

AD-A239 231



## MENTATION PAGE

Form Approved  
OMB No. 0704-0188

estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering the collection of information. Send comments regarding this burden estimate or any other aspect of this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Avenue, NE, Washington, DC 20004, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

REPORT DATE

3. REPORT TYPE AND DATES COVERED

THESIS/DESSERTATION

①

## 4. TITLE AND SUBTITLE

Numerical Calculation of Aspiration Efficiency of  
Aerosols into Thin Walled Sampling Inlets

## 5. FUNDING NUMBERS

## 6. AUTHOR(S)

Kevin M. Boyle, Captain

## 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

AFIT Student Attending: University of North Carolina

8. PERFORMING ORGANIZATION  
REPORT NUMBER

AFIT/CI/CIA-91-047

## 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFIT/CI  
Wright-Patterson AFB OH 45433-6583

10. SPONSORING/MONITORING  
AGENCY REPORT NUMBER

## 11. SUPPLEMENTARY NOTES

## 12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for Public Release IAW 190-1  
Distributed Unlimited  
ERNEST A. HAYGOOD, 1st Lt, USAF  
Executive Officer

## 12b. DISTRIBUTION CODE

## 13. ABSTRACT (Maximum 200 words)

DTIC  
ELECTE  
AUG 09 1991  
S D D

## 14. SUBJECT TERMS

## 15. NUMBER OF PAGES

119

## 16. PRICE CODE

17. SECURITY CLASSIFICATION  
OF REPORT18. SECURITY CLASSIFICATION  
OF THIS PAGE19. SECURITY CLASSIFICATION  
OF ABSTRACT

## 20. LIMITATION OF ABSTRACT

NUMERICAL CALCULATION OF INERTIAL ASPIRATION EFFICIENCY OF  
AEROSOLS INTO THIN WALLED SAMPLING INLETS

by

KEVIN M. BOYLE

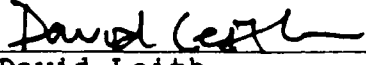
A technical report submitted to the Faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Master of Science in Environmental Engineering in the Department of Environmental Sciences and Engineering.

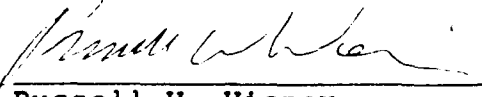
Chapel Hill

1991

Approved by:

  
Michael R. Flynn

  
David Leith

  
Russell W. Wiener

## ABSTRACT

### Numerical Calculation of Aspiration Efficiency of Aerosols into Thin Walled Sampling Inlets.

Efficiency of aspiration of particles from a flowing airstream into a thin walled sampling inlet is accurately predicted using a numerical model. The model combines the Boundary Integral Equation Method for predicting the air velocity field into the inlet with an analytical solution to the particle equations of motion to describe particle trajectories. A two and a three dimensional model are examined. Results for the three dimensional model correlate well with an empirical model of aspiration efficiency. The method can be generalized for a wide range of airstream, sampling inlet, and particle conditions.

Author: Kevin M. Boyle, Capt, USAF

Year: 1991

No of Pages: 119

Degree: Master of Science in Environmental Engineering

Name of Inst: University of North Carolina at Chapel Hill

|                    |                                     |
|--------------------|-------------------------------------|
| Accession For      |                                     |
| NTIS CRA&I         | <input checked="" type="checkbox"/> |
| DTIC TAB           | <input type="checkbox"/>            |
| Unannounced        | <input type="checkbox"/>            |
| Justification      |                                     |
| By                 |                                     |
| Distribution/      |                                     |
| Availability Codes |                                     |
| Dist               | Avail and/or Special                |
| A-1                |                                     |



## TABLE OF CONTENTS

|       |                  |     |
|-------|------------------|-----|
| I.    | BACKGROUND.....  | 1   |
| II.   | THEORY.....      | 7   |
| III.  | METHOD.....      | 13  |
| IV.   | RESULTS.....     | 31  |
| V.    | DISCUSSION.....  | 39  |
| VI.   | CONCLUSION.....  | 48  |
| VII.  | FUTURE WORK..... | 49  |
| VIII. | APPENDIX.....    | 53  |
| IX.   | REFERENCES.....  | 118 |

#### ACKNOWLEDGEMENTS

I would like to thank Dr. Michael R. Flynn, Department of Environmental Sciences and Engineering, for his guidance and constant encouragement throughout the course of this project. I am also grateful to Dr. David Leith and Dr. Russell W. Wiener for their comments and suggestions regarding the content and preparation of this report.

I extend my sincerest appreciation to my wife, Ellen, and daughter, Sarah, for their understanding and support throughout this project.

I would also like to thank the United States Air Force for providing the opportunity and financial support needed to complete my program of study.

This research was funded by the U.S. Environmental Protection Agency under grant NO. 44213.

# NUMERICAL CALCULATION OF INERTIAL ASPIRATION EFFICIENCY OF AEROSOLS INTO THIN-WALLED SAMPLING INLETS

by

Kevin M. Boyle

Unbiased sampling of airborne particulate from a flowing stream requires that the size distribution and concentration of aerosol collected be identical to that of the aerosol in the free stream. Sampling errors occur during aspiration of the aerosol from the free stream to the face of the inlet and during transmission of the aerosol along the sample tube. Additional losses or gains may occur due to particle bounce from the front edge of the sample tube. In this paper, a numerical model for determining the aspiration component of overall sampling efficiency for any arbitrarily shaped thin edged inlet is presented.

## BACKGROUND

Sampling errors due to aspiration occur when particles fail to enter the sampling inlet with the air. Particles have inertia which may prevent them from accelerating, decelerating, or changing their direction fast enough to move with the air (Agarwal and Liu, 1980). Particles also possess additional motion due to gravity. Aerosol aspiration efficiency,  $E_a$ , is defined in equation (1) as the ratio of particle concentration entering the sampling inlet,

$C_i$ , to the particle concentration at an undisturbed upstream location,  $C_o$ , (Belyaev and Levin, 1972).

$$E_a = \frac{C_i}{C_o} \quad (1)$$

One effect on aspiration efficiency would be loss or gains of particles due to bounce off the front of the sampling inlet. By modeling the system as a thin walled sampling inlet, particle rebound effects are assumed negligible and do not contribute to the overall aspiration efficiency (Belyaev and Levin, 1974).

Both numerical modeling and physical experiments have attempted to describe aspiration efficiency. Belyaev and Levin (1972, 1974) used flash illumination to photograph particle trajectories and develop equations to predict inertial aspiration efficiency. Durham and Lundgren (1980), developed aspiration equations from experimental data for aspiration efficiency during anisokinetic sampling with nozzle misalignment and sampler inlet velocity different from oncoming wind velocity. Their equations account for the projected area of the nozzle when it is misaligned. Jaysekera and Davies (1980) and Davies and Subari (1982) also performed experiments and derived aspiration equations which were similar to the equations of Belyaev and Levin.

Extensive measurements of sampling efficiency were made by Tufto and Willeke (1982), and Okazaki, Wiener, and Willeke (1987a, 1987b, 1987c) for both isoaxial and non-

isoaxial aerosol sampling conditions. Based on this data and work cited above, Hangal and Willeke (1990) proposed a unified model to predict aspiration efficiency,  $E_a$ . The general expression, equation(2), is:

$$E_a = 1 + (R \cos \theta - 1)f \quad (2)$$

where  $R$  is the velocity ratio,  $\theta$  is the sampling angle and  $f$  is an inertial parameter. For the case of  $0^\circ$  sampling angle the inertial parameter,  $f$ , is based on the work of Belyaev and Levin (1974) and is shown in equation (3).

$$f = 1 - 1 / [1 + (2 + 0.617/R) Stk] \quad (3)$$

$$R = U_o/U_i$$

$U_o$  is the free stream velocity

$U_i$  is the inlet velocity

$Stk$  is the Stokes number

For  $0^\circ$  to  $60^\circ$  sampling angle the parameter is based on the work of Durham and Lundgren (1980), equation (4).

$$f = \frac{[1 - 1/[1 + (2 + 0.617/R)stk']][1 - 1/[1 + 0.55stk' \exp(0.25stk')]]}{[1 - 1/[1 + (2.617)stk']]} \quad (4)$$

where

$$stk' = Stk \exp (0.022\theta) \quad (5)$$

To model the movement of particles in air it is necessary to model the airflow field and derive a method to track particles placed in the air stream. (Agarwal and Liu,

1980). Numerical models of air flow fields and particle trajectory paths have been developed by Agarwal and Liu (1980), Addlesee (1980), and Dunnett and Ingham (1986, 1988). Agarwal and Liu determined the theoretical flow field into a thin walled sampling inlet considering both particle inertia and gravitational settling. A numerical calculation procedure was used to solve the axisymmetric Navier-Stokes equations and particle motion equations.

Addlesee, (1980) studied the intake efficiency of a Casella cascade impactor. The impactor was modeled as a two dimensional infinite slot with no end effects. A solution to the flow field was developed by assuming that viscous forces were not significant and calculating stream lines using potential theory. From the flow field developed, the particle equations of motion were solved assuming that Stokes law is applicable ( $Re < 1$ ). The equations were put into finite difference form and solved by relaxation.

Dunnett and Ingham (1986) developed a mathematical theory to predict aspiration efficiency of two dimensional blunt bodies using the Boundary Integral Equation Method. Once the velocity field of the system was known, the particle equations of motion are solved assuming that the Reynolds number is always less than 1.0 and thus Stokes' drag could be applied. The equations were put in finite difference form and solved numerically. In 1988 they expanded the two dimensional solution to an axisymmetric blunt body sampler and numerically predicted aspiration

efficiency by particle tracking using finite difference techniques.

Flynn and Miller (1989) used the boundary integral equation method to model air flow into local exhaust hoods. Their solutions follow the same logic as Dunnett and Ingham and give good agreement with analytic solutions for flow into an infinitely flanged rectangular hood with a constant velocity at the hood face, and with other empirical models. The advantage of their model was that it reduced a 3 dimensional air flow solution to a 2D area solution along the boundary of a specified domain.

This research employs the model of Flynn and Miller to estimate the air flow into a sampler inlet, and then applies the solution for the particle equations of motion developed by Alenius (1989) to determine aspiration efficiency. The particle motion equations are described in the THEORY section which follows.

The referenced numerical models to calculate aspiration efficiency were limited by the geometry of the solution domain and employed finite difference techniques to solve the particle equations of motion. By combining the 3D airflow solution of Flynn and Miller with the analytic solution to the equations of motion proposed by Alenius (1989), this model can calculate particle trajectories into a thin edged inlet. This model has a significant advantage over other models as it does not require finite difference discretization to calculate particle trajectories and has

the flexibility to be developed for any arbitrarily shaped domain and sampling inlet over a wide range of particle size and velocity conditions.

## THEORY

To model particle flow, a method for determining the characteristics of the flow field into an arbitrarily shaped and oriented sampling inlet is required. Once the flow field is described, particles can be placed into the field, assigned initial conditions, and tracked. From the trajectories, an estimate of the aspiration efficiency can be made and compared with previous models or experimental data.

The boundary integral equation method (BIEM) is a method of solving a potential flow problem for any arbitrarily shaped domain. The assumptions in the development are inviscid, incompressible, irrotational flow which leads to the following expressions:

$$\text{grad} \cdot \mathbf{V} = 0 \quad (\text{continuity equation for incompressible flow}), \quad (6)$$

and;

$$\text{curl } \mathbf{V} = 0 \quad (\text{irrotational flow}). \quad (7)$$

Irrotational vector fields possess a potential function such that the vector is equal to the gradient of the potential,  $\phi$ , and thus

$$\mathbf{V} = \text{grad } \phi. \quad (8)$$

By combining the above equations

$$\begin{aligned} \text{grad} \cdot \mathbf{V} &= \text{grad} \cdot (\text{grad } \phi) \quad \text{or} \\ \text{grad}^2 \phi &= 0 \end{aligned} \quad (9)$$

Equation (9) is Laplace's equation. By applying the divergence theorem and Green's second identity, an equation can be derived which relates the velocity potential at any singular point (P) to the potential and normal derivatives of potential along the boundary of a specified domain.

The derivation (Liggett and Liu, 1983) yields the following equations which form the basis of the Boundary Integral Equation Method for two and three dimensional domains.

$$2D: \alpha\phi(P) = \int_{\Gamma} [(\phi/R) \delta R/\delta n - ((1/R) \delta\phi/\delta n)] dS \quad (10)$$

$$3D: -\alpha\phi(P) = \int_{\Omega} [(\phi(\delta(1/R)/\delta n)) - ((1/R)\delta\phi/\delta n)] dA \quad (11)$$

- $\phi$  = velocity potential
- $\delta\phi/\delta n$  = velocity normal to the boundary
- $\alpha$  = solid angle about (P)
- $R$  = distance from singular point (P) to a point on the boundary.
- $\Gamma$  = closed curve surrounding and defining the 2D domain
- $\Omega$  = closed surface surrounding and defining the 3D domain
- $dS$  = differential element on  $\Gamma$
- $dA$  = differential area on  $\Omega$

The above equations can be written in discrete form in either two (Liggett and Liu, 1983) or three dimensions (Flynn and Miller, 1989). After specifying either  $\phi$  or  $\delta\phi/\delta n$  at each node on a discretized boundary of  $N$  nodes, equation (10) or (11) is written for each node by successively placing the singular point (P) at each node on the boundary. This results in a system of  $N$  equations in  $N$  unknowns which are solved simultaneously to give the unknown

value of  $\phi$  or  $\delta\phi/\delta n$  at each node on the boundary. Once the boundary values are known, (P) can be located at any internal location and the left hand side of equation (10) or (11) differentiated to give the velocity at the internal point.

Once the velocity field is determined, particles can be placed in the air stream and their motion tracked. This model uses an analytical solution to the particle equations of motion developed by Alenius (1989). The general form of the equation is:

$$d^2\mathbf{r}/dt^2 - [Q(Re)/\tau(\mathbf{u}-(d\mathbf{r}/dt))]-\mathbf{g} = 0 \quad (12)$$

where bold face variables denote vector quantities.

In the above equation, the following notation is used:

$d^2\mathbf{r}/dt^2$  ( $=d\mathbf{v}/dt$ ) is the acceleration of the particle

$Q(Re)$  is a dimensionless factor based on the Reynolds number and the drag coefficient

$\tau$  is the particle relaxation time in air

$\mathbf{u}$  is the air velocity at the particle location

$d\mathbf{r}/dt=\mathbf{v}$  is the velocity of the particle

$\mathbf{g}$  is the earth's gravity

$\mathbf{0}$  is the null vector

$\mathbf{r}$  is the particle position vector

The general equation presented is not analytically solvable. The reason is that the factor  $Q(Re)$  and the air

velocity  $u$  vary along the trajectory of the particle and thus indirectly with time  $t$ . If the air velocity at the points where the particle is successively located can be described as a linear function of the time that has elapsed and  $Q(Re)$  is assumed to be constant over the time interval, an analytical solution to the general equation can be obtained. This condition is approximately met if the motion of the particle is successively calculated using sufficiently small time intervals. A complete derivation is presented in the appendix. The analytic solution is:

$$v(t+t') = u_a t' + u_b - (u_b - v(t)) \exp[-Q(Re)t'/\tau] \quad (13)$$

$$r(t+t') = r(t) + u_a/2 t'^2 + u_b t' - (u_b - v(t)) \tau / Q(Re) (1 - \exp Q(Re)t'/\tau) \quad (14)$$

where  $t'$  is the size of the time step and  $u_a$  and  $u_b$  have been added to simplify the appearance of the equation:

$$u_a = (u(t) - u(t-t'))/t' \quad (15)$$

$$u_b = u(t) - (u_a - g)\tau / Q(Re). \quad (16)$$

Particles can now be tracked in any domain in which the air field is known, and the efficiency of aspiration determined. Aspiration efficiency is calculated as the ratio of particles entering the inlet to particles in the undisturbed flow field far upstream, equation (1). Figure 1 shows a conceptual stream tube of particles flowing in an

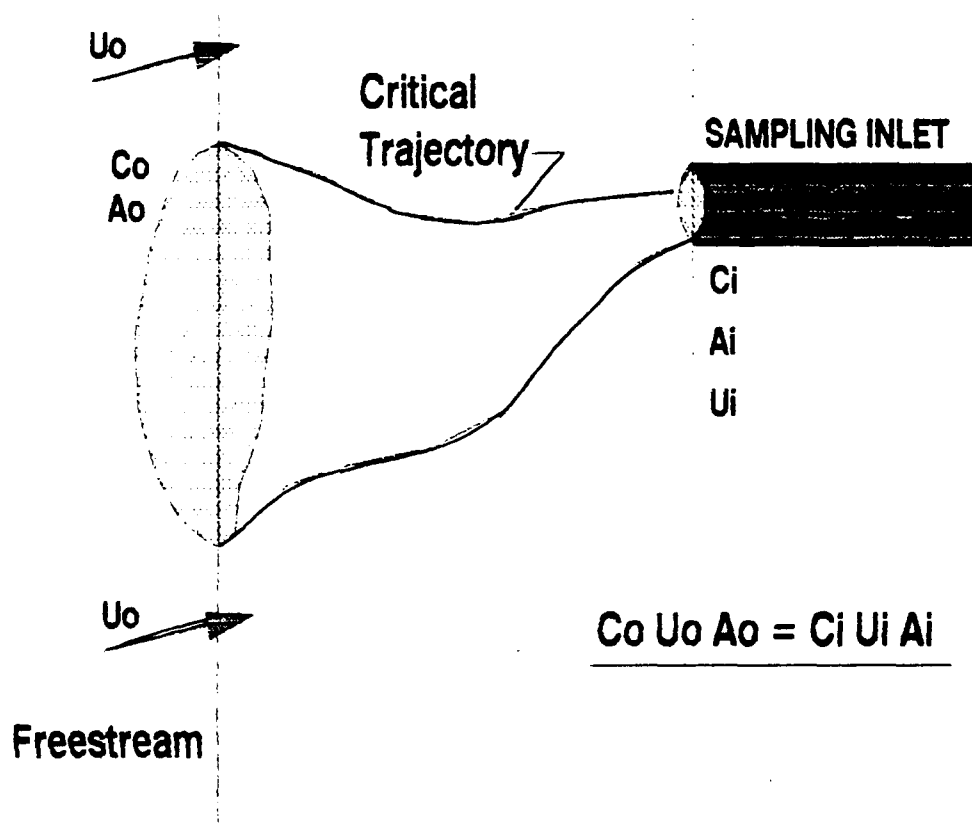


Figure 1. Conceptual Stream Tube. The flux of particles through the upstream plane equals that of particles passing the face of the sampler inlet.

airstream of given velocity,  $U_o$ , into an inlet sampling at a constant velocity,  $U_i$ . The boundaries of the stream tube represent all particle trajectories which just enter the tube. These trajectories are called critical trajectories. Particles located outside this tube would not be drawn into the inlet. The flux of particles passing through the upstream plane of the stream tube is particle concentration,  $C_o$ , multiplied by the stream tube cross sectional area,  $A_o$ , times the freestream velocity,  $U_o$ . By equating the flux of particles through the plane of the stream tube upstream from the inlet to that through the plane of the sampler face,  $C_i A_i U_i$ , the expression in equation (17) is obtained.

$$C_o A_o U_o = C_i A_i U_i \quad (17)$$

From equation (1),  $E_a = C_i/C_o$ , thus:

$$E_a = A_o U_o / A_i U_i \quad (18)$$

In equation (18), the characteristics of the sampler inlet,  $A_i$  and  $U_i$ , are known as well as the free stream velocity,  $U_o$ . By determining the coordinates of the critical trajectory start points in the free stream, an estimate of  $A_o$  and  $E_a$  can be made. This can be done by iteratively assigning particle locations in the free stream and tracking each particle in the domain until the critical trajectories are located.

## METHOD

The model was developed using a simple two dimensional problem which was then expanded to a more complex three dimensional case. Both isoaxial and non-isoaxial inlet orientations were modeled. Non-isoaxial conditions consisted of orienting the inlet at a pitch angle of  $30^\circ$  to the freestream.

### Two Dimensional Model

The two dimensional problem was modeled as an infinite slot. The coordinate system to visualize in two dimensions is an x-y cartesian system with the origin located at the center of the inlet. The positive x direction extends along the centerline of the hood face and the y direction is up and down, with gravity acting in the -y direction.

The air flow into the slot was modeled using the boundary integral equation method which provided a solution everywhere on the boundary of the domain for the potential and the normal derivative of potential. From the boundary solution, the exact velocity components at any internal point were calculated.

A FORTRAN computer program for the two dimensional boundary integral equation solution, written by Liggett and Liu (1983), was used for the 2D solution. The program, called GM8, returned the boundary values and the local air velocity components within the domain needed to calculate particle trajectories. GM8 was combined with a subroutine

written to solve the Alenius solution to the equations of motion, (11) and (12), and track particles within the domain. Figure 2 shows the flow of logic in the program.

The boundary of the domain in the simple 2-D case is shown in figure 3. The origin of the coordinate axes are located at the center of the sampling inlet. The upper lower, front and rear boundaries of the domain are at a distance of 20 inlet diameters from the inlet face. The remaining boundary consists of a rectangular indentation to represent the sampling nozzle of diameter equal to its width.

In figure 3, the boundary conditions are marked. The upper, lower, and rear boundary are situated far enough away from the sampling inlet to have potential due exclusively to the free stream velocity, or:

$$\phi = U_{ox}(x) + U_{oy}(y) \quad (19)$$

where  $U_{ox}$  and  $U_{oy}$  are the magnitude of the free stream velocity components and  $x, y$  are the location coordinates. The boundary condition on the line opposite the hood face is  $\delta\phi/\delta n = U_{ox}$ , the free stream velocity normal to the surface. On the upper and lower edges of the sampling inlet the boundary conditions are  $\delta\phi/\delta n = 0$ , no flow through the walls of the nozzle. At the face of the inlet  $\delta\phi/\delta n$  is equal to sampling velocity,  $U_i$ .

For the situation where an inlet pitch angle was introduced, the domain was not rotated  $30^\circ$  and a new

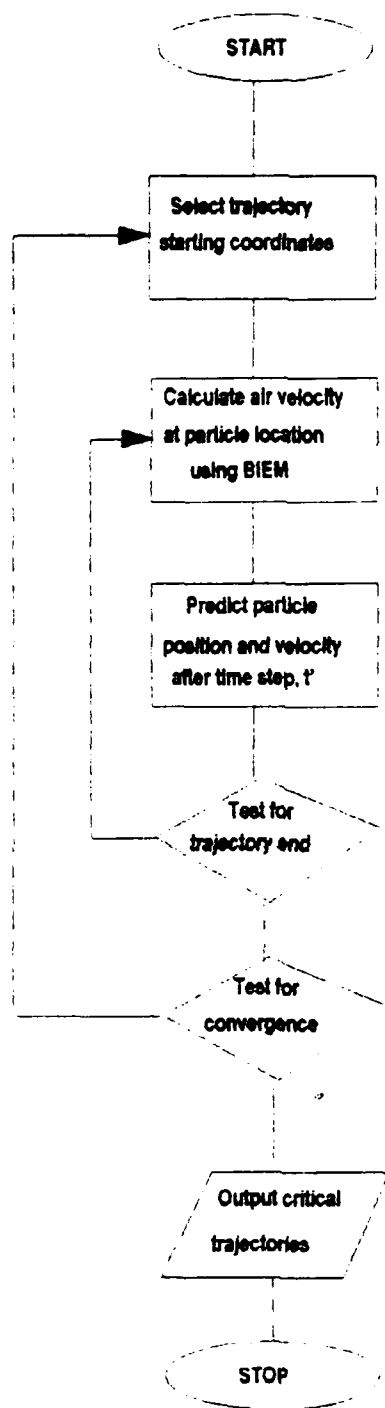


Figure 2. Flow diagram for the two dimension and 3 dimensional program logic.

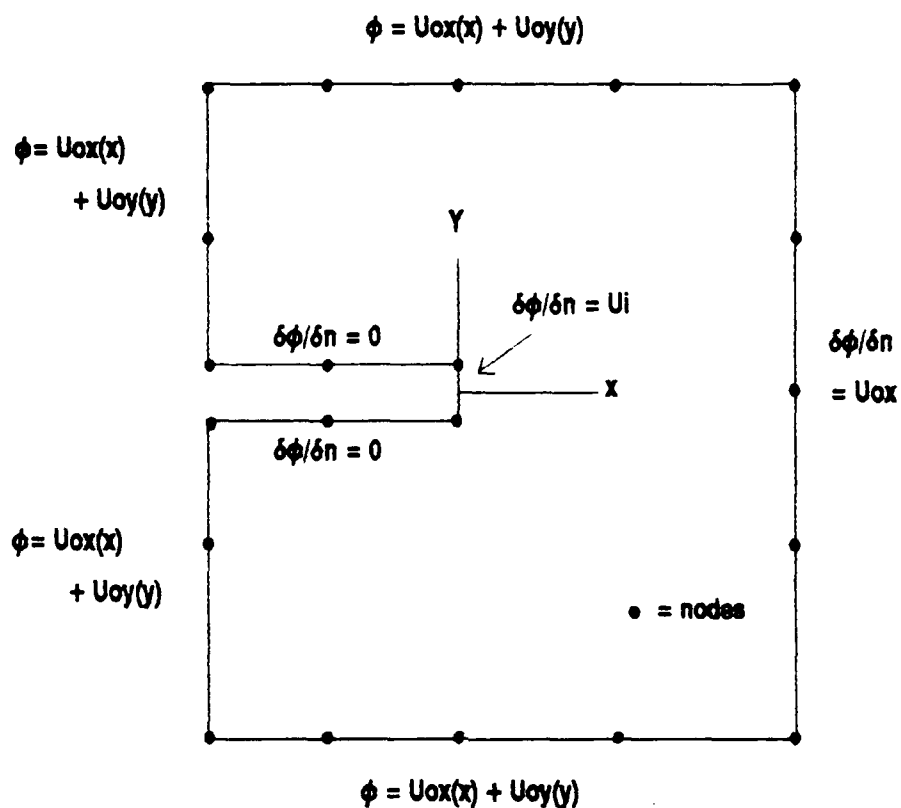


Figure 3. 2D model boundary with coordinate axes, boundary conditions, and sample discretization shown.  $\phi$  = potential,  $\delta\phi/\delta n$  = normal derivative of potential (air velocity).

boundary calculated. Rather boundary conditions were adjusted to represent the freestream,  $U_0$ , as the resultant of the free stream normal to the front boundary,  $U_{0x}$ , and a crossdraft,  $U_{0y}$ . This required a coordinate system conversion where the gravity vector in the x and y direction were  $-g \sin(\theta)$  and  $-g \cos(\theta)$  respectively,  $\theta$  was the angle of misalignment and  $g$  the magnitude of the gravity vector. The origin of this coordinate system was still centered at the inlet face.

The solution required the domain to be discretized into nodal elements as shown in figure 3. Much finer discretization than presented in the example is required because numerical accuracy during calculation of internal points suffers when the internal point is within 1 element length of the boundary.

The initial conditions assigned to the particles in the free stream are that they are falling at their terminal settling velocities and moving in the air stream (Addlesee, 1980) at the velocities calculated by the BIEM solution. Terminal settling velocity ( $V_{ts}$ ) was calculated using Stokes' Law:

$$V_{ts} = \tau g \quad (20)$$

where  $\tau$  is the particle relaxation time (Reist, 1984) and  $g$  is the acceleration due to gravity.

In two dimensions, it was necessary to define the location, in the free, stream of the upper and lower

critical trajectories to calculate aspiration efficiencies. This is because in two dimensions equation (18) reduces to:

$$Ea = d_o U_o / d_i U_i \quad (21)$$

where  $d_i$  is the inlet diameter, and  $d_o$  is the distance between the upper and lower critical trajectories. Internal air velocity coordinates were calculated using BIEM, then the particle position and velocity were predicted using the analytical solution and the process repeated. The critical trajectories were determined by selecting a point far upstream and tracking a particle from this point either into (capture) or past (miss) the inlet. Once a missed and captured trajectory were found the average of the two starting points determined the next point. Convergence occurred when the distance between the particle trajectory start point which is captured and that which escapes is less than a preset value. The program was designed to locate both an upper and lower critical trajectory corresponding to the upper and lower edges of the sampling inlet. The distance,  $s$ , between the upper and lower critical trajectories,  $d_o$ , was determined using the distance formula, equation (22).

$$s = ((x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2)^{1/2} \quad (22)$$

### Three Dimensional Model

In three dimensions,  $U_o$ ,  $A_i$ , and  $U_i$  are given and  $A_o$  is determined to calculate aspiration efficiency, equation (18).  $A_o$  is the area of the trajectory tube at an upstream location where the air flow is not influenced by the

sampling velocity. This area represents the boundary of the stream tube at an upstream location in which particles will enter the sampling inlet (Fig 1).

The shape of the three-dimensional domain is a square sided volume of half-width equal to 20 inlet widths. Extending from the rear wall of the volume into the center of the volume for a distance of 20 inlet widths is a sampling inlet with a square cross section. An equal area square is used to model a circular inlet to simplify the 3D boundary discretization.. The origin of the cartesian coordinate system is located at the center of the inlet. The x direction extends away from the hood face along its centerline. The z coordinate represents the height of the domain and the y coordinate represents the width.

For boundary conditions, the potential is specified on the front, rear, top, bottom and side faces of the 3D domain. The velocity is specified over the inlet face as  $U_i$ , and at the walls of the inlet tube as 0 cm/sec. As in 2D, the boundaries of the 3D domain are situated far enough away from the inlet to have potential due exclusively to the free stream:

$$\phi = U_{ox}(x) + U_{oz}(z) \quad (23)$$

Here,  $U_{ox}$  represents the velocity perpendicular to the front face of the domain and  $U_{oz}$  is the crossdraft vector used when the sampling angle did not equal zero. As in 2D, the gravity components in the coordinate system must be adjusted when a crossdraft is used to model the sampling angle.

The flow field is defined using the same theory as in two dimensions but using a FORTRAN program written by Flynn and Miller (1989) to solve the BIEM problem for three dimensions.

Particle trajectories were calculated in the same manner as in two dimensions with BIEM providing the local air velocity components at the particle location. The particles were assigned initial conditions as described in the 2D section and the equations of motion, with a third dimension added, were solved to predict the particle's next position and velocity. The program logic is the same as that shown in figure 2.

Because the domain is now in three dimensions, the region cannot be discretized into lines between nodes, rather, the boundary must be divided into areas. Flynn and Miller (1989) used triangular elements on the boundary. Examples of the discretization input files for the boundary are presented in the appendix.

The critical trajectories in 3D were determined by selecting a point far upstream and tracking a particle from this point either into the inlet (capture) or out of the domain (miss). The start point is a specified number of inlet diameters,  $XDIN$ , in front of the inlet and in the direction of the resultant velocity vector in the  $x$ - $z$  plane. For  $0^\circ$ , this point is along the  $x$  axis at  $x=XDIN$ ,  $y=0$ ,  $z=0$ , because  $U_o = U_{ox}$  for this condition. For all other angles

the first trajectory start point, figure 4, is calculated as:

$$\begin{aligned} x &= XDIN \cos(\theta) \\ y &= 0.0 \\ z &= XDIN \sin(\theta) \end{aligned} \quad (24)$$

The rationale for choosing this start point is to obtain a capture trajectory around which starting points for the critical trajectories can be centered. If this first point does not yield a capture, the start point is incrementally adjusted up or down along a line perpendicular to the free stream and the trajectory is recalculated. The process is repeated until a first capture identified.

Once a capture trajectory is found, a series of start points at which to begin critical trajectories are defined at a distance far enough away from the coordinates of the first capture to have a high probability of a miss. These points are located a specified distance, AA, from the coordinates of the first capture ( $x_1, y_1, z_1$ ) in incremental degree steps,  $\alpha$ , around a semicircle in the plane perpendicular to the free stream, figure 5. Critical trajectories for only one half the area were calculated due to the symmetry of the problem.

