SANDIA REPORT

SAND97-8250 • UC-406 **Unlimited Release** Printed April 1997

M97052749

# In Situ Bioremediation: A Network Model of **Diffusion and Flow in Granular Porous Media**

S. K. Griffiths, R. H. Nilson, R. W. Bradshaw

r1

Prepared by Sandia National Laboratories Albuquerque, New Mexico 87185 and Livermore, California 94550 for the United States Department of Energy under Contract DE-ACO4-94AL85000

Approved for public release, distribution is unlimited.

#### DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recom-



This report has been reproduced from the best available copy.

Available to DOE and DOE contractors from: Office of Scientific and Technical Information P.O. Box 62 Oak Ridge TN 37831

Prices available from (615) 576-8401, FTS 626-8401.

----

----

Available to the public from:

National Technical Information Service U.S. Department of Commerce 5285 Port Royal Rd. Springfield, VA 22161.

NTIS price codes Printed copy: A02 Microfiche copy: A01

n

¢.

α

## DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

#### UC-406

## SAND97-8250 Unlimited Release Printed April 1997

# IN SITU BIOREMEDIATION: A NETWORK MODEL OF DIFFUSION AND FLOW IN GRANULAR POROUS MEDIA

## S. K. Griffiths, R. H. Nilson and R. W. Bradshaw Sandia National Laboratories Livermore, California 94551-0969

In situ bioremediation is a potentially expedient, permanent and costeffective means of waste site decontamination. However, permeability reductions due to the transport and deposition of native fines or due to excessive microorganism populations may severely inhibit the injection of supplemental oxygen in the contamination zone. To help understand this phenomenon, we have developed a micro-mechanical network model of flow, diffusion and particle transport in granular porous materials. The model differs from most similar models in that the network is defined by particle positions in a numerically-generated particle array. The model is thus widely applicable to computing effective transport properties for both ordered and realistic random porous media. A laboratory-scale apparatus to measure permeability reductions has also been designed, built and tested.

#### INTRODUCTION

In situ bioremediation is a promising new technology for groundwater and soils decontamination, offering many potential benefits over both excavation and flushing [1,2,3]. Rather than extracting contaminants for subsequent treatment or permanent burial, this technique relies on microbial metabolisis to decompose contaminants in place, yielding biomass and harmless byproducts [4,5,]. Nearly all organic contaminants can be metabolized in this manner, including the recalcitrant pesticides, PCBs, and chlorinated dioxins, as well as the many common hydrocarbons [6,7,8].

Current estimates for the costs of waste site bioremediation range from \$50-\$80 per ton, as compared with \$200 per ton or more for landfill disposal, and \$250-\$600 per ton for incineration [9]. Despite the apparent economic benefits, however, bioremediation is often not the preferred method for site restoration. The reason for this is in part because the technology is relatively new and in part because reliable predictive capabilities do not yet exist. Site managers cannot be assured a priori of the cost, duration and efficacy of bioremediation and so often opt for more direct though more costly alternatives. Improved understanding of bioremediation processes and improved capabilities for accurately predicting the duration and extent of cleanup would significantly increase the utility of this technology.

Successful application of bioremediation usually depends on injecting air, oxygen, oxygenated water or hydrogen peroxide to provide supplemental oxygen within the contamination zone [4,7]. Only in rare instances is bioremediation based on anaerobic decomposition [10,11]. Effective oxygenation of the contamination zone requires that soil permeabilities remain relatively high during the course of treatment. The treatment processes, however, may induce dramatic reductions in formation permeabilities. This phenomenon, known as formation blocking, usually results from one of three mechanisms: pore bubble nucleation due to the accumulation of gaseous decomposition products; pore plugging due to the long range transport and concentration of native fines; or pore plugging by the biomass resulting from excessive microorganism populations [12,13]. The two latter phenomena are the topic of our current study.

State-of-the-art modeling of bioremediation processes is generally based on three-dimensional, multicomponent, multiphase codes employing well established relative permeability transport algorithms [14,15]. These codes have been developed over nearly two decades to solve the somewhat simpler problem of subsurface contaminant transport [16]. The primary shortcoming of these codes in modeling bioremediation is that they neglect micro-scale phenomena in order to address the multidimensional macro-scale processes of the entire waste site [17,18]. Convective transport of particles and microorganisms is generally neglected, as is the effect of biomass on formation permeability. Similarly, most laboratory studies of biodegradation and bioremediation have concentrated on identifying specialized microorganisms and chemical environments which accelerate contaminant decomposition. While these studies consistently show the importance of supplemental oxygen to support the microorganism population, only a few have focused on the transport processes necessary to provide this oxygen in the contamination zone [12,14,19,20].

In the present study, we have developed a micro-mechanical network model intended to describe several micro-scale or sub-grid processes that are not explicitly treated in large-scale continuum models. A source listing of the program is given in Appendix A. One unique aspect of this model is its emphasis on particle transport and its effect on permeability. A primary goal of this work is to improve the predictive capability of bioremediation modeling by helping to develop improved continuum correlations relating the macroscopic permeability and effective diffusivity to the evolved state of an interconnected pore structure. In addition, this research provides new capabilities for modeling fluid, species and particle motion in both ordered and random porous materials, with potential application to a wide range of micro-scale fluid transport and filtration problems.

In the experimental part of the present study, we have designed and built a laboratory-scale apparatus for measuring permeability reductions due to fines transport and deposition. Using this capability, we have made preliminary measurements of permeability reductions in glass spheres and in several common sands. The goal of this work was to provide permeability data on well characterized porous materials that would serve as a benchmark for the mathematical model.

#### NETWORK MODEL

Previous continuum models of plugging processes have failed to predict accurately the permeability reductions observed in laboratory and field experiments. This is true not only for bioremediation, but for more conventional filtration processes as well. Likewise, there has been very limited success with analytical models intended to describe plugging processes in terms of the reduction in throat size in a bundle of parallel capillary tubes. These classes of models fail to account for the interconnection among flow passages of differing sizes and the redistribution of flow that occurs as plugging proceeds.

To overcome these shortcomings we have developed a micro-mechanical network model that describes the internal structure of a random porous material as a system of interconnected tubes and nodes [21,22,23]. Tube networks such as these are generally constructed by placing tubes directly in the computational domain using a Poisson distribution or similar random process. Our approach differs from this in that the tube and node network is constructed about a collection of particles [24,25,26,27].

The computational domain is first symbolically packed with spherical particles from a specified size distribution. The size distribution is obtained by sampling spike, normal or log-normal distributions using a uniform random number generator. By this technique, a repeatable pseudo-random distribution of particle sizes of a specified mean and variance may be generated very quickly.

Once a particle size distribution is generated, the particles are assembled into a porous structure. To perform this task we have developed several packing algorithms. These appear in lines pages 3 through 19 of Appendix A. The first method employs a random number generator to place particles in a specified box, subject to the constraint that no particles may inter-penetrate. This constraint may be further tightened to prevent the placement of a particle within a specified distance of any other. Particles are added in this manner to the box until a specified fractional density is obtained. This method is reasonably fast, but fractional densities greater than about 40% cannot be obtained by this technique. This is because granular materials having greater densities are not at all random.

