary (x,y) is found. Then IX is compared with a standard ordered list (MASK, 1DFR).
COLI	Collision increment for cell L.
SQM	Estimates the maximum relative velocity in a given cell.
TRBKP	"Translate and Book-keep". This subroutine essentially controls computation of the convection increment.
INVRS	Computes the cell number L of the M-Th molecule in the molecule list.
UP, DOWN	Used in inserting, deleting or moving molecules in the molecule list.
TRS, TR	Transfer of coordinates.
FRP	Checks a given molecule for collision with boundaries and it moves it a distance appropriate to the time DT.
TRANS	Changes coordinates of a given molecule appropriate to the time interval DT.
LCHK	Checks whether collision occurs with a given boundary.
EMIT	Emits a fully accommodated molecule in the direction C,S.
REFL	Reflects a molecule in the direction C,S.
INPUT	Introduces CU and DT molecules through the input plane on the basis of the input profile defined by FNPR, DFNPR.

CBSPL	Cubic Spline interpolation subroutine.
TABU	Controls the accumulation of histograms and moments of the distribution function.
FIT	Inserts molecule at cell L.
SIFT	Auxiliary subroutine to accumulate histograms.
INTR	See §5.
RANDG	Standard normal random number generator.
RANDL	Uniform on [0,1] random number generator.
RANDR	Uniform on [-1,1].
UMAX	Finds the maximum element in an array.
ISGN	ISGN(X) = 1 if $x \geq 0$) 0 otherwise
DUMP.	Controls the output function of the program.
BAR	Sets up one histogram bar for printing.
HIST1	Sets up one histogram for printing.
HIST3	Sets up and prints the 3 x 3 histogram describing the distribution function in cell L.