The equations for these points are:

$$\begin{aligned} x &= x_1 + (AA \sin(\theta) \sin(\alpha)) \\ y &= AA \cos(\alpha) \\ z &= z_1 + (AA \cos(\theta) \sin(\alpha)) \end{aligned} \quad (25)$$

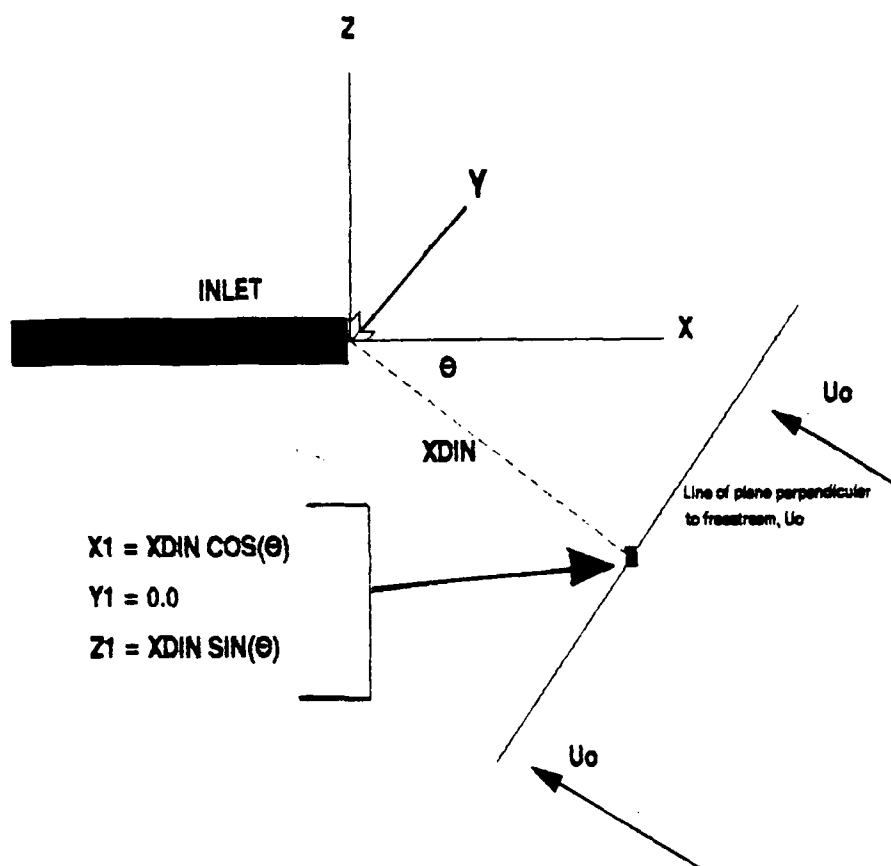
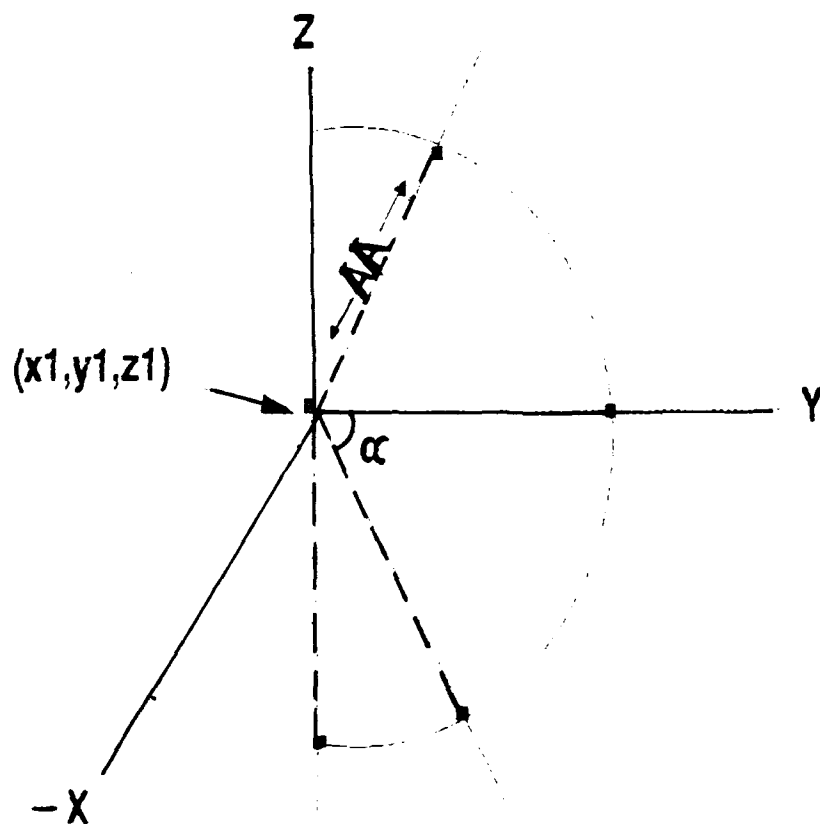


Figure 4. Location of initial trajectory start point.  $(X1, Y1, Z1)$  is initially placed at a distance  $XDIN$  in front of the inlet along a line perpendicular to  $U_0$  in the X-Z plane.



■ Trajectory start points

$$x = x1 + (AA \sin(\Theta) \sin(\alpha))$$

$$y = AA \cos(\alpha)$$

$$z = z1 + (AA \cos(\Theta) \sin(\alpha))$$

$\Theta$  = angle of misalignment

Figure 5. Trajectory start points are located on a semicircle centered around  $(X1, Y1, Z1)$  at degree increments,  $\alpha$ , along rays a distance  $AA$  from the center.  
 $(-90^\circ \leq \alpha \leq 90^\circ)$

If the start points defined in equation (25) result in a capture, the start point is incrementally adjusted farther from the initial capture point and the trajectory is recalculated until a miss identified.

Each first capture and initial miss coordinate combination results in a ray the end points of which define the upper and lower bounds of the critical trajectory and all rays are contained in the same plane perpendicular to the free stream. From this point, the starting coordinates of the last miss and last capture trajectory are averaged to determine the coordinates of the next start point on the ray. The process is repeated until the convergence criteria was satisfied. Critical trajectories are found for each starting point around the semicircle.

The output of the particle tracking routine is a series of points representing the critical trajectories which define  $A_o$ . The area,  $A_o$ , is calculated numerically with a FORTRAN program using the trapezoid rule. From  $A_o$ ,  $E_a$  is calculated using equation (18).

The models, both 2D and 3D, were verified by comparing the computed aspiration efficiency with the unified aspiration model proposed by Hangal and Willeke (1990), equation (2).

#### The Programs

The two dimensional program was compiled and run using Microsoft Fortran version 2. The entire aspiration efficiency calculation was run using one program. This

program initially calculates the boundary solution for the domain then calls subroutines to calculate particle velocity and position, and internal air velocity components at the particle's location. All 2D simulations were run on a Dell 386 personal computer and took approximately 15 minutes to complete. Each program run calculates approximately 30 separate trajectories with approximately 100 points calculated per trajectory.

Figure 6 shows an example of the output of the 2D program with particle positions plotted as trajectories. The particle trajectories were plotted using a simple routine written in BASIC (see appendix). The distance between the upper and lower critical trajectories was the basis of determining the aspiration efficiency as defined in equation (22). A commented version of the 2D program is contained in the appendix.

The 3D program is much more complex than the 2D version because of the addition of a third dimension and the resulting complexities of solving the BIEM equations. Because of this, 3D simulations were run on a Convex Super Computer. The 3D simulations were run using two FORTRAN programs. The first program, 4PTGQ.F, is the BIEM boundary solution program. The input for this program was a boundary node and element file for each combination of sampling velocity,  $U_i$ , wind velocity,  $U_o$ , and sampling angle,  $\theta$ . The input file contained 367 nodes,  $N$ , leading to  $N$  linear equations and  $N$  unknowns which were solved using a four

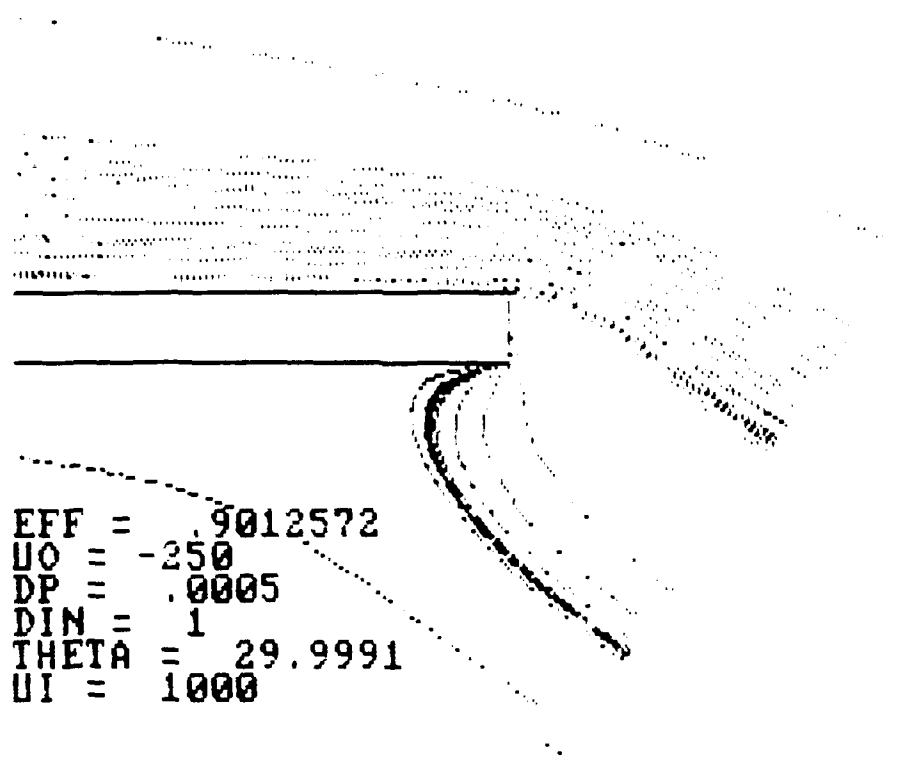


Figure 6. Sample of 2D trajectory plots. Trajectories originate in the plane of the freestream perpendicular to  $U_o$ .

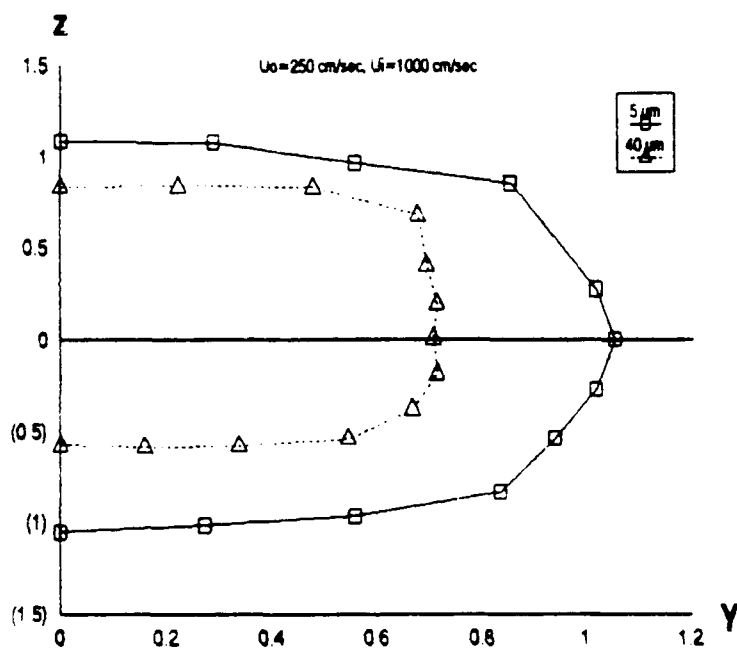
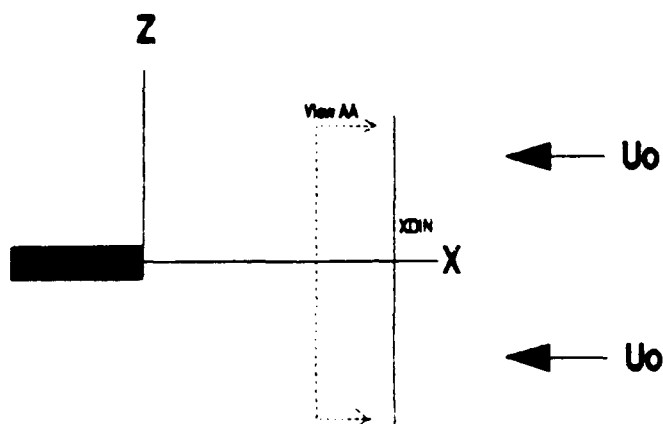
point gaussian quadrature. This program took 5.5 minutes to run on the Convex and returned potential and velocity values for each node on the boundary of the domain.

Once the output to 4PTGQ.F was received, it was imported as input to the 3D particle tracking program, TRK15.F. This program consists of a driver to specify particle location, test for convergence, and call subroutines to perform the BIEM internal air velocity, particle position, and particle velocity calculations. TRK15.F was designed to calculate 13 trajectories at  $15^\circ$  ( $\alpha = 15^\circ$  in equationkk (25)) increments around a semicircle surrounding the initial capture point (see figure 5). This leads to the calculation of approximately 10 trajectories per starting point, over the 13 starting points specified by the degree increment,  $\alpha$ , with 20-30 points along each trajectory to determine the fate of the particle. A typical combination of program variables shown below took 68 minutes of computer time on the Convex:

|                        |            |                 |
|------------------------|------------|-----------------|
|                        | $U_o =$    | 500 cm/sec      |
|                        | $U_i =$    | 250 cm/sec      |
| particle diameter =    |            | 5 $\mu\text{m}$ |
|                        | $\theta =$ | 30 degrees      |
| time step =            |            | 0.0005 seconds  |
| convergence criteria = |            | 0.05 cm         |

The output of TRK15.F is a series of points which defines a region in the freestream inside of which all particles are aspirated to the face of the sampling inlet. Figures 7 and 8 show an example of the 3D model output.

## 3D Isoaxial Sample Output



Plot of critical trajectories  
in freestream, view AA.

Figure 7. Plot of critical trajectory start points in their starting plane. The area inside the curves is the estimate of  $A_0$  in equation (14). The area is assumed symmetrical about  $Z = 0.0$ .

## 3D Non-Isoaxial Sample Output

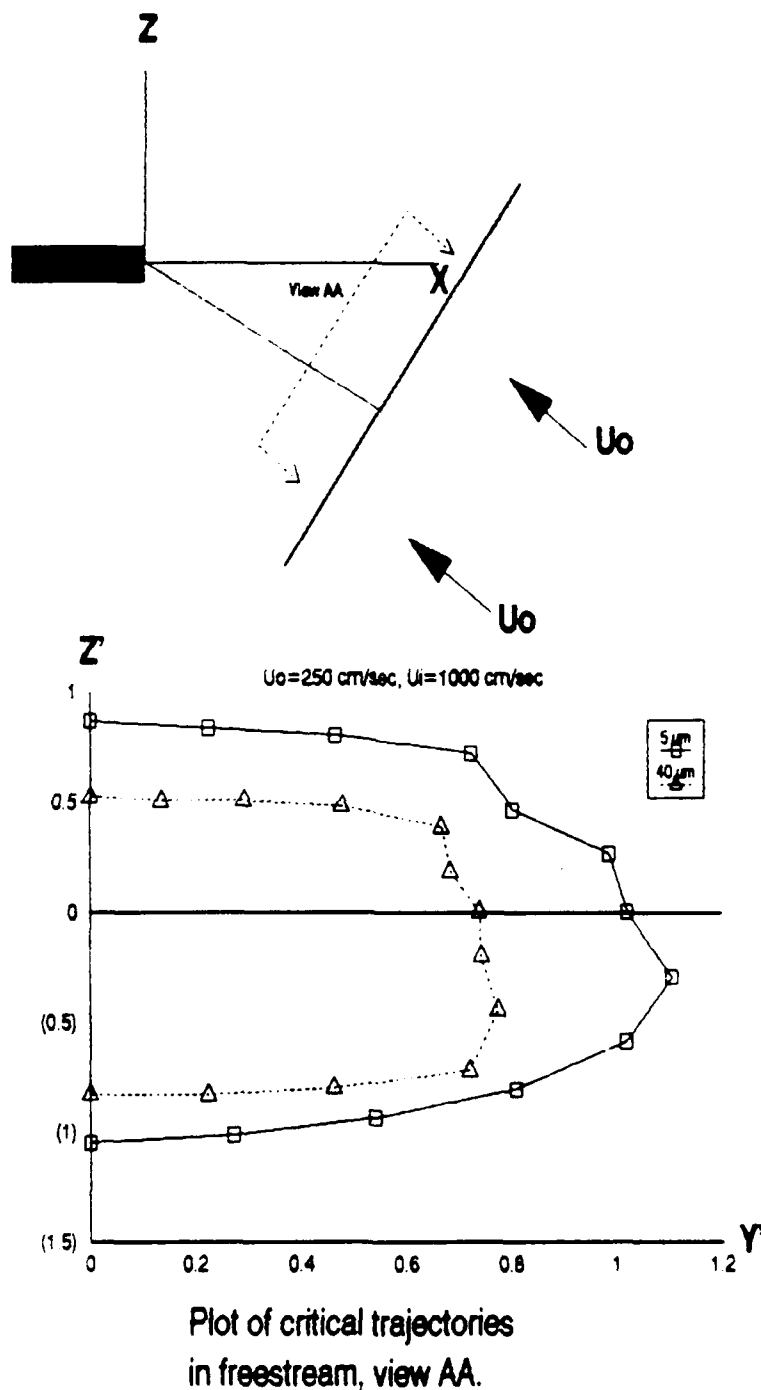


Figure 8. Plot of critical trajectory start point in their starting plane. The area inside the curves is the estimate of  $A_0$  in equation (14).  $Z'$ ,  $Y'$  represent the local coordinate system centered at  $(X_1, Y_1, Z_1)$ .

Figure 7 is an example of output for an isoaxial set of conditions. The plot consists of the location of the critical trajectories in the y-z plane at a constant value of x (distance in front of the sampling inlet) equal to XDIN in equation (24). In this case, x is constant because the sampling angle is 0° and the freestream is flowing in a direction perpendicular to the plane at  $x = XDIN$ .

The plot in figure 8 represents the area in the freestream,  $A_o$ , which is perpendicular to the freestream resultant velocity for the case when the sampling angle was 30°. In this case, a simple plot of the y-z coordinates would not represent the actual area of  $A_o$  as x was not constant over the plane of interest. To solve this problem the critical trajectory coordinates were converted to a local coordinate system centered around the coordinates of the initial capture point. First the distance, s, between the first capture point and the critical trajectory coordinate on each of the 13 rays defined by  $\alpha$ , the incremental angle, was calculated using the distance formula, equation (22). After the distances between the points were calculated, the x and y coordinates of the critical trajectories determined using the following relations:

$$Y' = s \cos \alpha \quad (26)$$

$$Z' = s \sin \alpha$$

The local coordinates were then used to numerically calculate the area using the FORTRAN program, AREA.F.

## RESULTS

Several combinations of free stream velocity,  $U_o$ , inlet size,  $d_i$ , inlet sampling velocity,  $U_i$ , sampling pitch angle (misalignment to the free stream), and particle aerodynamic diameter,  $d_{ae}$ , were run through the 2D and 3D programs.

Table 1 summarizes the input variables which coincide with input parameters used by Wiener (1987).

TABLE 1  
Model Input Parameters

| $d_i$<br>cm | $U_o$<br>cm/sec | $U_i$<br>cm/sec | $d_{ae}$<br>micron | theta<br>degrees |
|-------------|-----------------|-----------------|--------------------|------------------|
| .32         | 250             | 125             | 5.0                | 0                |
| 1.0         | 500             | 250             | 40.0               | 30               |
|             | 1000            | 500             |                    |                  |
|             |                 | 1000            |                    |                  |

The two dimensional program was run for 96 combinations of particle size ( $d_{ae}$ ), velocity ratio ( $U_o/U_i$ ), sampling angle ( $\theta$ ), and inlet diameter ( $d_i$ ). The results are summarized in the appendix.

A linear regression of the BIEM predicted aspiration efficiency on the empirical aspiration efficiency calculated from the unified aspiration model, equation (2), was performed for all 2D data collected, for all  $0^\circ$  combinations, and all  $30^\circ$  combinations. The results are summarized in table 2. Figures 9, 10, and 11 are plots of

the 2D aspiration simulation versus the unified aspiration model, equation (2).

TABLE 2

Two Dimensional Model Linear  
Regression Results

| CATEGORY | SLOPE | Y INT | LCL (1) | UCL (2) | R <sup>2</sup> (3) |
|----------|-------|-------|---------|---------|--------------------|
| All Data | 0.842 | 0.074 | 0.788   | 0.895   | 0.917              |
| All 0°   | 0.963 | 0.077 | 0.941   | 0.986   | 0.993              |
| All 30°  | 0.664 | 0.118 | 0.599   | 0.729   | 0.911              |

- (1) LCL - 95% lower confidence limit for slope  
 (2) UCL - 95% upper confidence limit for slope  
 (3) R<sup>2</sup> - correlation coefficient

The 3D results are presented in the appendix and summarized in figures 12, 13, and 14. Table 3 contains the linear regression results which compare the numerical prediction of aspiration efficiency with the unified model of Hangal and Willeke, 1990.

TABLE 3

Three Dimensional Model Linear  
Regression Results

| CATEGORY | SLOPE | Y INT  | LCL (1) | UCL (2) | R <sup>2</sup> (3) |
|----------|-------|--------|---------|---------|--------------------|
| All Data | 1.062 | -0.062 | 1.026   | 1.099   | 0.995              |
| All 0°   | 1.059 | -0.026 | 1.036   | 1.082   | 0.999              |
| All 30°  | 1.064 | -0.096 | 0.980   | 1.148   | 0.991              |

- (1) LCL - 95% lower confidence limit for slope  
 (2) UCL - 95% upper confidence limit for slope  
 (3) R<sup>2</sup> - correlation coefficient

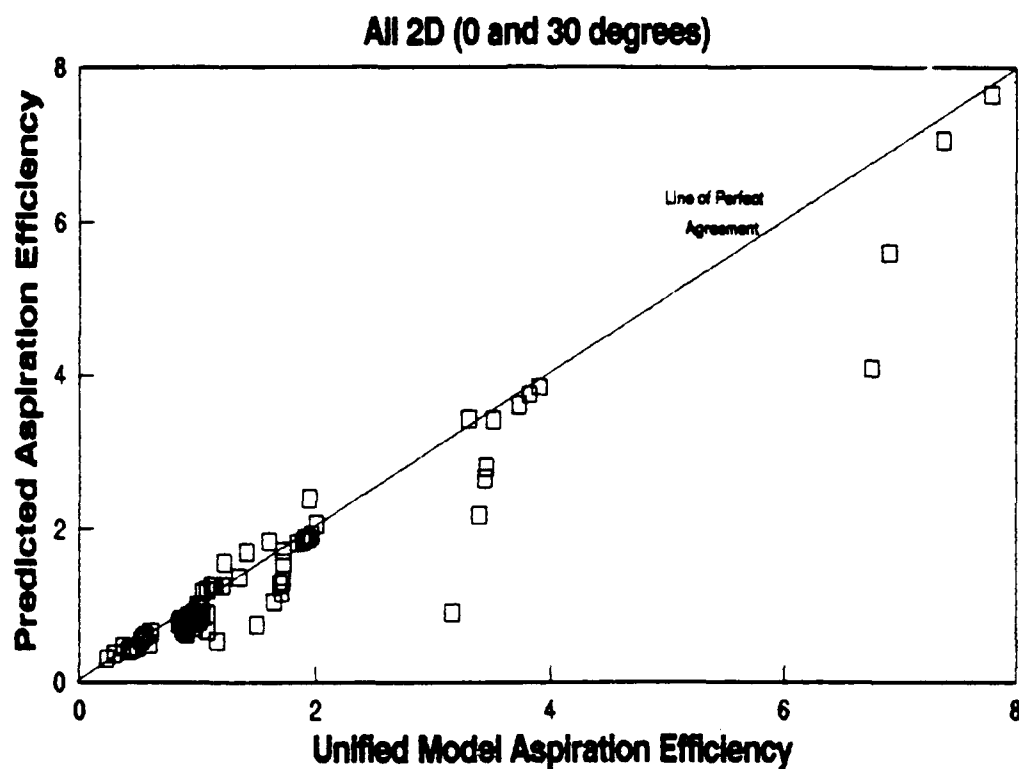


Figure 9. 2D results, all data. Plot of the predicted efficiency developed from the BIEM and Alenius theories against the empirical unified aspiration efficiency model of Hangal and Willeke (1990).

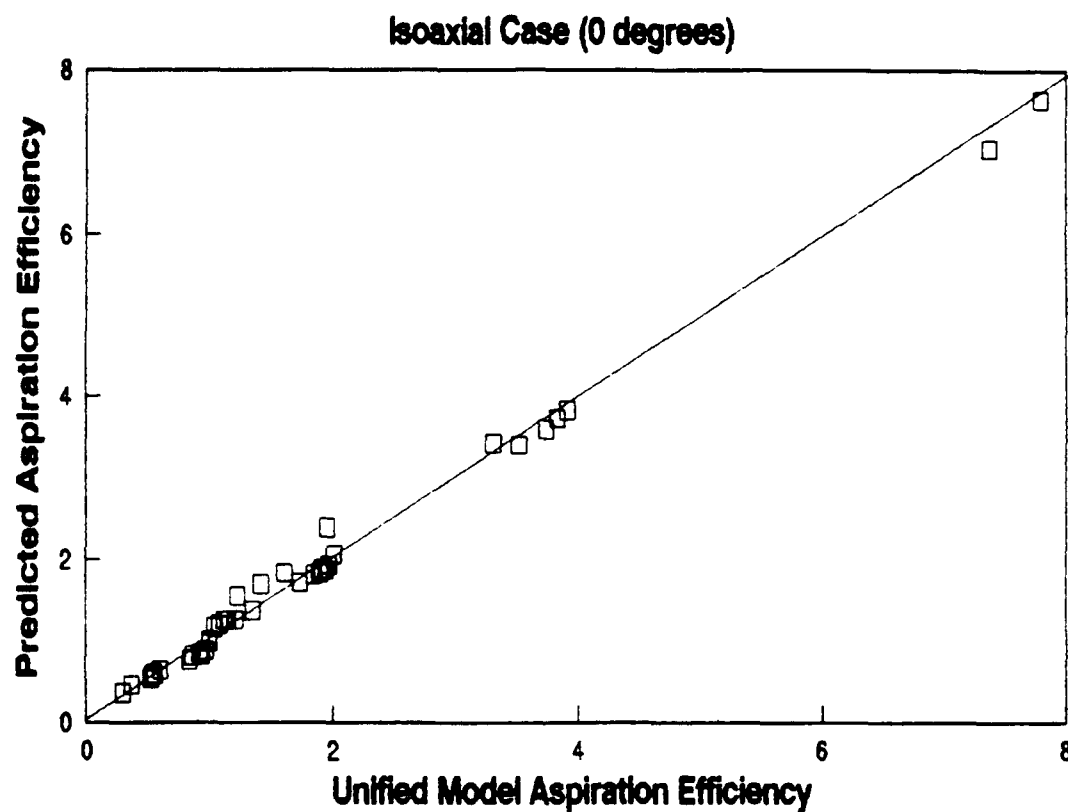


Figure 10. 2D results, isoaxial data. Plot of the predicted efficiency developed from the BIEM and Alenius theories against the empirical unified aspiration efficiency model of Hangal and Willeke (1990).

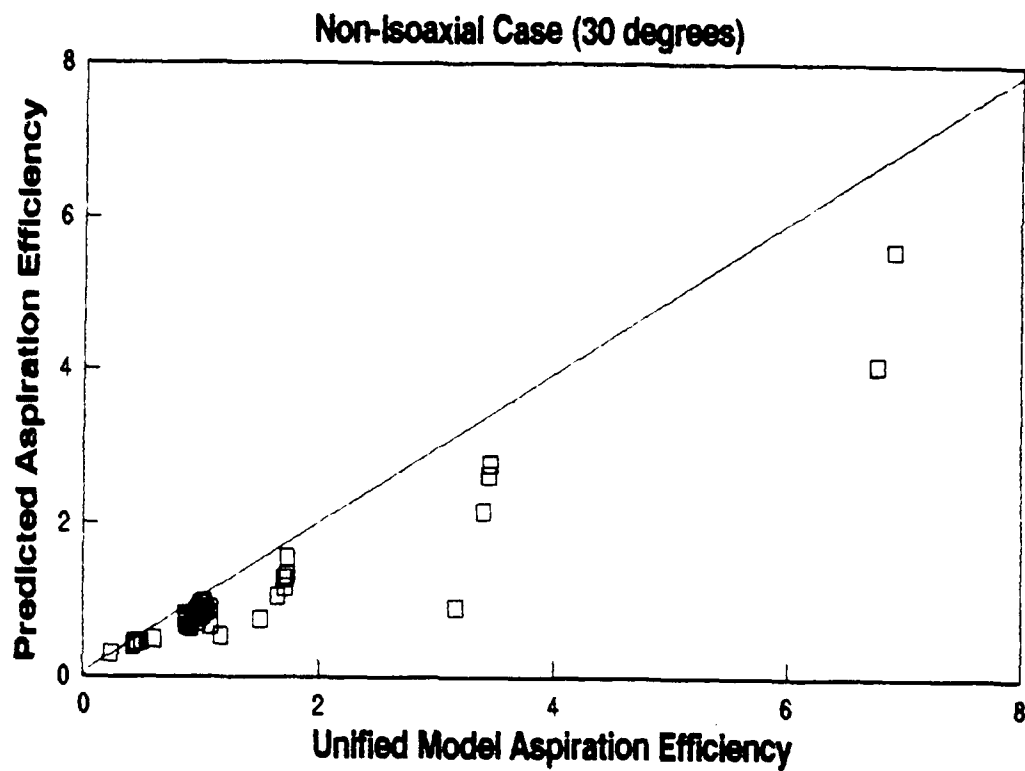


Figure 11. 2D results, non-isoaxial data. Plot of the predicted efficiency developed from the BIEM and Alenius theories against the empirical unified aspiration efficiency model of Hangal and Willeke (1990).

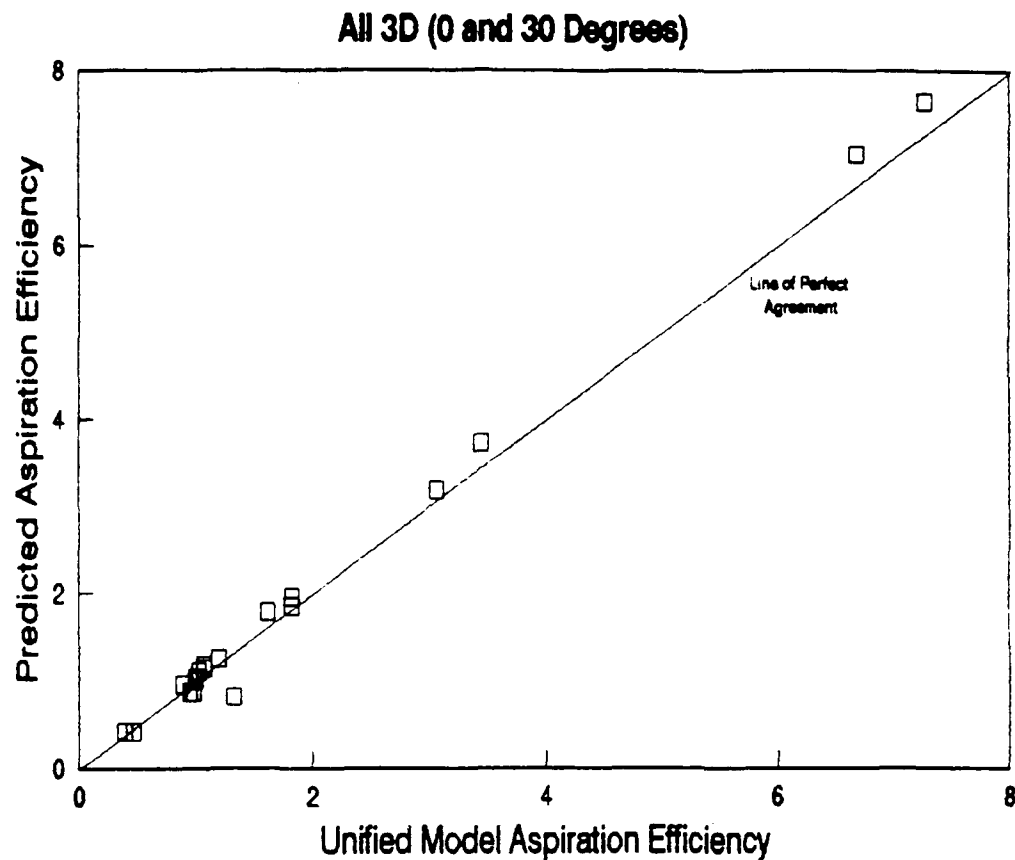


Figure 12. 3D results, all data. Plot of the predicted efficiency developed from the BIEM and Alenius theories against the empirical unified aspiration efficiency model of Hangal and Willeke (1990).