To obtain high fractional densities, we have developed a second packing algorithm in which particles are added randomly to the top of the box. These particles are then moved downward (as though by gravity) until they arrive at the upper surface of those particles previously placed. When a falling particle first contacts another, the region around that fixed particle is examined numerically to determine the minimum energy state in which the new particle can be placed. The processes of adding individual particles is repeated until the box is full. This algorithm is quite slow since the size and location of many other particles must be checked as each new particle is packed. Despite the large effort required by this method, and the seemingly high degree of consolidation that should be obtained, fractional densities obtained by this method rarely exceed about 65%, depending on the size distribution of the particle set.

To obtain still high packing densities, we have developed a third algorithm in which all particles are initially placed by deterministic methods to obtain the desired density. Large particles are placed first in a regular packing. Smaller and still smaller particles are added later within the voids formed by those already placed. The resulting structure is highly ordered, and does not provide a good representation of a random material. To achieve greater disorder, each particle in the final set is sent on a random, noninterfering walk through the box. This is an extremely time consuming process, since the current position of each particle must be checked against all others in the box to ensure that inter-penetration does not occur. For the large computational price, however, fractional densities as high as the size distribution will permit may be obtained by this method.

Once the size and position of each particle has been determined, the network model is constructed. For this we have employed a subdivision of the space via a Voronoi tessellation [28]. This tessellation assigns each region of the domain to the closest particle. The Voronoi tessellation results in a polygon bounding each particle. The sides of these polygons form the skeleton of the network of channels surrounding the particles. A sample calculation showing the network at this stage is shown in Fig. 1. Note that each particle is bounded by several network channels. The most frequent number of bounding channels is five, though in large particle arrays as few as three and as many as twelve bounding channels may be seen. Also note that network channels always form intersections of three, and that these intersections define a unique node location at the end of each channel. Figure 1. Schematic of particle array and corresponding Voronoi tessellation. The sides of the tessellation polygons form the network of channels through which fluid transport occurs.



Figure 2. Schematic of particle array and channel network. Channel intersections define the nodes of the computational domain. Although these channels are displayed as having parallel walls, the governing transport equations take into account that channels conform to particle surfaces.



After the Voronoi tessellation is computed and the skeleton of the network is known, the nodes identified by pore tube intersections are numbered and cross correlated. This cross-correlation table provides a numerical map of connected node pairs and corresponding tube numbers, making the network a useful computational domain for solving the transport equations. Following construction of the network skeleton, channel sizes for each segment of the network are computed using the known center positions and radii of the two particles defining each channel segment. The source listing of the tessellation algorithm and method of network construction appears in pages 19 through 35 of Appendix A.

A sample of a completed tube network is shown in Fig. 2. This network represents the pore structure in a two-dimensional slice through a three-dimensional particle array. In the sample shown, and in Fig. 1, the particles are uniform spheres. They appear to have nonuniform sizes only because of varying positions in the out-of-plane direction. The volume within this collection of tubes is the domain of the network model on which the conservation equations are solved.

#### CONSERVATION EQUATIONS

The transport equations governing fluid motion and species transport in a network model are much simpler than those usually encountered in multidimensional transport in porous materials. The reason for this is that all complexities of the pore geometry are explicitly described by the network itself. Phenomena such as reduced diffusivities due to the presence of a solids fraction and both longitudinal and transverse dispersion arise naturally as a consequence of the network geometry. As a result, these porous media transport phenomena, described by empirical correlations in conventional continuum models, are addressed in a network model on a first-principles basis, or nearly so.

For the isothermal transport of a species i through the tube network, conservation of mass is given simply by

$$V_k \frac{d}{dt}(\rho_k f_{i,k}) = \sum_{j=1}^3 q_{j,k} \bar{f}_{j,k} - S_k$$
(1)

where  $V_k$  is the portion of tube volume associated with node k,  $\rho_k$  is the fluid density at node k, and  $f_{i,k}$  is the local species fraction. The summation on the right of Eq. (1) is a sum over the species transport rates from the three tubes connected to node k, and  $\bar{f}_{j,k}$  is the mean species concentration in each tube. The final term on the right of Eq. (1) accounts for any sources or sinks do to surface or homogeneous reactions.

Momentum equations for the network model are likewise much simpler than those for continuum models of porous materials. Here, for low Reynolds number flows, the mean fluid velocity is related to the pressure gradient by conventional tube or channel correlations.

$$q_{j,k} = -\rho \frac{\overline{\delta_{j,k}^3}}{24} \frac{\Delta p_{j,k}}{\ell_{i,k}} \quad \text{or} \quad q_{j,k} = -\rho \pi \frac{\overline{\delta_{j,k}^4}}{64} \frac{\Delta p_{j,k}}{\ell_{i,k}} \tag{2a,b}$$

 $\Delta p_{j,k}$  is the pressure drop across the tube joining the j and k nodes,  $\ell$  is the tube length, and  $\overline{\delta}_{j,k}$  is the equivalent tube aperture yielding the correct mean flow rate. Equivalent apertures are discussed further in the following section. The first of these relations applies to a two-dimensional channel, appropriate for flow in two-dimensional geometries, while the second applies to a circular tube, appropriate for three-dimensional particle arrays.

To solve the governing transport equations, pressures are imposed on two external surfaces of the network domain. The other two boundaries are made impermeable by closing any tubes crossing these planes. Pressures for the tube network are then computed by solving Eq. (1) for each interior node. This system of coupled node equations is solved by a time marching algorithm. Using an appropriate equation-of-state relating the fluid pressure, temperature and density, either liquid or gas flows may be treated in this way. The algorithms for computing the pressure field are given in pages 35 through 48 of Appendix A.

Once the internal pressures and tube mass flow rates are computed, the permeability of the network is calculated by summing the flow contributions from each tube to obtain the total flow rate through the entire network. From this total flow rate, the fluid properties, size of the domain, and the specified boundary pressures, the macro-scale permeability of the medium can be computed. By letting tube diameters diminish as biomass accumulates or as channels become blocked by particulate fines, changes in the pore structure and the effect of the evolving structure on the flow rate can also be computed.

Finally, we note that Eq. (1) does not contain any contribution due to diffusive transport. Although this additional transport mechanism can be included in this manner, we have found that describing diffusive processes via tracer particles is generally a more direct means of utilizing the full capabilities of the network model. Tracer particle dynamics are addressed in a later section.

#### NETWORK SUB-SCALE MODELS

Although a network model resolves transport phenomena down to the particle scale, many important processes take place on still smaller scales. Diffusion and viscous dissipation, for example, both involve processes occurring at the molecular scale. Rather than explicitly modeling the details of molecular interactions, these sub-scale phenomena are treated here in the conventional continuum fashion. Further, by averaging the governing continuum equations over the channel crosssections and then integrating over their lengths, closed-form expressions for effective transport properties of each tube segment are obtained. These permit computation of the full pressure and concentration fields using state variables only at the network nodes.