APPENDIX I

PROGRAM LISTING

```
//FORT.SYSIN DD *
      COMMON /RCRD/ DRPR,DHPR,RPREL(50),HPREL(10)
      COMMON /GEOM/ PEI,DISNS,DIAMS,THLIP,SLO,SLI,XCR1,XCB2,XCR3,XCR4,
     X        SL11,SL12,XCRO,XCRI,XCR5,YCR1,YCR2,YCR3,YCR4,YCR5,YCR6,YCR7,
     X SLO1,SLO2,SLO3,SLO4,SLO5, SL13,SL14, RADIN,DISNI, XNI,YNI,
     X XNO,YNO, XNS,YNS, VOL(25),VWD(25)
      COMMON /NPREL/ ZNPR(10),FNPR(10),DFNPR(10)
      COMMON /CONS/ CU,SIA,SIB,CR,CT,ACCOM,PRES,CLWT(26),TCONS(25)
      COMMON /BEGIN/RR,RG,A,NST
      COMMON /TIME/ T,DT,DTM,TEO,DTSH,TEND,DTRP,TREP
      COMMON /STORE/ NCL,NCP,MAP(30),X(5000),Y(5000),U(5000),V(5000),
     X W(5000),ID(5000)
      COMMON /HIST/ FU(10,25),FV(10,25),FW(10,25),CFU(10,25),CFV(10,25),
     1CFW(10,25),UR(25),VR(25),WR(25),UUR(25),VVR(25) WWR(25),CORL(25)
      COMMON /SW/ KSAM,KREP
      COMMON /RAND/R,G
      COMMON /VARS/ XO,YO,XM,YM,UM,VM,WM,IUM
      INTEGER RR,RG
      RR=1795
      RG=1795
      NST=1000
      DTM=1./3.
      DTSH=1.
      TREP=0
      DTRP=1.
      CR=1.
      ACCOM=0.5
      NCL=25
      NCP=3000
      TEO=4.
      TEND=12.
      SIA=.1
      YCR7=1.6427679
      SIA=0.1
      ACCOM=0.5
      A=2.
      PRES=0.5
      PRES=1.
      ACCOM=1.
      ACCOM=0.
      CALL PROG(1,1)
      PRES=1.
      ACCOM=0.5
      CR=1.0
      SLO=0.67
      SLI=0.5
      A=0.5
      CALL PROG(1,1)
      A=1.
      CALL PROG(1,1)
      A=2.
      CALL PROG(1,1)
      A=4.
      CALL PROG(1,1)
      STOP
      END
      BLOCK DATA
      COMMON /CONS/ CU,SIA,SIB,CR,CT,ACCOM,PRES,CLWT(26),TCONS(25)
      COMMON /GEOM/ PEI,DISNS,DIAMS,THLIP,SLO,SLI,XCR1,XCR2,XCR3,XCR4,
     X        SL11,SL12,XCRO,XCRI,XCR5,YCR1,YCR2,YCR3,YCR4,YCR5,YCR6,YCR7,
     X SLO1,SLO2,SLO3,SLO4,SLO5, SL13,SL14, RADIN,DISNI, XNI,YNI,
     X XNO,YNO, XNS,YNS, VOL(25),VWD(25)
      DATA PEI/1.15/,DISNS/7.0/,DIAMS/0.55/,THLIP/0.1/,SLO/0.93/,
     X SLI/0.7/,XCR1/-1./,XCR2/0.5/,XCR3/1.5/,XCR4/3.0/,CU/25./,
     X SIA/0.0546/,CR/1./,ACCOM/0.5/,PRES/.5/,
     X CLWT/10*1.,290.25,39.0625,390.015625,7*1.,0./
      END
      SUBROUTINE PROG(NRPS,NARS)
C     SKINNER 1970
      COMMON /VARS/ XO,YO,XM,YM,UM,VM,WM,IUM
      COMMON /SW/ KSAM,KREP
      COMMON /TIME/ T,DT,DTM,TEO,DTSH,TEND,DTRP,TREP
      COMMON /STORE/ NCL,NCP,MAP(30),X(5000),Y(5000),U(5000),V(5000),
     X W(5000),ID(5000)
      CALL SETUP(NRPS)
    1 IF (T-TREP) 3,2,2
    2 KREP=2
      TREP=TREP+DTRP
      GO TO 4
    3 KREP=1
    4 IF(T-TEO) 6,5,5
    5 KSAM=2
      GO TO 7
    6 KSAM=1
    7 DT=0.
      DO 8 L=1,NCL
    8 CALL COLL(L)
      DT=DT/FLOAT(MAP(NCL))
      WRITE(6,99) DT,(MAP(I),I=1,25)
   99 FORMAT(' DT=',F7.5,'   MAP=',25I4)
      CALL TRANP(NARS)
      CALL INPUT
      GO TO (15,14), KSAM
   14 CALL TAND
   15 CONTINUE
      T=T+DT
      IF (T-TEND) 1,9,9
    9 CALL DUMP
      RETURN
      END
      SUBROUTINE SETUP(NRPS)
C     SKINNER 1970
      COMMON /HIST/ FU(10,25),FV(10,25),FW(10,25),CFU(10,25),CFV(10,25),
     1CFW(10,25),UR(25),VR(25),WR(25),UUR(25),VVR(25),WWR(25),CORL(25)
      COMMON /STORE/ NCL,NCP,MAP(30),X(5000),Y(5000),U(5000),V(5000),
     X W(5000),ID(5000)
      COMMON /NPREL/ ZNPR(10),FNPR(10),DFNPR(10)
      COMMON /BEGIN/RR,RG,A,NST
      COMMON /RAND/ R,G
      COMMON /GEOM/ PEI,DISNS,DIAMS,THLIP,SLO,SLI,XCR1,XCR2,XCR3,XCR4,
     X        SL11,SL12,XCRO,XCRI,XCR5,YCR1,YCR2,YCR3,YCR4,YCR5,YCR6,YCR7,
     X SLO1,SLO2,SLO3,SLO4,SLO5, SL13,SL14, RADIN,DISNI, XNI,YNI,
     X XNO,YNO, XNS,YNS, VOL(25),VWD(25)
      COMMON /TIME/ T,DT,DTM,TEO,DTSH,TEND,DTRP,TREP
      COMMON /CONS/ CU,SIA,SIB,CR,CT,ACCOM,PRES,CLWT(26),TCONS(25)
```

```
      COMMON /VARS/ XO,YO,XM,YM,UM,VM,WM,IUM
      COMMON /RCRD/ DRPR,DHPR,RPREL(50),HPREL(10)
      R=RR
      G=RG
      T=0.
C     CALCULATION OF GEOMETRICAL CONSTANTS
      YCR1=0.25*DIAMS
      YCR2=0.5*DIAMS
      DHPR=0.1*YCR2
      YCR3=YCR2+THLIP
      YCR4=DIAMS
      YCR5=1.5*DIAMS
      YCR6=2.5*DIAMS
      DISNI=DISNS+XCR1
      SL13=0.5*DIAMS/DISNS
      SL14=0.5*SL13
      SL12=2.*SL13
      SL11=2.*SL12
      SLO1=2.*SLO
      SLO2=3.*SLO
      SLO3=YCR4/DISNS
      SLO4=YCR5/DISNS
      SLO5=YCR6/DISNS
      RADIN=YCR6+SLO5*XCR1
      XCR5=(YCR6-YCR3)/(SLO-SLO5)
      DRPR=YCR7/50.
      XCRO=(YCR7-YCR3)/SLO
      XCRI=(YCR7-YCR2)/SLI
      HYP=SQRT(1.+SLO5*SLO5)
      XNS=SLO5/HYP
      YNS=-1./HYP
      HYP=SQRT(1.+SLO*SLO)
      XNO=-SLO/HYP
      YNO=1./HYP
      HYP=SQRT(1.+SLI*SLI)
      XNI=SLI/HYP
      YNI=-1./HYP
C     CONSTANTS FOR INPUT DENSITY PROFILE
      ZNPR(1)=0.
      ZNPR(2)=SL14*XCR1+YCR1
      ZNPR(3)=ZNPR(2)
      ZNPR(4)=SL13*XCR1+YCR2
      ZNPR(5)=ZNPR(4)
      ZNPR(6)=SLO3*XCR1+YCR4
      ZNPR(7)=ZNPR(6)
      ZNPR(8)=SLO4*XCR1+YCR5
      ZNPR(9)=ZNPR(8)
      ZNPR(10)=RADIN
      DIMENSION JWT(10)
      DATA JWT /1,1,2,2,11,11,13,13,16,16/
      DO 500 I=1,10
      CALL INPREF(R,DF,ZNPR(I))
      J=JWT(I)
      FNPR(I)=F*CLWT(J)
  500 DFNPR(I)=DF*CLWT(J)
      FNPRM=AMAX1(FNPR,10)
      DO 501 I=1,10
      FNPR(I)=FNPR(I)/FNPRM
  501 DFNPR(I)=DFNPR(I)/FNPRM
C     VOLUME COMPUTATION
      DIMENSION XVT(42),YVT(42)
      XVT(1)=XCR1
      XVT(2)=XCR1
      XVT(3)=XCR1
      XVT(4)=XCR1
      XVT(5)=XCR1
      XVT(6)=XCR1
      XVT(7)=0
      XVT(8)=(YCR5-YCR6)/(SLO2-SL14)
      XVT(9)=(YCR6-YCR6)/(SL12-SLO3)
      XVT(10)=(YCR2-YCR6)/(SLO2-SL13)
      XVT(11)=(YCR1-YCR6)/(SLO2-SL14)
      XVT(12)=-YCR6/SLO2
      XVT(13)=0
      XVT(14)=0
      XVT(15)=0
      XVT(16)=0
      XVT(17)=0
      XVT(18)=(YCR5-YCR4)/(SLO1-SLO4)
      XVT(19)=(YCR6-YCR4)/(SLO1-SLO5)
      XVT(20)=(YCR7-YCR4)/SLO5
      XVT(21)=(YCR7-YCR3)/SLO
      XVT(22)=(YCR3-YCR5)/(SLO4-SLO)
      XVT(23)=(YCR3-YCR4)/(SLO3-SLO)
      XVT(24)=XCR2
      XVT(25)=XCR2
      XVT(26)=XCR2
      XVT(27)=XCR2
      XVT(28)=XCR2
      XVT(29)=XCR3
      XVT(30)=XCR3
      XVT(31)=XCR3
      XVT(32)=XCR3
      XVT(33)=XCR3
      XVT(34)=XCR3
      XVT(35)=XCR3
      XVT(36)=(YCR7-YCR3)/SLI
      XVT(37)=XCR4
      XVT(38)=XCR4
      XVT(39)=XCR4
      XVT(40)=XCR4
      XVT(41)=XCR4
      XVT(42)=XCR4
      YVT(1)=0
      YVT(2)=0
      YVT(3)=0
      YVT(24)=0
      YVT(35)=0
      YVT(36)=0