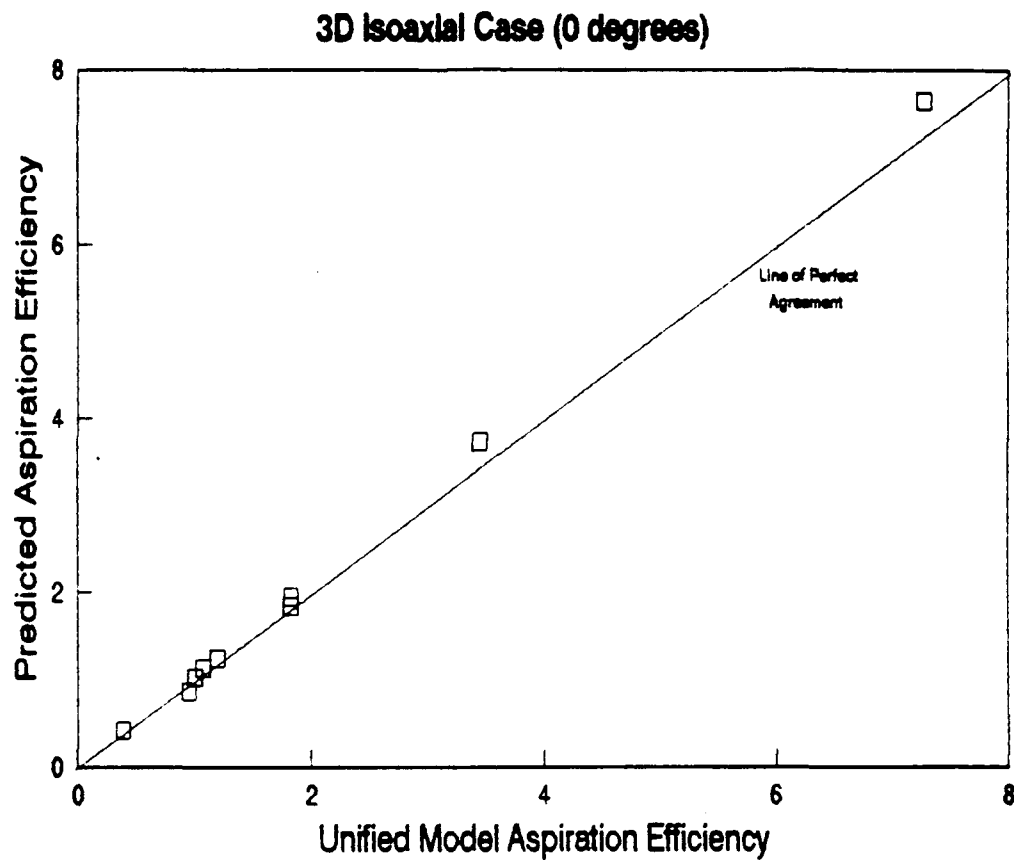


Figure 13. 3D results, isoaxial data. Plot of the predicted efficiency developed from the BIEM and Alenius theories against the empirical unified aspiration efficiency model of Hangal and Willeke (1990).

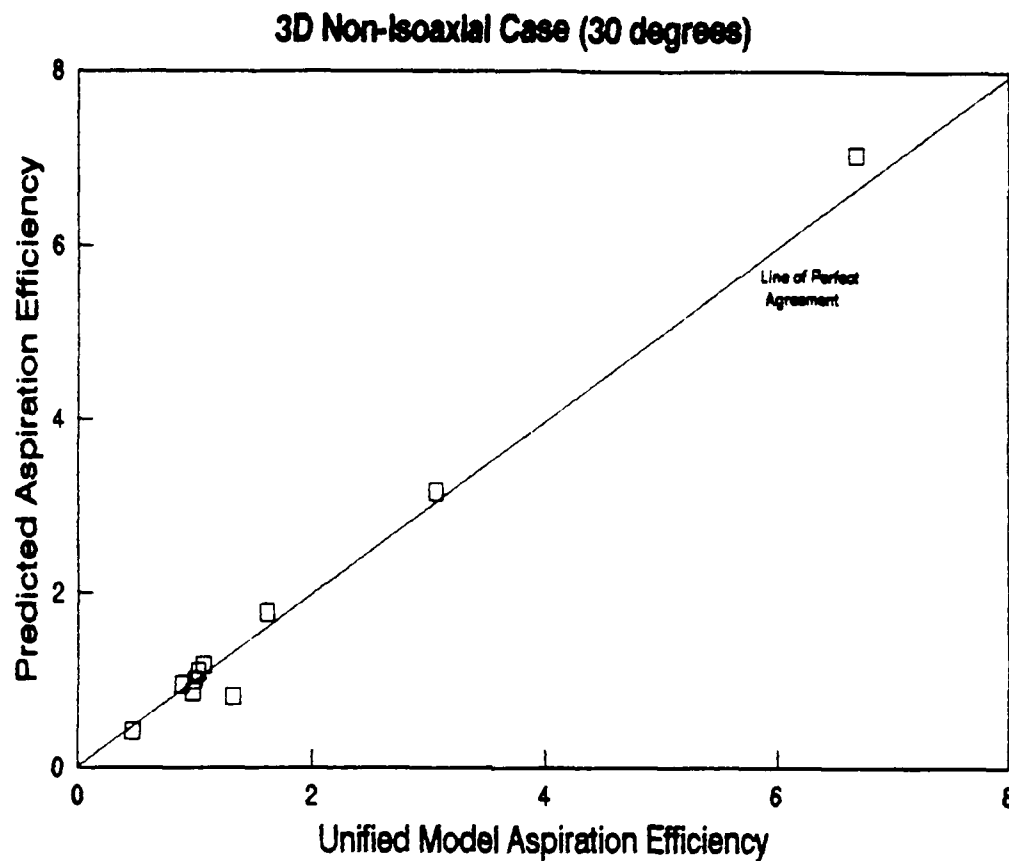


**Figure 14.** 3D results, non-isoaxial data. Plot of the predicted efficiency developed from the BIEM and Alenius theories against the empirical unified aspiration efficiency model of Hangal and Willeke (1990).

## DISCUSSION

The two dimensional model showed good agreement with the unified aspiration model for the isoaxial conditions. From table 2, the slope of the regression line for the  $0^\circ$  sampling angle is 0.96. A slope of 1.0 would indicate perfect agreement with the unified model. The largest error in the 2d model occurred when the velocity ratio was less than 1. In this case, the inlet velocity was 2 to 4 times greater than the wind velocity leading to large changes in air velocity and direction as the particles approached the inlet face.

The nonisoaxial 2D model was tested for a  $30^\circ$  sampling angle and showed poor agreement with the unified model. The percent difference between the unified model aspiration efficiency and the predicted aspiration efficiency was as large as -71%. The slope of the regression line for this set of conditions was .66.

Overall, the 2D model fairly represented the aspiration efficiency of aerosols into a thin edged inlet. The regression equation, slope of 0.84, was biased due to the poor agreement of the  $30^\circ$  model with the unified model.

As shown in table 3, the three dimensional solution performed significantly better in all categories when compared to the 2D model. The slopes of the regression lines were close to 1.0 for all cases indicating that the computer model accurately calculates aspiration efficiency

values when compared to the unified aspiration efficiency model. Table 4 shows a comparison of the 2D percent error with that of the 3D model for the same combinations of velocity ratios, sampling angle and particle diameter. As can be seen, in most cases, the 3D model led to an improvement in the aspiration efficiency value.

TABLE 4  
Comparison of 2D Model Error  
with the 3D Results

| UO   | UI   | THETA | DP | 2D ERR | 3D ERR |
|------|------|-------|----|--------|--------|
| 1000 | 125  | 0     | 5  | 22%    | 2%     |
| 1000 | 125  | 0     | 40 | 4%     | 5%     |
| 250  | 1000 | 0     | 5  | 12%    | -9%    |
| 250  | 1000 | 0     | 40 | 26%    | 9%     |
| 1000 | 125  | 30    | 40 | -39%   | 5%     |
| 250  | 1000 | 30    | 5  | -11%   | -11%   |
| 250  | 1000 | 30    | 40 | -3%    | -10%   |
| 500  | 500  | 0     | 5  | 1%     | 3%     |
| 500  | 500  | 30    | 5  | -10%   | 0%     |
| 500  | 250  | 0     | 5  | 11%    | 6%     |
| 500  | 250  | 30    | 5  | -6%    | 7%     |
| 500  | 125  | 0     | 5  | 26%    | 4%     |
| 500  | 125  | 30    | 5  | 17%    | 10%    |

Some error was introduced into the above comparison because the unified model of aspiration efficiency was developed for circular, thin edged inlets. This may have some effect on the comparison of calculated efficiency with the aspiration efficiency derived from the computer. This does not, however, affect the model calculation of efficiency because the actual areas using the square inlet were used in equation (18).

Boundary discretization, particle start point, convergence criteria, and time step ( $t'$ ) were the most important parameters in determining accurate solutions. Boundary discretization had to be dense enough so that calculations of internal velocities were performed at distances greater than the distance between nodes on the nearest boundary (Flynn and Miller, 1989). This was accomplished by developing a 2D boundary consisting of 75 nodes with dense discretization near the inlet face and along the inlet sides. The appendix contains a sample input file for the 2D boundary discretization. With this discretization, few aberrations in internal velocity calculations were seen. This was not the case for the 3D solution.

The three dimensional model boundary discretization was proportional to the 2D discretization with the sides of the triangular elements equal in length to the distance between nodes in the 2D discretization. This led to an input file containing 367 nodes defining 693 elements. As described by Flynn and Miller, the triangular elements are defined by numbering the element with the node numbers corresponding to the vertices of the triangular element in a clockwise fashion looking out of the domain. This was done for approximately 1/2 of the elements starting at the inlet face and along the walls of the inlet using a triangular element generating program written for this problem. The remainder of the element input file was fashioned using

LOTUS spreadsheets because of the complex interactions of common nodes along the edges of each facet of the cube and at its corners.

Even with this dense 3D boundary discretization, internal calculation of air velocities at points less than 0.05 cm from the face and walls of the inlet were extremely inaccurate with the values being orders of magnitude larger than the freestream velocity. This led to large errors in the prediction of particle location which is dependent in part on accurate calculation of air velocity at the particle's current location. To avoid this error, capture criteria for the particle was extended by a factor of 0.05 cm in front of the inlet face and around its edges. This in effect enlarged the face area,  $A_i$ , to 1.11 cm<sup>2</sup> for a square inlet of side equal to 1.0 cm. This enlarged area was the area used in the aspiration efficiency equation (18).

The effect of enlarging this area was to alter the actual average air velocity moving into the "inlet". Over the face of the inlet the air velocity was specified as  $U_i$ . Over the thin layer defining the addition to the actual inlet, the average velocity had a value which was a function of the free stream velocity and the sampling velocity. Because of this there is some error in the specification of  $U_i$  in equation (184) and hence  $E_a$ .

Another problem with the 3D boundary discretization involves the interface of a surface of the boundary where the velocity is specified with another surface where the

velocity is also specified, but is different than that of the first surface. In the final model, this occurred only at the nodes surrounding the edges of the face of the sampling inlet. Here, the velocity over the face was specified as the sampling velocity,  $U_i$ , and the velocity on the edge of the tube and along its length was specified as 0 cm/sec. This caused a discontinuity of velocity at the edges of the sampling inlet. The effect of the discontinuity was to cause inaccuracies in interpolation of velocities over the elements involved. This interpolation error was minimized by inserting a very thin row of elements just inside the edges of the inlet. For the final model the width of this small layer was 0.001 cm.

Particle start points were important because the aspiration efficiency calculation requires the critical trajectory start points to be in the free stream. It was found that at distances of 6 inlet diameters from the inlet, local air velocity and direction was within 5% of the specified free stream velocity. This agrees with the findings of Belyaev and Levin (1972) and held true for both 2D and 3D models.

When determining the location of critical trajectories, the distance between the last capture trajectory start point and last miss trajectory start point was compared to the convergence criteria. Once the distance reached a preset criteria, the points were averaged to give the start point of the critical trajectory. The convergence criteria for

the 2D model was set at 0.01 cm after iterations at criteria smaller than this caused a less than 1% change in the calculated efficiency. The convergence criteria for the 3D model led to good agreement with the unified model when set at 0.05 cm.

The time step was the most critical factor in developing accurate particle trajectories. Time steps that were too large resulted in predictions of trajectories that were too coarse to accurately predict the particle movement. Unnecessarily small time steps caused excessive computing time. Through a series of program tests, 0.0003 seconds was chosen as the time step for the 2D model with a further reduction by a factor of 10 once the particle came within 1 inlet diameter of the sampling inlet to account for the large velocity changes near the face of the inlet.

The time step and the convergence criteria for the 3D model were not as sensitive in the determination of the aspiration efficiency as they were in the 2D case. Since the 3D model more realistically represented the physics of the problem, accurate solutions were obtained for time steps ranging from 0.001 to 0.0003 seconds with the time step being halved when the particle was within 1 inlet diameter of the inlet face.

Because the size of the time step indirectly predicts the distance the particle will move; positions predicted when the air is turning and accelerating will not lie on the the actual trajectory. This is clearly seen in the 3D model

for velocity ratios of 2 and 4 where the estimate of  $E_a$  was consistently larger than the unified model prediction. In this case, the air is approaching the inlet faster than the inlet is sampling and must diverge around the inlet.

Particle trajectories will also diverge but the predicted position will lie slightly inside of the actual trajectory. This causes an overestimate of  $A_o$  leading to an increase in the prediction of  $E_a$ .

When running the model under certain conditions, the time step and capture criteria became very critical in determining whether a trajectory resulted in a miss or capture. In the three dimensional model, when  $U_o = 1000$ ,  $U_i = 125$  cm/sec, and  $\theta = 30^\circ$ , the best estimate of aspiration efficiency was off by 39% when compared to the unified model. Under these conditions the air flow is approaching the inlet at 8 times the sampling rate causing significant divergence of the air streamlines around the face of the inlet. This situation combined with the sampling angle results in particle trajectories which move virtually straight up in the positive  $z$  direction at locations within 0.1 cm of the inlet face when the particle is approaching the inlet from above the centerline. The air velocity in this area is also moving very fast. The combination of high velocity particle and steep angle of approach toward the sample inlet can easily lead to a predicted particle location which jumps across the numerical edges of the sampling inlet. A miss trajectory is recorded for this

situation when in actuality a capture should have occurred. This causes an underestimation of  $A_o$  leading to a prediction of  $E_a$  which is too small. This situation occurred for the last four rays around the start point semicircle and is the reason for the poor performance of the model under these conditions. This situation can be eliminated by reducing the size of the time step but would lead to a large increase in computing. The same problem was seen in the 2D model and prevented identification of an upper critical trajectory.

By specifying an incremental angle of  $15^\circ$  in the 3D model, the program was limited to locating 13 critical trajectories. The effect of decreasing  $\alpha$  has not been analyzed but would lead to a finer outline of the critical trajectory stream tube and a more accurate calculation of  $A_o$ . One anomaly in the output was seen when the ray specified coincided with the corners of the inlet. Critical trajectories associated with these rays tended to lie outside range of critical trajectories on either side of this point. In one program test, eliminating these rays from the program improved the model result from a 26% to a 4% error.

The results presented in this paper show that the solutions agree with the unified empirical model of Hangal and Willeke (1990). The advantage of this model is that it can be expanded to situations beyond that investigated. These situations include modeling a yaw angle, multiple inlets, and inlets of different geometry. Modeling a yaw

angle as opposed to a pitch angle would require only a shift in the direction of the gravity component in the currently available boundary file. Modeling of multiple inlets and differing inlet geometries could be accomplished by reworking boundary files to represent the new inlet configurations.

## CONCLUSION

The boundary integral equation method can be used to accurately describe the air flow field into a thin edged inlet. Once the flow is described, Alenius's (1989) analytical solution to the particle equations of motion can be used to locate critical aerosol trajectories and accurately predict aspiration efficiency. A simple two dimensional model described here can accurately predict aspiration efficiency for small particles when the velocity ratio is close to 1. In this case, the critical trajectories are symmetric about the center of the inlet. When an angle of misalignment is introduced and/or the velocity ratios differ significantly from 1.0, a three dimensional model is required to describe critical trajectories and thus aspiration efficiency.

The 3D model can accurately predict aspiration efficiencies which are in good agreement with the unified aspiration efficiency model for all forward sampling angles proposed by Hangal and Willeke (1990). Further refinements to the model should improve the accuracy of aspiration efficiency estimates.

Finally, this model has the capability of predicting aspiration efficiencies for inlets of differing shapes and orientations including multiple inlets, yaw angles, and inlets of differing geometry. By reworking the boundary input files, inlets of any geometry, location, sampling angle and size can be studied using these programs.

## FUTURE WORK

Successful calculation of particle trajectories depends on several adjustable variables in the program including boundary discretization, calculation time step, convergence criteria, and capture criteria. While this paper demonstrates that the method can accurately track particles for given conditions, the above variable have not been tested to determine their optimum values. A study of these variables will help determine the contribution of each to the overall error in the program.

Boundary discretization of the 3D domain required to solve the BIEM problem can be difficult due to the number of triangular elements required. Also, when constructing a closed domain, a number of common nodes appear along the edges of each face of the boundary and at the corners. This can lead to cases where one node is contained in up to 6 distinct triangular elements. Boundary files for this project were manually prepared for about 50% of the domain. Full automation of this process would greatly increase the applicability of this method.

Finer discretization at the face of the inlet and along the walls of the inlet will decrease the distance a particle can come within the inlet and inlet sides before internal air velocity calculations no longer represent the conditions. In this project, that distance was 0.05 cm. This will lead to a significant increase in the number of

nodes and triangular elements required to describe the boundary and may increase the size of the boundary file by as much as 50% and subsequently increase computing time. Finer discretization may be avoided by adding an interpolation scheme to the program. This subroutine would be invoked when the particle entered the volume around the face of the inlet where internal velocity calculations showed large error. While this will cause some error in the prediction of internal velocity, it is expected it will be less than that of assuming a captured trajectory has occurred when the particle enters this area.

Rediscretization of the inlet and walls to more closely represent a circular inlet may also lead to a better comparison of the model with the unified aspiration efficiency equation. This could be done by modelling the inlet as an equal area hexagon or octagon. Automation of the boundary file would be paramount in this case as the discretization would become extremely tedious.

It may be necessary to add a routine to the program to automatically decrease the time step if the distance between successive points becomes large or the particle jumps across the boundaries of the inlet. Alternatively, the path of the particle can be analyzed to see if it crossed the inlet and a routine added to the capture criteria which records a capture when this situation occurs.

The boundary input files and start point routines were designed for positive sampling angles only. The program is

not general enough to support negative (downward pointing) inlet orientations. However, by simply changing the sign on the gravity vector and representing the negative angle as a positive angle, predictions of aspiration efficiency may be obtained.

Combination of the 3D boundary solution program with the particle tracking routine and the area integration routine would further streamline the efficiency calculations. Currently, 3 FORTRAN programs are used to calculate aspiration efficiency.



**APPENDIX**

### Calculation of Particle Trajectories:

Particle transport is influenced by a gravity force and a drag force due to the friction of the surrounding air. By Newton's second law, particle mass times acceleration is equal to the sum of the forces on the particle,

$$\sum \mathbf{F} = m\mathbf{a} \quad (1)$$

where  $\mathbf{F}$  is the force,  $\mathbf{a}$  is the acceleration and bold face quantities represent vectors.

The frictional force can be expressed by the following equation:

$$\mathbf{F}_d = - \frac{C_d(\text{Re})}{CC} \frac{A_p \rho_l \mathbf{v}_r \mathbf{v}_r}{2} \quad (2)$$

where

$C_d(\text{Re})$  is the dimensionless friction factor  
dependent on the Reynolds number

$CC$  is the Cunningham slip correction

$A_p$  is the projected area of the particle

$\rho_l$  is the density of the medium (air)

$\mathbf{v}_r$  is the magnitude of the particle relative  
velocity

$v_r$  is the particle velocity relative to the air.

$C_d(Re)$  is dependent on the particle Reynold's number,

$$Re = \frac{v_r dp \rho_l}{\mu_l} \quad (3)$$

where

$dp$  is the particle diameter (assumed spherical)

$\mu_l$  is the viscosity of the air.

Several expressions for  $C_d(Re)$  can be found in the literature (Reist, 1984) with most yielding no major deviation (Alenius, 1987). For this project the expressions used by Alenius are incorporated into a simplifying expression:

$$C_d(Re) = 24/Re \cdot Q(Re)$$

where  $Q(Re)$  is defined as: (4)

$$\begin{aligned} Q(Re) &= 1 && \text{for } 0 < Re \leq 0.5 \\ Q(Re) &= 1 + 0.15 Re^{0.687} && \text{for } 0.5 < Re \leq 800 \\ Q(Re) &= 0.44 Re/24 && \text{for } 800 < Re < 20000. \end{aligned}$$

By combining equations (3) and (4) into equation (2) and introducing the expression for particle relaxation time,  $\tau$ , a simplified expression for the drag force becomes:

$$Fd/m = Q(Re)/\tau \quad \text{vr} \quad (5)$$

where

$$\tau = \frac{dp^2 \, pl \, CC}{18 \, \mu l} \quad (6)$$

From equations (1) and (5) the particle motion can be described as:

$$d^2r/dt^2 - [Q(Re)/\tau(u-(dr/dt))]-g = 0 \quad (7)$$

where

$r$  is the particle position vector

$g$  is the acceleration due to gravity

$u$  is the air velocity at the point where the particle is located.

Equation (7) is analytically solvable if the factor  $Q(Re)$  is constant and the air velocity  $u$  at the points where the particle is successively located can be described as a linear function of the time that has elapsed until the particle reached the point. This condition can be approximately met if the motion is calculated in sufficiently small time steps.

If we introduce an expression for the linear behavior of the air velocity over the time interval being studied, then:

$$u(t) = (u(t)-u(t-t'))/t'+b \quad (8)$$

where

t is time  
t' is the length of the time step  
b is some initial value of velocity

For ease of writing we define the first part of the above expression as ua:

$$ua = (u(t)-u(t-t'))/t' \quad (9)$$

Now if we let  $A=Q(Re)/\tau$  equation (7) can be rewritten as:

$$dv/dt - A(ua+b) + Av - g = 0$$

recognizing that  $d^2r/dt^2 = dv/dt$ ,

$$\text{or,} \quad dv/dt + Av = A(ua+b) - g \quad (10)$$

If equation (10) is multiplied through by  $e^{\lambda t}$  the left hand side of the equation will be a derivative.

$$e^{\lambda t}(dv/dt) + e^{\lambda t}(Av) = (e^{\lambda t})A(ua+b) - g(e^{\lambda t})$$

Integrating both sides of the above equation from  $t_1$  to  $t_2$  gives:

$$\begin{aligned} v(t_2)e^{\lambda t_2} - v(t_1)e^{\lambda t_1} &= [(ua t_2)e^{\lambda t_2} - (ua/A)e^{\lambda t_2} \\ &\quad + (b)e^{\lambda t_2} + (g/A)e^{\lambda t_2}] \\ &\quad - [(ua t_2)e^{\lambda t_1} - (ua/A)e^{\lambda t_1} \\ &\quad + (b)e^{\lambda t_1} + (g/A)e^{\lambda t_1}] \end{aligned} \quad (11)$$

Adding  $v(t_1)e^{At_1}$  to both side of the equation, dividing through by  $e^{At_2}$  and simplifying gives an expression for  $v(t_2)$ :

$$v(t_2) = uat_2 - ua/A + b + g/A - \{(uat_1) - ua/A + b + g/A - v(t_1)\}e^{A(t_1-t_2)} \quad (12)$$

If we define a simplifying term,

$$u_b = u(t) - (u_a - g)\tau/Q(Re)$$

and define  $t_1$  as  $t$ , and  $t_2$  as the time at the end of the times step,  $t'$  ( $t_2 = t + t'$ ), substitute the air velocity at time  $t$  for  $b$ , and replace  $A$  with  $Q(Re)/\tau$ , then the solution is:

$$v(t+t') = u_at' + u_b - (u_b - v(t))\exp-[Q(Re)t'/\tau] \quad (13)$$

Integrating the equation (13) again will give the particle position:

$$r(t+t') = r(t) + u_at'/2 + u_bt' - (u_b - v(t))\tau/Q(Re)(1 - \exp Q(Re)t'/\tau) \quad (14)$$

Equation (13) and (14) are analytical solutions to the particle equations of motion and are used in both the 2D and 3D particle tracking subroutines.

# 2D Boundary Discretization Sample:

Potential specified on top and rear sides.  
Velocity specified over front, bottom, inlet face  
and sides.

|       |             |                     |
|-------|-------------|---------------------|
| DIN   | 1 cm        | (inlet diameter)    |
| THETA | 30 degrees  | (nozzle angle)      |
| UI    | 1000 cm/sec | (sampling velocity) |
| UO    | -250 cm/sec | (wind velocity)     |
| UOX   | -216.5064   | (x component of UO) |
| UOY   | 125         | (y component)       |

| <u>NODE</u> | <u>X</u> | <u>Y</u> | <u>PHI</u> | <u>DPHI</u> | <u>ID</u> | <u>ICT</u> |
|-------------|----------|----------|------------|-------------|-----------|------------|
| 1           | 20       | 15       | 0          | -216.5064   | 3         | 0          |
| 2           | 20       | 10       | 0          | -216.5064   | 3         | 0          |
| 3           | 20       | 5        | 0          | -216.5064   | 3         | 0          |
| 4           | 20       | 0        | 0          | -216.5064   | 3         | 0          |
| 5           | 20       | -5       | 0          | -216.5064   | 3         | 0          |
| 6           | 20       | -10      | 0          | -216.5064   | 3         | 0          |
| 7           | 20       | -15      | 0          | -216.5064   | 3         | 0          |
| 8           | 20       | -20      | 0          | -216.5064   | 3         | 0          |
| 9           | 19.99    | -20      | 0          | -125        | 3         | 0          |
| 10          | 15       | -20      | 0          | -125        | 3         | 0          |
| 11          | 10       | -20      | 0          | -125        | 3         | 0          |
| 12          | 5        | -20      | 0          | -125        | 3         | 0          |
| 13          | 0        | -20      | 0          | -125        | 3         | 0          |
| 14          | -5       | -20      | 0          | -125        | 3         | 0          |
| 15          | -10      | -20      | 0          | -125        | 3         | 0          |
| 16          | -15      | -20      | 0          | -125        | 3         | 0          |
| 17          | -20      | -20      | 1830.13    | 0           | 2         | 1          |
| 18          | -20      | -15      | 2455.13    | 0           | 2         | 0          |
| 19          | -20      | -10      | 3080.13    | 0           | 2         | 0          |
| 20          | -20      | -5       | 3705.13    | 0           | 2         | 0          |
| 21          | -20      | -0.5     | 4267.63    | 0           | 2         | 2          |
| 22          | -17.5    | -0.5     | 0          | 0           | 3         | 0          |
| 23          | -15      | -0.5     | 0          | 0           | 3         | 0          |
| 24          | -12.5    | -0.5     | 0          | 0           | 3         | 0          |
| 25          | -10      | -0.5     | 0          | 0           | 3         | 0          |
| 26          | -7.5     | -0.5     | 0          | 0           | 3         | 0          |
| 27          | -5       | -0.5     | 0          | 0           | 3         | 0          |
| 28          | -4.5     | -0.5     | 0          | 0           | 3         | 0          |
| 29          | -4       | -0.5     | 0          | 0           | 3         | 0          |
| 30          | -3.5     | -0.5     | 0          | 0           | 3         | 0          |
| 31          | -3       | -0.5     | 0          | 0           | 3         | 0          |
| 32          | -2.5     | -0.5     | 0          | 0           | 3         | 0          |
| 33          | -2       | -0.5     | 0          | 0           | 3         | 0          |
| 34          | -1.5     | -0.5     | 0          | 0           | 3         | 0          |
| 35          | -1       | -0.5     | 0          | 0           | 3         | 0          |

|    |       |        |         |           |   |   |
|----|-------|--------|---------|-----------|---|---|
| 36 | -0.5  | -0.5   | 0       | 0         | 3 | 0 |
| 37 | 0     | -0.5   | 0       | 0         | 3 | 0 |
| 38 | 0     | -0.499 | 0       | 1000      | 3 | 0 |
| 39 | 0     | -0.375 | 0       | 1000      | 3 | 0 |
| 40 | 0     | -0.25  | 0       | 1000      | 3 | 0 |
| 41 | 0     | -0.125 | 0       | 1000      | 3 | 0 |
| 42 | 0     | 0      | 0       | 1000      | 3 | 0 |
| 43 | 0     | 0.125  | 0       | 1000      | 3 | 0 |
| 44 | 0     | 0.25   | 0       | 1000      | 3 | 0 |
| 45 | 0     | 0.375  | 0       | 1000      | 3 | 0 |
| 46 | 0     | 0.499  | 0       | 1000      | 3 | 0 |
| 47 | 0     | 0.5    | 0       | 0         | 3 | 0 |
| 48 | -0.5  | 0.5    | 0       | 0         | 3 | 0 |
| 49 | -1    | 0.5    | 0       | 0         | 3 | 0 |
| 50 | -1.5  | 0.5    | 0       | 0         | 3 | 0 |
| 51 | -2    | 0.5    | 0       | 0         | 3 | 0 |
| 52 | -2.5  | 0.5    | 0       | 0         | 3 | 0 |
| 53 | -3    | 0.5    | 0       | 0         | 3 | 0 |
| 54 | -3.5  | 0.5    | 0       | 0         | 3 | 0 |
| 55 | -4    | 0.5    | 0       | 0         | 3 | 0 |
| 56 | -4.5  | 0.5    | 0       | 0         | 3 | 0 |
| 57 | -5    | 0.5    | 0       | 0         | 3 | 0 |
| 58 | -7.5  | 0.5    | 0       | 0         | 3 | 0 |
| 59 | -10   | 0.5    | 0       | 0         | 3 | 0 |
| 60 | -12.5 | 0.5    | 0       | 0         | 3 | 0 |
| 61 | -15   | 0.5    | 0       | 0         | 3 | 0 |
| 62 | -17.5 | 0.5    | 0       | 0         | 3 | 0 |
| 63 | -20   | 0.5    | 4392.63 | 0         | 2 | 1 |
| 64 | -20   | 5      | 4955.13 | 0         | 2 | 0 |
| 65 | -20   | 10     | 5580.13 | 0         | 2 | 0 |
| 66 | -20   | 15     | 6205.13 | 0         | 2 | 0 |
| 67 | -20   | 20     | 6830.13 | 0         | 2 | 0 |
| 68 | -15   | 20     | 5747.6  | 0         | 2 | 0 |
| 69 | -10   | 20     | 4665.06 | 0         | 2 | 0 |
| 70 | -5    | 20     | 3582.53 | 0         | 2 | 0 |
| 71 | 0     | 20     | 2500    | 0         | 2 | 0 |
| 72 | 5     | 20     | 1417.47 | 0         | 2 | 0 |
| 73 | 10    | 20     | 334.94  | 0         | 2 | 0 |
| 74 | 15    | 20     | -747.6  | 0         | 2 | 0 |
| 75 | 20    | 20     | 0       | -216.5064 | 3 | 2 |

3D Boundary Discretization Sample:

INPUT FILE FOR 3D CASE, 367 NODE, 693 ELEMENTS

Potential specified at all faces except at boudary of tube  
and at the face. ~~Four~~ corners specified with potential.