To account for sub-scale processes in the network model, we define four equivalent apertures relating the tube volume, effective permeability, diffusivity and fluid transit time to the tube length. The simplest of these is the equivalent aperture for tube volume,  $\delta_V$ , defined by

$$V = \ell \,\delta_V \tag{3}$$

where  $\ell$  is the tube length, and V is its volume. It is staightforward to show that in this case the equivalent aperture is given by

$$\delta_V = \frac{2}{\ell} \int_0^{\ell/2} \delta dz \tag{4}$$

where  $\delta$  is the local tube aperture.

The second equivalent aperture is that of the permeability. In this case, the equivalent aperture for two-dimensional flow is defined by

$$q = -\frac{\delta^3}{12} \frac{dp}{dz} = -\frac{\delta_k}{12} \frac{\Delta p}{\ell}$$
(5)

where q is the constant volumetric flow rate through the tube, p is the local pressure and  $\Delta p$  is the total pressure drop along the tube. Note that we make no distinction here between compressible and incompressible flow. The reason for this is that each tube is very short, thus the ratio of the pressure drop to the mean pressure is always very small. In light of this, the effects of compressibility may be neglected on this scale. To solve for  $\delta_k$  requires that the local pressure is integrated over the tube length to obtain the total pressure drop. Substituting that result into Eq. (5) then yields

$$\frac{1}{\delta_k^3} = \frac{2}{\ell} \int_0^{\ell/2} \frac{dz}{\delta^3} \tag{6}$$



Figure 3. Diagram of a particle pair and flow channel. Equivalent apertures for pore volume, flow, diffusion, and through-channel transit time are computed using the channel geometry.

for the equivalent aperture providing the correct viscous forces within each tube.

The third equivalent aperture is that for the diffusivity. This is defined such that the constant diffusive transport rate, f, through a tube is correctly specified by

$$f = \delta D \frac{dc}{dz} = \delta_D D \frac{\Delta c}{\ell} \tag{7}$$

where D is the coefficient of binary diffusion, c is the local concentration of the diffusing species and  $\Delta c$  is the total variation of the concentration along the tube length. Now integrating Eq. (7) over the tube length to obtain the total variation in the concentration, and substituting that result back into Eq. (7) gives

$$\frac{1}{\delta_D} = \frac{2}{\ell} \int_0^{\ell/2} \frac{dz}{\delta} \tag{8}$$

for the equivalent tube aperture based on diffusion.

The final aperture to be defined is that for the mean fluid velocity or transit time. This yields the correct time a tracer particle is resident within a tube and so is useful in tracking particle motion within the network. In this case, the equivalent aperture is defined by

$$t = 2 \int_0^{\ell/2} \frac{dz}{u} = \frac{\ell}{\bar{u}} \tag{9}$$

where t is the tube transit time, and u and  $\bar{u}$  are the local and mean fluid speeds, respectively. These are given by

$$u = -\frac{\delta^2}{12} \frac{dp}{dz}$$
 and  $\bar{u} = -\frac{\delta_u^2}{12} \frac{\Delta p}{\ell}$  (10a,b)

Substituting Eqs. (10a) and (10b) into Eq. (9) and rearranging slightly yields

$$\frac{1}{\delta_u^2} = \frac{2}{\ell} \int_0^{\ell/2} \frac{\delta}{\delta_k^3} dz = \frac{\delta_V}{\delta_k^3} \tag{11}$$

Thus the equivalent aperture for mean fluid speed and transit time can be written simply in terms of those for the permeability,  $\delta_k$ , and the tube volume,  $\delta_V$ .

To apply these definitions of the four equivalent apertures, we must now take into account a specific particle geometry. Consider two two-dimensional particles having centers  $(x_1, y_1)$  and  $(x_2, y_2)$ , and radii  $R_1$  and  $R_2$ . As shown in Fig. 3, the local aperture between these particles is given by

$$\delta = \delta_0 + R_1 + R_2 - \sqrt{R_1^2 - z^2} - \sqrt{R_2^2 - z^2}$$
(12)

where the minimum aperture, occurring along the line joining the two centers, is

$$\delta_0 = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} - R_1 - R_2 \tag{13}$$

and z is the distance measured in a direction orthogonal to the line of centers. Now taking

$$R_2 = R_1 (1 + \epsilon)$$
 and  $z = R_1 \sin \theta$  (14a,b)

and substituting these results into Eq. (12), the local aperture can be expressed in terms of only  $R_1$ ,  $\epsilon$  and  $\theta$ .

$$\delta = \delta_0 + R_1 \left[ \left[ 2 + \epsilon - \cos\theta - (1 + \epsilon) \left[ 1 - \frac{\sin^2 \theta}{(1 + \epsilon)^2} \right]^{1/2} \right] \right]$$
$$\approx \delta_0 + R_1 \left[ 1 - 2\cos\theta + \epsilon(1 - \sec\theta) \right] \quad (15a,b)$$

Note that the latter of these relations is based on a one-term expansion for small values of  $\epsilon$ .

Substituting Eq. (12) into Eq. (2) and performing the indicated integration yields

$$\frac{\delta_V}{R_1} = 2\kappa + \epsilon - \frac{1}{2} \left[ (1+\epsilon)^2 - \sin^2 \theta \right]^{1/2} - \frac{R_1}{\ell} (1+\epsilon)^2 \sin^{-1} \left( \frac{\sin \theta}{1+\epsilon} \right) - \frac{1}{2} \left( 1 - \sin^2 \theta \right)^{1/2} - \frac{R_1}{\ell} \theta \qquad (16a,b)$$

for the equivalent volume aperture. The new parameter,  $\kappa,$  appearing here is given by

$$\kappa = 1 + \frac{\delta_0}{2R_1} \tag{17}$$

and in this case  $\theta$  is evaluated at the angular limit of integration

$$\theta = \sin^{-1}\left(\frac{\ell}{2R_1}\right) \tag{18}$$

corresponding to the end of the tube segment.