      YVT(2)=YVT(1)*XCR1+SL14*YVT(2)
      YVT(11)=YVT(1)*XCR1+SL14*YVT(11)
      YVT(14)=YVT(1)*XCR1+SL14*YVT(14)
      YVT(25)=YVT(1)*XCR1+SL14*YVT(25)
      YVT(36)=YVT(1)*XCR1+SL14*YVT(36)
      YVT(41)=YVT(1)*XCR1+SL14*YVT(41)
```

```
      YVT(3 )=YCR2+SLI3*XVT(3 )
      YVT(10)=YCR2+SLI3*XVT(10)
      YVT(15)=YCR2+SLI3*XVT(15)
      YVT(26)=YCR2+SLI3*XVT(26)
      YVT(33)=YCR2+SLI3*XVT(33)
      YVT(40)=YCR2+SLI3*XVT(40)
      YVT(4 )=YCR4+SLO3*XVT(4 )
      YVT(9 )=YCR4+SLO3*XVT(9 )
      YVT(17)=YCR4
      YVT(5 )=YCR5+SLO4*XVT(5 )
      YVT(8 )=YCR5+SLO4*XVT(8 )
      YVT(18)=YCR5+SLO4*XVT(18)
      YVT(37)=YCR7
      YVT(6 )=YCR6+SLO5*XVT(6 )
      YVT(7 )=YCR6+SLO5*XVT(7 )
      YVT(19)=YCR6+SLO5*XVT(19)
      YVT(20)=YCR6+SLO5*XVT(20)
      YVT(16)=YCR3+SLO*XVT(16)
      YVT(23)=YCR3+SLO*XVT(23)
      YVT(22)=YCR3+SLO*XVT(22)
      YVT(21)=YCR3+SLO*XVT(21)

      YVT(29)=YCR2+SLI*XVT(29)
      YVT(30)=YCR2+SLI*XVT(30)
      YVT(36)=YCR2+SLI*XVT(36)
      YVT(28)=YCR2+SLI1*XVT(28)
      YVT(31)=YCR2+SLI1*XVT(31)
      YVT(38)=YCR2+SLI1*XVT(38)
      YVT(27)=YCR2+SLI2*XVT(27)
      YVT(32)=YCR2+SLI2*XVT(32)
      YVT(39)=YCR2+SLI2*XVT(39)
      VOL(1 )=VPOL(XVT,YVT,4,12,11,2,1,0,DISNS,VWD(1))
      VOL(2 )=VPOL(XVT,YVT,4,11,10,3,2,0,DISNS,VWD(2))
      VOL(3 )=VPOL(XVT,YVT,4,13,14,11,12,0,DISNS,VWD(3))
      VOL(4 )=VPOL(XVT,YVT,4,14,15,10,11,0,DISNS,VWD(4))
      VOL(5 )=VPOL(XVT,YVT,4,24,25,14,13,0,DISNS,VWD(5))
      VOL(6 )=VPOL(XVT,YVT,4,25,26,15,14,0,DISNS,VWD(6))
      VOL(7 )=VPOL(XVT,YVT,4,35,34,25,24,0,DISNS,VWD(7))
      VOL(8 )=VPOL(XVT,YVT,4,34,33,26,25,0,DISNS,VWD(8))
      VOL(9 )=VPOL(XVT,YVT,4,42,41,34,35,0,DISNS,VWD(9))
      VOL(10)=VPOL(XVT,YVT,4,41,40,33,34,0,DISNS,VWD(10))
      VOL(11)=VPOL(XVT,YVT,4,10,9,4,3,0,DISNS,VWD(11))
      VOL(12)=VPOL(XVT,YVT,5,15,16,23,9,10,DISNS,VWD(12))
      VOL(13)=VPOL(XVT,YVT,4,9,8,5,4,0,DISNS,VWD(13))
      VOL(14)=VPOL(XVT,YVT,4,17,18,8,9,0,DISNS,VWD(14))
      VOL(15)=VPOL(XVT,YVT,4,23,22,18,17,0,DISNS,VWD(15))
      VOL(16)=VPOL(XVT,YVT,4,8,7,6,5,0,DISNS,VWD(16))
      VOL(17)=VPOL(XVT,YVT,4,8,18,19,7,0,DISNS,VWD(17))
      VOL(18)=VPOL(XVT,YVT,5,22,21,20,19,18,DISNS,VWD(18))
      VOL(19)=VPOL(XVT,YVT,3,26,29,15,0,0,DISNS,VWD(19))
      VOL(20)=VPOL(XVT,YVT,4,33,32,27,26,0,DISNS,VWD(20))
      VOL(21)=VPOL(XVT,YVT,4,40,39,32,33,0,DISNS,VWD(21))
      VOL(22)=VPOL(XVT,YVT,4,32,31,28,27,0,DISNS,VWD(22))
      VOL(23)=VPOL(XVT,YVT,4,39,38,31,32,0,DISNS,VWD(23))
      VOL(24)=VPOL(XVT,YVT,4,31,30,29,28,0,DISNS,VWD(24))
      VOL(25)=VPOL(XVT,YVT,5,38,37,36,30,31,DISNS,VWD(25))
C     END OF VOLUME COMPUTATION
      TVW=0.

      DO 1 I=1,18
1     TVW=TVW+VWD(I)*CLWT(I)
      DO 2 I=1,25
2     VWD(I)=VWD(I)*NST/TVW
      DO 201 I=1,NCL
201   TCONS(I)=2.*FLOAT(NST)*VOL(I)*CLWT(I)/A/TVW
      CU=0.
      DO 202 I=1,5
      K=JWT(2*I)
202   CU=CU+CLWT(K)*(YVT(I+1)**2-YVT(I)**2)
      CT=0.707*CR
      CU=(DISNS/DISN1)**2*3.14159*NST*CU/TVW
      SIR=0.707*SIA
      S=1./SIA
      DO 401 I=1,1675
401   FU(I,1)=0.
      DO 402 I=1,60
402   RPREL(I)=0.
      IF(NRPS.LT.0) RETURN
C LOAD NST MOLECULES
      DIMENSION TEMS(25)
      DO  300 I=19,25
      TEMS(I)=CLWT(I)
300   CLWT(I)=0
      DO 102 I=1,NCL
102   MAP(I)=0
      CLWTM=IMAX(CLWT,NCL)
      XCR41=XCR4-XCR1
      DCR4=DISNS+XCR4
      CONS=DISN1*DCR4
      DO 30 I=1,NST
10    XN=CONS/(DCR4-XCR41*RANDL(R))-DISNS
      YN=YCR7*SQRT(RANDL(R))
      LM=LOC(XN,YN)
      IF(CLWT(LM)/CLWTM-RANDL(R))10,10,20
20    XN=XN+DISNS
      HYP=SQRT(YN*YN+XN*XN)
      UN=XN/HYP+SIR*RANDG(G)
      VN=YN/HYP+SIR*RANDG(G)
      WN= SIR*RANDG(G)
      IDN=0
30    CALL FIT(LM,&40)
      GO TO 60
40    WRITE(6,50)
50    FORMAT(14H OVERFLOW, SETUP)
60    CONTINUE
      DO 301 I=19,25
301    CLWT(I)=TEMS(I)
      RETURN
      END
      FUNCTION VPOL(X,Y,NV,N1,N2,N3,N4,N5,C,FM)
      DIMENSION N(5),X(1),Y(1)
      FUN(Z)=ALOG(1.+(S*D/Z)**2)
      N(1)=N1
      N(2)=N2
      N(3)=N3
      N(4)=N4
```

```
      N(5)=N5
      F=0.
      V=0
      DO 1 I=1,NV
      NI=N(I)
      J=MOD(I,NV)+1
      NJ=N(J)
      A=C+X(NI)
      B=C+X(NJ)
      DX=B-A
      IF(ABS(DX).LT.1.E-3) GO TO 1
      S=(Y(NJ)-Y(NI))/DX
      D=(B*Y(NI)-A*Y(NJ))/DX
      DX=DX/10.
      FI=(FUN(A)+FUN(B))/2.
      DO 2 K=1,9
2     FI=FI+FUN(A+K*DX)
      F=F-FI*DX
1     V=V+1.0472*(Y(NI)**2+Y(NI)*Y(NJ)+Y(NJ)**2)*(X(NI)-X(NJ))
C     1.0472=PI/3
      FM=3.14159*C*C*F
      VPOL=V
      RETURN
      END
      SUBROUTINE INPR(F,DF,R)
C      SKINNER 1970
      COMMON /GEOM/ PFI,DISNS,DIAMS,THLIP,SLO,SLI,XCR1,XCR2,XCR3,XCR4,
     X      SLI1,SLI2,XCR0,XCRI,XCR5,YCR1,YCR2,YCR3,YCR4,YCR5,YCR6,YCR7,
     X SLO1,SLO2,SLO3,SLO4,SLO5, SLI3,SLI4, RADIN,DISN1, XN1,YN1,
     X XN0,YN0, XN5,YN5, VOL(25)
      A=ATAN(R/DISN1)
      A2=2.*A
      AP=PFI*A
      WA=1.+COS(A2)
      WB=1.+COS(AP)
      W=WA*WB
      F=R*W
      DA=DISN1/(DISN1*DISN1+R*R)
      DF=W-(2.*R*SIN(A2)*WB+PFI*R*WA*SIN(AP))*DA
      RETURN
      END
      FUNCTION LOC(X,Y)
C      SKINNER 1970
      INTEGER TA,TB,TC,TD,TE,TF,TG,TH,TI,TJ,TK,TL, TM,TN,TO,TP
      COMMON /GEOM/ PFI,DISNS,DIAMS,THLIP,SLO,SLI,XCR1,XCR2,XCR3,XCR4,
     X      SLI1,SLI2,XCR0,XCRI,XCR5,YCR1,YCR2,YCR3,YCR4,YCR5,YCR6,YCR7,
     X SLO1,SLO2,SLO3,SLO4,SLO5, SLI3,SLI4, RADIN,DISN1, XN1,YN1,
     X XN0,YN0, XN5,YN5, VOL(25)
      LOGICAL LIDER(25),LMASK(25),LIX,LIXM
      DIMENSION IDER(25),MASK(25)
      EQUIVALENCE (LIDER(1),IDER(1)),(LMASK(1),MASK(1)),
     X (LIX,IX),(LIXM,IXM)
      DATA MASK/Z10021,Z10023,Z8021,Z8023,ZC001,ZC003,Z6001,Z6003,Z3001,
     X Z3003,Z10026,Z80A6,Z1002C,Z6C,ZCC,Z10038,Z78,Z208,Z4102,
     X Z6802,Z3802,Z6C00,Z3C00,Z6500,Z3700/,
     X IDER/ Z10001,Z10002,Z21,Z22,ZC001,ZC002,Z2001,Z2002,Z1001,Z1002,
     X Z10004,Z84,Z10008,Z28,ZC8,Z10010,Z30,Z200,Z4100,
     X Z2800,Z1800,Z2400,Z1400,Z2100,Z1300/,

     X TA/Z2/,TB/Z4/,TC/Z8/,TD/Z10/,TE/Z20/,TF/Z40/,TG/Z80/,
     X TH/Z100/,TI/Z200/,TJ/Z400/,TK/Z800/,TL/Z1000/,TM/Z2000/,
     X TN/Z4000/,TO/Z8000/,TP/Z10000/
3      IX= ISGN(SLI4*X+YCR1-Y) + TA*ISGN(SLI3*X+YCR2-Y)
     X + TB*ISGN(SLO3*X+YCR4-Y) + TC*ISGN(SLO4*X+YCR5-Y)
     X +TD*ISGN(SLO5*X+YCR6-Y) + TE*ISGN(SLO2*X+YCR6-Y)