8

|       |             |                     |
|-------|-------------|---------------------|
| DIN   | 1 cm        | (inlet diameter)    |
| THETA | 30 degrees  | (nozzle angle)      |
| UI    | 1000 cm/sec | (sampling velocity) |
| UO    | -250 cm/sec | (wind velocity)     |
| UOX   | -216.5064   | (x component of UO) |
| UOY   | 0           | (y component)       |
| UOZ   | 125         | (z component)       |

| NODE | X | Y      | Z      | PHI | DPHI | ID | ICT |
|------|---|--------|--------|-----|------|----|-----|
| 1    | 0 | -0.5   | -0.5   | 0   | 0    | 2  | 0   |
| 2    | 0 | -0.5   | -0.499 | 0   | 0    | 2  | 0   |
| 3    | 0 | -0.5   | -0.375 | 0   | 0    | 2  | 0   |
| 4    | 0 | -0.5   | -0.25  | 0   | 0    | 2  | 0   |
| 5    | 0 | -0.5   | -0.125 | 0   | 0    | 2  | 0   |
| 6    | 0 | -0.5   | 0      | 0   | 0    | 2  | 0   |
| 7    | 0 | -0.5   | 0.125  | 0   | 0    | 2  | 0   |
| 8    | 0 | -0.5   | 0.25   | 0   | 0    | 2  | 0   |
| 9    | 0 | -0.5   | 0.375  | 0   | 0    | 2  | 0   |
| 10   | 0 | -0.5   | 0.499  | 0   | 0    | 2  | 0   |
| 11   | 0 | -0.5   | 0.5    | 0   | 0    | 2  | 0   |
| 12   | 0 | -0.499 | -0.5   | 0   | 0    | 2  | 0   |
| 13   | 0 | -0.499 | -0.499 | 0   | 1000 | 2  | 0   |
| 14   | 0 | -0.499 | -0.375 | 0   | 1000 | 2  | 0   |
| 15   | 0 | -0.499 | -0.25  | 0   | 1000 | 2  | 0   |
| 16   | 0 | -0.499 | -0.125 | 0   | 1000 | 2  | 0   |
| 17   | 0 | -0.499 | 0      | 0   | 1000 | 2  | 0   |
| 18   | 0 | -0.499 | 0.125  | 0   | 1000 | 2  | 0   |
| 19   | 0 | -0.499 | 0.25   | 0   | 1000 | 2  | 0   |
| 20   | 0 | -0.499 | 0.375  | 0   | 1000 | 2  | 0   |
| 21   | 0 | -0.499 | 0.499  | 0   | 1000 | 2  | 0   |
| 22   | 0 | -0.499 | 0.5    | 0   | 0    | 2  | 0   |
| 23   | 0 | -0.375 | -0.5   | 0   | 0    | 2  | 0   |
| 24   | 0 | -0.375 | -0.499 | 0   | 1000 | 2  | 0   |
| 25   | 0 | -0.375 | -0.375 | 0   | 1000 | 2  | 0   |
| 26   | 0 | -0.375 | -0.25  | 0   | 1000 | 2  | 0   |
| 27   | 0 | -0.375 | -0.125 | 0   | 1000 | 2  | 0   |
| 28   | 0 | -0.375 | 0      | 0   | 1000 | 2  | 0   |
| 29   | 0 | -0.375 | 0.125  | 0   | 1000 | 2  | 0   |
| 30   | 0 | -0.375 | 0.25   | 0   | 1000 | 2  | 0   |
| 31   | 0 | -0.375 | 0.375  | 0   | 1000 | 2  | 0   |
| 32   | 0 | -0.375 | 0.499  | 0   | 1000 | 2  | 0   |
| 33   | 0 | -0.375 | 0.5    | 0   | 0    | 2  | 0   |
| 34   | 0 | -0.25  | -0.5   | 0   | 0    | 2  | 0   |
| 35   | 0 | -0.25  | -0.499 | 0   | 1000 | 2  | 0   |
| 36   | 0 | -0.25  | -0.375 | 0   | 1000 | 2  | 0   |
| 37   | 0 | -0.25  | -0.25  | 0   | 1000 | 2  | 0   |
| 38   | 0 | -0.25  | -0.125 | 0   | 1000 | 2  | 0   |
| 39   | 0 | -0.25  | 0      | 0   | 1000 | 2  | 0   |
| 40   | 0 | -0.25  | 0.125  | 0   | 1000 | 2  | 0   |
| 41   | 0 | -0.25  | 0.25   | 0   | 1000 | 2  | 0   |
| 42   | 0 | -0.25  | 0.375  | 0   | 1000 | 2  | 0   |
| 43   | 0 | -0.25  | 0.499  | 0   | 1000 | 2  | 0   |
| 44   | 0 | -0.25  | 0.5    | 0   | 0    | 2  | 0   |
| 45   | 0 | -0.125 | -0.5   | 0   | 0    | 2  | 0   |
| 46   | 0 | -0.125 | -0.499 | 0   | 1000 | 2  | 0   |
| 47   | 0 | -0.125 | -0.375 | 0   | 1000 | 2  | 0   |
| 48   | 0 | -0.125 | -0.25  | 0   | 1000 | 2  | 0   |
| 49   | 0 | -0.125 | -0.125 | 0   | 1000 | 2  | 0   |
| 50   | 0 | -0.125 | 0      | 0   | 1000 | 2  | 0   |
| 51   | 0 | -0.125 | 0.125  | 0   | 1000 | 2  | 0   |
| 52   | 0 | -0.125 | 0.25   | 0   | 1000 | 2  | 0   |
| 53   | 0 | -0.125 | 0.375  | 0   | 1000 | 2  | 0   |

|     |   |        |        |   |      |   |   |
|-----|---|--------|--------|---|------|---|---|
| 54  | 0 | -0.125 | 0.499  | 0 | 1000 | 2 | 0 |
| 55  | 0 | -0.125 | 0.5    | 0 | 0    | 2 | 0 |
| 56  | 0 | 0      | -0.5   | 0 | 0    | 2 | 0 |
| 57  | 0 | 0      | -0.499 | 0 | 1000 | 2 | 0 |
| 58  | 0 | 0      | -0.375 | 0 | 1000 | 2 | 0 |
| 59  | 0 | 0      | -0.25  | 0 | 1000 | 2 | 0 |
| 60  | 0 | 0      | -0.125 | 0 | 1000 | 2 | 0 |
| 61  | 0 | 0      | 0      | 0 | 1000 | 2 | 0 |
| 62  | 0 | 0      | 0.125  | 0 | 1000 | 2 | 0 |
| 63  | 0 | 0      | 0.25   | 0 | 1000 | 2 | 0 |
| 64  | 0 | 0      | 0.375  | 0 | 1000 | 2 | 0 |
| 65  | 0 | 0      | 0.499  | 0 | 1000 | 2 | 0 |
| 66  | 0 | 0      | 0.5    | 0 | 0    | 2 | 0 |
| 67  | 0 | 0.125  | -0.5   | 0 | 0    | 2 | 0 |
| 68  | 0 | 0.125  | -0.499 | 0 | 1000 | 2 | 0 |
| 69  | 0 | 0.125  | -0.375 | 0 | 1000 | 2 | 0 |
| 70  | 0 | 0.125  | -0.25  | 0 | 1000 | 2 | 0 |
| 71  | 0 | 0.125  | -0.125 | 0 | 1000 | 2 | 0 |
| 72  | 0 | 0.125  | 0      | 0 | 1000 | 2 | 0 |
| 73  | 0 | 0.125  | 0.125  | 0 | 1000 | 2 | 0 |
| 74  | 0 | 0.125  | 0.25   | 0 | 1000 | 2 | 0 |
| 75  | 0 | 0.125  | 0.375  | 0 | 1000 | 2 | 0 |
| 76  | 0 | 0.125  | 0.499  | 0 | 1000 | 2 | 0 |
| 77  | 0 | 0.125  | 0.5    | 0 | 0    | 2 | 0 |
| 78  | 0 | 0.25   | -0.5   | 0 | 0    | 2 | 0 |
| 79  | 0 | 0.25   | -0.499 | 0 | 1000 | 2 | 0 |
| 80  | 0 | 0.25   | -0.375 | 0 | 1000 | 2 | 0 |
| 81  | 0 | 0.25   | -0.25  | 0 | 1000 | 2 | 0 |
| 82  | 0 | 0.25   | -0.125 | 0 | 1000 | 2 | 0 |
| 83  | 0 | 0.25   | 0      | 0 | 1000 | 2 | 0 |
| 84  | 0 | 0.25   | 0.125  | 0 | 1000 | 2 | 0 |
| 85  | 0 | 0.25   | 0.25   | 0 | 1000 | 2 | 0 |
| 86  | 0 | 0.25   | 0.375  | 0 | 1000 | 2 | 0 |
| 87  | 0 | 0.25   | 0.499  | 0 | 1000 | 2 | 0 |
| 88  | 0 | 0.25   | 0.5    | 0 | 0    | 2 | 0 |
| 89  | 0 | 0.375  | -0.5   | 0 | 0    | 2 | 0 |
| 90  | 0 | 0.375  | -0.499 | 0 | 1000 | 2 | 0 |
| 91  | 0 | 0.375  | -0.375 | 0 | 1000 | 2 | 0 |
| 92  | 0 | 0.375  | -0.25  | 0 | 1000 | 2 | 0 |
| 93  | 0 | 0.375  | -0.125 | 0 | 1000 | 2 | 0 |
| 94  | 0 | 0.375  | 0      | 0 | 1000 | 2 | 0 |
| 95  | 0 | 0.375  | 0.125  | 0 | 1000 | 2 | 0 |
| 96  | 0 | 0.375  | 0.25   | 0 | 1000 | 2 | 0 |
| 97  | 0 | 0.375  | 0.375  | 0 | 1000 | 2 | 0 |
| 98  | 0 | 0.375  | 0.499  | 0 | 1000 | 2 | 0 |
| 99  | 0 | 0.375  | 0.5    | 0 | 0    | 2 | 0 |
| 100 | 0 | 0.499  | -0.5   | 0 | 0    | 2 | 0 |
| 101 | 0 | 0.499  | -0.499 | 0 | 1000 | 2 | 0 |
| 102 | 0 | 0.499  | -0.375 | 0 | 1000 | 2 | 0 |
| 103 | 0 | 0.499  | -0.25  | 0 | 1000 | 2 | 0 |
| 104 | 0 | 0.499  | -0.125 | 0 | 1000 | 2 | 0 |
| 105 | 0 | 0.499  | 0      | 0 | 1000 | 2 | 0 |
| 106 | 0 | 0.499  | 0.125  | 0 | 1000 | 2 | 0 |
| 107 | 0 | 0.499  | 0.25   | 0 | 1000 | 2 | 0 |

|     |      |       |        |   |      |   |   |
|-----|------|-------|--------|---|------|---|---|
| 108 | 0    | 0.499 | 0.375  | 0 | 1000 | 2 | 0 |
| 109 | 0    | 0.499 | 0.499  | 0 | 1000 | 2 | 0 |
| 110 | 0    | 0.499 | 0.5    | 0 | 0    | 2 | 0 |
| 111 | 0    | 0.5   | -0.5   | 0 | 0    | 2 | 0 |
| 112 | 0    | 0.5   | -0.499 | 0 | 0    | 2 | 0 |
| 113 | 0    | 0.5   | -0.375 | 0 | 0    | 2 | 0 |
| 114 | 0    | 0.5   | -0.25  | 0 | 0    | 2 | 0 |
| 115 | 0    | 0.5   | -0.125 | 0 | 0    | 2 | 0 |
| 116 | 0    | 0.5   | 0      | 0 | 0    | 2 | 0 |
| 117 | 0    | 0.5   | 0.125  | 0 | 0    | 2 | 0 |
| 118 | 0    | 0.5   | 0.25   | 0 | 0    | 2 | 0 |
| 119 | 0    | 0.5   | 0.375  | 0 | 0    | 2 | 0 |
| 120 | 0    | 0.5   | 0.499  | 0 | 0    | 2 | 0 |
| 121 | 0    | 0.5   | 0.5    | 0 | 0    | 2 | 0 |
| 122 | -0.5 | -0.5  | -0.5   | 0 | 0    | 2 | 0 |
| 123 | -1   | -0.5  | -0.5   | 0 | 0    | 2 | 0 |
| 124 | -1.5 | -0.5  | -0.5   | 0 | 0    | 2 | 0 |
| 125 | -2   | -0.5  | -0.5   | 0 | 0    | 2 | 0 |
| 126 | -2.5 | -0.5  | -0.5   | 0 | 0    | 2 | 0 |
| 127 | -3   | -0.5  | -0.5   | 0 | 0    | 2 | 0 |
| 128 | -3.5 | -0.5  | -0.5   | 0 | 0    | 2 | 0 |
| 129 | -4   | -0.5  | -0.5   | 0 | 0    | 2 | 0 |
| 130 | -4.5 | -0.5  | -0.5   | 0 | 0    | 2 | 0 |
| 131 | -5   | -0.5  | -0.5   | 0 | 0    | 2 | 0 |
| 132 | -0.5 | 0     | -0.5   | 0 | 0    | 2 | 0 |
| 133 | -1   | 0     | -0.5   | 0 | 0    | 2 | 0 |
| 134 | -1.5 | 0     | -0.5   | 0 | 0    | 2 | 0 |
| 135 | -2   | 0     | -0.5   | 0 | 0    | 2 | 0 |
| 136 | -2.5 | 0     | -0.5   | 0 | 0    | 2 | 0 |
| 137 | -3   | 0     | -0.5   | 0 | 0    | 2 | 0 |
| 138 | -3.5 | 0     | -0.5   | 0 | 0    | 2 | 0 |
| 139 | -4   | 0     | -0.5   | 0 | 0    | 2 | 0 |
| 140 | -4.5 | 0     | -0.5   | 0 | 0    | 2 | 0 |
| 141 | -5   | 0     | -0.5   | 0 | 0    | 2 | 0 |
| 142 | -0.5 | 0.5   | -0.5   | 0 | 0    | 2 | 0 |
| 143 | -1   | 0.5   | -0.5   | 0 | 0    | 2 | 0 |
| 144 | -1.5 | 0.5   | -0.5   | 0 | 0    | 2 | 0 |
| 145 | -2   | 0.5   | -0.5   | 0 | 0    | 2 | 0 |
| 146 | -2.5 | 0.5   | -0.5   | 0 | 0    | 2 | 0 |
| 147 | -3   | 0.5   | -0.5   | 0 | 0    | 2 | 0 |
| 148 | -3.5 | 0.5   | -0.5   | 0 | 0    | 2 | 0 |
| 149 | -4   | 0.5   | -0.5   | 0 | 0    | 2 | 0 |
| 150 | -4.5 | 0.5   | -0.5   | 0 | 0    | 2 | 0 |
| 151 | -5   | 0.5   | -0.5   | 0 | 0    | 2 | 0 |
| 152 | -0.5 | 0.5   | 0      | 0 | 0    | 2 | 0 |
| 153 | -1   | 0.5   | 0      | 0 | 0    | 2 | 0 |
| 154 | -1.5 | 0.5   | 0      | 0 | 0    | 2 | 0 |
| 155 | -2   | 0.5   | 0      | 0 | 0    | 2 | 0 |
| 156 | -2.5 | 0.5   | 0      | 0 | 0    | 2 | 0 |
| 157 | -3   | 0.5   | 0      | 0 | 0    | 2 | 0 |
| 158 | -3.5 | 0.5   | 0      | 0 | 0    | 2 | 0 |
| 159 | -4   | 0.5   | 0      | 0 | 0    | 2 | 0 |
| 160 | -4.5 | 0.5   | 0      | 0 | 0    | 2 | 0 |
| 161 | -5   | 0.5   | 0      | 0 | 0    | 2 | 0 |

|     |       |      |      |      |   |   |   |
|-----|-------|------|------|------|---|---|---|
| 162 | -0.5  | 0.5  | 0.5  | 0    | 0 | 2 | 0 |
| 163 | -1    | 0.5  | 0.5  | 0    | 0 | 2 | 0 |
| 164 | -1.5  | 0.5  | 0.5  | 0    | 0 | 2 | 0 |
| 165 | -2    | 0.5  | 0.5  | 0    | 0 | 2 | 0 |
| 166 | -2.5  | 0.5  | 0.5  | 0    | 0 | 2 | 0 |
| 167 | -3    | 0.5  | 0.5  | 0    | 0 | 2 | 0 |
| 168 | -3.5  | 0.5  | 0.5  | 0    | 0 | 2 | 0 |
| 169 | -4    | 0.5  | 0.5  | 0    | 0 | 2 | 0 |
| 170 | -4.5  | 0.5  | 0.5  | 0    | 0 | 2 | 0 |
| 171 | -5    | 0.5  | 0.5  | 0    | 0 | 2 | 0 |
| 172 | -0.5  | 0    | 0.5  | 0    | 0 | 2 | 0 |
| 173 | -1    | 0    | 0.5  | 0    | 0 | 2 | 0 |
| 174 | -1.5  | 0    | 0.5  | 0    | 0 | 2 | 0 |
| 175 | -2    | 0    | 0.5  | 0    | 0 | 2 | 0 |
| 176 | -2.5  | 0    | 0.5  | 0    | 0 | 2 | 0 |
| 177 | -3    | 0    | 0.5  | 0    | 0 | 2 | 0 |
| 178 | -3.5  | 0    | 0.5  | 0    | 0 | 2 | 0 |
| 179 | -4    | 0    | 0.5  | 0    | 0 | 2 | 0 |
| 180 | -4.5  | 0    | 0.5  | 0    | 0 | 2 | 0 |
| 181 | -5    | 0    | 0.5  | 0    | 0 | 2 | 0 |
| 182 | -0.5  | -0.5 | 0.5  | 0    | 0 | 2 | 0 |
| 183 | -1    | -0.5 | 0.5  | 0    | 0 | 2 | 0 |
| 184 | -1.5  | -0.5 | 0.5  | 0    | 0 | 2 | 0 |
| 185 | -2    | -0.5 | 0.5  | 0    | 0 | 2 | 0 |
| 186 | -2.5  | -0.5 | 0.5  | 0    | 0 | 2 | 0 |
| 187 | -3    | -0.5 | 0.5  | 0    | 0 | 2 | 0 |
| 188 | -3.5  | -0.5 | 0.5  | 0    | 0 | 2 | 0 |
| 189 | -4    | -0.5 | 0.5  | 0    | 0 | 2 | 0 |
| 190 | -4.5  | -0.5 | 0.5  | 0    | 0 | 2 | 0 |
| 191 | -5    | -0.5 | 0.5  | 0    | 0 | 2 | 0 |
| 192 | -0.5  | -0.5 | 0    | 0    | 0 | 2 | 0 |
| 193 | -1    | -0.5 | 0    | 0    | 0 | 2 | 0 |
| 194 | -1.5  | -0.5 | 0    | 0    | 0 | 2 | 0 |
| 195 | -2    | -0.5 | 0    | 0    | 0 | 2 | 0 |
| 196 | -2.5  | -0.5 | 0    | 0    | 0 | 2 | 0 |
| 197 | -3    | -0.5 | 0    | 0    | 0 | 2 | 0 |
| 198 | -3.5  | -0.5 | 0    | 0    | 0 | 2 | 0 |
| 199 | -4    | -0.5 | 0    | 0    | 0 | 2 | 0 |
| 200 | -4.5  | -0.5 | 0    | 0    | 0 | 2 | 0 |
| 201 | -5    | -0.5 | 0    | 0    | 0 | 2 | 0 |
| 202 | -7.5  | -0.5 | -0.5 | 0    | 0 | 2 | 0 |
| 203 | -10   | -0.5 | -0.5 | 0    | 0 | 2 | 0 |
| 204 | -12.5 | -0.5 | -0.5 | 0    | 0 | 2 | 0 |
| 205 | -15   | -0.5 | -0.5 | 0    | 0 | 2 | 0 |
| 206 | -17.5 | -0.5 | -0.5 | 0    | 0 | 2 | 0 |
| 207 | -20   | -0.5 | -0.5 | 4268 | 0 | 1 | 1 |
| 208 | -7.5  | 0.5  | -0.5 | 0    | 0 | 2 | 0 |
| 209 | -10   | 0.5  | -0.5 | 0    | 0 | 2 | 0 |
| 210 | -12.5 | 0.5  | -0.5 | 0    | 0 | 2 | 0 |
| 211 | -15   | 0.5  | -0.5 | 0    | 0 | 2 | 0 |
| 212 | -17.5 | 0.5  | -0.5 | 0    | 0 | 2 | 0 |
| 213 | -20   | 0.5  | -0.5 | 4268 | 0 | 1 | 1 |
| 214 | -7.5  | 0.5  | 0.5  | 0    | 0 | 2 | 0 |
| 215 | -10   | 0.5  | 0.5  | 0    | 0 | 2 | 0 |

|     |       |        |        |       |   |   |   |
|-----|-------|--------|--------|-------|---|---|---|
| 216 | -12.5 | 0.5    | 0.5    | 0     | 0 | 2 | 0 |
| 217 | -15   | 0.5    | 0.5    | 0     | 0 | 2 | 0 |
| 218 | -17.5 | 0.5    | 0.5    | 0     | 0 | 2 | 0 |
| 219 | -20   | 0.5    | 0.5    | 4393  | 0 | 1 | 1 |
| 220 | -7.5  | -0.5   | 0.5    | 0     | 0 | 2 | 0 |
| 221 | -10   | -0.5   | 0.5    | 0     | 0 | 2 | 0 |
| 222 | -12.5 | -0.5   | 0.5    | 0     | 0 | 2 | 0 |
| 223 | -15   | -0.5   | 0.5    | 0     | 0 | 2 | 0 |
| 224 | -17.5 | -0.5   | 0.5    | 0     | 0 | 2 | 0 |
| 225 | -20   | -0.5   | 0.5    | 4393  | 0 | 1 | 1 |
| 226 | -20   | -20    | -20    | 1830  | 0 | 1 | 0 |
| 227 | -20   | -20    | -10    | 3080  | 0 | 1 | 0 |
| 228 | -20   | -20    | 0      | 4330  | 0 | 1 | 0 |
| 229 | -20   | -20    | 10     | 5580  | 0 | 1 | 0 |
| 230 | -20   | -20    | 20     | 6830  | 0 | 1 | 0 |
| 231 | -20   | -10    | -20    | 1830  | 0 | 1 | 0 |
| 232 | -20   | -10    | -10    | 3080  | 0 | 1 | 0 |
| 233 | -20   | -10    | 0      | 4330  | 0 | 1 | 0 |
| 234 | -20   | -10    | 10     | 5580  | 0 | 1 | 0 |
| 235 | -20   | -10    | 20     | 6830  | 0 | 1 | 0 |
| 236 | -20   | 0      | -20    | 1830  | 0 | 1 | 0 |
| 237 | -20   | -0.501 | 0.501  | 4393  | 0 | 1 | 0 |
| 238 | -20   | -0.501 | 0      | 4330  | 0 | 1 | 0 |
| 239 | -20   | 0.501  | -0.501 | 4268  | 0 | 1 | 0 |
| 240 | -20   | 0      | 10     | 5580  | 0 | 1 | 0 |
| 241 | -20   | 0      | 20     | 6830  | 0 | 1 | 0 |
| 242 | -20   | 0      | -10    | 3080  | 0 | 1 | 0 |
| 243 | -20   | 0.501  | 0.501  | 4393  | 0 | 1 | 0 |
| 244 | -20   | 0.501  | 0      | 4330  | 0 | 1 | 0 |
| 245 | -20   | -0.501 | -0.501 | 4268  | 0 | 1 | 0 |
| 246 | -20   | 10     | -20    | 1830  | 0 | 1 | 0 |
| 247 | -20   | 10     | -10    | 3080  | 0 | 1 | 0 |
| 248 | -20   | 10     | 0      | 4330  | 0 | 1 | 0 |
| 249 | -20   | 10     | 10     | 5580  | 0 | 1 | 0 |
| 250 | -20   | 10     | 20     | 6830  | 0 | 1 | 0 |
| 251 | -20   | 20     | -20    | 1830  | 0 | 1 | 0 |
| 252 | -20   | 20     | -10    | 3080  | 0 | 1 | 0 |
| 253 | -20   | 20     | 0      | 4330  | 0 | 1 | 0 |
| 254 | -20   | 20     | 10     | 5580  | 0 | 1 | 0 |
| 255 | -20   | 20     | 20     | 6830  | 0 | 1 | 0 |
| 256 | 20    | -20    | 20     | -1830 | 0 | 1 | 0 |
| 257 | 10    | -20    | 20     | 334.9 | 0 | 1 | 0 |
| 258 | 0     | -20    | 20     | 2500  | 0 | 1 | 0 |
| 259 | -10   | -20    | 20     | 4665  | 0 | 1 | 0 |
| 260 | 20    | -20    | 10     | -3080 | 0 | 1 | 0 |
| 261 | 10    | -20    | 10     | -915  | 0 | 1 | 0 |
| 262 | 0     | -20    | 10     | 1250  | 0 | 1 | 0 |
| 263 | -10   | -20    | 10     | 3415  | 0 | 1 | 0 |
| 264 | 20    | -20    | 0      | -4330 | 0 | 1 | 0 |
| 265 | 10    | -20    | 0      | -2165 | 0 | 1 | 0 |
| 266 | 0     | -20    | 0      | 0     | 0 | 1 | 0 |
| 267 | -10   | -20    | 0      | 2165  | 0 | 1 | 0 |
| 268 | 20    | -20    | -10    | -5580 | 0 | 1 | 0 |
| 269 | 10    | -20    | -10    | -3415 | 0 | 1 | 0 |

|     |        |     |     |       |   |   |   |
|-----|--------|-----|-----|-------|---|---|---|
| 270 | 0      | -20 | -10 | -1250 | 0 | 1 | 0 |
| 271 | -10    | -20 | -10 | 915.1 | 0 | 1 | 0 |
| 272 | 20     | -20 | -20 | -6830 | 0 | 1 | 0 |
| 273 | 10     | -20 | -20 | -4665 | 0 | 1 | 0 |
| 274 | 0      | -20 | -20 | -2500 | 0 | 1 | 0 |
| 275 | -10    | -20 | -20 | -335  | 0 | 1 | 0 |
| 276 | 20     | -10 | 20  | -1830 | 0 | 1 | 0 |
| 277 | 20     | -10 | 10  | -3080 | 0 | 1 | 0 |
| 278 | 20     | -10 | 0   | -4330 | 0 | 1 | 0 |
| 279 | 20     | -10 | -10 | -5580 | 0 | 1 | 0 |
| 280 | 20     | -10 | -20 | -6830 | 0 | 1 | 0 |
| 281 | 20     | 0   | 20  | -1830 | 0 | 1 | 0 |
| 282 | 20     | 0   | 10  | -3080 | 0 | 1 | 0 |
| 283 | 20     | 0   | 0   | -4330 | 0 | 1 | 0 |
| 284 | 20     | 0   | -10 | -5580 | 0 | 1 | 0 |
| 285 | 20     | 0   | -20 | -6830 | 0 | 1 | 0 |
| 286 | 20     | 10  | 20  | -1830 | 0 | 1 | 0 |
| 287 | 20     | 10  | 10  | -3080 | 0 | 1 | 0 |
| 288 | 20     | 10  | 0   | -4330 | 0 | 1 | 0 |
| 289 | 20     | 10  | -10 | -5580 | 0 | 1 | 0 |
| 290 | 20     | 10  | -20 | -6830 | 0 | 1 | 0 |
| 291 | 20     | 20  | 20  | -1830 | 0 | 1 | 0 |
| 292 | 20     | 20  | 10  | -3080 | 0 | 1 | 0 |
| 293 | 20     | 20  | 0   | -4330 | 0 | 1 | 0 |
| 294 | 20     | 20  | -10 | -5580 | 0 | 1 | 0 |
| 295 | 20     | 20  | -20 | -6830 | 0 | 1 | 0 |
| 296 | 10     | -10 | -20 | -4665 | 0 | 1 | 0 |
| 297 | 0      | -10 | -20 | -2500 | 0 | 1 | 0 |
| 298 | -10    | -10 | -20 | -335  | 0 | 1 | 0 |
| 299 | 10     | 0   | -20 | -4665 | 0 | 1 | 0 |
| 300 | 0      | 0   | -20 | -2500 | 0 | 1 | 0 |
| 301 | -10    | 0   | -20 | -335  | 0 | 1 | 0 |
| 302 | 10     | 10  | -20 | -4665 | 0 | 1 | 0 |
| 303 | 0      | 10  | -20 | -2500 | 0 | 1 | 0 |
| 304 | -10    | 10  | -20 | -335  | 0 | 1 | 0 |
| 305 | 10     | 20  | -20 | -4665 | 0 | 1 | 0 |
| 306 | 0      | 20  | -20 | -2500 | 0 | 1 | 0 |
| 307 | -10    | 20  | -20 | -335  | 0 | 1 | 0 |
| 308 | 0      | -20 | -20 | -2500 | 0 | 1 | 0 |
| 309 | 19.999 | -10 | -20 | -6830 | 0 | 1 | 0 |
| 310 | 19.999 | 0   | -20 | -6830 | 0 | 1 | 0 |
| 311 | 19.999 | 10  | -20 | -6830 | 0 | 1 | 0 |
| 312 | 19.999 | 20  | -20 | -6830 | 0 | 1 | 0 |
| 313 | 10     | 20  | -10 | -3415 | 0 | 1 | 0 |
| 314 | 0      | 20  | -10 | -1250 | 0 | 1 | 0 |
| 315 | -10    | 20  | -10 | 915.1 | 0 | 1 | 0 |
| 316 | 10     | 20  | 0   | -2165 | 0 | 1 | 0 |
| 317 | 0      | 20  | 0   | 0     | 0 | 1 | 0 |
| 318 | -10    | 20  | 0   | 2165  | 0 | 1 | 0 |
| 319 | 10     | 20  | 10  | -915  | 0 | 1 | 0 |
| 320 | 0      | 20  | 10  | 1250  | 0 | 1 | 0 |
| 321 | -10    | 20  | 10  | 3415  | 0 | 1 | 0 |
| 322 | 10     | 20  | 20  | 334.9 | 0 | 1 | 0 |
| 323 | 0      | 20  | 20  | 2500  | 0 | 1 | 0 |

|     |     |         |         |       |   |   |   |
|-----|-----|---------|---------|-------|---|---|---|
| 324 | -10 | 20      | 20      | 4665  | 0 | 1 | 0 |
| 325 | 10  | 10      | 20      | 334.9 | 0 | 1 | 0 |
| 326 | 0   | 10      | 20      | 2500  | 0 | 1 | 0 |
| 327 | -10 | 10      | 20      | 4665  | 0 | 1 | 0 |
| 328 | 10  | 0       | 20      | 334.9 | 0 | 1 | 0 |
| 329 | 0   | 0       | 20      | 2500  | 0 | 1 | 0 |
| 330 | -10 | 0       | 20      | 4665  | 0 | 1 | 0 |
| 331 | 10  | -10     | 20      | 334.9 | 0 | 1 | 0 |
| 332 | 0   | -10     | 20      | 2500  | 0 | 1 | 0 |
| 333 | -10 | -10     | 20      | 4665  | 0 | 1 | 0 |
| 334 | 20  | -19.999 | 19.999  | -1830 | 0 | 1 | 0 |
| 335 | 20  | -19.999 | 10      | -3080 | 0 | 1 | 0 |
| 336 | 20  | -19.999 | 0       | -4330 | 0 | 1 | 0 |
| 337 | 20  | -19.999 | -10     | -5580 | 0 | 1 | 0 |
| 338 | 20  | -19.999 | -19.999 | -6830 | 0 | 1 | 0 |
| 339 | 20  | -10     | -19.999 | -6830 | 0 | 1 | 0 |
| 340 | 20  | 0       | -19.999 | -6830 | 0 | 1 | 0 |
| 341 | 20  | 10      | -19.999 | -6830 | 0 | 1 | 0 |
| 342 | 20  | 19.999  | -19.999 | -6830 | 0 | 1 | 0 |
| 343 | 20  | 19.999  | -10     | -5580 | 0 | 1 | 0 |
| 344 | 20  | 19.999  | 0       | -4330 | 0 | 1 | 0 |
| 345 | 20  | 19.999  | 10      | -3080 | 0 | 1 | 0 |
| 346 | 20  | 19.999  | 19.999  | -1830 | 0 | 1 | 0 |
| 347 | 20  | 10      | 19.999  | -1830 | 0 | 1 | 0 |
| 348 | 20  | 0       | 19.999  | -1830 | 0 | 1 | 0 |
| 349 | 20  | -10     | 19.999  | -1830 | 0 | 1 | 0 |
| 350 | -20 | 0       | 0.501   | 4393  | 0 | 1 | 0 |
| 351 | -20 | 0       | -0.501  | 4268  | 0 | 1 | 0 |
| 352 | -20 | -19.999 | -19.999 | 1830  | 0 | 1 | 0 |
| 353 | -20 | -19.999 | -10     | 3080  | 0 | 1 | 0 |
| 354 | -20 | -19.999 | 0       | 4330  | 0 | 1 | 0 |
| 355 | -20 | -19.999 | 10      | 5580  | 0 | 1 | 0 |
| 356 | -20 | -19.999 | 19.999  | 6830  | 0 | 1 | 0 |
| 357 | -20 | -10     | -19.999 | 1830  | 0 | 1 | 0 |
| 358 | -20 | -10     | 19.999  | 6830  | 0 | 1 | 0 |
| 359 | -20 | 0       | -19.999 | 1830  | 0 | 1 | 0 |
| 360 | -20 | 0       | 19.999  | 6830  | 0 | 1 | 0 |
| 361 | -20 | 10      | -19.999 | 1830  | 0 | 1 | 0 |
| 362 | -20 | 10      | 19.999  | 6830  | 0 | 1 | 0 |
| 363 | -20 | 19.999  | -19.999 | 1830  | 0 | 1 | 0 |
| 364 | -20 | 19.999  | -10     | 3080  | 0 | 1 | 0 |
| 365 | -20 | 19.999  | 0       | 4330  | 0 | 1 | 0 |
| 366 | -20 | 19.999  | 10      | 5580  | 0 | 1 | 0 |
| 367 | -20 | 19.999  | 19.999  | 6830  | 0 | 1 | 0 |