Similarly, the equivalent aperture yielding the correct viscous drag is obtained by substituting Eq. (15b) into Eq. (6) and integrating the result. Using a tedious though straightforward change of variables, this gives

$$\frac{\delta_k^3}{\ell R_1^2} = 4 \frac{\kappa^2 - 1}{f(\kappa, \theta)} \left[ 1 + \frac{3\epsilon I(\kappa, \theta)(\kappa^2 - 1)}{2f(\kappa, \theta)} \right]$$
(19)

where

$$I(\kappa,\theta) = \frac{1}{3(\kappa+1)} \left[ \frac{\sin\theta}{(\kappa-\cos\theta)^3} - \frac{3h(\kappa,\theta)}{2(\kappa^2-1)} + \frac{2f(\kappa,\theta)}{2(\kappa^2-1)} \right]$$
(20)

where the functions  $h(\kappa, \theta)$  and  $f(\kappa, \theta)$  are

$$h(\kappa,\theta) = \frac{\sin\theta}{(\kappa - \cos\theta)^2} + \frac{3\kappa\sin\theta}{(\kappa^2 - 1)(\kappa - \cos\theta)} + 2\frac{2\kappa^2 + 1}{(\kappa^2 - 1)^{3/2}}\tan^{-1}\left[\left(\frac{\kappa + 1}{\kappa - 1}\right)\tan\left(\frac{\theta}{2}\right)\right]$$
(21)

and

$$f(\kappa,\theta) = \frac{(2+\kappa^2)\sin\theta}{2(\kappa^2-1)(\kappa-\cos\theta)} + \frac{\kappa\sin\theta}{2(\kappa-\cos\theta)^2} + \frac{3\kappa}{(\kappa^2-1)^{3/2}}\tan^{-1}\left[\left(\frac{\kappa+1}{\kappa-1}\right)^{1/2}\tan\left(\frac{\theta}{2}\right)\right]$$
(22)

respectively.

Finally, we consider the equivalent aperture appropriate for computing diffusive transport through the tube network. Now substituting Eq. (15b) into Eq. (8) and again performing the integration gives

$$\frac{\delta_D}{\ell} = \frac{1}{g(\kappa,\theta)} \left[ 1 + \frac{\epsilon \psi(\kappa,\theta)}{2(\kappa+1)g(\kappa,\theta)} \right]$$
(23)

where

$$g(\kappa,\theta) = -\theta + \frac{2\kappa}{(\kappa^2 - 1)^{1/2}} \tan^{-1} \left[ \left(\frac{\kappa + 1}{\kappa - 1}\right)^{1/2} \tan\left(\frac{\theta}{2}\right) \right]$$
(24)

and

$$\psi(\kappa,\theta) = \frac{\sin\theta}{\kappa - \cos\theta} - \frac{2}{(\kappa^2 - 1)^{1/2}} \tan^{-1} \left[ \left( \frac{\kappa + 1}{\kappa - 1} \right)^{1/2} \tan\left( \frac{\theta}{2} \right) \right]$$
(25)

For the special case in which all particles within the particle array are the same size, the results above become greatly simplified. Rewriting Eq. (16) for  $\epsilon = 0$  yields

$$\frac{\delta_V}{R} = 2\kappa - \left(1 - \sin^2\theta\right)^{1/2} - 2\frac{R}{\ell}\theta \tag{26}$$

for the equivalent volume aperture. Similarly, Eq. (19) in this limit reduces to

$$\frac{\delta_k^3}{\ell R^2} = \frac{4\left(\kappa^2 - 1\right)}{f(\kappa, \theta)} \tag{27}$$

for the equivalent aperture for viscous fluid flow. Finally, the equivalent aperture for diffusion, given by Eq. (23) above, becomes

$$\frac{\delta_D}{\ell} = \frac{1}{g(\kappa, \theta)} \tag{28}$$

for this case of uniform particle size. As before, the angular limit of integration,  $\theta$ , is taken as that corresponding to the tube length.

#### EFFECTIVE TRANSPORT PROPERTIES

The sub-grid models described above are intended for use in a network model generally capable of describing transport in disordered porous materials. We can, however, also apply these sub-grid relations directly to regular particle arrays. For such regular arrays, the geometry of the corresponding pore network is easily specified. In addition, the permeability and diffusivity have been computed by direct solution of the Navier-Stokes and diffusion equations for several regular patterns of two and three-dimensional particles [29,30,31]. Comparing the present results with



Figure 4. Unit cell of uniform hexagonal array of cylindrical particles. Particle size and minimum aperture determine porosity, diffusivity and permeability.

these numerical solutions provides a useful check on the accuracy of the network approach and on the range of applicability of correlations derived from the equations outlined earlier.

To this end, we now consider the simple problem of flow and diffusion through a regular hexagonal array of circular cylinders. As shown in Fig. 4, the particle size and geometry of the unit cell uniquely determine all transport properties of the array. From the definitions of the equivalent apertures, the effective permeability of this array is given by

$$k = \frac{1}{12} \frac{\delta_k^3}{h\sqrt{\tau}} \tag{29}$$

where  $h = \sqrt{3}S/2$  is the height of the unit cell, and  $\tau = 4/3$  is the tortuosity for flow from left to right in the geometry shown. Again,  $\delta_k$  is the equivalent aperture for viscous flow. Using Eq. (27), this result may be expressed as

$$k = \frac{1}{12} \frac{\left(\kappa^2 - 1\right)}{f(\kappa, \theta)} \frac{\ell d^2}{h\sqrt{\tau}} \tag{30}$$

where  $f(\kappa, \theta)$  is given by Eq. (22). Again from the geometry we obtain

$$\frac{\ell}{h} = \frac{2}{3}$$
 and  $\kappa = 1 + \frac{\delta_0}{d} = \sqrt{\frac{1 - \phi_0}{1 - \phi}}$  (31a,b)

where  $\phi$  is the porosity (void volume fraction), and  $\phi_0 = 1 - \pi/2\sqrt{3} \approx 0.093$  is the porosity at the percolation threshold. The percolation threshold is the condition at which interconnected pores just marginally span the material sample of interest. Also from geometry, the angular limit of integration is

$$\theta = \sin^{-1}\left(\frac{\ell}{d}\right) \quad \text{where} \quad \frac{\ell}{d} = \frac{\kappa}{\sqrt{3}}$$
 (32a,b)

Now defining a reference permeability as the area of the unit cell,

$$k_r = \frac{\pi d^2}{4(1-\phi)} \tag{33}$$

the normalized permeability may be expressed as

$$k^* = \frac{k}{k_r} = \frac{\phi - \phi_0}{3\sqrt{3}\pi f(\kappa, \theta)} \tag{34}$$

We note that exactly this result is also obtained for vertical flow through the unit cell of Fig. 4. The application of equivalent apertures to that problem is somewhat more difficult, however, since the unit cell for flow in that direction involves the confluence of two tubes into one.

A comparison between these results and an exact analytical solution to the problem [32] is shown in Fig. 5. For this simplified geometry of a regular hexagonal array, Eq. (34) agrees to within 15% for all porosities above the percolation threshold and below  $\phi = 0.9$ . Based on this agreement, we conclude that the expressions describing the equivalent aperture for viscous flow in two-dimensional channels should be suitable for network modeling of flow through disordered materials. Similar agreement has been obtained between the equivalent aperture formulas and exact solutions for two-dimensional square arrays and for the more complex problem of three-dimensional flow through regular arrays of spheres.