     X + TF*ISGN(SLO1*X+YCR4-Y) + TG*ISGN(Y-YCR3-SLO*X)
     X + TH*ISGN(SLI*X+YCR2-Y) + TI*ISGN(YCR7-Y)
     X + TJ*ISGN(SLI1*X+YCR2-Y) + TK*ISGN(SLI2*X+YCR2-Y)
     X + TL*ISGN(XCR4-X) + TM*ISGN(XCR3-X) + TN*ISGN(XCR2-X)
     X + TO*ISGN(X) + TP*ISGN(X-XCR1)
      DO 1 I=1,25
      LIXM=LIX.AND.LMASK(I)
      II=I
      IF (IXM-IDER(I)) 1,2,1
    1 CONTINUE
      LOC=26
      RETURN
    2 LOC=II
      RETURN
      END
      SUBROUTINE COLI(L)
C     SKINNER 1970
      COMMON /STORE/ NCL,NCP,MAP(30),X(5000),Y(5000),U(5000),V(5000),
     X W(5000),ID(5000)
      COMMON /CONS/ CU,SIA,SIR,CR,CT,ACCOM,PRES,CLWT(26),TCONS(25)
      COMMON /RAND/R,G
      COMMON /TIME/ T,DT,DTM,TEO,DTSM,TEND,DTRP,TREP
      NAMELIST/CHEK/ KA,KB,MA,MB,ITERY,UA,VA,WA,UB,VB,WB,MAP
      DDT=0.
      IF (L-1) 1,1,2
    1 KA=0
      GO TO 3
    2 KA=MAP(L-1)
    3 KB=MAP(L)
      FKBA=FLOAT(KB-KA)
      KA=KA+1
      IF (KB-KA) 10,10,4
4     CCT=TCONS(L)/FKBA/FKBA
      VMAX=SQRT(SQM(KA,KB,U)+SQM(KA,KB,V)+SQM(KA,KB,W))
      IF(VMAX) 10,10,5
    5 MA=KA+IFIX(FKBA*RANDL(R))
      MB=KA+IFIX(FKBA*RANDL(R))
      IF (MB-MA) 6,5,6
    6 UA=U(MA)
      VA=V(MA)
      WA=W(MA)
      UB=U(MB)
      VB=V(MB)
      WB=W(MB)
      URA=UB-UA
      VRA=VB-VA
      WRA=WB-WA
      VREL=SQRT(URA*URA+VRA*VRA+WRA*WRA)
      IF(VREL.LT.VMAX*RANDL(R)) GO TO 5
      TCC=CCT/VREL
      DDT=DDT+TCC
```

```
      IF(DDT.LT.DTM) GO TO 7
      IF(RANDL(R).GT.(DDT-DTM)/TCR) GO TO 7
      DDT=DDT-TCR
      GO TO 10
    7 FU=RANDN(R)
      FV=RANDN(R)
      FW=RANDN(R)
      F=SQRT(FU*FU+FV*FV+FW*FW)
      IF (F) 7,7,8
    8 IF (F-1.) 9,9,7
    9 FU=VREL*FU/F
      FV=VREL*FV/F
      FW=VREL*FW/F
      U(MA)=0.5*(UA+UB-FU )
      V(MA)=0.5*(VA+VB-FV)
      W(MA)=0.5*(WA+WB-FW)
      ID(MA)=ID(MA)+2
      U(MB)=0.5*(UA+UB+FU)
      V(MB)=0.5*(VA+VB+FV)
      W(MB)=0.5*(WA+WB+FW)
      ID(MB)=ID(MB)+2
      ITERY=1
      IF(VREL.EQ.0.) WRITE(6,CHEK)
      IF (DDT-DTM) 5,10,10
   10 DT=DT+DDT*EKRA
      RETURN
      END
      FUNCTION SOM(L,M,X)
C     SKINNER 1970
      DIMENSION X(1)
      A=X(L)
      B=A
      DO 4 I=L,M
      XM=X(I)
      IF (XM-A) 1,2,2
    1 A=XM
    2 IF (XM-B) 4,4,3
    3 B=XM
    4 CONTINUE
      SOM=(B-A)*(B-A)
      RETURN
      END
      SUBROUTINE TRBKP(NABS)
C     SKINNER 1970
      COMMON /TIME/ T,DT,DTM,TEO,DTSM,TEND,DTRP,TREP
      COMMON /STORE/ NCL,MCP,MAP(30),X(5000),Y(5000),U(5000),V(5000),
     X W(5000),ID(5000)
      COMMON /CONS/ CU,SIA,SIB,CR,CT,ACCOM,PRES,CLWT(26),TCONS(25)
      COMMON /VARS/ XO,YO,XM,YM,UM,VM,WM,IDM
      DATA IC/16384/
      M=1
    1 MPCL=MAP(NCL)
    2 IF (M-MPCL) 4,4,3
    3  RETURN
    4 IDM=ID(M)
      IF (IDM/IC) 6,6,5
    5 ID(M)=IDM-IC
      M=M+1
      GO TO 2
    6 LO=INVRS(MAP,NCL,M)
      XO=X(M)
      YO=Y(M)
      UM=U(M)
      VM=V(M)
      WM=W(M)
      CALL ERP(NABS)
   16 LM=LOC(XM,YM)
      CLO=CLWT(LM)/CLWT(LO)
      ICLO=INTR(CLO)
      IF (ICLO) 7,7,14
   14 IF (LM-LO) 9,12,10
    9 CALL DOWN(MAP,M,LO-1,LM)
      M=M+ICLO
      GO TO 11
   10 CALL UP(MAP,M,LO,LM)
      IDM=IDM+IC
   11  MM=MAP(LM)
       GO TO 13
   12 MM=M
       M=M+1
   13 CALL TRS(MM)
      IF(ICLO-1) 1,1,15
   15 DO 151 I=2,ICLO
  151 CALL FIT(LM,G1)
      GO TO 1
    7 CALL UP(MAP,M,[illegible],NCL)
      MAP(NCL)=MPCL-1
      GO TO 1
       END
      FUNCTION INVRS(MAP,L,M)
      DIMENSION MAP(1)
      I=0
      J=L
    9 IF (J-I-1) 10,10,6
   10 INVRS=J
      RETURN
    6 K=(I+J)/2
      IF (M-MAP(K)) 7,7,8
    7 J=K
      GO TO 9
    8  I=K
      GO TO 9
      END
      SUBROUTINE UP(MAP,M,L,K)
      DIMENSION MAP(1)
      LX=L
      MA=M
    1 MB=MAP(LX)
       CALL TR(MA,MB)
       IF (LX-K) 3,2,2
    2  RETURN
    3 MAP(LX)=MB-1
      LX=LX+1
      MA=MB
      GO TO 1
      END
       SUBROUTINE DOWN(MAP,M,L,K)
      DIMENSION MAP(1)
       LX=L
      MA=M
    1 MB=MAP(LX)+1
      CALL TR(MA,MB)
      MAP(LX)=MB
      IF (LX-K) 2,2,3
    2 RETURN
    3 LX=LX-1
      MA=MB
      GO TO 1
      END
      SUBROUTINE TRS(M)
      COMMON /VARS/ XO,YO,XM,YM,UM,VM,WM,IDM
      COMMON /STORE/ NCL,MCP,MAP(30),X(5000),Y(5000),U(5000),V(5000),
     X W(5000),ID(5000)
      X(M)=XM
      Y(M)=YM
      U(M)=UM
      V(M)=VM
      W(M)=WM
      ID(M)=IDM
      RETURN
       END
      SUBROUTINE TR(M,N)
      COMMON /STORE/ NCL,MCP,MAP(30),X(5000),Y(5000),U(5000),V(5000),
     X W(5000),ID(5000)
      X(M)=X(N)
      Y(M)=Y(N)
      U(M)=U(N)
      V(M)=V(N)
      W(M)=W(N)
      ID(M)=ID(N)
      RETURN
      END
      SUBROUTINE ERP(NABS)
C     SKINNER 1970
      COMMON /GEOM/ PEI,DISNS,DIAMS,THLIP,SLO,SLI,XCR1,XCR2,XCR3,XCR4,
     X       SLI1,SLI2,XCBO,XCB1,XCB5,YCB1,YCB2,YCB3,YCB4,YCB5,YCB6,YCB7,
     X SLO1,SLO2,SLO3,SLO4,SLO5, SLI3,SLI4, RADIN,DISNI, XNI,YNI,
     X XNO,YNO, XN5,YN5, VOL(25)
      COMMON /VARS/ XO,YO,XM,YM,UM,VM,WM,IDM
      COMMON /CONS/ CU,SIA,SIB,CR,CT,ACCOM,PRES,CLWT(26),TCONS(25)
      COMMON /RAND/ R,G
      COMMON /RCRD/ DBPR,DHPR,BPREL(50),HPREL(10)
      COMMON /TIME/ T,DT,DTM,TEO,DTSM,TEND,DTRP,TREP
      COMMON /SW/ KSAM,KREP
      TL=DT
   31 CALL LCHK(SLO5,YCB6,XCB1,XCB5,-1.,E5,TL)
      CALL LCHK(0.,YCB7,XCB5,XCBO,-1.,E7,TL)
   32 CALL LCHK(SLO,YCB3,0.,XCBO,1.,E9,TL)
      CALL LCHK(SLI,YCB2,0.,XCB1,-1.,E12,TL)
C     CHECK LIP
      IF(UM) 19,19,1
    1 IF(XO) 2,2 ,21
    2 TT=-XO/UM
      IF(TL-TT) 19,3,3
    3 CALL TRANS(TT,0)
      IF(YCB2-YM) 4,4,18
    4 IF(YCB3-YM) 19,15,15
C     BEAM PROFILE
   21 TT=(XCB4-XO)/UM
      IF(TL-TT) 19,22,22
   22 GO TO(19,23),KSAM
   23 CALL TRANS(TT,0)
      CALL SIFT(0.,DBPR,BPREL,50,1.,YM)
      GO TO 19
    5 CALL REFL(XN5,YN5)
      UM=UM-PRES*XN5
      VM=VM-PRES*YN5
      GO TO 32
    7 CALL REFL(0.,-1.)
      VM=VM+PRES
      GO TO 32
    9 IF(ACCOM-RANDL(R)) 10,10,11
   10 CALL REFL(XNO,YNO)
      IF(NABS.LT.0) YO=10000.
      GO TO 31
   11 CALL EMIT(XNO,YNO)
      IF(NABS.LT.0) YO=10000.
      GO TO 31
   12 IF(ACCOM-RANDL(R)) 13,13,14
   13 CALL REFL(XNI,YNI)
      IF(NABS.LT.0) YO=10000.
      GO TO 19
   14 CALL EMIT(XNI,YNI)
      IF(NABS.LT.0) YO=10000.
      GO TO 19
   15 TL=TL-TT
      IF (ACCOM-RANDL(R)) 16,16,17