| <u>ELEM</u> | <u>NODE1</u> | <u>NODE2</u> | <u>NODE3</u> | <u>NODE4</u> |
|-------------|--------------|--------------|--------------|--------------|
| 1           | 1            | 2            | 13           | 13           |
| 2           | 2            | 3            | 13           | 13           |
| 3           | 3            | 4            | 14           | 14           |
| 4           | 4            | 5            | 15           | 15           |
| 5           | 5            | 6            | 16           | 16           |
| 6           | 6            | 7            | 17           | 17           |
| 7           | 7            | 8            | 18           | 18           |
| 8           | 8            | 9            | 19           | 19           |
| 9           | 9            | 10           | 20           | 20           |
| 10          | 10           | 11           | 21           | 21           |
| 11          | 1            | 13           | 12           | 13           |
| 12          | 3            | 14           | 13           | 13           |
| 13          | 4            | 15           | 14           | 14           |
| 14          | 5            | 16           | 15           | 15           |
| 15          | 6            | 17           | 16           | 16           |
| 16          | 7            | 18           | 17           | 17           |
| 17          | 8            | 19           | 18           | 18           |
| 18          | 9            | 20           | 19           | 19           |
| 19          | 10           | 21           | 20           | 20           |
| 20          | 11           | 22           | 21           | 21           |
| 21          | 12           | 13           | 23           | 13           |
| 22          | 13           | 14           | 24           | 24           |
| 23          | 14           | 15           | 25           | 25           |
| 24          | 15           | 16           | 26           | 26           |
| 25          | 16           | 17           | 27           | 27           |
| 26          | 17           | 18           | 28           | 28           |
| 27          | 18           | 19           | 29           | 29           |
| 28          | 19           | 20           | 30           | 30           |
| 29          | 20           | 21           | 31           | 31           |
| 30          | 21           | 22           | 32           | 32           |
| 31          | 13           | 24           | 23           | 24           |
| 32          | 14           | 25           | 24           | 24           |
| 33          | 15           | 26           | 25           | 25           |
| 34          | 16           | 27           | 26           | 26           |
| 35          | 17           | 28           | 27           | 27           |
| 36          | 18           | 29           | 28           | 28           |
| 37          | 19           | 30           | 29           | 29           |
| 38          | 20           | 31           | 30           | 30           |
| 39          | 21           | 32           | 31           | 31           |
| 40          | 22           | 33           | 32           | 32           |
| 41          | 23           | 24           | 34           | 24           |
| 42          | 24           | 25           | 35           | 35           |
| 43          | 25           | 26           | 36           | 36           |
| 44          | 26           | 27           | 37           | 37           |
| 45          | 27           | 28           | 38           | 38           |
| 46          | 28           | 29           | 39           | 39           |
| 47          | 29           | 30           | 40           | 40           |
| 48          | 30           | 31           | 41           | 41           |
| 49          | 31           | 32           | 42           | 42           |
| 50          | 32           | 33           | 43           | 43           |
| 51          | 24           | 35           | 34           | 35           |
| 52          | 25           | 36           | 35           | 35           |
| 53          | 26           | 37           | 36           | 36           |

| <u>ELEM</u> | <u>NODE1</u> | <u>NODE2</u> | <u>NODE3</u> | <u>NODE4</u> |
|-------------|--------------|--------------|--------------|--------------|
| 54          | 27           | 38           | 37           | 37           |
| 55          | 28           | 39           | 38           | 38           |
| 56          | 29           | 40           | 39           | 39           |
| 57          | 30           | 41           | 40           | 40           |
| 58          | 31           | 42           | 41           | 41           |
| 59          | 32           | 43           | 42           | 42           |
| 60          | 33           | 44           | 43           | 43           |
| 61          | 34           | 35           | 45           | 35           |
| 62          | 35           | 36           | 46           | 46           |
| 63          | 36           | 37           | 47           | 47           |
| 64          | 37           | 38           | 48           | 48           |
| 65          | 38           | 39           | 49           | 49           |
| 66          | 39           | 40           | 50           | 50           |
| 67          | 40           | 41           | 51           | 51           |
| 68          | 41           | 42           | 52           | 52           |
| 69          | 42           | 43           | 53           | 53           |
| 70          | 43           | 44           | 54           | 54           |
| 71          | 35           | 46           | 45           | 46           |
| 72          | 36           | 47           | 46           | 46           |
| 73          | 37           | 48           | 47           | 47           |
| 74          | 38           | 49           | 48           | 48           |
| 75          | 39           | 50           | 49           | 49           |
| 76          | 40           | 51           | 50           | 50           |
| 77          | 41           | 52           | 51           | 51           |
| 78          | 42           | 53           | 52           | 52           |
| 79          | 43           | 54           | 53           | 53           |
| 80          | 44           | 55           | 54           | 54           |
| 81          | 45           | 46           | 56           | 46           |
| 82          | 46           | 47           | 57           | 57           |
| 83          | 47           | 48           | 58           | 58           |
| 84          | 48           | 49           | 59           | 59           |
| 85          | 49           | 50           | 60           | 60           |
| 86          | 50           | 51           | 61           | 61           |
| 87          | 51           | 52           | 62           | 62           |
| 88          | 52           | 53           | 63           | 63           |
| 89          | 53           | 54           | 64           | 64           |
| 90          | 54           | 55           | 65           | 65           |
| 91          | 46           | 57           | 56           | 57           |
| 92          | 47           | 58           | 57           | 57           |
| 93          | 48           | 59           | 58           | 58           |
| 94          | 49           | 60           | 59           | 59           |
| 95          | 50           | 61           | 60           | 60           |
| 96          | 51           | 62           | 61           | 61           |
| 97          | 52           | 63           | 62           | 62           |
| 98          | 53           | 64           | 63           | 63           |
| 99          | 54           | 65           | 64           | 64           |
| 100         | 55           | 66           | 65           | 65           |
| 101         | 56           | 57           | 67           | 57           |
| 102         | 57           | 58           | 68           | 68           |
| 103         | 58           | 59           | 69           | 69           |
| 104         | 59           | 60           | 70           | 70           |
| 105         | 60           | 61           | 71           | 71           |
| 106         | 61           | 62           | 72           | 72           |

| <u>ELEM</u> | <u>NODE1</u> | <u>NODE2</u> | <u>NODE3</u> | <u>NODE4</u> |
|-------------|--------------|--------------|--------------|--------------|
| 107         | 62           | 63           | 73           | 73           |
| 108         | 63           | 64           | 74           | 74           |
| 109         | 64           | 65           | 75           | 75           |
| 110         | 65           | 66           | 76           | 76           |
| 111         | 57           | 68           | 67           | 68           |
| 112         | 58           | 69           | 68           | 68           |
| 113         | 59           | 70           | 69           | 69           |
| 114         | 60           | 71           | 70           | 70           |
| 115         | 61           | 72           | 71           | 71           |
| 116         | 62           | 73           | 72           | 72           |
| 117         | 63           | 74           | 73           | 73           |
| 118         | 64           | 75           | 74           | 74           |
| 119         | 65           | 76           | 75           | 75           |
| 120         | 66           | 77           | 76           | 76           |
| 121         | 67           | 68           | 78           | 68           |
| 122         | 68           | 69           | 79           | 79           |
| 123         | 69           | 70           | 80           | 80           |
| 124         | 70           | 71           | 81           | 81           |
| 125         | 71           | 72           | 82           | 82           |
| 126         | 72           | 73           | 83           | 83           |
| 127         | 73           | 74           | 84           | 84           |
| 128         | 74           | 75           | 85           | 85           |
| 129         | 75           | 76           | 86           | 86           |
| 130         | 76           | 77           | 87           | 87           |
| 131         | 68           | 79           | 78           | 79           |
| 132         | 69           | 80           | 79           | 79           |
| 133         | 70           | 81           | 80           | 80           |
| 134         | 71           | 82           | 81           | 81           |
| 135         | 72           | 83           | 82           | 82           |
| 136         | 73           | 84           | 83           | 83           |
| 137         | 74           | 85           | 84           | 84           |
| 138         | 75           | 86           | 85           | 85           |
| 139         | 76           | 87           | 86           | 86           |
| 140         | 77           | 88           | 87           | 87           |
| 141         | 78           | 79           | 89           | 79           |
| 142         | 79           | 80           | 90           | 90           |
| 143         | 80           | 81           | 91           | 91           |
| 144         | 81           | 82           | 92           | 92           |
| 145         | 82           | 83           | 93           | 93           |
| 146         | 83           | 84           | 94           | 94           |
| 147         | 84           | 85           | 95           | 95           |
| 148         | 85           | 86           | 96           | 96           |
| 149         | 86           | 87           | 97           | 97           |
| 150         | 87           | 88           | 98           | 98           |
| 151         | 79           | 90           | 89           | 90           |
| 152         | 80           | 91           | 90           | 90           |
| 153         | 81           | 92           | 91           | 91           |
| 154         | 82           | 93           | 92           | 92           |
| 155         | 83           | 94           | 93           | 93           |
| 156         | 84           | 95           | 94           | 94           |
| 157         | 85           | 96           | 95           | 95           |
| 158         | 86           | 97           | 96           | 96           |
| 159         | 87           | 98           | 97           | 97           |

| <u>ELEM</u> | <u>NODE1</u> | <u>NODE2</u> | <u>NODE3</u> | <u>NODE4</u> |
|-------------|--------------|--------------|--------------|--------------|
| 160         | 88           | 99           | 98           | 98           |
| 161         | 89           | 90           | 100          | 90           |
| 162         | 90           | 91           | 101          | 101          |
| 163         | 91           | 92           | 102          | 102          |
| 164         | 92           | 93           | 103          | 103          |
| 165         | 93           | 94           | 104          | 104          |
| 166         | 94           | 95           | 105          | 105          |
| 167         | 95           | 96           | 106          | 106          |
| 168         | 96           | 97           | 107          | 107          |
| 169         | 97           | 98           | 108          | 108          |
| 170         | 98           | 99           | 109          | 109          |
| 171         | 90           | 101          | 100          | 101          |
| 172         | 91           | 102          | 101          | 101          |
| 173         | 92           | 103          | 102          | 102          |
| 174         | 93           | 104          | 103          | 103          |
| 175         | 94           | 105          | 104          | 104          |
| 176         | 95           | 106          | 105          | 105          |
| 177         | 96           | 107          | 106          | 106          |
| 178         | 97           | 108          | 107          | 107          |
| 179         | 98           | 109          | 108          | 108          |
| 180         | 99           | 110          | 109          | 109          |
| 181         | 100          | 101          | 111          | 101          |
| 182         | 101          | 102          | 112          | 102          |
| 183         | 102          | 103          | 113          | 103          |
| 184         | 103          | 104          | 114          | 104          |
| 185         | 104          | 105          | 115          | 105          |
| 186         | 105          | 106          | 116          | 106          |
| 187         | 106          | 107          | 117          | 107          |
| 188         | 107          | 108          | 118          | 108          |
| 189         | 108          | 109          | 119          | 109          |
| 190         | 109          | 121          | 120          | 109          |
| 191         | 101          | 112          | 111          | 101          |
| 192         | 102          | 113          | 112          | 102          |
| 193         | 103          | 114          | 113          | 103          |
| 194         | 104          | 115          | 114          | 104          |
| 195         | 105          | 116          | 115          | 105          |
| 196         | 106          | 117          | 116          | 106          |
| 197         | 107          | 118          | 117          | 107          |
| 198         | 108          | 119          | 118          | 108          |
| 199         | 109          | 120          | 119          | 109          |
| 200         | 109          | 110          | 121          | 109          |
| 201         | 122          | 1            | 56           | 56           |
| 202         | 123          | 122          | 132          | 132          |
| 203         | 124          | 123          | 133          | 133          |
| 204         | 125          | 124          | 134          | 134          |
| 205         | 126          | 125          | 135          | 135          |
| 206         | 127          | 126          | 136          | 136          |
| 207         | 128          | 127          | 137          | 137          |
| 208         | 129          | 128          | 138          | 138          |
| 209         | 130          | 129          | 139          | 139          |
| 210         | 131          | 130          | 140          | 140          |
| 211         | 132          | 122          | 56           | 56           |
| 212         | 133          | 123          | 132          | 132          |

| <u>ELEM</u> | <u>NODE1</u> | <u>NODE2</u> | <u>NODE3</u> | <u>NODE4</u> |
|-------------|--------------|--------------|--------------|--------------|
| 213         | 134          | 124          | 133          | 133          |
| 214         | 135          | 125          | 134          | 134          |
| 215         | 136          | 126          | 135          | 135          |
| 216         | 137          | 127          | 136          | 136          |
| 217         | 138          | 128          | 137          | 137          |
| 218         | 139          | 129          | 138          | 138          |
| 219         | 140          | 130          | 139          | 139          |
| 220         | 141          | 131          | 140          | 140          |
| 221         | 132          | 56           | 111          | 111          |
| 222         | 133          | 132          | 142          | 142          |
| 223         | 134          | 133          | 143          | 143          |
| 224         | 135          | 134          | 144          | 144          |
| 225         | 136          | 135          | 145          | 145          |
| 226         | 137          | 136          | 146          | 146          |
| 227         | 138          | 137          | 147          | 147          |
| 228         | 139          | 138          | 148          | 148          |
| 229         | 140          | 139          | 149          | 149          |
| 230         | 141          | 140          | 150          | 150          |
| 231         | 142          | 132          | 111          | 111          |
| 232         | 143          | 133          | 142          | 142          |
| 233         | 144          | 134          | 143          | 143          |
| 234         | 145          | 135          | 144          | 144          |
| 235         | 146          | 136          | 145          | 145          |
| 236         | 147          | 137          | 146          | 146          |
| 237         | 148          | 138          | 147          | 147          |
| 238         | 149          | 139          | 148          | 148          |
| 239         | 150          | 140          | 149          | 149          |
| 240         | 151          | 141          | 150          | 150          |
| 241         | 142          | 111          | 116          | 116          |
| 242         | 143          | 142          | 152          | 152          |
| 243         | 144          | 143          | 153          | 153          |
| 244         | 145          | 144          | 154          | 154          |
| 245         | 146          | 145          | 155          | 155          |
| 246         | 147          | 146          | 156          | 156          |
| 247         | 148          | 147          | 157          | 157          |
| 248         | 149          | 148          | 158          | 158          |
| 249         | 150          | 149          | 159          | 159          |
| 250         | 151          | 150          | 160          | 160          |
| 251         | 152          | 142          | 116          | 116          |
| 252         | 153          | 143          | 152          | 152          |
| 253         | 154          | 144          | 153          | 153          |
| 254         | 155          | 145          | 154          | 154          |
| 255         | 156          | 146          | 155          | 155          |
| 256         | 157          | 147          | 156          | 156          |
| 257         | 158          | 148          | 157          | 157          |
| 258         | 159          | 149          | 158          | 158          |
| 259         | 160          | 150          | 159          | 159          |
| 260         | 161          | 151          | 160          | 160          |
| 261         | 152          | 116          | 121          | 121          |
| 262         | 153          | 152          | 162          | 162          |
| 263         | 154          | 153          | 163          | 163          |
| 264         | 155          | 154          | 164          | 164          |
| 265         | 156          | 155          | 165          | 165          |

| <u>ELEM</u> | <u>NODE1</u> | <u>NODE2</u> | <u>NODE3</u> | <u>NODE4</u> |
|-------------|--------------|--------------|--------------|--------------|
| 266         | 157          | 156          | 166          | 166          |
| 267         | 158          | 157          | 167          | 167          |
| 268         | 159          | 158          | 168          | 168          |
| 269         | 160          | 159          | 169          | 169          |
| 270         | 161          | 160          | 170          | 170          |
| 271         | 162          | 152          | 121          | 121          |
| 272         | 163          | 153          | 162          | 162          |
| 273         | 164          | 154          | 163          | 163          |
| 274         | 165          | 155          | 164          | 164          |
| 275         | 166          | 156          | 165          | 165          |
| 276         | 167          | 157          | 166          | 166          |
| 277         | 168          | 158          | 167          | 167          |
| 278         | 169          | 159          | 168          | 168          |
| 279         | 170          | 160          | 169          | 169          |
| 280         | 171          | 161          | 170          | 170          |
| 281         | 162          | 121          | 66           | 66           |
| 282         | 163          | 162          | 172          | 172          |
| 283         | 164          | 163          | 173          | 173          |
| 284         | 165          | 164          | 174          | 174          |
| 285         | 166          | 165          | 175          | 175          |
| 286         | 167          | 166          | 176          | 176          |
| 287         | 168          | 167          | 177          | 177          |
| 288         | 169          | 168          | 178          | 178          |
| 289         | 170          | 169          | 179          | 179          |
| 290         | 171          | 170          | 180          | 180          |
| 291         | 172          | 162          | 66           | 66           |
| 292         | 173          | 163          | 172          | 172          |
| 293         | 174          | 164          | 173          | 173          |
| 294         | 175          | 165          | 174          | 174          |
| 295         | 176          | 166          | 175          | 175          |
| 296         | 177          | 167          | 176          | 176          |
| 297         | 178          | 168          | 177          | 177          |
| 298         | 179          | 169          | 178          | 178          |
| 299         | 180          | 170          | 179          | 179          |
| 300         | 181          | 171          | 180          | 180          |
| 301         | 172          | 66           | 11           | 11           |
| 302         | 173          | 172          | 182          | 182          |
| 303         | 174          | 173          | 183          | 183          |
| 304         | 175          | 174          | 184          | 184          |
| 305         | 176          | 175          | 185          | 185          |
| 306         | 177          | 176          | 186          | 186          |
| 307         | 178          | 177          | 187          | 187          |
| 308         | 179          | 178          | 188          | 188          |
| 309         | 180          | 179          | 189          | 189          |
| 310         | 181          | 180          | 190          | 190          |
| 311         | 182          | 172          | 11           | 11           |
| 312         | 183          | 173          | 182          | 182          |
| 313         | 184          | 174          | 183          | 183          |
| 314         | 185          | 175          | 184          | 184          |
| 315         | 186          | 176          | 185          | 185          |
| 316         | 187          | 177          | 186          | 186          |
| 317         | 188          | 178          | 187          | 187          |
| 318         | 189          | 179          | 188          | 188          |

| <u>ELEM</u> | <u>NODE1</u> | <u>NODE2</u> | <u>NODE3</u> | <u>NODE4</u> |
|-------------|--------------|--------------|--------------|--------------|
| 319         | 190          | 180          | 189          | 189          |
| 320         | 191          | 181          | 190          | 190          |
| 321         | 182          | 11           | 6            | 6            |
| 322         | 183          | 182          | 192          | 192          |
| 323         | 184          | 183          | 193          | 193          |
| 324         | 185          | 184          | 194          | 194          |
| 325         | 186          | 185          | 195          | 195          |
| 326         | 187          | 186          | 196          | 196          |
| 327         | 188          | 187          | 197          | 197          |
| 328         | 189          | 188          | 198          | 198          |
| 329         | 190          | 189          | 199          | 199          |
| 330         | 191          | 190          | 200          | 200          |
| 331         | 192          | 182          | 6            | 6            |
| 332         | 193          | 183          | 192          | 192          |
| 333         | 194          | 184          | 193          | 193          |
| 334         | 195          | 185          | 194          | 194          |
| 335         | 196          | 186          | 195          | 195          |
| 336         | 197          | 187          | 196          | 196          |
| 337         | 198          | 188          | 197          | 197          |
| 338         | 199          | 189          | 198          | 198          |
| 339         | 200          | 190          | 199          | 199          |
| 340         | 201          | 191          | 200          | 200          |
| 341         | 192          | 6            | 1            | 1            |
| 342         | 193          | 192          | 122          | 122          |
| 343         | 194          | 193          | 123          | 123          |
| 344         | 195          | 194          | 124          | 124          |
| 345         | 196          | 195          | 125          | 125          |
| 346         | 197          | 196          | 126          | 126          |
| 347         | 198          | 197          | 127          | 127          |
| 348         | 199          | 198          | 128          | 128          |
| 349         | 200          | 199          | 129          | 129          |
| 350         | 201          | 200          | 130          | 130          |
| 351         | 122          | 192          | 1            | 1            |
| 352         | 123          | 193          | 122          | 122          |
| 353         | 124          | 194          | 123          | 123          |
| 354         | 125          | 195          | 124          | 124          |
| 355         | 126          | 196          | 125          | 125          |
| 356         | 127          | 197          | 126          | 126          |
| 357         | 128          | 198          | 127          | 127          |
| 358         | 129          | 199          | 128          | 128          |
| 359         | 130          | 200          | 129          | 129          |
| 360         | 131          | 201          | 130          | 130          |
| 361         | 202          | 131          | 151          | 151          |
| 362         | 203          | 202          | 208          | 208          |
| 363         | 204          | 203          | 209          | 209          |
| 364         | 205          | 204          | 210          | 210          |
| 365         | 206          | 205          | 211          | 211          |
| 366         | 207          | 206          | 212          | 212          |
| 367         | 208          | 202          | 151          | 151          |
| 368         | 209          | 203          | 208          | 208          |
| 369         | 210          | 204          | 209          | 209          |
| 370         | 211          | 205          | 210          | 210          |
| 371         | 212          | 206          | 211          | 211          |

| <u>ELEM</u> | <u>NODE1</u> | <u>NODE2</u> | <u>NODE3</u> | <u>NODE4</u> |
|-------------|--------------|--------------|--------------|--------------|
| 372         | 213          | 207          | 212          | 212          |
| 373         | 208          | 151          | 171          | 171          |
| 374         | 209          | 208          | 214          | 214          |
| 375         | 210          | 209          | 214          | 214          |
| 376         | 211          | 210          | 216          | 216          |
| 377         | 212          | 211          | 217          | 217          |
| 378         | 213          | 212          | 218          | 218          |
| 379         | 214          | 208          | 171          | 171          |
| 380         | 215          | 209          | 214          | 214          |
| 381         | 216          | 210          | 215          | 215          |
| 382         | 217          | 211          | 216          | 216          |
| 383         | 218          | 212          | 217          | 217          |
| 384         | 219          | 213          | 218          | 218          |
| 385         | 214          | 171          | 191          | 191          |
| 386         | 215          | 214          | 220          | 220          |
| 387         | 216          | 215          | 221          | 221          |
| 388         | 217          | 216          | 222          | 222          |
| 389         | 218          | 217          | 223          | 223          |
| 390         | 219          | 218          | 224          | 224          |
| 391         | 220          | 214          | 191          | 191          |
| 392         | 221          | 215          | 220          | 220          |
| 393         | 222          | 216          | 221          | 221          |
| 394         | 223          | 217          | 222          | 222          |
| 395         | 224          | 218          | 223          | 223          |
| 396         | 225          | 219          | 224          | 224          |
| 397         | 220          | 191          | 131          | 131          |
| 398         | 221          | 220          | 202          | 202          |
| 399         | 222          | 221          | 203          | 203          |
| 400         | 223          | 222          | 204          | 204          |
| 401         | 224          | 223          | 205          | 205          |
| 402         | 225          | 224          | 206          | 206          |
| 403         | 202          | 220          | 131          | 131          |
| 404         | 203          | 221          | 202          | 202          |
| 405         | 204          | 222          | 203          | 203          |
| 406         | 205          | 223          | 204          | 204          |
| 407         | 206          | 224          | 205          | 205          |
| 408         | 207          | 225          | 206          | 206          |
| 409         | 352          | 353          | 357          | 357          |
| 410         | 353          | 354          | 232          | 232          |
| 411         | 354          | 355          | 233          | 233          |
| 412         | 355          | 356          | 234          | 234          |
| 413         | 353          | 232          | 357          | 232          |
| 414         | 354          | 233          | 232          | 232          |
| 415         | 355          | 234          | 233          | 233          |
| 416         | 356          | 358          | 234          | 234          |
| 417         | 357          | 232          | 359          | 359          |
| 418         | 232          | 245          | 242          | 242          |
| 419         | 233          | 237          | 238          | 233          |
| 420         | 234          | 358          | 240          | 240          |
| 421         | 232          | 242          | 359          | 242          |
| 422         | 232          | 233          | 245          | 233          |
| 423         | 233          | 234          | 237          | 233          |
| 424         | 358          | 360          | 240          | 240          |

| <u>ELEM</u> | <u>NODE1</u> | <u>NODE2</u> | <u>NODE3</u> | <u>NODE4</u> |
|-------------|--------------|--------------|--------------|--------------|
| 425         | 242          | 351          | 239          | 242          |
| 426         | 234          | 240          | 237          | 237          |
| 427         | 242          | 239          | 247          | 247          |
| 428         | 350          | 240          | 243          | 240          |
| 429         | 359          | 242          | 361          | 242          |
| 430         | 239          | 248          | 247          | 247          |
| 431         | 243          | 249          | 248          | 248          |
| 432         | 240          | 360          | 249          | 249          |
| 433         | 242          | 247          | 361          | 246          |
| 434         | 244          | 243          | 248          | 248          |
| 435         | 243          | 240          | 249          | 249          |
| 436         | 360          | 362          | 249          | 249          |
| 437         | 361          | 247          | 363          | 247          |
| 438         | 247          | 248          | 364          | 248          |
| 439         | 248          | 249          | 365          | 248          |
| 440         | 249          | 362          | 366          | 249          |
| 441         | 247          | 364          | 363          | 247          |
| 442         | 248          | 365          | 364          | 248          |
| 443         | 249          | 366          | 365          | 249          |
| 444         | 362          | 367          | 366          | 366          |
| 445         | 256          | 257          | 260          | 260          |
| 446         | 257          | 258          | 261          | 261          |
| 447         | 258          | 259          | 262          | 262          |
| 448         | 259          | 230          | 263          | 263          |
| 449         | 257          | 261          | 260          | 260          |
| 450         | 258          | 262          | 261          | 261          |
| 451         | 259          | 263          | 262          | 262          |
| 452         | 230          | 229          | 263          | 263          |
| 453         | 260          | 261          | 264          | 264          |
| 454         | 261          | 262          | 265          | 265          |
| 455         | 262          | 263          | 266          | 266          |
| 456         | 263          | 229          | 267          | 267          |
| 457         | 261          | 265          | 264          | 264          |
| 458         | 262          | 266          | 265          | 265          |
| 459         | 263          | 267          | 266          | 266          |
| 460         | 229          | 228          | 267          | 267          |
| 461         | 264          | 265          | 268          | 268          |
| 462         | 265          | 266          | 269          | 269          |
| 463         | 266          | 267          | 270          | 270          |
| 464         | 267          | 228          | 271          | 271          |
| 465         | 265          | 269          | 268          | 268          |
| 466         | 266          | 270          | 269          | 269          |
| 467         | 267          | 271          | 270          | 270          |
| 468         | 228          | 227          | 271          | 271          |
| 469         | 268          | 269          | 272          | 269          |
| 470         | 269          | 270          | 273          | 273          |
| 471         | 270          | 271          | 274          | 274          |
| 472         | 271          | 227          | 275          | 275          |
| 473         | 269          | 273          | 272          | 269          |
| 474         | 270          | 274          | 273          | 273          |
| 475         | 271          | 275          | 274          | 274          |
| 476         | 227          | 226          | 275          | 275          |
| 477         | 334          | 335          | 349          | 349          |

| <u>ELEM</u> | <u>NODE1</u> | <u>NODE2</u> | <u>NODE3</u> | <u>NODE4</u> |
|-------------|--------------|--------------|--------------|--------------|
| 478         | 335          | 336          | 277          | 277          |
| 479         | 336          | 337          | 278          | 278          |
| 480         | 337          | 338          | 279          | 279          |
| 481         | 349          | 335          | 277          | 277          |
| 482         | 336          | 278          | 277          | 277          |
| 483         | 337          | 279          | 278          | 278          |
| 484         | 338          | 339          | 279          | 279          |
| 485         | 348          | 349          | 277          | 277          |
| 486         | 277          | 278          | 282          | 282          |
| 487         | 278          | 279          | 283          | 283          |
| 488         | 279          | 339          | 284          | 284          |
| 489         | 277          | 282          | 348          | 282          |
| 490         | 278          | 283          | 282          | 282          |
| 491         | 279          | 284          | 283          | 283          |
| 492         | 339          | 340          | 284          | 284          |
| 493         | 347          | 348          | 282          | 282          |
| 494         | 282          | 283          | 287          | 287          |
| 495         | 283          | 284          | 288          | 288          |
| 496         | 284          | 340          | 289          | 289          |
| 497         | 282          | 287          | 347          | 287          |
| 498         | 283          | 288          | 287          | 287          |
| 499         | 284          | 289          | 288          | 288          |
| 500         | 340          | 341          | 289          | 289          |
| 501         | 347          | 287          | 346          | 287          |
| 502         | 287          | 288          | 345          | 288          |
| 503         | 344          | 288          | 289          | 289          |
| 504         | 341          | 343          | 289          | 289          |
| 505         | 345          | 346          | 287          | 287          |
| 506         | 344          | 345          | 288          | 288          |
| 507         | 343          | 344          | 289          | 289          |
| 508         | 341          | 342          | 343          | 343          |
| 509         | 308          | 273          | 309          | 308          |
| 510         | 273          | 274          | 296          | 296          |
| 511         | 274          | 275          | 297          | 297          |
| 512         | 275          | 226          | 298          | 298          |
| 513         | 273          | 296          | 309          | 309          |
| 514         | 274          | 297          | 296          | 296          |
| 515         | 275          | 298          | 297          | 297          |
| 516         | 226          | 231          | 298          | 298          |
| 517         | 309          | 296          | 310          | 310          |
| 518         | 296          | 297          | 299          | 299          |
| 519         | 297          | 298          | 300          | 300          |
| 520         | 298          | 231          | 301          | 301          |
| 521         | 296          | 299          | 310          | 310          |
| 522         | 297          | 300          | 299          | 299          |
| 523         | 298          | 301          | 300          | 300          |
| 524         | 231          | 236          | 301          | 301          |
| 525         | 310          | 299          | 311          | 311          |
| 526         | 299          | 300          | 302          | 302          |
| 527         | 300          | 301          | 303          | 303          |
| 528         | 301          | 236          | 304          | 304          |
| 529         | 299          | 302          | 311          | 311          |
| 530         | 300          | 303          | 302          | 302          |

| <u>ELEM</u> | <u>NODE1</u> | <u>NODE2</u> | <u>NODE3</u> | <u>NODE4</u> |
|-------------|--------------|--------------|--------------|--------------|
| 531         | 301          | 304          | 303          | 303          |
| 532         | 236          | 246          | 304          | 304          |
| 533         | 311          | 302          | 312          | 312          |
| 534         | 302          | 303          | 305          | 305          |
| 535         | 303          | 304          | 306          | 306          |
| 536         | 304          | 246          | 307          | 307          |
| 537         | 302          | 305          | 312          | 312          |
| 538         | 303          | 306          | 305          | 305          |
| 539         | 304          | 307          | 306          | 306          |
| 540         | 246          | 251          | 307          | 307          |
| 541         | 272          | 308          | 280          | 308          |
| 542         | 308          | 309          | 280          | 309          |
| 543         | 280          | 309          | 285          | 309          |
| 544         | 309          | 310          | 285          | 309          |
| 545         | 285          | 310          | 290          | 310          |
| 546         | 310          | 311          | 290          | 310          |
| 547         | 290          | 311          | 295          | 311          |
| 548         | 311          | 312          | 295          | 311          |
| 549         | 295          | 305          | 294          | 305          |
| 550         | 305          | 306          | 313          | 313          |
| 551         | 306          | 307          | 314          | 314          |
| 552         | 307          | 251          | 315          | 315          |
| 553         | 305          | 313          | 294          | 313          |
| 554         | 306          | 314          | 313          | 314          |
| 555         | 307          | 315          | 314          | 315          |
| 556         | 251          | 252          | 315          | 315          |
| 557         | 294          | 313          | 293          | 313          |
| 558         | 313          | 314          | 316          | 316          |
| 559         | 314          | 315          | 317          | 317          |
| 560         | 315          | 252          | 318          | 318          |
| 561         | 313          | 316          | 293          | 316          |
| 562         | 314          | 317          | 316          | 316          |
| 563         | 315          | 318          | 317          | 317          |
| 564         | 252          | 253          | 318          | 318          |
| 565         | 293          | 316          | 292          | 316          |
| 566         | 316          | 317          | 319          | 319          |
| 567         | 317          | 318          | 320          | 320          |
| 568         | 318          | 253          | 321          | 321          |
| 569         | 316          | 319          | 292          | 319          |
| 570         | 317          | 320          | 319          | 319          |
| 571         | 318          | 321          | 320          | 320          |
| 572         | 253          | 254          | 321          | 321          |
| 573         | 292          | 319          | 291          | 319          |
| 574         | 319          | 320          | 322          | 322          |
| 575         | 320          | 321          | 323          | 323          |
| 576         | 321          | 254          | 324          | 324          |
| 577         | 319          | 322          | 291          | 322          |
| 578         | 320          | 323          | 322          | 322          |
| 579         | 321          | 324          | 323          | 323          |
| 580         | 254          | 255          | 324          | 324          |
| 581         | 291          | 322          | 286          | 322          |
| 582         | 322          | 323          | 325          | 325          |
| 583         | 323          | 324          | 326          | 326          |