We now consider the use of equivalent apertures for computing the effective diffusivity of a regular hexagonal array. From Eqs. (8) and (28), the effective diffusivity may be written as

$$D^* = \frac{D_e}{D} = \frac{\delta_D}{h\sqrt{\tau}} = \frac{\ell}{hg(\kappa,\theta)\sqrt{\tau}}$$
(35)

where  $D_e$  is the apparent diffusivity in the porous array, and D is the coefficient of diffusion. This expression can be evaluated directly using the values of  $\ell/h$ ,  $\kappa$ ,  $\theta$  and  $\tau$  given above, along with Eq. (24) for  $g(\kappa, \theta)$ . Again, although this result applies to diffusion from left to right in the geometry shown in Fig. 4, the same



result is obtained for diffusion in the vertical direction, and for that matter, in any arbitrary direction through the unit cell.

Figure 5. Comparison between network results and

exact solutions from previous analyses of the normal-

ized permeability and effec-

tive diffusivity for a regular

hexagonal array of cylindri-

cal particles.

A comparison between this result and analytical solutions to the diffusion problem [33] is also shown in Fig. 5. In this case, Eq. (35) agrees to within 10% for all porosities above the percolation threshold and below  $\phi = 0.8$ . Again, a similar approach has been applied to other geometries with comparable agreement between the results of the network model and direct numerical solutions.

Although the diffusivity and permeability of a well ordered medium can be described analytically, numerical simulations are usually needed to determine the transport properties of random materials. To deduce the permeability of a particular network model, it is only necessary to set pressure boundary conditions on a pair of opposing faces and calculate the resulting flow rate. By seeding tracer particles into the flow and observing their motion it is then possible to observe the process of hydrodynamic dispersion that results from differences in fluid speed along different streamlines. The dispersivity of a given medium is influenced by the distribution of tube sizes and lengths, the degree of connectivity, and the degree of mixing that occurs at tube junctions. Dispersion also occurs on the scale of a single flow tube owing to differences in fluid speed between wall and center streamlines. All of these factors are accounted for explicitly in the network model by means of tracer particles.

#### PARTICLE TRANSPORT

Tracer particles may serve either as fictitious markers of the fluid motion or as physical particles of finite size. In the first capacity they are used to represent diffusive and dispersive contributions to the transport of reactive species carried by the mean flow [34,35,36]. In the latter role, physical particles of a finite size can be advected through the network to describe the advection, deposition, and accumulation of particles in the void space [37,38,39]. Blockage or partial blockage of individual tubes is straightforward to compute since the full geometry of the tube network is known and may be evolved in time. Thus, the reduction in permeability due to the relocation and deposition of biomass and native fines can be calculated from fundamental considerations.

To track the motion of tracer particles requires a knowledge not only of their streamline positions within tubes but also of their trajectories at nodal interconnections. Two basic configurations of nodes are possible: (1) a node having one in-flow and two out-flow tubes: and (2) a node having two in-flow and one out-flow tubes. These are the only possible configurations for a network constructed via Voronoi tessellation. In assigning particles to outflow tubes it is simplest to assume that all junctions are well mixed and to assign probabilities to the outflow channels and their streamlines based on their respective flow rates. However, it is generally much more realistic to require that particles follow continuous paths that smoothly interconnect the incoming and outgoing streamlines. That substantially more difficult approach is implemented in the present network model.

To compute particle motion through a network node, we assume a parabolic fluid velocity profile over the cross-section of each of the three tubes forming the node. Given the spatial position of a particle as it exits a tube, the node entrance streamline based on the parabolic velocity distribution can be determined. The corresponding node exit streamline can then be computed based on the total node in-flow and out-flow and knowledge of whether the node currently posses one or two in-flow tubes. Having computed the node exit streamline, the particle is placed at the correct radial position (in the correct exit tube, if more than one exit tube exists) again based on the parabolic velocity profile. Using this approach, real or tracer particles may be transported through the entire tube network following a single streamline through the repeated branching and confluence of tubes. For low Reynolds number creeping flows, typical of those in the applications of interest, turbulent transport of particles across streamlines does not exist. The only mechanism for this process is particle diffusion. It is this diffusion, along with the velocity profile within each tube and the variation in mean tube velocities, that gives rise to both longitudinal and transverse dispersion in flows in porous material.

Dispersion in a porous material is a complex process involving simultaneous flow and diffusion [40,41]. To model this process, we first compute the mean flow

field for the entire tube network. This yields the local fluid speed at each radial and longitudinal position within each network tube. We then inject a large number of particles at the inflow boundary and track their progress through the network for a fixed time. Particles are partitioned among tubes on the inlet boundary based on the total inlet flow rate and the contribution to this total from each inlet tube. This ensures that particles enter the network inlet tubes in correct proportions. Once an inlet tube is selected, the particle is placed at a radial position using a probability distribution that reflects the parabolic velocity profile. This ensures that particles entering a given tube are correctly distributed in accordance with local fluid speeds. Under these procedures, particles enter the network as though they were supplied from a reservoir of fluid containing a uniform particle concentration.

As the injected particles enter the tube network, they are advected by the local mean flow and diffuse about this mean speed. The diffusive portion of this transport is described by a noninterfering random walk [21,34]. At each time step, an advective displacement is computed from the local fluid speed and the size of the time step,  $\delta t$ . The time step may be constant or may be obtained by sampling a uniform random distribution. For the same time period, a random diffusive displacement,  $\delta \ell_D$  is computed from the time step size and the specified diffusivity, D. This is given by

$$\delta\ell_D = \xi \sqrt{2D\delta t} \tag{36}$$

where  $\xi$  is a distribution function that may be unity, random or Gaussian. All three give very similar results, provided that the step size is adjusted such that the mean step size is consistent with Eq. (36). The direction of the diffusive step is then computed by sampling a uniform random distribution, and the combined advective and diffusive steps are taken. If the resulting particle position is outside the tube network, the step is recomputed.

Since all of the tracer particles advance with different speeds, they tend to spread apart as they traverse the medium. To quantify this longitudinal dispersion, each of the particles is transported through the network for a fixed period, t, and the final position of each is noted. Following the injection and transport of a large number of particles, the cumulative distribution of final particle positions is fit with an error function to obtain the mean and variance of the particle positions. The error function is used because it is the solution to the continuum dispersion equations. The dispersivity,  $D^*$  is then be calculated from the variance,  $\sigma^2$  of the particle positions by

$$D^* = \frac{\phi}{D} \frac{\sigma^2}{2t} \tag{37}$$

where  $\phi$  is the porosity of the medium. The dispersivity is a property of the material, and so should be independent of the time interval, t, provided that the interval is large enough to sample a statistically significant portion of the network. The procedure above is used to compute longitudinal dispersion and the longitudinal dispersivity. Transverse dispersivities are computed by a similar method, except that particles are injected into only a single entrance tube. In this case, the particles are tracked and the final transverse position of each is noted. These positions are fit using a Gaussian distribution, and the transverse dispersivity is computed from the variance, again using Eq. (37).

A random walk is also used, in the absence a net fluid motion, to compute the effective diffusivity in the network model. As in the earlier simulations of hydrodynamic dispersion, particles are introduced into tubes along a vertical or horizontal line through the network. In this case, however, the starting line is generally centered within the medium since there will be no net displacement of the particle front. From their initial positions, the particles are again sent on random walks. After a specified time period, the effective diffusivity of the network is computed by matching a Gaussian distribution to the computed spatial distribution of the final tracer positions.