   16 CALL TRANS(TT,1)
      CALL REFL(-1.,0.)
      IF(NABS.LT.0) YO=10000.
      GO TO 31
   17 CALL EMIT(-1.,0.)
      IF(NABS.LT.0) YO=10000.
      GO TO 31
   18 GO TO (19,24), KSAM
   24 CALL SIFT(0.,DHPR,HPREL,10,1.,YM)
   19 CALL TRANS(TL,1)
      RETURN
      END
      SUBROUTINE TRANS(DT,III)
      COMMON /VARS/ X,R,XNEW,RNEW,VX,VR,VO
      VVR=VR
      VVO=VO
      TVO=DT*VO
      RM=R+DT*VR
      XNEW=X+DT*VX
      RNEW=SQRT(RM*RM+TVO*TVO)
      IF (RNEW) 1,2,1
    1 IF(III.EQ.0) RETURN
      SN=TVO/RNEW
```

```
      CS=WH/VMEA
      VM=SN*VU+CS*VVR
      VU=CS*VU-SN*VVR
      RETURN
    2 VM=-VM
      RETURN
      END
      SUBROUTINE ICHK(S,B,XI,XU,XI,*,TLEFT)
      COMMON /VARS/ XO,YO,XM,YM,UM,VM,WM,IUM
C     THIS SUBROUTINE CHECKS WHETHER A MOLECULE COLLIDES WITH THE SEGMENT
C     FROM XI TO XU OF THE LINE Y=S*X+B .  II IS A PARAMETER DESIGNATING
C     SIDE OF LINE FROM WHICH COLLISION CAN OCCUR
      DT=TLEFT
      TMAX=0.
      TT=0
      AL=S*XO+B
      IF((VM-S*UM)*XI) 2,17,17
    2 IF((YO-AL)*XI) 17,16,3
    3 C1=VM*VM+UM*UM -(S*UM)**2
      C2=YO*VM-AL*S*UM
      C3=YO*YO-AL*AL
      IF(C1) 7,4,7
    4 IF(C2) 6,17,6
    6 TT=-0.5*C3/C2
      GO TO 14
    7 C2=C2/C1
      C3=C3/C1
      DISCR=C2*C2-C3
      IF(DISCR) 17,9,9
    9 RTDIS=SQRT(DISCR)
      TMAX=RTDIS-C2
      TMIN=-RTDIS-C2
      IF (TMAX.LT.0) GO TO 17
      IF (TMIN.LT.0) GO TO 12
      TT=TMIN
      GO TO 14
   12 TT=TMAX
   14 IF(DT-TT) 17,15,15
   15 CALL TRANS(TT,0)
      IF(XI .LT.XM .AND. XM.LT.XU) GO TO 16
      IF (TT.LT.TMAX) GO TO 12
   17 RETURN
   16 TLEFT=DT-TT
      CALL TRANS(TT,1)
      RETURN1
      END
      SUBROUTINE EMIT(C,S)
C     SKINNER 1970
      COMMON /CONS/ CU,SIA,SIB,CR,CT,ACCOM,PRES,CLMT(25),TCONS(25)
      COMMON /VARS/ XO,YO,XM,YM,UM,VM,WM,IUM
      DATA IC/16384/
      COMMON /RAND/ R,G
      VN=CT*SQRT(-ALOG(RANDL(R)))
      VT=CR*RANDG(G)
      UM=C*VN-S*VT
      VM=C*VT+S*VN
      WM=CR*RANDG(G)
      XO=XM
      YO=YM
      IUM=IC*(IUM/IC)
      RETURN
      END
      SUBROUTINE REFL(XN,YN)
      COMMON /VARS/ XO,YO,XM,YM,UM,VM,WM,IUM
      DATA IC/16384/
      VNORM=2.*(UM*XN+VM*YN)
      UM=UM-VNORM*XN
      VM=VM-VNORM*YN
      XO=XM
      YO=YM
      IUM=IC*(IUM/IC)
      RETURN
      END
      SUBROUTINE INPUT
C     SKINNER 1970
      INTEGER R,G
      COMMON /GEOM/ PEL,DISNS,DIANS,THLIP,SLO,SLI,XCB1,XCB2,XCB3,XCB4,
     X       SLI1,SLI2,XCB0,XCH1,XCH5,YCH1,YCB2,YCB3,YCB4,YCB5,YCB6,YCB7,
     X SLO1,SLO2,SLO3,SLO4,SLO5, SLI3,SLI4, RADIN,DISNI, XN1,YN1,
     X XNO,YNO, XN5,YN5, VOL(25)
      COMMON /VARS/ XO,YO,XM YM,UM,VM,WM,IUM
      COMMON /CONS/ CU,SIA,SIB,CR,CT,ACCOM,PRES,CLMT(26),TCONS(25)
      COMMON /NPREL/ ZNPR(10),FNPR(10),DFNPR(10)
      COMMON /TIME/ T,DT,DTM,TEO,DTSM,TEND,DTRP,TREP
      COMMON /RAND/ R,G
      NUM=INTR(CU*DT)
      DO 6 I=1,NUM
  818 XO=XCB1
    1 YO=RADIN*RANDL(R)
      IF(CASPL(ZNPR,FNPR,DFNPR,10,YO)-RANDL(R)) 1,4,4
    4 HYP=SQRT(DISNI*DISNI+YO*YO)
      VXI=DISNI/HYP
      VRI=YO/HYP
    8 Z=VXI+SIB*RANDG(G)
      IF(Z-1.5*RANDL(R)) 8,5,5
    5 UM=Z
      VM=VRI+SIA*RANDG(G)
      WM=SIB*RANDG(G)
      IUM=0
      CALL TRANS(DT*RANDL(R))
      LTJ=LOC(XM,YM)
      IF(LTJ.LE.25) GO TO 6
      WRITE(6,914)
  914 FORMAT(1H ,*ERROR IN LOC*)
      GO TO 818
    6 CALL FIT(LTJ,&7)
    7 RETURN
      END
      FUNCTION CASPL(Z,F,DF,I,X)
C     SKINNER 1970
C     SUPPOSE F IS A FUNCTION WE WISH TO COMPUTE AND WE ARE GIVEN A VECT
C     OF DIMENSION I OF ARGUMENT POINTS, A VECTOR G CONTAINING THE FUNCT
C     F( )=ZF( ) AND A VECTOR H CONTAINING THE FUNCTION 3F( )-F( ).
C     CASPL COMPUTES AN INTERPOLATED VALUE OF F AT X.
      DIMENSION Z(1),F(1),DF(1)
C     BINARY SEARCH FOR INTERVAL Z(I),Z(J)  (J=I+1) WHERE X OCCURS
      I=1
      J=N
    6 K=(I+J)/2
      IF(X-Z(K)) 7,7,8
    7 J=K
      GO TO 9
    8 I=K
    9 IF(J-I-1) 10,10,6
C     END OF BINARY SEARCH
   10 ZI=Z(I)
      ZJ=Z(J)
      ZIJ=ZJ-ZI
      XI=(X-ZI)/ZIJ
      XJ=(ZJ-X)/ZIJ
      FI=F(I)
      FJ=F(J)
      DFI=-ZIJ*DF(I)
      DFJ=ZIJ*DF(J)
      GI=DFI-2.*FI
      GJ=DFJ-2.*FJ
      HI=3.*FI-DFI
      HJ=3.*FJ-DFJ
      CASPL=(GJ*XI+HJ)*XI*XI+(GI*XJ+HI)*XJ*XJ
      RETURN
      END
      SUBROUTINE TABU
      COMMON /STORE/ NCL,MCP,MAP(30),X(5000),Y(5000),U(5000),V(5000),
     X W(5000),ID(5000)
      COMMON /VARS/ XO,YO,XM,YM,UM,VM,WM,IUM
      COMMON /TIME/ T,DT,DTM,TEO,DTSM,TEND,DTRP,TREP
      COMMON /HIST/ FU(10,25),FV(10,25),FW(10,25),CFU(10,2     ,25),
     1CFW(10,25),UB(25),VB(25),WB(25),UUB(25),VVB(25),WWB(     ,25)
      MLAST=0
      DO 6 I=1,NCL
      MFIRST=MLAST+1
      MLAST=MAP(I)
      LSIZE=MLAST+1-MFIRST
      IF(LSIZE) 6,6,7
    7 D=DT/FLOAT(LSIZE)
      DO 8 M=MFIRST,MLAST
      UM=U(M)
      VM=V(M)
      WM=W(M)
      MTAG=MOD(ID(M),32)/2
      IF (MTAG) 1,1,2
      CALL SIFT(-1.5,0.3,FU(1,I),10,DT,UM)
      CALL SIFT(-1.5,0.3,FV(1,I),10,DT,VM)
      CALL SIFT(-1.5,0.3,FW(1,I),10,DT,WM)
      GO TO 5
    2 IF(MTAG/2) 4,4,3
    3 CORG(I)=CORG(I)+DT
    4 CALL SIFT(-1.5,0.3,CFU(1,I),10,DT,UM)
      CALL SIFT(-1.5,0.3,CFV(1,I),10,DT,VM)
      CALL SIFT(-1.5,0.3,CFW(1,I),10,DT,WM)
    5 UB(I)=UB(I)+UM*D
      VB(I)=VB(I)+VM*D
      WB(I)=WB(I)+WM*D
      UUB(I)=UUB(I)+UM*UM*D
      VVB(I)=VVB(I)+VM*VM*D
      WWB(I)=WWB(I)+WM*WM*D
    8 CONTINUE
    6 CONTINUE
      RETURN
      END
      SUBROUTINE FIT(L,*)
      COMMON /VARS/ XO,YO,XM,YM,UM,VM,WM,IUM
      COMMON /TIME/ T,DT,DTM,TEO,DTSM,TEND,DTRP,TREP
      COMMON /STORE/ NCL,MCP,MAP(30),X(5000),Y(5000),U(5000),V(5000),
     X W(5000),ID(5000)
      MC=MAP(NCL)+1
      IF (MC-MCP)  3,3,1
    1 WRITE (6,2)
    2 FORMAT (17H STORAGE OVERFLOW)
      TEND=T
      RETURN 1
    3 CALL MOVM (MAP,MC,NCL,L)
      M=MAP(L)
      CALL INS(M)
      RETURN
      END
      SUBROUTINE SIFT(A,D,X,N,Z,W)
C     TO ACCUMULATE A HISTOGRAM FOR THE VARIABLE W WHERE X IS THE
C     HISTOGRAM ARRAY OF DIMENSION N. IF A+(K-1)D .LE. W .LT. A+KD,
C     Z IS ADDED TO X(K). IF W FALLS OUTSIDE ARRAY RANGE IT CONTRIBUTES
C     TO THE NEAREST END CELL.
      DIMENSION X(N)
      K=IFIX((W-A)/D)+1
      IF(K) 2,2,1
    1 IF (N-K) 3,4,4
    2 K=1