| <u>ELEM</u> | <u>NODE1</u> | <u>NODE2</u> | <u>NODE3</u> | <u>NODE4</u> |
|-------------|--------------|--------------|--------------|--------------|
| 584         | 324          | 255          | 327          | 327          |
| 585         | 322          | 325          | 286          | 322          |
| 586         | 323          | 326          | 325          | 325          |
| 587         | 324          | 327          | 326          | 326          |
| 588         | 255          | 250          | 327          | 327          |
| 589         | 286          | 325          | 281          | 325          |
| 590         | 325          | 326          | 328          | 328          |
| 591         | 326          | 327          | 329          | 329          |
| 592         | 327          | 250          | 330          | 330          |
| 593         | 325          | 328          | 281          | 325          |
| 594         | 326          | 329          | 328          | 328          |
| 595         | 327          | 330          | 329          | 329          |
| 596         | 250          | 241          | 330          | 330          |
| 597         | 281          | 328          | 276          | 328          |
| 598         | 328          | 329          | 331          | 331          |
| 599         | 329          | 330          | 332          | 332          |
| 600         | 330          | 241          | 333          | 333          |
| 601         | 328          | 331          | 276          | 328          |
| 602         | 329          | 332          | 331          | 331          |
| 603         | 330          | 333          | 332          | 332          |
| 604         | 241          | 235          | 333          | 333          |
| 605         | 276          | 331          | 256          | 331          |
| 606         | 331          | 332          | 257          | 257          |
| 607         | 332          | 333          | 258          | 258          |
| 608         | 333          | 235          | 259          | 259          |
| 609         | 331          | 257          | 256          | 256          |
| 610         | 332          | 258          | 257          | 257          |
| 611         | 333          | 259          | 258          | 258          |
| 612         | 235          | 230          | 259          | 259          |
| 613         | 256          | 335          | 334          | 334          |
| 614         | 260          | 264          | 335          | 335          |
| 615         | 264          | 268          | 336          | 336          |
| 616         | 268          | 272          | 337          | 337          |
| 617         | 349          | 256          | 334          | 334          |
| 618         | 256          | 260          | 335          | 335          |
| 619         | 264          | 336          | 335          | 335          |
| 620         | 268          | 337          | 336          | 336          |
| 621         | 272          | 338          | 337          | 337          |
| 622         | 276          | 256          | 349          | 349          |
| 623         | 338          | 272          | 339          | 339          |
| 624         | 272          | 280          | 339          | 339          |
| 625         | 281          | 276          | 349          | 349          |
| 626         | 281          | 349          | 348          | 348          |
| 627         | 339          | 280          | 340          | 340          |
| 628         | 280          | 285          | 340          | 340          |
| 629         | 286          | 281          | 348          | 348          |
| 630         | 286          | 348          | 347          | 347          |
| 631         | 340          | 285          | 341          | 341          |
| 632         | 285          | 290          | 341          | 341          |
| 633         | 291          | 286          | 347          | 347          |
| 634         | 291          | 347          | 346          | 346          |
| 635         | 295          | 342          | 341          | 341          |
| 636         | 290          | 295          | 341          | 341          |

| <u>ELEM</u> | <u>NODE1</u> | <u>NODE2</u> | <u>NODE3</u> | <u>NODE4</u> |
|-------------|--------------|--------------|--------------|--------------|
| 637         | 291          | 346          | 345          | 345          |
| 638         | 292          | 291          | 345          | 345          |
| 639         | 344          | 292          | 345          | 345          |
| 640         | 293          | 292          | 344          | 344          |
| 641         | 293          | 344          | 343          | 343          |
| 642         | 294          | 293          | 343          | 343          |
| 643         | 295          | 294          | 343          | 343          |
| 644         | 295          | 343          | 342          | 342          |
| 645         | 226          | 353          | 352          | 352          |
| 646         | 226          | 227          | 353          | 353          |
| 647         | 227          | 228          | 353          | 353          |
| 648         | 228          | 354          | 353          | 353          |
| 649         | 228          | 229          | 354          | 354          |
| 650         | 229          | 355          | 354          | 354          |
| 651         | 229          | 230          | 355          | 355          |
| 652         | 230          | 356          | 355          | 355          |
| 653         | 231          | 226          | 357          | 357          |
| 654         | 226          | 352          | 357          | 357          |
| 655         | 356          | 230          | 358          | 358          |
| 656         | 230          | 235          | 358          | 358          |
| 657         | 236          | 231          | 357          | 357          |
| 658         | 236          | 357          | 359          | 359          |
| 659         | 358          | 235          | 360          | 360          |
| 660         | 235          | 241          | 360          | 360          |
| 661         | 246          | 236          | 359          | 359          |
| 662         | 246          | 359          | 361          | 361          |
| 663         | 360          | 241          | 362          | 362          |
| 664         | 241          | 250          | 362          | 362          |
| 665         | 251          | 246          | 361          | 361          |
| 666         | 251          | 361          | 363          | 363          |
| 667         | 255          | 367          | 362          | 362          |
| 668         | 250          | 255          | 362          | 362          |
| 669         | 251          | 363          | 364          | 364          |
| 670         | 252          | 251          | 364          | 364          |
| 671         | 252          | 364          | 365          | 365          |
| 672         | 253          | 252          | 365          | 365          |
| 673         | 366          | 253          | 365          | 365          |
| 674         | 254          | 253          | 366          | 366          |
| 675         | 255          | 254          | 366          | 366          |
| 676         | 255          | 366          | 367          | 367          |
| 677         | 207          | 245          | 238          | 238          |
| 678         | 225          | 207          | 238          | 238          |
| 679         | 225          | 238          | 237          | 237          |
| 680         | 225          | 237          | 350          | 350          |
| 681         | 219          | 225          | 350          | 350          |
| 682         | 219          | 350          | 243          | 243          |
| 683         | 219          | 243          | 244          | 244          |
| 684         | 213          | 219          | 244          | 244          |
| 685         | 213          | 244          | 239          | 239          |
| 686         | 351          | 213          | 239          | 239          |
| 687         | 207          | 213          | 351          | 351          |
| 688         | 350          | 237          | 240          | 240          |
| 689         | 245          | 351          | 242          | 242          |

| <u>ELEM</u> | <u>NODE1</u> | <u>NODE2</u> | <u>NODE3</u> | <u>NODE4</u> |
|-------------|--------------|--------------|--------------|--------------|
| 690         | 272          | 269          | 308          | 269          |
| 691         | 295          | 312          | 294          | 312          |
| 692         | 239          | 244          | 248          | 248          |
| 693         | 245          | 233          | 238          | 233          |

### Two Dimensional Model Output

Time Step = 0.003 seconds

Convergence Criteria = 0.001

| <u>U0,cm/sec</u> | <u>Ui,cm/sec</u> | <u>Theta</u> | <u>d, um</u> | <u>D, cm</u> | MODEL<br><u>Eff.</u> | UNIFIED<br><u>MODEL EFF</u> |
|------------------|------------------|--------------|--------------|--------------|----------------------|-----------------------------|
| 250              | 125              | 0            | 5            | 1.00         | 1.17                 | 1.0420                      |
| 250              | 125              | 0            | 40           | 1.00         | 1.71                 | 1.7372                      |
| 500              | 125              | 0            | 5            | 1.00         | 1.55                 | 1.2268                      |
| 500              | 125              | 0            | 40           | 1.00         | 3.40                 | 3.5187                      |
| 1000             | 125              | 0            | 5            | 1.00         | 2.39                 | 1.9534                      |
| 1000             | 125              | 0            | 40           | 1.00         | 7.04                 | 7.3689                      |
| 250              | 250              | 0            | 5            | 1.00         | 0.99                 | 1.0000                      |
| 250              | 250              | 0            | 40           | 1.00         | 1.00                 | 1.0000                      |
| 500              | 250              | 0            | 5            | 1.00         | 1.20                 | 1.0806                      |
| 500              | 250              | 0            | 40           | 1.00         | 1.81                 | 1.8487                      |
| 1000             | 250              | 0            | 5            | 1.00         | 1.69                 | 1.4217                      |
| 1000             | 250              | 0            | 40           | 1.00         | 3.58                 | 3.7384                      |
| 250              | 500              | 0            | 5            | 1.00         | 0.89                 | 0.9710                      |
| 250              | 500              | 0            | 40           | 1.00         | 0.64                 | 0.6013                      |
| 500              | 500              | 0            | 5            | 1.00         | 0.99                 | 1.0000                      |
| 500              | 500              | 0            | 40           | 1.00         | 1.00                 | 1.0000                      |
| 1000             | 500              | 0            | 5            | 1.00         | 1.26                 | 1.1492                      |
| 1000             | 500              | 0            | 40           | 1.00         | 1.87                 | 1.9182                      |
| 250              | 1000             | 0            | 5            | 1.00         | 0.82                 | 0.9412                      |
| 250              | 1000             | 0            | 40           | 1.00         | 0.46                 | 0.3665                      |
| 500              | 1000             | 0            | 5            | 1.00         | 0.86                 | 0.9453                      |
| 500              | 1000             | 0            | 40           | 1.00         | 0.60                 | 0.5564                      |
| 1000             | 1000             | 0            | 5            | 1.00         | 0.99                 | 1.0000                      |
| 1000             | 1000             | 0            | 40           | 1.00         | 1.00                 | 1.0000                      |
| 250              | 125              | 0            | 5            | 0.32         | 1.25                 | 1.1205                      |
| 250              | 125              | 0            | 40           | 0.32         | 1.83                 | 1.8976                      |
| 500              | 125              | 0            | 5            | 0.32         | 1.83                 | 1.6106                      |
| 500              | 125              | 0            | 40           | 0.32         | 3.72                 | 3.8271                      |
| 1000             | 125              | 0            | 5            | 0.32         | 3.41                 | 3.3106                      |
| 1000             | 125              | 0            | 40           | 0.32         | 7.64                 | 7.7848                      |
| 250              | 250              | 0            | 5            | 0.32         | 0.98                 | 1.0000                      |
| 250              | 250              | 0            | 40           | 0.32         | 0.98                 | 1.0000                      |
| 500              | 250              | 0            | 5            | 0.32         | 1.25                 | 1.2150                      |
| 500              | 250              | 0            | 40           | 0.32         | 1.87                 | 1.9460                      |
| 1000             | 250              | 0            | 5            | 0.32         | 2.05                 | 2.0147                      |
| 1000             | 250              | 0            | 40           | 0.32         | 3.82                 | 3.9110                      |
| 250              | 500              | 0            | 5            | 0.32         | 0.85                 | 0.9194                      |
| 250              | 500              | 0            | 40           | 0.32         | 0.58                 | 0.5376                      |
| 500              | 500              | 0            | 5            | 0.32         | 0.98                 | 1.0000                      |
| 500              | 500              | 0            | 40           | 0.32         | 1.00                 | 1.0000                      |
| 1000             | 500              | 0            | 5            | 0.32         | 1.37                 | 1.3540                      |
| 1000             | 500              | 0            | 40           | 0.32         | 1.92                 | 1.9723                      |
| 250              | 1000             | 0            | 5            | 0.32         | 0.76                 | 0.8426                      |
| 250              | 1000             | 0            | 40           | 0.32         | 0.36                 | 0.2917                      |
| 500              | 1000             | 0            | 5            | 0.32         | 0.81                 | 0.8612                      |
| 500              | 1000             | 0            | 40           | 0.32         | 0.54                 | 0.5195                      |

# Two Dimensional Model Output

Time Step = 0.003 seconds

Convergence Criteria = 0.001

| <u>U0,cm/sec</u> | <u>U1,cm/sec</u> | <u>Theta</u> | <u>d, um</u> | <u>D, cm</u> | MODEL<br><u>Eff.</u> | UNIFIED<br><u>MODEL EFF</u> |
|------------------|------------------|--------------|--------------|--------------|----------------------|-----------------------------|
| 1000             | 1000             | 0            | 5            | 0.32         | 0.98                 | 1.0000                      |
| 1000             | 1000             | 0            | 40           | 0.32         | 0.99                 | 1.0000                      |
| 250              | 125              | 30           | 5            | 0.32         | 0.87                 | 1.0405                      |
| 250              | 125              | 30           | 40           | 0.32         | 1.28                 | 1.6996                      |
| 0                |                  | 30           | 0            | 0.32         |                      | ERR                         |
| 500              | 125              | 30           | 40           | 0.32         | 2.63                 | 3.4431                      |
| 0                |                  | 30           | 0            | 0.32         |                      | ERR                         |
| 1000             | 125              | 30           | 40           | 0.32         | 5.58                 | 6.9082                      |
| 250              | 250              | 30           | 5            | 0.32         | 0.89                 | 0.9918                      |
| 250              | 250              | 30           | 40           | 0.32         | 0.72                 | 0.8711                      |
| 500              | 250              | 30           | 5            | 0.32         | 0.67                 | 1.0796                      |
| 500              | 250              | 30           | 40           | 0.32         | 1.32                 | 1.7273                      |
| 0                |                  | 30           | 0            | 0.32         |                      | ERR                         |
| 1000             | 250              | 30           | 40           | 0.32         | 2.79                 | 3.4572                      |
| 250              | 500              | 30           | 5            | 0.32         | 0.85                 | 0.9595                      |
| 250              | 500              | 30           | 40           | 0.32         | 0.43                 | 0.4494                      |
| 500              | 500              | 30           | 5            | 0.32         | 0.76                 | 0.9842                      |
| 500              | 500              | 30           | 40           | 0.32         | 0.74                 | 0.8664                      |
| 0                |                  | 30           | 0            | 0.32         |                      | ERR                         |
| 1000             | 500              | 30           | 40           | 0.32         | 1.55                 | 1.7308                      |
| 250              | 1000             | 30           | 5            | 0.32         | 0.81                 | 0.9299                      |
| 250              | 1000             | 30           | 40           | 0.32         | 0.30                 | 0.2305                      |
| 500              | 1000             | 30           | 5            | 0.32         | 0.77                 | 0.9241                      |
| 500              | 1000             | 30           | 40           | 0.32         | 0.45                 | 0.4320                      |
| 1000             | 1000             | 30           | 5            | 0.32         | 0.70                 | 0.9704                      |
| 1000             | 1000             | 30           | 40           | 0.32         | 0.81                 | 0.8660                      |
| 250              | 125              | 30           | 5            | 1            | 0.79                 | 1.0130                      |
| 250              | 125              | 30           | 40           | 1            | 0.75                 | 1.5025                      |
| 500              | 125              | 30           | 5            | 1            | 0.89                 | 1.0825                      |
| 500              | 125              | 30           | 40           | 1            | 0.90                 | 3.1660                      |
| 1000             | 125              | 30           | 0            | 1            |                      | 1.3858                      |
| 1000             | 125              | 30           | 40           | 1            | 4.09                 | 6.7622                      |
| 250              | 250              | 30           | 5            | 1            | 0.95                 | 0.9973                      |
| 250              | 250              | 30           | 40           | 1            | 0.63                 | 0.9063                      |
| 500              | 250              | 30           | 5            | 1            | 0.97                 | 1.0260                      |
| 500              | 250              | 30           | 40           | 1            | 1.05                 | 1.6474                      |
| 1000             | 250              | 30           | 0            | 1            | 0.53                 | 1.1649                      |
| 1000             | 250              | 30           | 40           | 1            | 2.16                 | 3.3993                      |
| 250              | 500              | 30           | 5            | 1            | 0.90                 | 0.9863                      |
| 250              | 500              | 30           | 40           | 1            | 0.48                 | 0.5927                      |
| 500              | 500              | 30           | 5            | 1            | 0.89                 | 0.9947                      |
| 500              | 500              | 30           | 40           | 1            | 0.66                 | 0.8803                      |
| 1000             | 500              | 30           | 0            | 1            | 0.85                 | 1.0516                      |
| 1000             | 500              | 30           | 40           | 1            | 1.17                 | 1.7150                      |
| 250              | 1000             | 30           | 5            | 1            | 0.87                 | 0.9748                      |
| 250              | 1000             | 30           | 40           | 1            | 0.41                 | 0.4184                      |

### Two Dimensional Model Output

Time Step = 0.003 seconds

Convergence Criteria = 0.001

| <u>U0,cm/sec</u> | <u>Ui,cm/sec</u> | <u>Theta</u> | <u>d, um</u> | <u>D, cm</u> | <u>MODEL Eff.</u> | <u>UNIFIED MODEL EFF</u> |
|------------------|------------------|--------------|--------------|--------------|-------------------|--------------------------|
| 500              | 1000             | 30           | 5            | 1            | 0.87              | 0.9733                   |
| 500              | 1000             | 30           | 40           | 1            | 0.45              | 0.4862                   |
| 1000             | 1000             | 30           | 5            | 1            | 0.86              | 0.9896                   |
| 1000             | 1000             | 30           | 40           | 1            | 0.69              | 0.8685                   |

Uo = wind velocity

Ui = inlet velocity

theta = sampling angle

d = particle diameter

D = inlet diameter

3D RESULTS FOR SQUARE INLET OF 1/2 AREA = TO .5555 CM<sup>2</sup>  
 DIN = DIAMETER OF EQUAL AREA CIRCLE AND IS USED TO CALCULATE  
 STOKES NUMBER

INLET AREA: 0.56

| UO   | UI   | THETA | DP | DT<br>SEC | CVC   | TRAJ<br>AREA | BIEM<br>Ai | UNIFIED<br>Ai | %<br>DIFF |
|------|------|-------|----|-----------|-------|--------------|------------|---------------|-----------|
| 1000 | 125  | 0     | 5  | 0.0003    | 0.001 | 0.112        | 1.6158     | 1.8193        | -11%      |
| 1000 | 125  | 0     | 5  | 0.001     | 0.05  | 0.128        | 1.8483     | 1.8193        | 2%        |
| 1000 | 125  | 0     | 5  | 0.0003    | 0.05  | 0.191        | 2.7491     | 1.8193        | 51%       |
| 1000 | 125  | 0     | 40 | 0.0003    | 0.001 | 0.53         | 7.6309     | 7.2619        | 5%        |
| 1000 | 125  | 0     | 40 | 0.001     | 0.001 | 0.458        | 6.5928     | 7.2619        | -9%       |
| 250  | 1000 | 0     | 5  | 0.001     | 0.01  | 1.733        | 0.7799     | 0.9500        | -18%      |
| 250  | 1000 | 0     | 5  | 0.0005    | 0.05  | 1.928        | 0.8677     | 0.9500        | -9%       |
| 250  | 1000 | 0     | 40 | 0.001     | 0.01  | 0.963        | 0.4334     | 0.3846        | 13%       |
| 250  | 1000 | 0     | 40 | 0.0005    | 0.01  | 0.944        | 0.4248     | 0.3846        | 10%       |
| 250  | 1000 | 0     | 40 | 0.0003    | 0.05  | 0.929        | 0.4183     | 0.3846        | 9%        |
| 1000 | 125  | 30    | 5  | 0.0003    | 0.01  | 0.053        | 0.7574     | 1.3243        | -43%      |
| 1000 | 125  | 30    | 5  | 0.0005    | 0.05  | 0.056        | 0.8134     | 1.3243        | -39%      |
| 1000 | 125  | 30    | 40 | 0.0005    | 0.05  | 0.488        | 7.0240     | 6.6768        | 5%        |
| 250  | 1000 | 30    | 5  | 0.002     | 0.05  | 1.654        | 0.7446     | 0.9786        | -24%      |
| 250  | 1000 | 30    | 5  | 0.0005    | 0.05  | 1.933        | 0.8698     | 0.9786        | -11%      |
| 250  | 1000 | 30    | 40 | 0.001     | 0.01  | 0.58         | 0.2611     | 0.4622        | -44%      |
| 250  | 1000 | 30    | 40 | 0.0005    | 0.01  | 0.928        | 0.4177     | 0.4622        | -10%      |
| 250  | 1000 | 30    | 40 | 0.0003    | 0.05  | 0.929        | 0.4183     | 0.4622        | -10%      |
| 500  | 500  | 0     | 5  | 0.0005    | 0.05  | 0.57         | 1.0267     | 1.0000        | 3%        |
| 500  | 500  | 30    | 5  | 0.0005    | 0.05  | 0.554        | 0.9976     | 0.9955        | 0%        |
| 500  | 250  | 0     | 5  | 0.0005    | 0.05  | 0.314        | 1.1321     | 1.0686        | 6%        |
| 500  | 250  | 30    | 5  | 0.0005    | 0.05  | 0.305        | 1.0972     | 1.0219        | 7%        |
| 500  | 125  | 0     | 5  | 0.0005    | 0.05  | 0.209        | 1.5023     | 1.1936        | 26%       |
| 500  | 125  | 0     | 5  | 0.0005    | 0.05  | 0.172        | 1.2408     | 1.1936        | 4%        |
| 500  | 125  | 30    | 5  | 0.0005    | 0.05  | 0.164        | 1.1801     | 1.0693        | 10%       |
| 500  | 500  | 0     | 40 | 0.0005    | 0.05  | 0.573        | 1.0322     | 1.0000        | 3%        |
| 500  | 500  | 30    | 40 | 0.0005    | 0.05  | 0.528        | 0.9506     | 0.8856        | 7%        |
| 500  | 250  | 0     | 40 | 0.0005    | 0.05  | 0.544        | 1.9568     | 1.8251        | 7%        |
| 500  | 250  | 30    | 40 | 0.0005    | 0.05  | 0.496        | 1.7873     | 1.6177        | 10%       |
| 500  | 125  | 0     | 40 | 0.0005    | 0.05  | 0.516        | 3.7151     | 3.4445        | 8%        |
| 500  | 125  | 30    | 40 | 0.0005    | 0.05  | 0.439        | 3.1646     | 3.0644        | 3%        |

Uo = wind velocity

Ui = inlet velocity

theta = sampling angle

DP = particle diameter

DT = time step

cvc = convergence criteria

TRAJ AREA = area Ao in freestream

BIEM Ai = aspiration efficiency from model

UNIFIED Ai = aspiration efficiency from Hangel (1990)

## Two Dimensional Particle Tracking Program

C THIS PROGRAM COMBINES THE 2 DIMENSIONAL BIEM SOLUTION PROGRAM  
 C WRITTEN BY LIGGET AND LIU (1983) WITH THE ANALYTICAL SOLUTION  
 C TO THE PARTICLE EQUATIONS OF MOTION PROPOSED BY ALENIOUS  
 C (1989) TO PLOT PARTICLE TRAJECTORIES INTO AN INLET.  
 C

```

      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION RLN(100,100),RN(100,100),F(100),ID(100),
    $TITLE(20),CRITX(2),CRITY(2)
      COMMON H(100,100),RHS(100),N,X(100),Y(100),XXNOW,
    $YYNOW,P(100),PN(100),PN1(5),PN2(5),ICT(100),
    $DPDY(600),DPDX(600),A,L,N1,N2,UO,DIN,DP,THETA,UI,XCRIT
    $,YCRIT,DT,XXNEW,YYNEW,XNOW1,YNOW1
      READ (7,7) TITLE
    7 FORMAT (20A4)
      PRINT 7777,TITLE
    7777 FORMAT (//1X,20A4)
      READ (7,*) KEY
      READ (7,*) N
      MP=N
      N1=1
      IF (KEY .EQ. 1) PRINT 6933,N
    6933 FORMAT('  NUMBER OF NODES  =' ,I3/)
      IF (KEY .EQ. 1) PRINT 6938
    6938 FORMAT('.....KEY TO INPUT DATA AND ID NUMBERS.....
    1 '/' X,Y ARE THE NODAL COORDINATES '/' P,PN ARE THE
    2POTENTIAL AND THE NORMAL DERIVATIVE AT THE NODE')
      IF (KEY .EQ. 1) PRINT 6939
    6939 FORMAT ('  ID: IDENTIFIER FOR THE NODAL UNKNOWNNS'/,
    1 '                                =2 FOR P GIVEN'/,
    2 '                                =3 FOR PN GIVEN'/)
      IF (KEY .EQ. 1) PRINT 6941
    6941 FORMAT('  ICT: NODAL CORNER TYPE IDENTIFIER'/,
    1 '                                =1 FOR PN GIVEN APPROACHING NODE'/,
    2 '                                =2 FOR PN GIVEN LEAVING NODE'/,
    3 '.....ALL MOVEMENTS IN CLOCWISE SENSE.....'/)

```

C  
 C NODAL DATA I/O  
 C

```

    6932 L=N+1
      IF (KEY.EQ.1) PRINT 9953
    9953 FORMAT('  POINT',7X,'X-COOD',6X,'Y-COOD',5X,
    1'POTENTIAL',3X,'NORM DERV',3X,'ID',8X,'ICT')
      DO 6946 I=2,L
      READ (7,*) X(I),Y(I),P(I),PN(I),ID(I),ICT(I)
      IM1=I-1
    6946 IF (KEY.EQ.1) PRINT 9950,IM1,X(I),Y(I),P(I),PN(I),
    $ID(I),ICT(I)
    9950 FORMAT(I5,5X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,I5,
    15X,I5)

```

C

C CLEARING THE GLOBAL MATRIX  
C

```
DO 17 I=1,N
DO 16 J=1,L
16 H(I,J)=0.0
17 F(I)=0.0
```

C  
C BOUNDARY INTEGRATION  
C

```
X(1)=X(L)
X(L+1)=X(2)
Y(1)=Y(L)
Y(L+1)=Y(2)
DO 200 I=2,L
II1=1
II2=1
PNJP=0.0
PJP=0.0
DO 140 J=2,L
IF(I.EQ.J) GOTO 130
IF(I.EQ.J+1) GOTO 135
IF(J.NE.L) GOTO 98
IF(I.EQ.2) GO TO 135
98 R1=DSQRT((X(J+1)-X(J))**2+(Y(J+1)-Y(J))**2)
CO=(X(J+1)-X(J))/R1
SI=(Y(J+1)-Y(J))/R1
XIA=(Y(J)-Y(I))*SI+(X(J)-X(I))*CO
XIB=(Y(J+1)-Y(I))*SI+(X(J+1)-X(I))*CO
ETA=DABS((Y(I)-Y(J))*CO-(X(I)-X(J))*SI)
100 IF(ETA.LT.1.E-5) GO TO 102
BTN=DATAN(XIB/ETA)
ATN=DATAN(XIA/ETA)
105 ASQ=XIA**2+ETA**2
ALN=DLOG(ASQ)
BSQ=XIB**2+ETA**2
106 BLN=DLOG(BSQ)
107 ONE=BSQ*(BLN-1.)-ASQ*(ALN-1.)
TWO=XIB*BLN-XIA*ALN-2.*(XIB-XIA)+2.*ETA*(BTN-ATN)
SIGNRN=-(X(J)-X(I))*(Y(J+1)-Y(J))+(X(J+1)-X(J))*(Y(J)-
$Y(I))
PJ=(XIB*(BTN-ATN)-.5*ETA*(BLN-ALN))/(XIB-XIA)
PJ=DSIGN(PJ,SIGNRN)
XBXA=DABS(XIB-XIA)
PNJ=TWO*XIB/(2.*XBXA)-ONE/(4.*XBXA)
IF(ICT(J).NE.1) GOTO 321
PN1(II1)=PN(J)
F(I-1)=F(I-1)+PNJP*PN(J)
II1=II1+1
PNJP=0.0
321 IF(ICT(J).NE.2) GOTO 432
F(I-1)=F(I-1)+PNJ*PN(J)
PN2(II2)=PN(J)
II2=II2+1
PNJ=0.0
```

```

432 RN(I,J)=PJ+PJP
    RLN(I,J)=PNJ+PNJP
    PJP=(ETA*(BLN-ALN)*0.5-XIA*(BTN-ATN))/(XIB-XIA)
    PJP=DSIGN(PJP,SIGNRN)
    PNJP=ONE/(4.*XBXA)-TWO*XIA/(2.*XBXA)
    GO TO 140
102 BTN=0.0
    ATN=0.0
    GO TO 105
130 ALN=0.0
    BTN=0.0
    ATN=0.0
    XIB=DSQRT((X(I+1)-X(I))**2+(Y(I+1)-Y(I))**2)
    XIA=0.0
    ETA=0.0
    ASQ=0.0
    BSQ=XIB**2
    GO TO 106
135 XIA=-DSQRT((X(I)-X(J))**2+(Y(I)-Y(J))**2)
    BTN=0.0
    ATN=0.0
    XIB=0.0
    BLN=0.0
    BSQ=0.0
    ASQ=XIA**2
    ALN=DLOG(ASQ)
    ETA=0.0
    GOTO 107
140 CONTINUE
    RN(I,2)=RN(I,2)+PJP
    IF(ICT(2).NE.1) GOTO 544
    F(I-1)=F(I-1)+PNJP*PN1(1)
    PNJP=0.0
544 RLN(I,2)=RLN(I,2)+PNJP
    ALPHA=((X(I)-X(I-1))**2+(Y(I)-Y(I-1))**2+
$ (X(I)-X(I+1))**2+(Y(I)-Y(I+1))**2-(X(I+1)-
$ X(I-1))**2-(Y(I+1)-Y(I-1))**2)/(2.*DSQRT((X(I)
$ -X(I-1))**2+(Y(I)-Y(I-1))**2)*(X(I)-X(I+1
$ ))**2+(Y(I)-Y(I+1))**2)))
    IF(DABS(ALPHA).LT.0.9999) GOTO 141
    ALPHA=3.14159265
    GOTO 170
141 ALPHA=DACOS(ALPHA)
    IF (DABS(X(I)-X(I-1)).GT.1.E-5) GOTO 144
    IF(Y(I).LT.Y(I-1)) GOTO 142
    IF(X(I+1).LT.X(I)) ALPHA=6.28318531-ALPHA
    GOTO 170
142 IF (X(I+1).GT.X(I)) ALPHA=6.28318531-ALPHA
    GOTO 170
144 A=(Y(I-1)-Y(I))/(X(I-1)-X(I))
    B=Y(I)-A*X(I)
    YA=A*X(I+1)+B
    IF(Y(I) .LT. Y(I-1)) GOTO 150
    IF(X(I) .LT. X(I-1)) GOTO 146

```

```

        IF(YA .LT. Y(I+1))ALPHA=6.28318531-ALPHA
        GOTO 170
146  IF(YA .GT. Y(I+1))ALPHA=6.28318531-ALPHA
        GOTO 170
150  IF(X(I) .GT. X(I-1)) GOTO 152
        IF(YA .GT. Y(I+1))ALPHA=6.28318532-ALPHA
        GOTO 170
152  IF(YA .LT. Y(I+1))ALPHA=6.28318531-ALPHA
170  RN(I,I)=-ALPHA
200  CONTINUE
    13  FORMAT(//)
C
C      ASSEMBLING THE GLOBAL EQUATIONS
C
    DO 1000 I=1,N
    DO 2000 J=1,N
    IF(ID(J+1) .EQ. 2)H(I,J)=-RLN(I+1,J+1)
    IF(ID(J+1) .EQ. 3)H(I,J)=RN(I+1,J+1)
2000  CONTINUE
C
C      COMPUTING R-H-S VECTOR
C
    A=0.0
    DO 2001 J=1,N
    IF(ICT(J) .NE. 0)PN(J+1)=0.0
2001  A=A-P(J+1)*RN(I+1,J+1)+PN(J+1)*RLN(I+1,J+1)
    H(I,L)=A+F(I)
1000  CONTINUE
C
C      SOLVING GLOBAL EQS FOR NODAL UNKNOWNNS
C
    CALL EQSOL
C
C      ALLOCATING EACH SOLUTION TO THE PROPER NODAL UNKNOWN
C
    JJ1=1
    DO 8003 J=2,L
    IF(ID(J) .EQ. 2) GOTO 8005
    P(J)=RHS(JJ1)
    JJ1=JJ1+1
    GOTO 8003
8005  PN(J)=RHS(JJ1)
    JJ1=JJ1+1
8003  CONTINUE
C
C      OUTPUT BOUNDARY SOLUTION
C
    WRITE (6,13)
    WRITE (6,30)
30  FORMAT('.....COMPUTED RESULTS.....')
    do 7654 j=2,l
    write (6,*) j-1,pn(j),p(j)
7654  continue
C