A number of options may be exercised in computing diffusivities. The computational algorithm permits either a random walk on the tube network or directly on the void volume defined by the particles comprising the granular material. Also, when computing diffusivities, the distinct contributions of ordinary and Knudsen diffusion may be determined by varying the mean diffusive step size relative to the characteristic pore diameter. Knudsen diffusion becomes dominant when the mean free path (diffusive step size) of the tracer particles become comparable to that of the pore size. In this regime, most diffusive steps result in collisions with a particle of the porous structure. Although Knudsen diffusion is not usually important in bioremediation applications, this capability of the network model has been employed in computing effective diffusivities for porous fiber preforms used in composites manufacturing by low-pressure chemical vapor infiltration.

Particle transport simulations are particularly advantageous in computing the transport of reactive chemical species [35,36]. Using minor variations on the methods above, species concentrations can be computed from particle concentrations during continuous particle injection. Particle lifetimes defined by reaction probabilities, can be used to account for both homogeneous and surface reactions. Surface reaction rates are especially easy to compute by this technique since the surface impingement rate, defined by a particle displacement to a region outside the tube network, is already monitored as a necessary part of the particle advective and diffusive motion. Details of the mathematical methods used in particle transport appear in pages 48 through 58 of Appendix A.

#### SAMPLE CALCULATIONS

To demonstrate the unique capabilities of a network model, we now present the results of calculations for several sample problems. These sample problems include the pressure field for an incompressible flow, permeability reductions due to uniform film growth on particles of a granular medium, particle-scale fingering and the associated longitudinal dispersion, and the effects of the Peclet number on transverse dispersion.

The first step in solving all transport problems using a network model is to compute the pressure field. This is done by solving the coupled continuity equations for all network nodes by means of a relaxation technique, subject to the desired boundary conditions. The result is a two or three-dimensional spatial distribution of pressures. A sample pressure field is shown in Fig. 6. The horizontal axis in this plot is the spatial position along the direction of flow. The vertical axis is the normalized pressure, where the normalization is such that the inlet pressure is unity and the exit value is zero. Boundary conditions for this sample problem are fixed pressures on the inlet and exit and impermeable boundaries on the top and bottom of the domain. The inlet and exit conditions are imposed by identifying those nodes lying on these boundaries and assigning the appropriate value, which is then held fixed through computation. Conditions on the impermeable boundaries are imposed by identifying those tubes crossing the top or bottom of the domain, and assigning to these tubes an effective aperture of zero. In this manner, no flow may cross the top or bottom boundaries. These particular boundary conditions are frequently used because they permit a direct calculation of the directional permeability once the pressure and flow fields have been determined.

This plot shows one particularly interesting feature of a typical pressure field. Although the mean gradient of the pressure is always negative, corresponding to flow from the inlet toward the exit, local pressure gradients are sometimes positive. That is, on the scale of particles within a porous granular material, local fluid velocities may oppose the mean flow direction. This condition arises naturally in disordered materials and is important because such local variations in both the magnitude and direction of fluid speeds contribute significantly to the very large apparent diffusivities associated with longitudinal dispersion.

Another interesting feature of Fig. 6 is the degree to which the mean node pressures deviate from the linear gradient obtained from a continuum model. The gradient of mean node pressures is about 20% above the linear value near the inlet and about 20% below at the exit. This deviation from the continuum result arises because the permeability of the network is locally lower than the average value near the inlet and locally higher near the exit. Since the total flow rate through the network is the same at all axial positions, low local permeabilities give high pressure gradients, while high permeabilities give relatively lower gradients of the mean node

Figure 6. Normalized pressures at nodes inside a tube network. Dashed curves indicate interconnected nodes. The solid diagonal line shows the linear pressure gradient that satisfies the continuum equations for low Reynolds number flow of an incompressible fluid through a homogeneous porous medium.

Normalized Pressure



Logitudinal Position

pressure. This behavior is a result of the inherent nonuniformity of disordered porous materials. On larger domains, this effect is still more pronounced. This is important to the problem of bioremediation, since low permeabilities are associated with large specific surface area, and large surface areas lead to high contaminant retention. Thus regions most likely to require decontamination are also the most difficult to supply with the oxygen and nutrients needed for rapid treatment by this method.

We now consider the problem of the evolving pore structure and associated reductions in permeability due to accumulation of solids on particles of a granular porous solid. To generate the pore networks for this type of problem, we first generate a collection of particles by one of the three methods previously described. Then, a region of uniform thickness around each particle is excluded from the void space to account for that portion of the initial void occupied by the deposited material. This numerical process mimics that of biofilm growth on the surfaces of granular solids when the particle size is much larger than that of the microbe. Finally, the tube network is constructed about the particles and accumulated mass to obtain a network representation of the remaining void volume.

Three sample networks constructed in this way are shown if Fig. 7. Figure 7A represents the initial formation, while Figs. 7B and 7C represent the same collection

of particles with 33% and 66% of the initial void filled. The initial network (7A) at a porosity of 0.6 consists of 223 tubes and has a normalized permeability of about  $k^* = 1.1 \times 10^{-3}$ . When the porosity is reduced to 0.4, the resulting network (7B) still contains 166 tubes, but the normalized permeability has dropped by over an order of magnitude to only  $k^* = 8.0 \times 10^{-5}$ . This large reduction in permeability is due only in a small part to the fact that the remaining tubes have reduced diameters. The more important reason for this large effect is that the network void volume is losing connectivity. In the initial configuration, at a porosity of 0.6, over 97% of all the nodes are connected to two other nodes. At a porosity 0.4, only 37% of the nodes are still connected to two other nodes, over 50% are connected only to one, and about 10% are no longer connected to the network. Finally, at a porosity of 0.2, only 14% of the nodes remain doubly connected, 55% possess a single connection, and over 25% have become altogether isolated. It is this dramatic decrease in the number of interconnected pores that accounts for most of the large drop in permeability.

Despite the fact that the network still retains a large number of node connections at a porosity of 0.2, the permeability of the network shown in Fig. 7C is zero. At this porosity, the network is just below the percolation threshold, as a continuous path connecting the entrance and exit planes no longer exists.

To illustrate the use of tracer particles in computing effective transport properties, we now consider the problems of longitudinal and lateral dispersion. Dispersion in a porous material is a complex process involving both flow and diffusion. Variations in local fluid velocities yield varying particle speeds as tracer particles traverse the pore network. Diffusion is important to this process because only by diffusion can particles move from one streamline to another. High diffusion coefficients give rise to a wider sampling of fast and slow streamlines in large and small pores, yielding smaller variations in average fluid speeds. Thus, contrary to intuition and to many statements made in the literature, an increase in molecular diffusivity leads to a decrease in longitudinal dispersivity.