      GO TO 4
    3 K=N
    4 X(K)=X(K)+Z
      RETURN
      END
      FUNCTION INTR(X)
      COMMON /RAND/ R
      I=INT(X)
      IF (X-FLOAT(I)-RANDL(R)) 2,1,1
    1 I=I+1
    2 INTR=I
      RETURN
      END
      FUNCTION RANDG(IX)
      A=0.0
      DO 50 I=1,12
   50 A=A+RANDL(IX)
      RANDG=A-6.0
      RETURN
      END
      FUNCTION RANDL(IX)
      IY=IX*65539
      IF(IY) 5,6,6
    5 IY=IY+2147483647+1
```

```
6     YFL=IY
      YFL=YFL*.4656613 E-9
      IX=IY
      RANDL=YFL
      RETURN
      END
      FUNCTION RANDR(X)
      RANDR    =2.*RANDL(X)-1.
      RETURN
      END
      FUNCTION UMAX(X,L)
      DIMENSION X(1)
      Y=X(1)
      DO 1 I=2,L
      IF(Y-X(I)) 2,1,1
2     Y=X(I)
1     CONTINUE
      UMAX=Y
      RETURN
      END
      FUNCTION ISGN(X)
      ISGN=0
      IF(A.GE.0.) ISGN=1
      RETURN
      END
      SUBROUTINE OUR2
      COMMON /STORE/ NCL,MCP,P(120,50)
      COMMON /RCRD/ DBPR,DHPR,BPREL(50),HPREL(10)
      COMMON /GEOM/ PEI,DISNS,DIAMS,THLIP,SLO,SLI,XCB1,XCB2,XCB3,XCB4,
     X       SLI1,SLI2,XCB0,XCBI,XCB5,YCB1,YCB2,YCB3,YCB4,YCB5,YCB6,YCB7,
     X SLO1,SLO2,SLO3,SLO4,SLO5, SLI3,SLI4, RADIN,DISNI, XNI,YNI,
     X XNO,YNO, XN5,YN5, VOL(25),VHD(25)
      COMMON /CONS/ CU,SIA,SIB,CR,CT,ACCOM,PRES,CLWT(26),TCONS(25)
      COMMON /BEGIN/BR,BG,A,NST
      COMMON /TIME/ T,DT,DTM,TEO,DTSM,TEND,DTRP,TREP
      COMMON /HIST/ FU(10,25),FV(10,25),FW(10,25),CFU(10,25),CFV(10,25),
     1CFW(10,25),UB(25),VB(25),WB(25),UUB(25),VVB(25),WWB(25),CORG(25)
      COMMON /RAND/R,G
      INTEGER BR,BG  ,R,G
      NAMELIST/GEO/ PEI,DISNS,DIAMS,THLIP,SLO,SLI,XCB1,XCB2,XCB3,XCB4,
     X       SLI1,SLI2,XCB0,XCBI,XCB5,YCB1,YCB2,YCB3,YCB4,YCB5,YCB6,YCB7,
     X SLO1,SLO2,SLO3,SLO4,SLO5, SLI3,SLI4, RADIN,DISNI, XNI,YNI,
     X XNO,YNO, XN5,YN5, VOL
      NAMELIST/CON/ CU,SIA,SIB,CR,CT,ACCOM,PRES,CLWT    ,TCONS
      NAMELIST/BEG/ BR,BG,A,NST,R,G
      NAMELIST/TIM/ T,DT,DTM,TEO,DTSM,TEND,DTRP,TREP
      WRITE(6,1)
1     FORMAT('1',49X,'SKINNER 1971'//)
      WRITE(6,GEO)
      WRITE(6,CON)
      WRITE(6,BEG)
      WRITE(6,TIM)
      FLT=T-TEO
C PREPARATION OF HISTOGRAMS FOR OUTPUT
      TBPR=0.
      DO 21 I=1,50
      L=LOC(XCB4,I*DBPR)
      IF(L.LT.11) TBPR=BPREL(I)/FLT+TBPR
21    BPREL(I)=BPREL(I)/(CLWT(L)*FLT*6.28318*(I-0.5)*DBPR)
      THPR=0
      DO 22 I=1,10
      L=LOC(0,I*DHPR)
      THPR=THPR+HPREL(I)/CLWT(L)/FLT
22    HPREL(I)=HPREL(I)/(CLWT(L)*FLT*6.28318*(I-0.5)*DHPR)
      TNI=1.57079*YCB2*YCB2*A*TCONS(1)/VOL(1)/CLWT(1)
      DO 24 L=1,NCL
      FTWT=FLT*CLWT(L)
      DO 23 I=1,10
      FU(I,L)=FU(I,L)/FTWT
      FV(I,L)=FV(I,L)/FTWT
      FW(I,L)=FW(I,L)/FTWT
      CFU(I,L)=CFU(I,L)/FTWT
      CFV(I,L)=CFV(I,L)/FTWT
23    CFW(I,L)=CFW(I,L)/FTWT
      CORG(L)=CORG(L)/FTWT
      UB(L)=UB(L)/FLT
      VB(L)=VB(L)/FLT
      WB(L)=WB(L)/FLT
      UUB(L)=UUB(L)/FLT
      VVB(L)=VVB(L)/FLT
24    WWB(L)=WWB(L)/FLT
C PRINTING OF HISTOGRAMS
      WRITE(6,41)
41    FORMAT(//,30X,'FLUX PROFILE AT ORIFICE',//,' DISTANCE FROM AXIS',
     X  45X, 'NUMBER PER UNIT TIME',//)
      CALL HIST1(HPREL,10,P,2,40,TOTH)
      DO 42 J=1,10
42    P(1,J)=J*DHPR
      WRITE(6,43)  ((P(I,J),I=1,42),J=1,10)
43    FORMAT(10(F10.5,5X,40A1,F10.5/))
      WRITE(6,44)            TOTH,THPR,TNI
44    FORMAT(' TOTAL',48X,F10.5//,' NUMBER/TIME THRU ORIFICE =',F10.5,
     X ', THEORETICAL FIGURE FOR NO SKINNER INTERACTION =',F10.5)
      WRITE(6,45)
45    FORMAT(1H1,36X,'BEAM FLUX PROFILE AT END OF FLOWFIELD',//,
     X ' DISTANCE FROM AXIS',80X,'NUMBER PER UNIT TIME' ,//)
      CALL HIST1(BPREL,50,P,2,100,TOTB)
      DO 46    J=1,50
46    P(1,J)=J*DBPR
      WRITE(6,47) ((P(I,J),I=1,102),J=1,50)
47    FORMAT(50(F10.5 ,100A1,F10.5/))
      WRITE(6,48) TBPR,TOTB
48    FORMAT(/,' NUMBER/TIME IN BEAM',F10.5,75X,'TOTAL',F10.5)
C DISTRIBUTION FUNCTION PART
      DO 72 L=1,NCL
      WRITE(6,6) L
6     FORMAT
     X  (1H1,45X,'VELOCITY HISTOGRAMS FOR CELL ',I4/,' THE FIRST HISTOGR
     XAM TRIPLE REFERS TO MOLECULES WHICH HAVE UNDERGONE NO COLLISIONS I
     XN THE FLOWFIELD,'/         ' THE SECOND TO MOLECULES WHICH HAVE UNDE
     XRGONE AT LEAST ONE COLLISION, AND THE THIRD GIVES THE SUM OF THE F
     XIRST TWO.'/            ' LOWER CELL BOUNDARY VALUES APPEAR DOWN THE
     XLEFT. TIME AVERAGED NUMBER OF MOLECULES APPEARS AT THE RIGHT OF EA
     XCH BAR.'//)
      CALL HIST3(P,FU,FV,FW,L)
```

```
      CALL HIST3(P,CFU,CFV,CFW,L)
      DO 71 I=1,10
      FU(I,L) = FU(I,L)+CFU(I,L)
      FV(I,L) = FV(I,L)+CFV(I,L)
71    FW(I,L) = FW(I,L)+CFW(I,L)
      CALL HIST3(P,FU,FV,FW,L)
      WRITE(6,74) VHD(L)
72    WRITE(6,73) [illegible] ,VVB(L),WWB(L),CORG(L)
73    FORMAT(' MEAN VELOCITY VECTOR',3F10.5,   ' MEAN SQUARE VELOCITY VE
     XCTOR',3F10.5/,          ' TIME AVERAGED NUMBER OF MOLECULES WHICH HAV
     XE UNDERGONE MORE THAN ONE COLLISION IN THE FLOWFIELD ',F10.5)
74    FORMAT( /,' THEORETICAL NO INTERACTION TOTAL =',F10.5//)
      WRITE(6,1)
      RETURN
      END
      SUBROUTINE BAR(P,L,X)
      DIMENSION P(1)
      DATA STAR,BLANK/4H*       ,4H    
      N=IFIX((FLOAT(L)-1.E-6)*X)+1
      DO 1 I=1,N
1     P(I)=STAR
      N=N+1
      DO 2 I=N,L
2     P(I)=BLANK
      RETURN
      END
      SUBROUTINE HIST1(F,N,P,K,L,T)
C F IS THE N-VECTOR WHERE THE FIGURES ARE STORED
C P IS A 120XN ARRAY WHERE THE HISTOGRAM BARS WILL BE SET UP
C K IS THE INITIAL CHAR LOCATION OF EACH BAR
C L IS THE LENGTH OF THE BARS
C T IS THE LOCATION WHERE THE TOTAL WILL BE STORED
      DIMENSION P(120,1),F(1)
      U=UMAX(F,N)
      IF(U) 2,3,2
3     U=1.
2     S=0
      KL=K+L
      DO 1 I=1,N
      Z=F(I)
      S=S+Z
      CALL BAR(P(K,I),L,Z/U)
1     P(KL,I)=Z
      T=S
      RETURN
      END
      SUBROUTINE HIST3(P,A,B,C,L)
      DIMENSION P(120,1), A(10,1),B(10,1),C(10,1)
      DO 4 J=1,10
4     P(1,J)=-1.8+0.3*FLOAT(J)
      CALL HIST1(A(1,L),10,P,2,20,TA)
      CALL HIST1(B(1,L),10,P,23,20,TB)
      CALL HIST1(C(1,L),10,P,44,20,TC)
      WRITE(6,1) ((P(I,J),I=1,64),J=1,10)
1     FORMAT(10(F10.5,5X,20A1,F10.5,5X,20A1,F10.5,5X,20A1,F10.5/) )
      WRITE(6,2) TA,TB,TC
2     FORMAT(7H TOTALS,28X,F10.5,25X,F10.5,25X,F10.5/  )
      RETURN
      END
```