```

```

C      PERFORM TRAJECTORY CALCULATION
C
C      INPUT PARTICLE AND AIRSTREAM VARIABLES AND CONV. PARAMETERS
C
      DO 4000 K=1,2
        IF (K .EQ. 1) THEN
          A=1.
        ELSE IF (K .EQ. 2) THEN
          A=-1.
        ELSE
          ENDIF
        CALL PART
        CRITX(K)=XCRIT
        CRITY(K)=YCRIT
4000 CONTINUE
      EFF = (((CRITX(1)-CRITX(2))**2+(CRITY(1)-CRITY(2))**2)**.5
      $*(-1)*UO)/(UI*DIN)
      WRITE (3,*) UO,UI,DP,DIN,THETA,EFF
      PRINT *, EFF
      STOP
      END

C
C      LINEAR EQUATION SOLVER SUBROUTINE
C
      SUBROUTINE EQSOL
      IMPLICIT REAL *8(A-H,O-Z)
      DIMENSION M(100)
      COMMON A(100,100),X(100),N,X1(100),Y(100),XXNOW,
      $YYNOW,P(100),PN(100),PN1(5),PN2(5),ICT(100),
      $DPDY(600),DPDX(600),A1,L,N1,N2,UO,DIN,DP,THETA,UI
      $,XCRIT,YCRIT,DT,XXNEW,YYNEW,XNOW1,YNOW1
      DO 5 I=1,N
        M(I)=1
        AMAX=A(I,1)
        DO 2 J=2,N
          IF(DABS(A(I,J)) .LE. DABS(AMAX)) GOTO 2
100    AMAX=A(I,J)
        M(I)=J
      2 CONTINUE
        IF(AMAX .EQ. 0) GOTO 98
      3 NN=N+1
        DO 4 J=1,NN
          A(I,J)=A(I,J)/AMAX
          DO 5 IP=1,N
            IF(IP .EQ. I) GOTO 5
            MMM=M(I)
            ZMULT=A(IP,MMM)
            DO 6 J=1,NN
              IF(J .NE. MMM) GOTO 9
            8 A(IP,J)=0.0
              GOTO 6
            9 A(IP,J)=A(IP,J)-ZMULT*A(I,J)
            6 CONTINUE
            5 CONTINUE

```

```

      DO 7 I=1,N
      NO=M(I)
      7 X(NO)=A(I,NN)
      RETURN
98  CONTINUE
      STOP
      END

```

C  
C  
C

SUBROUTINE FOR PARTICLE TRAJ. CALC., 2D

```

SUBROUTINE PART
  IMPLICIT REAL *8(A-H,O-Z)
  DIMENSION DPDXOL(600),DPDYOL(600),MAXI(150)
  COMMON H(100,100),RHS(100),N,X(100),Y(100),XXNOW,
$YYNOW,P(100),PN(100),PN1(5),PN2(5),ICT(100),
$DPDY(600),DPDX(600),A,L,N1,N2,UO,DIN,DP,THETA,UI
$,XCRIT,YCRIT,DT,XXNEW,YYNEW,XNOW1,YNOW1

```

C C C C C C C C C C C C C C C C

VARIABLE IDENTIFICATION

UO = INITIAL FREE STREAM VELOCITY

G = GRAVITY

DP = PARTICLE DIAMETER, CM

AV = AIR VISCOSITY

DIN = INLET DIAMETER

DT = TIME STEP LENGTH

PDENS = PARTICLE DENSITY

ADENS = AIR DENSITY

VR = PARTICLE RELATIVE VELOCITY

RE = REYNOLDS NUMBER

DPDX, DPDY = AIR VELOCITY COMPONENTS

VXP, VYP = PARTICLE VELOCITY COMPONENTS

XX, YY = PARTICLE LOCATION COORDINATES

THETA = NOZZLE ANGLE

```
READ (8,*) UO,DP,DIN,DT,THETA,UI,XDIN
```

READ (8,\*) CVC

```
PRINT *, UO, DP, DIN, DT, THETA, UI, XDIN, CVC
```

$$G = -980.$$
$$dt_1 = dt$$
 $AV = .000183$ 

PDENS=1.0

ADENS=.001205

$$DPL = 10000.0 * DP$$
$$CC=1+((2.*0.07/DPL)*(1.257+(0.4*EXP(-0.55*DPL/0.07))))$$
$$TAU = (DP^{**}2 * PDENS * CC) / (18 * AV)$$
$$\text{THETA1} = -.01745277 * \text{THETA}$$
$$\text{THETA} = .01745277 * \text{THETA}$$
$$GX = G * \sin(\theta)$$
$$GY = G * \cos(\theta)$$

MANY=599

N MANY=50

$$N_2 = 0$$

```
IF (A .GT. 0.) THEN
```

```

YYNOW=XDIN*DIN*SIN(THETA1)
XXNOW=XDIN*DIN*COS(THETA)
ELSE
YYNOW= XDIN*DIN*SIN(THETA1)+(.67*XDIN*DIN*COS(THETA)*A)
XXNOW=XDIN*DIN*COS(THETA)+(.67*XDIN*DIN*SIN(THETA)*A)
ENDIF
NCAPF=0
IF (A .GT. 0) NFCAP=0
NMISS=0
CONV=9999
N1=1

```

C  
C  
C

# PARTICLE POSITION AND VELOCITY COMPONENTS

```

DO 5000 J=1,NMANY
  NCAP=0
  N1=1
  N2=N2+1
  DO 4300 I=1,MANY
    CALL INSOLN
    IF (I .EQ. 1) THEN
      VXNOW=DPDX(I)+(TAU*GX)
      VYNOW=DPDY(I)+(TAU*GY)
    ENDIF
    VR=((VXNOW-DPDX(I))**2+(VYNOW-DPDY(I))**2)**0.5
    RE=VR*DP*ADENS/AV
    IF (RE .LE. 0.5) THEN
      QRE=1
    ELSE IF ((RE .GT. 0.5) .AND. (RE .LE. 800.)) THEN
      QRE=1+(0.15*RE**0.687)
    ELSE
      QRE=0.44*RE/24
    ENDIF
    IF (I .EQ. 1) THEN
      DPDXOL(I)=DPDX(I)
      DPDYOL(I)=DPDY(I)
    ELSE
      DPDXOL(I)=DPDX(I-1)
      DPDYOL(I)=DPDY(I-1)
    ENDIF
    if ((xxnow .lt. 1.5*din) .and. (xxnow .gt. -1.5*din)
$      .and. (yynew .lt. 1.5*din) .and.
$      (yynew .gt. -1.5*din)) then
      dt=dt1/10.
    else
      dt=dt1
    endif
    UAX=(DPDX(I)-DPDXOL(I))/DT
    UAY=(DPDY(I)-DPDYOL(I))/DT
    UBX=DPDX(I)-((UAX-GX)*(TAU/QRE))
    UBY=DPDY(I)-((UAY-GY)*(TAU/QRE))
    VXNEW=(UAX*DT)+UBX-((UBX-VXNOW)*
$      EXP(-QRE*DT/TAU))
  
```

```

VYNEW=(UAY*DT)+UBY-((UBY-VYNOW)*
$   EXP(-QRE*DT/TAU))
XXNEW=XXNOW+(UAX/2*DT**2)+(UBX*DT)-
$   ((UBX-VXNOW)*(TAU/QRE)*(1-EXP(-QRE*DT/TAU)))
YYNEW=YYNOW+(UAY/2*DT**2)+(UBY*DT)-
$   ((UBY-VYNOW)*(TAU/QRE)*(1-EXP(-QRE*DT/TAU)))
IF (I .EQ. 1) THEN
XNOW1=XXNOW
YNOW1=YYNOW
ENDIF
5002 WRITE (4,5002) XXNOW,YYNOW
      FORMAT (2F15.10)
      IF (XXNEW .LT. -1.1*XDIN/2*DIN) THEN
        NCAP=2
        PRINT *, 'PARTICLE EXIT REAR BOUNDARY, MISS'
        GOTO 4301
      ELSE IF ((YYNEW .GT. 1.1*XDIN*DIN) .OR. (YYNEW .LT.
$   -1.1*XDIN*DIN)) THEN
        NCAP=2
        PRINT *, 'PARTICLE EXIT TOP/BOTTOM, MISS'
        GOTO 4301
      ELSE IF ((XXNEW .LT. 0.0) .AND. (XXNOW .GT. 0.0)
$   .AND. (ABS(YYNEW) .LT. DIN/2)) THEN
        NCAP=1
        PRINT *, 'CAPTURE AT FACE'
        GOTO 4301
      ELSE IF ((ABS(YYNEW) .LT. DIN/2) .AND.
$   (ABS(YYNOW) .GT. DIN/2) .AND. (XXNEW .LT. 0))
$   THEN
        NCAP=2
        PRINT *, 'CONTACT TUBE, MISS'
        GOTO 4301
      ELSE
        NCAP=0
      ENDIF
      N1=N1+1
      XXNOW=XXNEW
      YYNOW=YYNEW
      VXNOW=VXNEW
      VYNOW=VYNEW
4300 CONTINUE
      PRINT *, 'ERROR, DID NOT EXIT LOOP'
      PRINT *, XNOW1,YNOW1,N1
4301 CONTINUE
      MAXI(J)=N1
      IF (NFCAP .GT. 0) THEN
        GOTO 50
      ELSE IF (NCAP .EQ. 1) THEN
        NFCAP=1
        YCAP=YNOW1
        XCAP=XNOW1
        YYNOW=YNOW1+(.67*XDIN*DIN*COS(THETA))
        XXNOW=XNOW1-((YNOW1-YYNOW)*TAN(THETA))
        GOTO 4999

```

```

ELSE IF ((NCAP .EQ. 2) .AND. (YYNOW .LE. -DIN/2.)) THEN
  YYNOW=YNOW1+(.1*DIN)
  XXNOW=XNOW1+((YYNOW-YNOW1)*TAN(THETA))
  GOTO 4999
ELSE IF ((NCAP .EQ. 2) .AND. (YYNOW .GE. DIN/2.)) THEN
  YYNOW=YNOW1-(.1*DIN)
  XXNOW=XNOW1-((YNOW1-YYNOW)*TAN(THETA))
  GOTO 4999
ELSE
ENDIF
50 IF (NMISS .GT. 0) THEN
  GOTO 10
ELSE IF (NCAP .EQ. 2) THEN
  NMISS=1
  YMISS=YNOW1
  XMISS=XNOW1
  GOTO 10
ELSE IF (NCAP .EQ. 1) THEN
  YYNOW=YNOW1+(A*.1*DIN)
  XXNOW=XNOW1+((YYNOW-YNOW1)*TAN(THETA))
  XDIN=XDIN+(.1*XDIN)
  GOTO 4999
ELSE
ENDIF
10 IF (NCAP .EQ. 1) THEN
  YCAP=YNOW1
  XCAP=XNOW1
ELSE IF (NCAP .EQ. 2) THEN
  YMISS=YNOW1
  XMISS=XNOW1
ELSE
ENDIF
CONV=((XMISS-XCAP)**2+(YMISS-YCAP)**2)**.5
IF (YCAP .GT. YMISS) THEN
  YYNOW=(CONV/2*COS(THETA))+YMISS
  XXNOW=(CONV/2*SIN(THETA))+XMISS
ELSE IF (YMISS .GT. YCAP) THEN
  YYNOW=(CONV/2*COS(THETA))+YCAP
  XXNOW=(CONV/2*SIN(THETA))+XCAP
ELSE
ENDIF
IF (CONV .LT. CVC) THEN
  XCRIT=XXNOW
  YCRIT=YYNOW
  GOTO 5001
ELSE
ENDIF
4999 PRINT *, N2,NCAP,YNOW1,DPDX(1),DPDY(1)
5000 CONTINUE
5001 PRINT *, 'TRAJECTORIES HAVE CONVERGED'
PRINT *, UC,DP,DIN,THETA,XCRIT,YCRIT
RETURN
STOP
END

```

```

C
C      SUBROUTINE TO CALC. VELOCITY AT PARTICLE LOCATION
C
      SUBROUTINE INSOLN
      IMPLICIT REAL *8(A-H,O-Z)
      DIMENSION U(600)
      COMMON H(100,100),RHS(100),N,X(100),Y(100),XXNOW,
$YYNOW,P(100),PN(100),PN1(5),PN2(5),ICT(100),
$DPDY(600),DPDX(600),A,L,N1,N2,UO,DIN,DP,THETA,UI,
$XCRIT,YCRIT,DT,XXNEW,YYNEW,XNOW1,YNOW1
C
C      PERFORMING BOUNDARY INTEG. FOR INTERNAL SOLUTION
C
      DO 4200 I=N1,N1
      II1=1
      II2=1
      U(I)=0.0
      DPDX(I)=0.0
      DPDY(I)=0.0
      PNJP=0.0
      PNXP=0.0
      PNYJP=0.0
      PJP=0.0
      PXJP=0.0
      PYJP=0.0
      DO 4140 J=2,L
      R2=DSQRT((X(J+1)-X(J))**2+(Y(J+1)-Y(J))**2)
      CO=(X(J+1)-X(J))/R2
      SI=(Y(J+1)-Y(J))/R2
      ETA=DABS((YYNOW-Y(J))*CO-(XXNOW-X(J))*SI)
      XIA=(Y(J)-YYNOW)*SI+(X(J)-XXNOW)*CO
      XIB=(Y(J+1)-YYNOW)*SI+(X(J+1)-XXNOW)*CO
      XBXA=XIB-XIA
      SIGNRN=-(X(J)-XXNOW)*(Y(J+1)-Y(J))+(X(J+1)-X(J))*(Y(J)
$ -YYNOW)
      IF(SIGNRN .LT. 0)ETA=-ETA
      IF(DABS(ETA) .LT. 1.E-5) GOTO 4102
      BTN=DATAN(XIB/ETA)
      ATN=DATAN(XIA/ETA)
      GOTO 4105
4102  BTN=0.0
      ATN=0.0
4105  ASQ=XIA**2+ETA**2
      ALN=DLOG(ASQ)
      BSQ=XIB**2+ETA**2
      BLN=DLOG(BSQ)
      ONE=BSQ*(BLN-1.)-ASQ*(ALN-1.)
      TWO=XIB*BLN-XIA*ALN-2.*(XIB-XIA)+2.*ETA*(BTN-ATN)
      PJ=(XIB*(BTN-ATN)-0.5*ETA*(BLN-ALN))/(XIB-XIA)
      PXJ=(-0.5*SI*(BLN-ALN+2.)-CO*(BTN-
$ ATN)+SI*(ETA**2+XIA*XIB)/ASQ)/(XIB-XIA)+ETA*CO/ASQ
      PYJ=(0.5*CO*(BLN-ALN+2.)-SI*(BTN- ATN)-CO*(ETA**2+XIA*
$ XIB)/ASQ)/(XIB-XIA)+ETA*SI/ASQ
      PNJ=TWO*XIB/(2.*XBXA)-ONE/(4.*XBXA)

```

```

PNXJ=-CO+((ETA*CO-XIB*SI)*(BTN-ATN)+0.5*(XIB*CO+ETA
$*SI)*(BLN-ALN))/XBXA
PNYJ=-SI+((ETA*SI+XIB*CO)*(BTN-ATN)+0.5*(XIB*SI-
$ETA*CO)*(BLN-ALN))/XBXA
IF(ICT(J) .NE. 1) GOTO 4321
U(I)=U(I)-PNJP*PN1(II1)
DPDX(I)=DPDX(I)+PNXJP*PN1(II1)
DPDY(I)=DPDY(I)+PNYJP*PN1(II1)
II1=II1+1
PNJP=0.0
PNXJP=0.0
PNYJP=0.0
4321 IF(ICT(J) .NE. 2) GOTO 4322
U(I)=U(I)-PNJ*PN2(II2)
DPDX(I)=DPDX(I)+PNXJ*PN2(II2)
DPDY(I)=DPDY(I)+PNYJ*PN2(II2)
II2=II2+1
PNJ=0.0
PNXJ=0.0
PNYJ=0.0
4322 U(I)=U(I)+(PJ+FJP)*P(J)-(PNJ+PNJP)*PN(J)
DPDX(I)=DPDX(I)+(PXJ+PXJP)*P(J)+(PNXJ+PNXJP)*PN(J)
DPDY(I)=DPDY(I)+(PYJ+PYJP)*P(J)+(PNYJ+PNYJP)*PN(J)
PJP=(ETA*(BLN-ALN)*0.5-XIA*(BTN-ATN))/(XIB-XIA)
PXJP=(0.5*SI*(BLN-ALN-2.)+CO*(BTN-ATN)+SI*(ETA**2+
$XIA*XIB)/BSQ)/(XIB-XIA)-ETA*CO/BSQ
PYJP=(-0.5*CO*(BLN-ALN-2.)+SI*(BTN-ATN)-CO*(ETA**2+
$XIA*XIB)/BSQ)/(XIB-XIA)-ETA*SI/BSQ
PNJP=ONE/(4.*XBXA)-TWO*XIA/(2.*XBXA)
PNXJP=CO+((-ETA*CO+XIA*SI)*(BTN-ATN)-0.5*(XIA*CO+ETA*
$SI)*(BLN-ALN))/XBXA
PNYJP=SI+(-(ETA*SI+XIA*CO)*(BTN-ATN)-0.5*(XIA*SI-ETA*
$CO)*(BLN-ALN))/XBXA
4140 CONTINUE
U(I)=U(I)+PJP*P(2)
DPDX(I)=DPDX(I)+PXJP*P(2)
DPDY(I)=DPDY(I)+PYJP*P(2)
IF(ICT(2) .NE. 1) GOTO 4195
U(I)=U(I)-PNJP*PN1(1)
DPDX(I)=DPDX(I)+PNXJP*PN1(1)
DPDY(I)=DPDY(I)+PNYJP*PN1(1)
PNJP=0.0
PNXJP=0.0
PNYJP=0.0
4195 CONTINUE
U(I)=U(I)-(PNJP*PN(2))
DPDX(I)=DPDX(I)+PNXJP*PN(2)
DPDY(I)=DPDY(I)+PNYJP*PN(2)
U(I)=U(I)/6.283185308
DPDX(I)=DPDX(I)/6.283185308
DPDY(I)=DPDY(I)/6.283185308
4200 CONTINUE
RETURN
END

```

Three Dimensional Model Programs:

PROGRAM 4PTGQ.F

```

C
C   THIS IS A PROGRAM TO CALCULATE THE THREE DIMENSIONAL
C   VELOCITY FIELD INTO LOCAL EXHAUST OPENINGS USING THE
C   BIEM METHOD. WRITTEN BY MIKE FLYNN IN CONJUNCION
C   WITH CASEY MILLER IN MARCH 1988
C
C
C   IMPLICIT REAL*8(A-H,O-Z)
C   DIMENSION PHI(367),DPHI(367),ALPHA(367),
C   1ID(367),PREV(3),AE(691),VX(100),VY(100),VZ(100),R(367),
C   2H(367,367),U(367),X(367),Y(367),Z(367),NE(691,4),ICT(367),
C   3XX(100),YY(100),ZZ(100),ALTDPHI(691,3)
C   COMMON H,U,MP,X,Y,Z,NE,XX,YY,ZZ,I,J,K,L,A,B,AE
C   OPEN (5,FILE='INPUT.DAT')
C   OPEN (3,FILE='INDAT2')
C   OPEN (6,FILE='PRN')
C   READ (5,*) N
C   READ (5,*) M
C   LL=N+1
C   MP=N
C   DO 1001 I=1,N
C   READ (5,*) X(I), Y(I), Z(I), PHI(I), DPHI(I), ID(I), ICT(I)
1001 CONTINUE
C   READ (5,*)((NE(I,II),II=1,4),I=1,M)
C   PRINT *, NE(691,1), NE(691,2), NE(691,3), NE(691,4)
C   CLOSE (3)
C
C
C   THE I LOOP GENERATES VALUES FOR THE COEFFICIENT MATRIX
C   H AND THE KNOWN COLUMN VECTOR R. DIFFERENT VALUES FOR I
C   CORRESPOND TO THE BASE POINT BEING POSITIONED AT A
C   DIFFERENT NODE. VALUES FOR THE UNKNOWN COLUMN VECTOR
C   U ARE ALLOCATED DEPENDING ON THE ID CODE AT THE START
C   OF THE LOOP
C
C
C   DO 10 I=1,N
C   R(I)=0.0
C   DO 20 J=1,N
C   H(I,J)=0.0
20 CONTINUE
10 CONTINUE
C   DO 100 I=1,N
C   IF(ID(I).EQ.1) THEN
C   U(I)=DPHI(I)
C   ELSE
C   U(I)=PHI(I)
C   ENDIF
C   ALPHA(I)=0.0

```

```

C
C   THE J LOOP CONTAINS THE SUMMATIONS OVER EACH ELEMENT
C   INITIALLY THE AREA OF THE JTH ELEMENT IS CALCULATED
C   THIS IS AE(J).
C
DO 200 J=1,M
  DY21=Y(NE(J,2))-Y(NE(J,1))
  DY31=Y(NE(J,3))-Y(NE(J,1))
  DZ21=Z(NE(J,2))-Z(NE(J,1))
  DZ31=Z(NE(J,3))-Z(NE(J,1))
  DX21=X(NE(J,2))-X(NE(J,1))
  DX31=X(NE(J,3))-X(NE(J,1))
  AE1=DY21*DZ31-DZ21*DY31
  AE2=DZ21*DX31-DX21*DZ31
  AE3=DX21*DY31-DY21*DX31
  AE(J)=0.5*(DSQRT(AE1**2+AE2**2+AE3**2))
  ASUM=0.0
C
C
C   THE K LOOP CALLS THE PROGRAMS THAT PERFORM THE BOUNDARY
C   INTEGRATIONS FOR A SINGLE TRIANGULAR ELEMENT, SPINT
C   IS USED IF THE BASE POINT COINCIDES WITH THE ELEMENT
C   NODE THAT IS BEING CONSIDERED, GQUAD IS THE STANDARD
C   GAUSSIAN QUADRATURE. BOTH PROGRAMS RETURN A AND B.
C
C
DO 300 K=1,3
  IF(I.EQ.NE(J,K)) THEN
    CALL SPINT
  ELSE
    CALL GQUAD
  ENDIF
  IF(ID(NE(J,K)).EQ.1.AND.ICT(NE(J,K)).EQ.0) THEN
    H(I,NE(J,K))=H(I,NE(J,K))-B
    R(I)=R(I)-A*PHI(NE(J,K))
  ELSEIF(ICT(NE(J,K)).EQ.1.AND.ID(NE(J,K)).EQ.1.AND.
$ ID(NE(J,4)).EQ.2) THEN
    H(I,NE(J,K))=H(I,NE(J,K))
    R(I)=R(I)+B*DPHI(NE(J,4))-A*PHI(NE(J,K))
    ALTDPHI(J,K)=DPHI(NE(J,4))
  ELSEIF(ICT(NE(J,K)).EQ.1.AND.ID(NE(J,K)).EQ.1.AND.
$ ID(NE(J,4)).EQ.1) THEN
    H(I,NE(J,K))=H(I,NE(J,K))-B
    R(I)=R(I)-A*PHI(NE(J,K))
  ELSE
    H(I,NE(J,K))=H(I,NE(J,K))+A
    R(I)=R(I)+B*DPHI(NE(J,K))
  ENDIF
  ASUM=ASUM+A
C
C
300 CONTINUE
  ALPHA(I)=-ASUM+ALPHA(I)
200 CONTINUE

```

```

      IF(ID(I).EQ.1) THEN
        R(I)=R(I)-ALPHA(I)*PHI(I)
      ELSE
        H(I,I)=H(I,I)+ALPHA(I)
      ENDIF
100  CONTINUE
      DO 111 I=1,N
        H(I,LL)=R(I)
111  CONTINUE
      CALL EQSOL

C
C   AFTER RETURNING FROM EQSOL U(I) HAS BEEN WRITTEN
C   OVER WITH THE SOLUTIONS, THESE ARE THEN ALLOCATED TO
C   PHI(I) OR DPHI(I) AS FOLLOWS:
C
C   OPEN (4,FILE='BOUND.OUT')
      DO 1000 I=1,N
        IF(ID(I).EQ.1) THEN
          DPHI(I)=U(I)
        ELSE
          PHI(I)=U(I)
        ENDIF
        WRITE (6,678) X(I),Y(I),Z(I),PHI(I),DPHI(I),ID(I),ICT(I)
678   FORMAT (F15.5,' ',F15.5,' ',F15.5,' ',F15.5,' ',F15.5,' ',
      $I5,' ',I5)
1000 CONTINUE
      CLOSE (4)

C
C   READ IN THE INTERNAL POINTS FOR VELOCITY CALCULATIONS
C
      READ (5,*) NN
      DO 222 I=1,NN
        READ (5,*) XX(I), YY(I), ZZ(I)
222  CONTINUE

C
C   DECLARE AND INITIALIZE
C
      DO 1100 I=1,NN
        PVX=0.0
        PVY=0.0
        PVZ=0.0
      DO 2000 J=1,M
        PREVX=0.0
        PREVY=0.0
        PREVZ=0.0
      DO 3000 K=1,3
      DO 4000 L=1,3
        CALL INTERGQ
        IF(ICT(NE(J,K)).EQ.1.AND.ID(NE(J,K)).EQ.1
      $ .AND.ID(NE(J,4)).EQ.2) THEN
          DPHIDE=ALTDPHI(J,K)
        ELSE
          DPHIDE=DPHI(NE(J,K))
        ENDIF

```

```

PREV(L)=(A*PHI(NE(J,K))-B*DPHIDE)
4000 CONTINUE
PREVX=PREVX+PREV(1)
PREVY=PREVY+PREV(2)
PREVZ=PREVZ+PREV(3)
3000 CONTINUE
PVX=PVX+PREVX
PVY=PVY+PREVY
PVZ=PVZ+PREVZ
2000 CONTINUE
COEF1=-0.079577471
VX(I)=COEF1*PVX
VY(I)=COEF1*PVY
VZ(I)=COEF1*PVZ
1100 CONTINUE
C
C
C   OUTPUT INTERNAL DATA TO FILE
C   print *, 'internal velocities:'
C
C   OPEN (4,FILE='INT.OUT')
C   DO 444 I=1,NN
C     WRITE (6,*) VX(I),VY(I),VZ(I)
444 CONTINUE
C   CLOSE (4)
C   CONTINUE
C   STOP
C   END
C
C
SUBROUTINE GQUAD
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION W(13),CHI(13),ETA(13),F(13),
1H(367,367),U(367),X(367),Y(367),Z(367),
2NE(691,4),XX(100),YY(100),ZZ(100),AE(691)
COMMON H,U,MP,X,Y,Z,NE,XX,YY,ZZ,I,J,K,L,A,B,AE
A=0.D00
B=0.D00
W(1)=-0.5626000000000000
W(2)=.5208333333333333
W(3)=W(2)
W(4)=W(2)
CHI(1)=.3333333333333333
CHI(2)=.6000000000000000
CHI(3)=.2000000000000000
CHI(4)=CHI(3)
ETA(1)=CHI(1)
ETA(2)=CHI(3)
ETA(3)=CHI(2)
ETA(4)=CHI(3)
DO 4400 II=1,4
IF(K.EQ.1) THEN
F(II)=CHI(II)
ELSEIF (K.EQ.2) THEN

```

```

      F(II)=ETA(II)
    ELSE
      F(II)=1.D00-ETA(II)-CHI(II)
    ENDIF
    A1=(X(NE(J,1))*CHI(II)+X(NE(J,2))*ETA(II)+X(NE(J,3))
    $*(1.D00-CHI(II)-ETA(II))-X(I))
    B1=(Y(NE(J,1))*CHI(II)+Y(NE(J,2))*ETA(II)+Y(NE(J,3))
    $*(1.D00-CHI(II)-ETA(II))-Y(I))
    C1=(Z(NE(J,1))*CHI(II)+Z(NE(J,2))*ETA(II)+Z(NE(J,3))
    $*(1.D00-CHI(II)-ETA(II))-Z(I))
    R=DSQRT(A1**2+B1**2+C1**2)
    FF=F(II)*(1.D00/R)*W(II)
    B=B+FF
    IF (I.EQ.NE(J,1).OR.I.EQ.NE(J,2).OR.I.EQ.NE(J,3)) THEN
      GOTO 4400
    ELSE
      CONTINUE
    ENDIF
    AY1=Y(NE(J,1))-B1-Y(I)
    AY2=Y(NE(J,2))-B1-Y(I)
    AZ1=Z(NE(J,1))-C1-Z(I)
    AZ2=Z(NE(J,2))-C1-Z(I)
    AX1=X(NE(J,1))-A1-X(I)
    AX2=X(NE(J,2))-A1-X(I)
    A2=AY1*AZ2-AZ1*AY2
    B2=AZ1*AX2-AX1*AZ2
    C2=AX1*AY2-AY1*AX2
    G1=A1*A2+B1*B2+C1*C2
    G2=DSQRT(A2**2+B2**2+C2**2)
    G=F(II)*W(II)*(-G1/(R**3*G2))
    A=A+G
4400 CONTINUE
    A=AE(J)*A
    B=AE(J)*B
    RETURN
  END

```

C  
C  
C  
C  
C  
C  
C  
C  
C  
C  
C  
C  
C  
C

NOTE THE SIGN FOR G IS CORRECT HERE ONLY IF  
THE ELEMENTS HAVE BEEN NUMBERED COUNTERCLOCKWISE  
WHEN LOOKING OUT OF THE DOMAIN THIS CONVENTION  
MUST BE OBSERVED WHEN CREATING E(J,K)

THE SUBROUTINE SPINT IS DESIGNED TO HANDLE THE SPECIAL  
INTEGRATIONS THAT OCCUR WHEN AN ELEMENT NODE COINCIDES  
WITH THE BASE POINT

```

SUBROUTINE SPINT
  IMPLICIT REAL*8(A-H,O-Z)
  DIMENSION W(13),ETA(13),CHI(13),F(13),
  1H(367,367),U(367),X(367),Y(367),Z(367),
  2NE(691,4),XX(100),YY(100),ZZ(100),AE(691)

```

```

COMMON H,U,MP,X,Y,Z,NE,XX,YY,ZZ,I,J,K,L,A,B,AE
A=0.D00
B=0.D00
BP=0.D00
W(1)=-0.5626000000000000
W(2)=.5208333333333333
W(3)=W(2)
W(4)=W(2)
CHI(1)=.3333333333333333
CHI(2)=.6000000000000000
CHI(3)=.2000000000000000
CHI(4)=CHI(3)
ETA(1)=CHI(1)
ETA(2)=CHI(3)
ETA(3)=CHI(2)
ETA(4)=CHI(3)
DO 500 II=1,4
  IF (K.EQ.1) THEN
    F(II)=CHI(II)
  ELSEIF (K.EQ.2) THEN
    F(II)=ETA(II)
  ELSE
    F(II)=1.D00-ETA(II)-CHI(II)
  ENDIF
  A1=(X(NE(J,1))*CHI(II)+X(NE(J,2))*ETA(II)+X(NE(J,3))
  $*(1.D00-CHI(II)-ETA(II))-X(I))
  B1=(Y(NE(J,1))*CHI(II)+Y(NE(J,2))*ETA(II)+Y(NE(J,3))
  $*(1.D00-CHI(II)-ETA(II))-Y(I))
  C1=(Z(NE(J,1))*CHI(II)+Z(NE(J,2))*ETA(II)+Z(NE(J,3))
  $*(1-CHI(II)-ETA(II))-Z(I))
  R=DSQRT(A1**2+B1**2+C1**2)
  BPRM31=W(II)*(F(II)-1.D00)*(1.D00/R)
  BP=BP+BPRM31
500 CONTINUE
  P1P2=DSQRT((X(NE(J,1))-X(NE(J,2)))**2+
  $(Y(NE(J,1))-Y(NE(J,2)))**2+(Z(NE(J,1))-Z(NE(J,2)))**2)
  P1P3=DSQRT((X(NE(J,1))-X(NE(J,3)))**2+
  $(Y(NE(J,1))-Y(NE(J,3)))**2+(Z(NE(J,1))-Z(NE(J,3)))**2)
  P2P3=DSQRT((X(NE(J,2))-X(NE(J,3)))**2+
  $(Y(NE(J,2))-Y(NE(J,3)))**2+(Z(NE(J,2))-Z(NE(J,3)))**2)
  IF (K.EQ.1) THEN
    BS=DABS(P1P3)
  ELSEIF (K.EQ.2) THEN
    BS=DABS(P1P2)
  ELSE
    BS=DABS(P2P3)
  ENDIF
  XXX1=0.D00
  YYY2=0.D00
  XXX2=BS
  IF (K.EQ.1) THEN
    BSS=DABS(P1P2)
  ELSEIF (K.EQ.2) THEN
    BSS=DABS(P2P3)

```

```

ELSE
  BSS=DABS(P1P3)
ENDIF
IF(K.EQ.1) THEN
  BSSS=DABS(P2P3)
ELSEIF(K.EQ.2) THEN
  BSSS=DABS(P1P3)
ELSE
  BSSS=DABS(P1P2)
ENDIF
THETA1=DACOS((BS**2+BSS**2-BSSS**2)/(2.D00*BS*BSS))
YYY3=BSS*DSIN(THETA1)
XXX3=BSS*DCOS(THETA1)
QSQS=DABS(XXX3-XXX2)
IF(QSQS.LT.0.0001) THEN
  PREBS=BSS
  BSS=BS
  BS=PREBS
  XXX2=BS
  YYY3=BSS*DSIN(THETA1)
  XXX3=BSS*DCOS(THETA1)
ELSE
  CONTINUE
ENDIF
EM=(YYY3-YYY2)/(XXX3-XXX2)
BB=-EM*BS
2121 ALP1=(1.D00/(DSQRT(1.D00+EM**2)))
ALP2=(-EM/(DSQRT(1.D00+EM**2)))
TNALF2=ALP2/(1.D00+ALP1)
TNHT1=DSIN(THETA1)/(1.D00+DCOS(THETA1))
B3NUMR=(TNHT1+TNALF2)/(1.D00-TNHT1*TNALF2)
BSTR3=((BB/(DSQRT(1.D00+EM**2)))*DLOG(DABS(B3NUMR/TNALF2)))
B=(AE(J)*BP)+(BSTR3)
RETURN
END