Fig. 8 shows the instantaneous fluid interface during intrusion of a fluid into the pore network of a three-dimensional random packing of polydisperse particles. The interface is tracked across the network by a large number of tracer particles injected into the boundary between the two fluids. As the invading fluid fills progressively more of the pore volume, the interface roughens due to the varying mean fluid speeds along the various paths. This roughness of the advancing front is equivalent to a diffusion process in which the two fluids intermix along the plane of the intrusion front. This is not a true diffusion, however, and is referred to instead as dispersion.

A pronounced feature of Fig. 8 is the finger-like structure of the fluid interface. Such structures are widely known to occur in flows in porous media when the viscosity of the invading fluid is lower than that of the fluid initially occupying the pore volume [42,43]. In that case, roughness of the interface results from an inherent





7A

7B

Figure 7. Evolution of pore structure due to uniform accumulation on particles. 7A shows original pore structure, 7B shows 10% accumulation by total volume, and 7C shows 20% accumulation. The network of 7B has a permeability more than an order of magnitude below that of the original. The portion of the network shown in 7C is just below the percolation threshold.





 $\mathbf{24}$ 

Figure 8. Particle-scale fingers in a pore network. Finger pattern results from local variations in permeability due to local variations in the size of interconnected pores. Unlike traditional viscous fingers, the pattern produced in a given pore network is largely deterministic in nature.



instability, and the resulting structures are known as Saffman-Taylor fingers [44]. The fingers in Fig. 8 do not have this origin. Instead, these fingers result only from the statistical nature of the distribution of pore sizes and connectivity of the pore network. For a given network, the formation of these fingers is almost entirely deterministic and depends only on the magnitude of the Peclet number indicating the relative importance of advective and diffusive transport.

Quantitative values of the dispersivity are extracted by analyzing the spatial distribution of the tracer particles used to map the intrusion interface. This is illustrated in Fig. 9. Here the final position of all tracer particles is shown on a slice through the particle array. The jagged solid curve is the cumulative distribution of these positions starting from the right boundary of the domain. The dashed curve is an error function, fit to the cumulative distribution in a least-squares sense by selecting the best values of the mean and standard deviation. The error function is used for this purpose because it is a solution to the continuum equations describing diffusion about the fluid interface. The dispersivity can be computed directly from the mean and variance obtained from this fit by the relation  $D^* = \sigma^2/2t = \bar{u}\sigma^2/2\bar{x}$ , where  $\bar{u}$  is the mean fluid speed,  $\sigma^2$  is the variance, and  $\bar{x}$  is the mean particle position. Note that the very good agreement between the cumulative distribution

Figure 9. Tracer particle final positions. Particles are injected into the left boundary and carried by the mean flow for a specified period. Fitting the cumulative distribution of their final positions (dots and solid curve) using an error function (dashed curve) yields the dispersivity of the network.



of the particles positions and the error function fit indicates that dispersion in the network does indeed mimic a diffusion process.

The last sample problem concerns lateral dispersion. This is an important process in both bioremediation and contaminant transport since it strongly influences the vertical and lateral extent of the plume formed as fluids are transported downstream of a source. As with longitudinal dispersion, lateral dispersion involves a coupling between advective and diffusive transport. To examine this process, we have again computed the moving interface between two fluids during fluid injection. This time, however, the fluid is injected into a single tube on the left boundary of the network, rather than along its entire length. This is illustrated in Fig. 10. As before, a large number of particles are initially placed at the fluid interface. These are then carried into the network by the mean flow.

The results shown in Fig. 10 are for a special case in which there is no diffusion of the tracer particles between streamlines. In this case, lateral spreading of the plume is limited to a few pore diameters above and below the point of injection. The reason for this is that all of the particles are confined to the streamline on which they were injected. For low Reynolds numbers there is relatively little mixing of streamlines, even in highly disordered materials, so there is no mechanism to produce significant lateral spreading of the plume. As with longitudinal dispersion, a quantitative Figure 10. Particle-scale lateral dispersion in a pore network. At high Peclet numbers, lateral dispersion is limited because species do not move readily across streamlines within the pore volume. Results shown are for asymptotically large Peclet number.



Figure 11. Lateral dispersion at low Peclet number. Large coefficient of diffusion allows tracer particles to cross streamlines and follow mean flow into an increasing lateral extent of the network. Results shown are for Pe = 1.



estimate of the lateral dispersivity can be obtained by fitting the distribution of final particle positions within a slab vertical slab, using in this case a Gaussian profile.

In contrast to longitudinal dispersion, lateral dispersion becomes more pronounced when particles are allowed to diffuse between streamlines. This is illustrated in Fig. 11. Here, the intrusion interface is tracked through the same network used for Fig. 10. In this case, however, the coefficient of diffusion is set to a value to give a Peclet number based on the pore diameter of  $Pe = \rho \bar{u} d/D = 1$ . The result is a dramatic increase in the extent of the lateral spread of the plume. The top and bottom boundaries of the plume now grow away from the centerline in proportion to the square-root of the longitudinal distance from the injection point. Note that the larger longitudinal extent of the plume is due to longitudinal dispersion, which is also present here but was absent in the results of Fig. 10.

It is important to recognize that the large lateral extent of the plume in Fig. 11 does not result directly from diffusion. For the diffusion coefficient used to obtain Pe = 1, the maximum lateral extent of the plume would be only about half as large if diffusion were the only mechanism for lateral transport.

#### LABORATORY APPARATUS

The basic experimental approach for investigating deposition of fine suspended solid particles during fluid flow through a porous media was to determine the increases in the local hydraulic gradients within the porous media caused by the deposition of particles. These particles accumulate on the surfaces of the porous material over a considerable depth compared to the dimensions of a typical pore. Thus, pressure gradients are distributed over a relatively large distance within the porous media rather than being localized, e.g., at the surface of a filtering element or plate. The experimental protocol was to measure and record the local pressure distributions within the porous filtration media as a function of time. These data were then used to calculate the hydraulic gradient as a function of position in the packed bed. The experimental variables included the dimensions of the packing material used to create the porous media, the flow rate of the suspension, and the concentration of particulate material. Another objective of this work was to determine the local loading of deposited suspension that caused the observed increases in hydraulic gradients.

A schematic diagram of the experimental apparatus is shown in Fig. 12. An aqueous suspension was pumped from a continuously stirred reservoir to a cylindrical packed column that was operated in the upflow mode. Tubing ports along the axis of the column were connected to a differential pressure gauge to measure the Figure 12. Diagram of experimental apparatus. Suspension is pumped through bottom of column against constant pressure of the reservoir at the top. Pressures are measured along the length of the column.



local hydraulic pressure relative to the prevailing atmospheric pressure in the laboratory. These pressure data were recorded and stored digitally during the course of an experiment. The liquid exited from the column by overflowing from a small tank that served to provide a constant hydrostatic head during the experiment. Photographs of the experimental apparatus are shown in Figs. 13 and 14..