APPENDIX 2

Typical Output

Note: The "Histograms for cell - " is done for each of the 25 cells.

```
DT=0.33241   MAP=  17 107 125 169 185 262 310 424 497 687 732 826 869 890 904 948 96610001000100010001000100010001000
ERROR IN LOC
DT=0.34343   MAP=  21 111 127 202 220 273 314 442 514 676 756 815 847 879 900 938 9631006101010161025102610261026 1026
DT=0.32498   MAP=  16 116 136 229 244 316 350 473 549 699 776 835 877 897 926 953 986104110441054107210771077107810 78
DT=0.33012   MAP=  22 101 127 232 249 328 354 462 532 682 779 819 869 891 923 961 98710431054107111021114111311191119
DT=0.31247   MAP=  22 105 125 193 209 259 286 408 465 613 712 794 820 851 877 920 94710031019103110701083109311011101
DT=0.30550   MAP=  22 118 138 200 219 261 284 389 442 587 672 728 775 802 932 885 911 966 992 9951041105510751094 1095
DT=0.33530   MAP=  31 100 127 207 228 275 299 383 430 561 672 723 780 812 847 888 922 968 986 9981044105710841112 1119
DT=0.40380   MAP=  28 113 135 209 231 304 326 386 432 567 671 726 782 823 860 904 934 99210041013106010711109 11451158
DT=0.43577   MAP=  31 127 163 214 240 311 336 404 439 555 670 745 799 833 869 916 945 98910061015105410701109 11391164
DT=0.28454   MAP=  30 149 168 245 266 311 343 437 466 551 673 753 792 815 850 898 925 965 976 9371015103410721113 1150
DT=0.51370   MAP=  29 123 146 238 259 298 326 405 430 521 661 743 778 801 833 870 904 933 957 963 990100310371086 1129
DT=0.29056   MAP=  19 116 153 244 274 317 340 401 427 494 602 685 733 759 788 830 854 885 901 909 934 953 98810491100
DT=0.43122   MAP=  27 120 145 242 268 323 344 395 424 497 591 671 727 749 772 821 844 874 896 904 929 944 97510441100
DT=0.27430   MAP=  24 125 161 240 249 274 293 345 373 442 549 607 652 692 716 760 781 815 837 846 871 881 903 9701061
DT=0.28063   MAP=  36 124 148 230 245 295 313 346 366 429 551 610 652 685 717 761 781 815 833 839 863 877 902 9681067
ERROR IN LOC
DT=0.29127   MAP=  28  99 130 215 238 282 296 334 355 406 522 603 654 681 710 756 778 814 837 841 864 878 905 9691066
DT=0.27059   MAP=  37 127 151 241 258 307 319 352 369 417 529 603 655 692 712 754 775 812 835 839 859 875 902 9681068
DT=0.48684   MAP=  25 135 175 255 277 321 337 380 393 425 514 586 631 663 696 741 762 800 826 930 850 963 890 9591050
DT=0.37829   MAP=  31 105 133 258 268 331 347 401 417 437 506 567 621 646 677 740 764 808 836 843 858 872 896 9691050
DT=0.33496   MAP=  26 100 138 207 244 327 340 391 405 433 553 607 659 690 720 781 905 852 875 888 898 913 931101010[illegible]
DT=0.26375   MAP=  27 119 145 224 265 327 349 408 423 451 570 632 672 714 746 796 835 878 902 905 920 945 95910411144
DT=0.37233   MAP=  23 108 144 221 243 306 342 407 424 457 564 626 674 699 751 812 845 889 910 920 936 957 97310591162
DT=0.38265   MAP=  24 106 131 182 205 268 297 375 389 425 528 613 654 682 712 765 800 843 873 883 898 929 95310351141
DT=0.39911   MAP=  24 113 147 231 248 295 319 391 408 451 555 630 668 693 727 778 814 856 883 891 911 931 95510531172
ERROR IN LOC
DT=0.36406   MAP=  24 104 133 245 263 320 341 387 398 452 563 626 673 712 747 794 830 875 899 919 931 954 98810911211
DT=0.29700   MAP=  32 109 152 221 245 310 330 381 395 452 578 631 679 707 738 784 818 869 896 909 925 944 97910791213
DT 0.30441   MAP=  29 120 149 220 239 280 308 360 381 430 547 606 646 679 702 747 777 831 860 869 894 914 94410421197
DT=0.25078   MAP=  23 120 149 225 240 283 310 369 395 429 551 608 652 672 716 768 802 947 974 879 897 919 94910491200
DT=0.31977   MAP=  27 115 131 211 233 279 305 348 367 413 541 601 656 687 716 762 793 833 859 877 891 904 93410261186
DT=0.26965   MAP=  31 129 155 248 264 303 326 366 383 439 548 634 693 715 750 799 933 870 893 904 921 936 96910531214
DT=0.25812   MAP=  24 107 146 250 274 315 329 368 385 443 564 642 692 721 763 819 856 885 910 921 936 953 99810721223
DT=0.37864   MAP=  24 107 147 228 248 292 302 333 354 404 502 593 650 677 710 769 801 835 869 877 895 913 94310311165
DT=0.32447   MAP=  27  95 124 196 212 253 266 296 315 355 478 551 608 642 671 728 760 900 837 846 865 879 91710091138
DT=0.37647   MAP=  26 115 152 201 224 258 273 304 321 356 484 559 615 645 670 733 762 801 837 842 858 873 904 9971124
ERROR IN LOC
DT=0.43511   MAP=  35 137 155 240 256 307 324 358 368 394 522 596 653 685 710 767 806 842 871 878 887 900 93510321151
DT=0.34669   MAP=  47 170 189 302 320 371 386 442 448 471 569 630 683 711 742 796 835 875 896 901 913 928 96010361154
```