```

C  
C  
C

```

SUBROUTINE INTERGQ
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION W(13),ETA(13),CHI(13),F(13),
1H(367,367),U(367),X(367),Y(367),Z(367),
2NE(691,4),XX(100),YY(100),ZZ(100),AE(691)
COMMON H,U,MP,X,Y,Z,NE,XX,YY,ZZ,I,J,K,L,A,B,AE
A=0.0
B=0.0
W(1)=-0.5626000000000000
W(2)=.5208333333333333
W(3)=W(2)
W(4)=W(2)
CHI(1)=.3333333333333333
CHI(2)=.6000000000000000
CHI(3)=.2000000000000000
CHI(4)=CHI(3)

```

```

ETA(1)=CHI(1)
ETA(2)=CHI(3)
ETA(3)=CHI(2)
ETA(4)=CHI(3)
DO 400 II=1,4
  IF(K.EQ.1) THEN
    F(II)=CHI(II)
  ELSEIF (K.EQ.2) THEN
    F(II)=ETA(II)
  ELSE
    F(II)=1.-ETA(II)-CHI(II)
  ENDIF
  A1=(X(NE(J,1))*CHI(II)+X(NE(J,2))*ETA(II)+X(NE(J,3))
  $*(1-CHI(II)-ETA(II))-XX(I))
  B1=(Y(NE(J,1))*CHI(II)+Y(NE(J,2))*ETA(II)+Y(NE(J,3))
  $*(1-CHI(II)-ETA(II))-YY(I))
  C1=(Z(NE(J,1))*CHI(II)+Z(NE(J,2))*ETA(II)+Z(NE(J,3))
  $*(1-CHI(II)-ETA(II))-ZZ(I))
  R=DSQRT(A1**2+B1**2+C1**2)
  IF(L.EQ.1) THEN
    RNUM=A1
  ELSEIF(L.EQ.2) THEN
    RNUM=B1
  ELSE
    RNUM=C1
  ENDIF
  FF=F(II)*(RNUM/R**3)*W(II)
  B=B+FF
  AY1=Y(NE(J,1))-B1-YY(I)
  AY2=Y(NE(J,2))-B1-YY(I)
  AZ1=Z(NE(J,1))-C1-ZZ(I)
  AZ2=Z(NE(J,2))-C1-ZZ(I)
  AX1=X(NE(J,1))-A1-XX(I)
  AX2=X(NE(J,2))-A1-XX(I)
  A2=AY1*AZ2-AZ1*AY2
  B2=AZ1*AX2-AX1*AZ2
  C2=AX1*AY2-AY1*AX2
  GG1=DSQRT(A2**2+B2**2+C2**2)
  IF(L.EQ.1) THEN
    CF1=A2
  ELSEIF(L.EQ.2) THEN
    CF1=B2
  ELSE
    CF1=C2
  ENDIF
  T1=1./(R**3*GG1)
  T3=T1**2
  T6=A1*A2+B1*B2+C1*C2
  G=F(II)*W(II)*((-3*RNUM*R*GG1*T6*T3)+(CF1*T1))
  A=A+G
400 CONTINUE
A=AE(J)*A
B=AE(J)*B
RETURN

```

END

C  
C

```

SUBROUTINE EQSOL
  IMPLICIT REAL*8 (A-H,O-Z)
  DIMENSION M(367),H(367,367),U(367)
  COMMON H,U,MP,X,Y,Z,NE,XX,YY,ZZ,I,J,K,L,A,B
  DO 5 II=1,MP
    M(II)=1
    AMAX=H(II,1)
    DO 2 JJ=2,MP
      IF(DABS(H(II,JJ)).LE.DABS(AMAX)) GOTO 2
100  AMAX=H(II,JJ)
      M(II)=JJ
    2 CONTINUE
    IF(AMAX.EQ.0) GOTO 98
    3 NN=MP+1
    DO 4 JJ=1,NN
      4 H(II,JJ)=H(II,JJ)/AMAX
      DO 5 IP=1,MP
        IF (IP.EQ.II) GOTO 5
        MMM=M(II)
        ZMULT=H(IP,MMM)
        DO 6 JJ=1,NN
          IF(JJ.NE.MMM) GOTO 9
        8 H(IP,JJ)=0.0
        GOTO 6
      9 H(IP,JJ)=H(IP,JJ)-ZMULT*H(II,JJ)
    6 CONTINUE
    5 CONTINUE
    DO 7 II=1,MP
      NO=M(II)
      7 U(NO)=H(II,NN)
    RETURN
98 PRINT 'no solution'
  STOP
  END

```

## Program TRK15.F

C THIS PROGRAM CALCULATES THE EFFICIENCY OF ASPIRATION  
C OF PARTICLES INTO AN ARBITRARY SHAPED SAMPLING INLET  
C USING BIEM TO CALCULATE THE AIR FLOW PATTERNS AND THE  
C EQUATIONS BY ALENIOUS TO PREDICT PARTICLE LOCATION AND  
C AND VELOCITY AFTER A SMALL TIME STEP HAS ELAPSED.

C  
C THE PROGRAM REQUIRES THE OUTPUT OF THE BIEM BOUNDARY  
C SOLUTION (4PTGQ.F) AS INPUT IN ORDER TO USE THE SHELL  
C SUBROUTINE TO CALCULATE INTERNAL VELOCITIES AT THE  
C PARTICLE LOCATION.

C  
C THE SIZES OF THE ARRAYS FOR THE ARRAY VARIABLES IS  
C DEPENDENT ON THE NUMBER OF NODES AND ELEMENTS USED  
C TO FORMULATE THE BOUNDARY OF THE DOMAIN.

C  
C  
C IMPLICIT REAL \*8(A-H,O-Z)  
C DIMENSION X(367),Y(367),Z(367),PHI(367)  
C \$,DPHI(367),ID(367),ICT(367),NE(693,4),AE(693)  
C COMMON UO,UI,DIN,DP,THETA1,DTA,VX,VY,VZ,XX1,YY1,ZZ1,  
C \$NCAP,XX,YY,ZZ,MP,X,Y,Z,NE,I,J,K,L,A,B,AE,XDIN,  
C \$PHI,DPHI,ID,ICT,N,M

C  
C \*\*\*\*\*

## C INPUT VARIABLES:

C UO = WIND VELOCITY, CM/SEC  
C DP = PARTICLE DIAMETER, CM  
C DIN = INLET DIAMETER, CM  
C DTA = INITIAL TIME STEP, SEC  
C THETA1 = ANGLE OF MISALIGNMENT, DEGREES  
C UI = INLET SAMPLING VELOCITY, CM/SEC  
C XDIN = NUMBER OF INLET DIAMETERS AWAY FROM FACE  
C FOR TRAJECTORY START PTS  
C N = NUMBER OF NODES  
C M = NUMBER OF ELEMENTS  
C X,Y,Z = COORDINATES OF NODES  
C PHI, DPHI = VALUE OF POTENTIAL OR NORMAL DERIVATIVE  
C 0 IF UNKNOWN  
C ID, ICT = BOUNDARY CODES, SEE 4PTGQ.F  
C NE = ARRAY NAME FOR ALL THE TRIANGULAR ELEMENTS. EACH  
C ELEMENT IS IDENTIFIED BY THE NODE NUMBER OF  
C EACH CORNER LISTED IN A CLOCKWISE DIRECTION  
C LOOKING OUT OF THE DOMAIN. A FOURTH INTERGER  
C IN THIS ARRAY SPECIFIES WHETHER POTENTIAL OF  
C VELOCITY IS KNOWN OVER THE ELEMENT.

C X,Y,Z,PHI,DPHI,ID, AND ICT CORRESPOND TO THE OUTPUT OF  
C THE BIEM BOUNDARY SOLUTION FROM 4PTGQ.

C \*\*\*\*\*

C  
C READ REQUIRED INPUT DATA

```

C      READ (5,*) UO,DP,DIN,DTA,THETA1,UI,XDIN
      READ (5,*) CVC,CINC
      READ (5,*) N
      READ (5,*) M
      DO 1001 I=1,N
      READ (5,*) X(I), Y(I), Z(I), PHI(I), DPHI(I), ID(I),
$      ICT(I)
1001  CONTINUE
      READ (5,*)((NE(I,II),II=1,4),I=1,M)
      THETA=THETA1*.017453292

C
C
C      THE J LOOP CONTAINS THE SUMMATIONS OVER EACH ELEMENT
C      INITIALLY THE AREA OF THE JTH ELEMENT IS CALCULATED
C      THIS IS AE(J).
C
      DO 200 J=1,M
      DY21=Y(NE(J,2))-Y(NE(J,1))
      DY31=Y(NE(J,3))-Y(NE(J,1))
      DZ21=Z(NE(J,2))-Z(NE(J,1))
      DZ31=Z(NE(J,3))-Z(NE(J,1))
      DX21=X(NE(J,2))-X(NE(J,1))
      DX31=X(NE(J,3))-X(NE(J,1))
      AE1=DY21*DZ31-DZ21*DY31
      AE2=DZ21*DX31-DX21*DZ31
      AE3=DX21*DY31-DY21*DX31
      AE(J)=0.5*(DSQRT(AE1**2+AE2**2+AE3**2))
200  CONTINUE

C      CALCULATE FIRST STARTING POINT
C
      NCAP=0
      NTRAJ=0
      XX=XDIN*DIN*COS(THETA1*(-1))
      YY=0.0
      ZZ=XDIN*DIN*SIN(THETA1*(-1))
250  CONTINUE
      NTRAJ=NTRAJ+1
      IF (NTRAJ.EQ. 15) THEN
      PRINT *, "first Capture not found", NTRAJ
      GOTO 3000
      ENDIF
      CALL TRAJ

C
C      CHECK FOR FIRST CAPTURE
C
      IF (NCAP.EQ. 1) THEN
      GOTO 1000
      ELSE IF ((NCAP.EQ. 2).AND. (ZZ.GT. 0.0)) THEN
      XX=XX1-(CINC*SIN(THETA1))
      YY=0.0
      ZZ=ZZ1-(CINC*COS(THETA1))
      GOTO 250

```

```

ELSE
    XX=XX1+(CINC*SIN(THETA1))
    YY=0.0
    ZZ=ZZ1+(CINC*COS(THETA1))
    GOTO 250
ENDIF
1000 PRINT *, "NO. OF TRAJ FCAP =",NTRAJ
C
C BEGIN CRITICAL TRAJECTORY CALCULATIONS
C
XFCAP=XX1
YFCAP=YY1
ZFCAP=ZZ1
PRINT *, "1st cap:",",",",xfcap,",",",yfcap,",",",zfcap
XCAP=XX1
YCAP=YY1
ZCAP=ZZ1
C
C STEP = NUMBER OF DEGREES BETWEEN RAYS ON SEMICIRCLE
C OF STARTING POINTS, NSTEP = NUMBER OF STEPS
C
STEP=15.
NSTEP=13
nsum=35
NCOUNT=0
C
C INITIAL CAPTURE TRAJECTORY HAS BEEN FOUND, THIS SECTION OF
THE PROGRAM CALCULATES TRAJECTORY STARTING POINTS AT SUCCESSIVE
C LOCATIONS AROUND A SEMICIRCLE ABOUT THE COORDINATES OF THE
C FIRST CAPTURE
C
DO 3000 JJ=1,NSTEP
    NCOUNT=NCOUNT+1
    theta2=(-90+((ncount-1)*step))*0.017453292
    AA=.67*XDIN*DIN
    XX=XFCAP+(AA*SIN(THETA2)*SIN(THETA1))
    YY=(AA*COS(THETA2))*(-1)
    ZZ=ZFCAP+(AA*SIN(THETA2)*COS(THETA1))
    XCAP=XFCAP
    YCAP=YFCAP
    ZCAP=ZFCAP
    NFMIS=0
C
C ONCE A START POINT IS CALCULATED, THE KK LOOP PLOTS
C SUCCESSIVE TRAJECTORIES UNTIL THE CONVERGENCE CRITERIA
C IS SATISFIED.
C
DO 2900 KK=1,NSUM
1500 CALL TRAJ
    IF ((NFMIS .EQ. 0) .AND. (NCAP .EQ. 1)) THEN
        AA=CINC+AA
        XX=XFCAP+(AA*SIN(THETA2)*SIN(THETA1))
        YY=(AA*COS(THETA2))*(-1)

```

```

ZZ=ZFCAP+(AA*SIN(THETA2)*COS(THETA1))
GOTO 1500
ELSE
  NFMISS=1
ENDIF
IF (NCAP .EQ. 2) THEN
  XMISS=XX1
  YMISS=YY1
  ZMISS=ZZ1
ELSE
  XCAP=XX1
  YCAP=YY1
  ZCAP=ZZ1
ENDIF

C
C   S IS THE DISTANCE BETWEEN THE LAST CAPTURE AND LAST
C   MISS. S IS COMPARED TO CVC AND LOOP EXITED IF S < CVC.
C
$   S=SQRT((XCAP-XMISS)**2+(YCAP-YMISS)**2+
      (ZCAP-ZMISS)**2)
   IF (S .LT. CVC) THEN
     GOTO 2990
   ELSE
     XX=(XMISS+XCAP)/2
     YY=(YMISS+YCAP)/2
     ZZ=(ZMISS+ZCAP)/2
   ENDIF
   if (kk .eq. nsum) print *, "no convergence, s =",s
2900  CONTINUE
2990  XCRIT=(XMISS+XCAP)/2
      YCRIT=(YMISS+YCAP)/2
      ZCRIT=(ZMISS+ZCAP)/2

C
C   EACH CRITICAL TRAJECTORY COORDINATE IS WRITTEN TO THE
C   OUTPUT UNIT.
C
write (6,*)"crit",",",",ncount",",",",xcrit",",",",ycrit",",",",zcrit
3000  CONTINUE
      STOP
      END

C
C
C
SUBROUTINE TRAJ

C
C   THIS SUBROUTINE TAKES A SINGLE PARTICLE STARTING
C   POINT, ASSIGNS THE INITIAL VELOCITY VALUES TO THE
C   PARTICLE, AND TRACKS THE PARTICLE TO THE FACE OF THE
C   INLET (CAPTURE) OR OUT OF THE BOUNDARY (MISS)
C
C   IMPLICIT REAL *8(A-H,O-Z)
C   DIMENSION VXOL(1600),VYOL(1600),VZOL(1600),
C1X(367),Y(367),Z(367),PHI(367),DPHI(367),ID(367)
C2,ICT(367),NE(693,4),AE(693)

```

```

COMMON UO,UI,DIN,DP,THETA1,DTA,VX,VY,VZ,XX1,YY1,ZZ1,
$NCAP,XX,YY,ZZ,MP,X,Y,Z,NE,I,J,K,L,A,B,AE,XDIN,
$PHI,DPHI,ID,ICT,N,M

```

```

C
C   print *, M
C   G=-980.
C   DT=DTA
C   AV=.000183
C   PDENS=1.0
C   ADENS=.001205
C   DPL=10000.0*DP
C   CC=1+((2.*0.07/DPL)*(1.257+(0.4*EXP(-0.55*DPL/0.07))))
C   TAU=(DP**2*PDENS*CC)/(18*AV)
C   GX=G*SIN(THETA1)
C   GZ=G*COS(THETA1)
C   GY=0
C
C
C   MANY=9999
C   NCAP=0
C   N1=1
C   DO 4300 I=1,MANY
C       CALL SHELL
C       IF (I .EQ. 1) THEN
C           VXNOW=VX+(TAU*GX)
C           VYNOW=VY+(TAU*GY)
C           VZNOW=VZ+(TAU*GZ)
C           XX1=XX
C           YY1=YY
C           ZZ1=ZZ
C       ENDIF
C       VR=((VXNOW-VX)**2+(VZNOW-VZ)**2
$       +(VYNOW-VY)**2)**0.5
C       RE=VR*DP*ADENS/AV
C       IF (RE .LE. 0.5) THEN
C           QRE=1
C       ELSE IF ((RE .GT. 0.5) .AND. (RE .LE. 800.)) THEN
C           QRE=1+(0.15*RE**0.687)
C       ELSE
C           QRE=0.44*RE/24
C       ENDIF
C       IF (I .EQ. 1) THEN
C           VXOL(I)=VX
C           VYOL(I)=VY
C           VZOL(I)=VZ
C       ELSE
C           VXOL(I)=VXOL(I-1)
C           VZOL(I)=VZOL(I-1)
C           VYOL(I)=VYOL(I-1)
C       ENDIF
$       if ((XX .lt. 1.5*din) .and. (XX .gt. (-1.5)*din)
$       .and. (YY .lt. 1.5*din) .and. (YY .gt. (-1.5)*din)
$       .AND. (ZZ .LT. 1.5*DIN) .AND. (ZZ .GT. (-1.5)*DIN))
$       then

```

```

      DT=DTA/2.
    else
      DT=DTA
    endif
    UAX=(VX-VXOL(I))/DT
    UAZ=(VZ-VZOL(I))/DT
    UAY=(VY-VYOL(I))/DT
    UBX=VX-((UAX-GX)*(TAU/QRE))
    UBY=VY-((UAY-GY)*(TAU/QRE))
    UBZ=VZ-((UAZ-GZ)*(TAU/QRE))
    VXNEW=(UAX*DT)+UBX-((UBX-VXNOW)*
$      EXP(-QRE*DT/TAU))
    VYNEW=(UAY*DT)+UBY-((UBY-VYNOW)*
$      EXP(-QRE*DT/TAU))
    VZNEW=(UAZ*DT)+UBZ-((UBZ-VZNOW)*
$      EXP(-QRE*DT/TAU))
    XXNEW=XX+(UAX/2*DT**2)+(UBX*DT)-
$      ((UBX-VXNOW)*(TAU/QRE)*(1-EXP(-QRE*DT/TAU)))
    ZZNEW=ZZ+(UAZ/2*DT**2)+(UBZ*DT)-
$      ((UBZ-VZNOW)*(TAU/QRE)*(1-EXP(-QRE*DT/TAU)))
    YYNEW=YY+(UAY/2*DT**2)+(UBY*DT)-
$      ((UBY-VYNOW)*(TAU/QRE)*(1-EXP(-QRE*DT/TAU)))
    IF (XXNEW .LT. (-1.1)*XDIN/2*DIN) THEN
      NCAP=2
    c      PRINT *, 'PARTICLE EXIT REAR BOUNDARY, MISS'
      GOTO 4301
    ELSE IF ((ZZNEW .GT. 1.1*XDIN*DIN) .OR. (ZZNEW .LT.
$      (-1.1)*XDIN*DIN)) THEN
      NCAP=2
    c      PRINT *, 'PARTICLE EXIT TOP/BOTTOM, MISS'
      GOTO 4301
    ELSE IF ((YYNEW .GT. 1.1*XDIN*DIN) .OR. (YYNEW .LT.
$      (-1.1)*XDIN*DIN)) THEN
      NCAP=2
    c      PRINT *, 'PARTICLE EXIT SIDE, MISS'
      GOTO 4301
    ELSE IF ((XXNEW .LT. 0.05*din) .AND. (XX .GT.
$      0.05*din)
$      .AND. (ABS(ZZNEW) .LT. DIN/2+.05) .AND.
$      (ABS(YYNEW) .LT. DIN/2+.05)) THEN
      NCAP=1
    c      PRINT *, 'CAPTURE AT FACE'
      GOTO 4301
    ELSE IF ((ABS(ZZNEW) .LT. DIN/2+.05) .AND.
$      (ABS(YYNEW) .LT. DIN/2+.05) .AND.
$      (XXNEW .LT. 0.0))
$      THEN
      NCAP=2
    c      PRINT *, 'CONTACT TUBE, MISS'
      GOTO 4301
    ELSE
      NCAP=0
    ENDIF
    N1=N1+1

```

```

        XX=XXNEW
        YY=YYNEW
        ZZ=ZZNEW
        VXNOW=VXNEW
        VYNOW=VYNEW
        VZNOW=VZNEW
4300  CONTINUE
4301  RETURN
      END

C
C
C
      SUBROUTINE SHELL
C
C THIS SUBROUTINE CALCULATES THE INTERNAL AIR VELOCITY
C COMPONENTS IN 3 DIMENSIONS AT THE LOCATION THAT THE PARTICLE
C IS LOCATED USING THE BIEM INTERNAL SOLUTION ROUTINE. IT
C FURTHERCALLS SUBROUTINE INTERGQ TO PERFORM THE INTERNAL
C GAUSSIAN QUADRATURE.
C
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION PREV(3),AE(693),NE(693,4),
1X(367),Y(367),Z(367),PHI(367),DPHI(367),ID(367)
2,ICT(367),ALTDPHI(693,3)
      COMMON UO,UI,DIN,DP,THETA1,DTA,VX,VY,VZ,XX1,YY1,ZZ1,
$NCAP,XX,YY,ZZ,MP,X,Y,Z,NE,I,J,K,L,A,B,AE,XDIN,
$PHI,DPHI,ID,ICT,N,M
      MP=N

C
C
C
      DECLARE AND INITIALIZE

      PVX=0.0
      PVY=0.0
      PVZ=0.0
      DO 2000 J=1,M
        PREVX=0.0
        PREVY=0.0
        PREVZ=0.0
        DO 3000 K=1,3
          DO 4000 L=1,3
            CALL INTERGQ
            IF (ICT(NE(J,K)).EQ.1.AND.ID(NE(J,K)).EQ.1
$.AND.ID(NE(J,4)).EQ.2) THEN
              DPHIDE=DPHI(NE(J,4))
            ELSE
              DPHIDE=DPHI(NE(J,K))
            ENDIF
            PREV(L)=(A*PHI(NE(J,K))-B*DPHIDE)
4000  CONTINUE
            PREVX=PREVX+PREV(1)
            PREVY=PREVY+PREV(2)
            PREVZ=PREVZ+PREV(3)
          
```

```

3000 CONTINUE
    PVX=PVX+PREVX
    PVY=PVY+PREVY
    PVZ=PVZ+PREVZ
2000 CONTINUE
    COEF1=-0.079577471
    VX=COEF1*PVX
    VY=COEF1*PVY
    VZ=COEF1*PVZ
1100 CONTINUE
C
C
444 CONTINUE
C   CLOSE (4)
    RETURN
    END
C
C
C
SUBROUTINE INTERGQ
    IMPLICIT REAL*8(A-H,O-Z)
    DIMENSION W(13),ETA(13),CHI(13),F(13),NE(693,4),
    1X(367),Y(367),Z(367),PHI(367),DPHI(367),ID(367),ICT(367)
    2,AE(693)
    COMMON UO,UI,DIN,DP,THETA1,DTA,VX,VY,VZ,XX1,YY1,ZZ1,
    $NCAP,XX,YY,ZZ,MP,X,Y,Z,NE,I,J,K,L,A,B,AE,XDIN,
    $PHI,DPHI,ID,ICT,N,M
    A=0.0
    B=0.0
    W(1)=.3333333333333333
    W(2)=W(1)
    W(3)=W(2)
    CHI(1)=.6666666666666667
    CHI(2)=.1666666666666667
    CHI(3)=CHI(2)
    ETA(1)=.1666666666666667
    ETA(2)=.1666666666666667
    ETA(3)=.6666666666666667
    DO 400 II=1,3
    IF(K.EQ.1) THEN
        F(II)=CHI(II)
    ELSEIF (K.EQ.2) THEN
        F(II)=ETA(II)
    ELSE
        F(II)=1.-ETA(II)-CHI(II)
    ENDIF
    A1=(X(NE(J,1))*CHI(II)+X(NE(J,2))*ETA(II)+X(NE(J,3))
    $*(1-CHI(II)-ETA(II))-XX)
    B1=(Y(NE(J,1))*CHI(II)+Y(NE(J,2))*ETA(II)+Y(NE(J,3))
    $*(1-CHI(II)-ETA(II))-YY)
    C1=(Z(NE(J,1))*CHI(II)+Z(NE(J,2))*ETA(II)+Z(NE(J,3))
    $*(1-CHI(II)-ETA(II))-ZZ)
    R=DSQRT(A1**2+B1**2+C1**2)
    IF(L.EQ.1) THEN

```

```

      RNUM=A1
    ELSEIF(L.EQ.2) THEN
      RNUM=B1
    ELSE
      RNUM=C1
    ENDIF
    FF=F(II)*(RNUM/R**3)*W(II)
    B=B+FF
    AY1=Y(NE(J,1))-B1-YY
    AY2=Y(NE(J,2))-B1-YY
    AZ1=Z(NE(J,1))-C1-ZZ
    AZ2=Z(NE(J,2))-C1-ZZ
    AX1=X(NE(J,1))-A1-XX
    AX2=X(NE(J,2))-A1-XX
    A2=AY1*AZ2-AZ1*AY2
    B2=AZ1*AX2-AX1*AZ2
    C2=AX1*AY2-AY1*AX2
    GG1=DSQRT(A2**2+B2**2+C2**2)
    IF(L.EQ.1) THEN
      CF1=A2
    ELSEIF(L.EQ.2) THEN
      CF1=B2
    ELSE
      CF1=C2
    ENDIF
    T1=1./(R**3*GG1)
    T3=T1**2
    T6=A1*A2+B1*B2+C1*C2
    G=F(II)*W(II)*((-3*RNUM*R*GG1*T6*T3)+(CF1*T1))
    A=A+G
400 CONTINUE
    A=AE(J)*A
    B=AE(J)*B
    RETURN

```

END

Program AREA.FOR

C THIS CODE CALCULATE THE AREA UNDER A CURVE USING TRAPIZOIDAL  
RULE.

C

C INPUT

C N ; NUMBER OF INPUT POINTS

C X ; X-COORD. OF THE POINT

C Y ; Y-COORD. OF THE POINT

C

C OUTPUT ---> AREA ON THE SCREEN.

C

\$DEBUG

IMPLICIT REAL\*8(A-H,O-Z)

DIMENSION X(37),Y(37)

READ (7,\*) N

DO 100 I=1,N

READ (7,\*) y(i),x(i)

100 CONTINUE

SUM=0.0

DO 200 I=1,N-1

DX=X(I+1)-X(I)

IF(DX.GT.0.000000D0) THEN

T=1.0

ELSE

T=-1.0

ENDIF

TEMP=DABS(DX)\*DABS(Y(I+1)+Y(I))/2.D0

200 SUM=SUM+T\*TEMP

write(\*,1001) SUM

1001 format(e20.5)

STOP

END

Trajectory Plotting Program in BASIC

Program PART.BAS

```
10 OPEN "filename of traj coords" FOR INPUT AS #1
20 OPEN "filename of efficiency data" FOR INPUT AS #2
30 CLS
40 SCREEN 1
50 WINDOW (-6,-12)-(12,12)
60 INPUT #2, UO,UI,DP,DIN,THETA,EFF
70 THETA = 57.29577951#*THETA
80 LINE (0,DIN/2)-(0,-DIN/2)
90 LINE (0,DIN/2)-(-6,DIN/2)
100 LINE (0,-DIN/2)-(-6,-DIN/2)
110 WHILE NOT EOF(1)
120 INPUT #1, XX,YY
130 PSET (XX,YY)
140 WEND
150 CLOSE #1
160 CLOSE #2
170 KEY OFF
180 LOCATE 19,8:PRINT "EFF = ";EFF
190 LOCATE 20,8:PRINT "UO = ";UO
200 LOCATE 21,8:PRINT "DP = ";DP
210 LOCATE 22,8:PRINT "DIN = ";DIN
220 LOCATE 23,8:PRINT "THETA = ";THETA
230 LOCATE 24,8:PRINT "UI = ";UI
240 A$=INKEY$:IF A$="C" THEN 260 ELSE 250
250 IF A$="X" THEN 270 ELSE 240
260 LPRINT CHR$(12)
270 END
```

## REFERENCES

Addlesee, A. J., "Anisokinetic Sampling of Aerosols at a Slot Intake", Journal of Aerosol Science, Vol. 11, pp. 483-493 (1980).

Agarwal, J. K., Liu, B. Y. H., "A Criterion for Accurate Aerosol Sampling in Calm Air", American Industrial Hygiene Association JOURNAL, Vol. 41, pp. 191-198 (1980).

Alenius, S., "Calculation of Particle Motion in an Air Stream, Particularly Outside Exhausts," Study Report 1987:2, Swedish National Board of Occupational Safety and Health, Research Department, Technical Unit, Ventilation and Heating Section, 1987.

Alenius, S., Jansson, A., "Air Flow and Particle Transport into Local Exhaust Hoods, A Verified Computer Model", Arbete och Halsa 1989:34, Ventilation Division, National Institute of Occupational Health, S-171 84 Solna, Sweden, 1989.

Belyaev, S. P., Levin, L. M., "Investigation of Aerosol Aspiration by Photographing Particle Tracks Under Flash Illumination", Journal of Aerosol Science, Vol. 3, pp. 127-140 (1972).

Belyaev, S. P., Levin, L. M., "Techniques for Collection of Representative Aerosol Samples", Journal of Aerosol Science, Vol. 5, pp. 325-338 (1974).

Davies, C. N., Subari, M., "Aspiration Above Wind Velocity of Aerosols With Thin-Walled Nozzles Facing and at Right Angles to the Wind Direction", Journal of Aerosol Science, Vol. 13, No. 1, pp. 59-71 (1982).

Dunnett, S. J., Ingham, D. B., "A Mathematical Theory to Two-Dimensional Blunt Body Sampling", Journal of Aerosol Science, Vol. 17, No. 5, pp. 839-853 (1986).

Dunnett, S. J., Ingham, D. B., "The Human Head as a Blunt Aerosol Sampler", Journal of Aerosol Science, Vol. 19, No. 3, pp. 365-380 (1988).

Durham, M. D., Lundgren, D. A., "Evaluation of Aerosol Aspiration Efficiency as a Function of Stokes Number, Velocity Ratio and Nozzle Angle", Journal of Aerosol Science, Vol. 11, pp. 179-188 (1980).

Flynn, M. R., Miller, C. T., "The Boundary Integral Equation Method (BIEM) for Modeling Local EXhaust Hood Flow Fields", American Industrial Hygiene Association JOURNAL, Vol. 50(5), pp. 281-288 (1989).

Fuchs, N. A., "Review Papers Sampling of Aerosols", Atmospheric Environment, Vol. 9, pp. 697-707 (1975).

Hangal, S, Willeke, K, "Aspiration Efficiency: Unified Model for All Forward Sampling Angles", Environmental Science and Technology, Vol. 24, pp. 688-691 (1990).

Ingham, D. B., "The Entrance of Airborne Particles into a Blunt Sampling Head", Journal of Aerosol Science, Vol 12, No. 6, pp. 541-549 (1981).

Jayasekera, P. N., Davies, C. N., "Aspiration Below Wind Velocity of Aerosols with Sharp Edged Nozzles Facing the Wind," Journal of Aerosol Science, Vol. 11, pp. 535-547.

Liggett, J.A. and Liu, P.L.F., The Boundary Integral Equation Method for Porous Media Flow, G.A. Unwin, London, 1983.

Liu, B. Y. H., Zhang, Z. Q., Kuehn, T. H., "A Numerical Study of Inertial Errors in Anisokinetic Sampling", Journal of Aerosol Science, Vol. 20, No. 3, pp. 367-380 (1989).

Okazaki, K., Wiener, R. W., Willeke, K., "Isoaxial Aerosol Sampling: Nondimensional Representation of Overall Sampling Efficiency", Environmental Science and Technology, Vol. 21, pp. 178-182, (1987a).

Okazaki, K., Wiener, R. W., Willeke, K., "Non-Isoaxial Aerosol Sampling: Mechanisms Controlling the Overall Sampling Efficiency", Environmental Science and Technology, Vol. 21, pp. 183-187, (1987b).

Reist, P. C., Introduction to Aerosol Science, Macmillan, New York, 1984.

Okazaki, K., Wiener, R. W., Willeke, K., "The Combined Effect of Aspiration and Transmission on Aerosol Sampling Accuracy For Horizontal Isoaxial Sampling", Atmospheric Environment, Vol. 21, No. 5, pp. 1181-1185 (1987c).

Tufto, P. A., Willeke, K., "Dynamic Evaluation of Aerosol Sampling Inlets", Environmental Science and Technology, Vol. 16, pp. 607-609, (1982).

Weiner, R. W., "The Influence of Wind Condition and Inlet Size on Aerosol Particle Collection", disseration submitted to the Division of Graduate Studies of the University of Cincinnati, Cincinnati, OH, 1987, in partial fulfillment of the requirements for the degree of Doctor of Philosophy.