APPENDIX 2
TYPICAL OUTPUT

Note: The "Histograms for cell - " is done for each of the 25 cells

A2-1

```
&GEO
PFI=  1.1499996    ,DISNS=  7.0000000    ,DIAMS= 0.54999995    ,THLIP= 0.99999964E-01,SLO= 0.92999995    ,SLI= 0.69999999    ,XCBI=
 -1.0000000    ,XCB2= 0.50000000    ,XCB3=  1.5000000    ,XCB4=  3.0000000    ,SLI1= 0.15714276    ,SLI2= 0.78571379E-01,XCB0=
  1.3631897    ,XCBI=  1.9539518    ,XCB5=  1.3631916    ,YCB1= 0.13749999    ,YCB2= 0.27499998    ,YCB3= 0.37499994    ,YCB4=
 0.54999995    ,YCB5= 0.82499993    ,YCB6=  1.3749990    ,YCB7=  1.6427670    ,SLO1=  1.8599997    ,SLO2=  2.7499990    ,SLO3=
 0.78571379E-01,SLO4= 3.11785710    ,SLO5= 0.19642842    ,SLI3= 0.39285708E-01,SLI4= 0.19642852E-01,RADIN=  1.1785698    ,DISNI=
  6.0000000    ,XNI= 0.76188570    ,YNI=-0.81923199    ,XNO=-0.68101281    ,YNO= 0.73227185    ,XN5= 0.19274509    ,YN5=
-0.98124856    ,VOL= 0.25639497E-01, 0.83392024E-01, 0.25675233E-01, 0.70551932E-01, 0.31869690E-01, 0.95608950E-01, 0.77679157E-01,
 0.23303765    , 0.15591383    , 0.46774197    , 0.38058275    , 0.29136091    , 0.73771054    , 0.38943589    , 0.38877684    ,
  2.9284039    ,  1.7104807    ,  2.9427347    , 0.20664322    , 0.83638906E-01, 0.34364694    , 0.19979407    , 0.91639000    ,
  2.5216074    ,  10.137089
&END
&CON
CU=  404.85156    ,SIA= 0.99999964E-01,SIB= 0.70699930E-01,CR=  1.0000000    ,CT= 0.70699996    ,ACCOM=  1.0000000    ,PRFS=
  1.0000000    ,CLWT=  1.0000000    ,  1.0000000    ,  1.0000000    ,  1.0000000    ,  1.0000000    ,  1.0000000    ,
  1.0000000    ,  1.0000000    ,  1.0000000    ,  1.0000000    , 0.25000000    , 0.25000000    , 0.62500000E-01, 0.62500000E-01,
 0.62500000E-01, 0.15625000E-01, 0.15625000E-01, 0.15625000E-01,  1.0000000    ,  1.0000000    ,  1.0000000    ,  1.0000000    ,
  1.0000000    ,  1.0000000    ,  1.0000000    , 0.0    ,TCONS=  18.893417    ,  61.450516    ,  18.919754    ,
  51.988815    ,  23.484360    ,  70.453018    ,  57.240784    ,  171.72246    ,  114.89091    ,  344.67310    ,  70.111633    ,
  53.675018    ,  33.975586    ,  17.935638    ,  17.905273    ,  33.717224    ,  19.694229    ,  33.882233    ,  152.27275    ,
  61.632446    ,  253.22893    ,  146.48874    ,  675.27612    ,  1858.1392    ,  7469.8906
&END
&BEG
BR=      1795,BG=       1795,A=  2.0000000    ,NST=       1000,R=  853718289,G=  972282627
&END
&TIM
T=  12.303652    ,DT= 0.34669107    ,DTM= 0.33333331    ,TEQ=  4.0000000    ,DTSM=  1.0000000    ,TEND=  12.000000    ,DTRP=
  1.0000000    ,TREP=  12.000000
&END

                    FLUX PROFILE AT ORIFICE

DISTANCE FROM AXIS                                        NUMBER PER UNIT TIME

   0.02750     ***************************************        19.51538
   0.05500     ********************************************   22.30328
   0.08250     *************************************          18.67900
   0.11000     *************************************          18.51970
   0.13750     *******************************************    21.52887
   0.16500     *******************************                15.84042
   0.19250     ***********************************            17.47806
   0.22000     ***********************************            17.28505
   0.24750     *****************************                  14.51355
   0.27500     *******************                             9.53759

TOTAL                                                        175.20087

NUMBER/TIME THRU ORIFICE = 137.52980. THEORETICAL FIGURE FOR NO SKIMMER INTERACTION = 17[illegible].07092
```

BEAM FLUX PROFILE AT END OF FLOWFIELD

DISTANCE FROM AXIS	NUMBER PER UNIT TIME
0.03236	1.16674
0.06571	1.55566
0.09857	2.56684
0.13142	2.83352
0.16428	2.98168
0.19713	2.75776
0.22999	2.33349
0.26284	1.94457
0.29570	2.60802
0.32855	2.45630
0.36141	1.44454
0.39425	2.07985
0.42712	1.68011
0.45997	1.38281
0.49283	0.68395
0.52569	0.97856
0.55854	0.70712
0.59140	1.16674
0.62425	0.75681
0.65711	0.98725
0.68996	0.22766
0.72282	0.43414
0.75567	0.44077
0.78853	0.29789
0.82138	0.50003
0.85424	0.48042
0.88709	0.41927
0.91995	0.31920
0.95280	0.24563
0.98566	0.23730
1.01851	0.36341
1.05137	0.42595
1.08423	0.21540
1.11708	0.36570
1.14994	0.27055
1.18279	0.23006
1.21565	0.28769
1.24850	0.18668
1.28136	0.31820
1.31421	0.14769
1.34707	0.37451
1.37992	0.18274
1.41278	0.13770
1.44563	0.14752
1.47849	0.19664
1.51134	0.16668
1.54420	0.21328
1.57706	0.09825
1.60991	0.20448
1.64277	0.44319

NUMBER/TIME IN BEAM 28.54164

TOTAL 43.65318

A2-3

UNIVERSITY OF TORONTO

VELOCITY HISTOGRAMS FOR CELL 4

THE FIRST HISTOGRAM TRIPLE REFERS TO MOLECULES WHICH HAVE UNDERGONE NO COLLISIONS IN THE FLOWFIELD.
THE SECOND TO MOLECULES WHICH HAVE UNDERGONE AT LEAST ONE COLLISION, AND THE THIRD GIVES THE SUM OF THE FIRST TWO.
LOWER CELL BOUNDARY VALUES APPEAR DOWN THE LEFT. TIME AVERAGED NUMBER OF MOLECULES APPEARS AT THE RIGHT OF EACH BAR.

-1.50000	*	0.70648	*	0.84152	**	0.97441
-1.20000	*	0.79896	**	1.01412	*	0.25859
-0.90000	***	2.88463	**	0.95511	*	0.57928
-0.60000	**	1.95595	***	1.39375	**	0.84047
-0.30000	*	0.62503	**********	8.81504	*******************	12.96533
0.0	*	0.0	********************	18.62901	********************	14.70952
0.30000	*	0.0	*	0.08213	**	0.74295
0.60000	**	2.19485	*	0.04806	**	0.88111
0.90000	********************	23.98384	*	0.49769	*	0.28260
1.20000	*	0.0	*	0.37321	**	0.87541
TOTALS		33.14963		33.14960		33.14961

-1.50000	*	0.51958	*	0.0	*	0.14307
-1.20000	*	0.0	*	0.26465	*	0.61572
-0.90000	**	0.67336	*******	4.97926	***	2.00127
-0.60000	**	0.94652	********	5.33958	******	3.97033
-0.30000	******	3.67065	******** *******	11.95767	********************	14.43795
0.0	*************	6.98103	********************	14.99689	********************	14.58814
0.30000	*****************	10.14162	**********	6.87838	*************	9.04415
0.60000	********************	12.34117	*****	3.06251	****	2.27678
0.90000	*******************	11.70168	**	0.86644	**	0.78821
1.20000	***	1.54171	*	0.22189	*	0.65171
TOTALS		48.51729		48.51721		48.51730

-1.50000	*	1.22606	*	0.84152	*	1.11748
-1.20000	*	0.79886	*	1.27877	*	0.87431
-0.90000	**	3.55799	****	5.93437	**	2.38055
-0.60000	**	2.90247	*****	7.23333	****	4.81079
-0.30000	***	4.29568	**************	20.77269	********************	27.40327
0.0	****	6.98103	********************	33.62590	********************	29.29765
0.30000	******	10.14162	*****	6.91051	*******	9.82709
0.60000	**********	14.53602	**	3.11058	***	3.15789
0.90000	********************	35.68552	*	1.36412	*	1.07082
1.20000	*	1.54171	*	0.59510	**	1.52712
TOTALS		81.66693		81.66695		81.66695

THEORETICAL NO INTERACTION TOTAL = 55.20851

MEAN VELOCITY VECTOR 0.57278 -0.00041 0.04086 MEAN SQUARE VELOCITY VECTOR 0.69413 0.17106 0.18060
TIME AVERAGED NUMBER OF MOLECULES WHICH HAVE UNDERGONE MORE THAN ONE COLLISION IN THE FLOWFIELD 33.45020

APPENDIX 3

No Interaction Beam Intensity

We derive here a formula for the beam intensity at points close to the skimmer.

Let D be the nozzle skimmer distance, L be the skimmer-observation point distance and b be the skimmer radius. Consider a particle in the skimmer orifice plane at position (r cos θ, r sin θ) whose velocity has a probability density function proportional to

$$\exp-[(u-1)^2 + (v-r\cos\theta/D)^2 + (w-r\sin\theta/D)^2]/c^2$$

That is, the mean velocity vector points in the direction

$$(1, \frac{r\cos\theta}{D}, \frac{r\sin\theta}{D})$$

The impact position of the particle on the plane normal to the axis distance L downstream of the skimmer then has probability distribution

$$\frac{1}{L^2 d^2}\, e^{-\left(\frac{1}{Lc}\right)^2 [(x-\alpha r\cos\theta)^2 + (y-\alpha r\cos\theta)^2]}$$

where

$$\alpha = \frac{L+D}{D}$$

If the flux through the orifice is Ø particles per unit area then the flux through the point x,y is

$$\emptyset_L(x,y) = \emptyset \int_o^b r\,dr \int_o^{2\pi} d\theta\, \frac{1}{L^2c^2\pi}\, e^{-\left(\frac{1}{Lc}\right)^2 [(x-dr\cos\theta)^2 + (y-dr\sin\theta)^2]}$$

This integral is messy in general, but if we ask only for the centreline flux (x = y = o) we obtain

$$\emptyset_{Lo} = \int_o^b r\,dr \int_o^{2\pi} d\theta\, \frac{1}{Lc^2\pi}\, e^{-\left(\frac{1}{Lc}\right)^2 \alpha^2 r^2}$$

which one can easily evaluate to obtain

$$\emptyset_{Lo} = \left(\frac{D}{L+D}\right)^2 \left(1-e^{-[\frac{b}{c}(\frac{1}{L}+\frac{1}{D})]^2}\right).$$