The granular material that filtered the suspension was packed into cylindrical columns fabricated from acrylic tubing. The columns had an inside diameter of 50 mm (2.0 in.) and were approximately 0.51 m long. The packing material rested on a stainless steel wire mesh that was supported by a narrow ring of acrylic plastic held in place by a flange at the bottom (inlet end) of the column. This arrangement made virtually all of the column cross-section available for flow.

The detailed view of a packed column in Fig. 14 shows the arrangement for monitoring pressure during the flow experiments. A series of ports were drilled in the side wall of the column to allow access for measuring pressure. The pressure ports were spaced 10 mm apart near the inlet end of the columns, 20 mm apart in the center section, and 40 mm apart near the outlet (top end). The ports were spaced more closely near the inlet end as pressure gradients were expected to be largest there. The pressure ports consisted of stainless steel syringe tubing with Luer-Lok adapters that were used to connect the ports to the pressure-sensing device. The openings of the syringe tubes were aligned along the axis of the column. A fine piece

29

Figure 13. Photograph of the apparatus used to study flow in porous media, showing the packed bed column, reservoir, pump, data acquisition computer and pressure measuring system.



Figure 14. Photograph of a packed bed column as used for filtration experiments showing arrangement of the pressure monitoring ports. The white material at the bottom of the column is a deposit resulting from deposition of fines from a suspension.



| Packing Material  | Diameter ( $\mu$ m)   | Mesh Size                        |
|---|---|----------------------------------|
| P-170 Spheres*<br>A-055 Spheres*<br>Ottawa Sand**<br>A-150 Spheres* | $\begin{array}{c} 300-450\\ 500-600\\ 600-850\\ 1400-1700\end{array}$ | 40-50<br>30-35<br>20-30<br>12-14 |

\* Potter's Industries, Inc.,  $\rho = 2.45 - 2.50 \text{ gm/cc}$ 

\*\* Fisher Scientific Co.,  $\rho = 2.62$  gm/cc

Table 1. Dimensions of the packing materials used for the filtration experiments.

of stainless steel wire was inserted into each of the syringe tubes to prevent intrusion and clogging by the packing material. As each monitoring port tube was necessarily filled with water to equalize the hydrostatic head before flow was initiated, the time response to changes in pressure were slowed somewhat by the reduction in crosssectional area due to the inserted wire. However, pressure changes typically occurred quite slowly during these experiments, and the configuration used here responded to step pressure changes within 10 to 15 seconds.

Two types of materials were used to pack the filtration beds. The sizes of these packings are summarized in Table 1, where the particle size refers to the mean diameter. The primary packing material consisted of several grades of solid spherical beads of soda-lime glass (Potters Industries, Inc., Valley Forge, PA) that are classified into relatively narrow size ranges. Three sizes of spheres were used to provide a range of particle diameter that varied by a factor of about four, as indicated in Table 1. The second type of material was Ottawa sand (Fisher Scientific Corp., Pittsburgh, PA). It is a naturally-occurring silica sand that has been classified into a narrow size range of 600 to 850  $\mu$ m (diameter), which corresponds to 20-30 mesh fraction. The individual grains of sand are generally ellipsoidal in shape. The complete specifications of Ottawa sand are given by ASTM-C 778-80a.

The columns were packed by gradually adding weighed amounts of a given packing material to a column filled with water to avoid trapping air. As the beads settled, the column was agitated by tapping to ensure a uniform packing distribution. The uniformity of packing was subsequently confirmed by measuring the hydraulic gradient during flow of deionized and filtered water and to verify that the gradient was uniform. A peristaltic pump was used to supply a constant flow rate of suspension during these experiments. The suspension was pumped using a Masterflex Model 7523 peristaltic pump (Cole-Parmer Instrument Co., Vernon Hills, IL). Flow rates up to 270 milliliters per minute (ml/min) could be achieved. The flow rate was determined from the pump rotation speed, rather than in-line measurement, to avoid interference between the suspended particles and a flow sensor. The pumping rate was set by a digital speed control and was reproducible to  $\pm 2\%$ . Flow rates were calibrated by collecting known volumes of liquid during measured time intervals over the entire range of flow rates to be investigated using a variety of tubing sizes compatible with the pump heads.

Pulsations from the peristaltic pump were reduced, but not completely eliminated, by an in-line reservoir "pulse dampener," Cole-Parmer Instrument Co., Vernon Hills, IL) that established a trapped liquid/gas interface between the pump and the inlet to the column. This device required approximately 20 ml of liquid holdup and did not cause any loss of the suspended material in these tests. The amplitude of pulsations from the pump were also reduced quite effectively by using the smallest tubing diameter in the pump head that was capable of providing the desired flow rate. As smaller tubing required greater rotational speeds to achieve a given flow rate, the relatively high frequency of the pulses was found to reduce the peak-to-peak amplitude of pressure variations. The data acquisition system, described below, further reduced the influence of pulsations by using an averaging procedure to collect data. Experiments with clean water confirmed that hydraulic gradients could be measured accurately in the presence of larger pressure pulsations than actually occurred during filtration experiments.

The pressure measurement system was designed to determine relatively small hydraulic gradients and thus precise measurements of local pressures throughout the column were necessary. The measurement technique was based on eliminating the hydrostatic component of pressure, thereby increasing the resolution with which pressure changes due to fluid flow and, subsequently, the effect of particle deposition, could be determined. Measurements were made with thin-film pressure transducers (Omega Engineering Inc., Stamford, CT, Models PX-162 and PX-164) and waterfilled manometers. The differential pressure transducers had ranges from 0 to 5 in. (water) to 0 to 27 in. (water) and provided electrical output signals that could be interfaced with the digital data acquisition system. The liquid manometers were used for baseline measurements of the hydraulic gradients during flow of clean water, rather than flow of suspensions.

Each pressure port was periodically connected to the pressure transducer by a computer-controlled multi-position valve (Valco Instruments, Houston, TX). The valve ports were connected sequentially, starting each cycle with the ports at the upper end of the column, having the lowest differential pressure, and stepping to the inlet port, having the largest differential pressure. This procedure minimized the





offset inherent in the movement of the free surface of liquid between the manometer tubing and the valve/transducer interface. The first position of the valve was open to the atmosphere so that each cycle of pressure readings could re-establish an atmospheric (zero) reference pressure. Pressure readings were corrected for the slight displacement of the water-air interface in the manometer tubing by a calculation feature in the Workbench software. In addition, a time-averaging scheme in this software was used to minimize small fluctuations in pressure caused by the pump prior to capturing the data. The valve cycle was set to wait for one minute at each port, resulting in a total cycle period of 16 minutes.

Data acquisition and recording were performed using the software application, Workbench (Omega Engineering Corp., Stamford, CT), running on an Apple Macintosh SE computer. This software also provided the stepping control of the multiposition valve and the synchronizing signal to identify which location in the column corresponded to the pressure data that were recorded. Fig. 15 shows a typical display of the pressure data captured during several complete sequences of valve operation during a flow experiment. The stepped line indicates the values of pressure at various locations in the column. The axial separation of pressure monitoring ports along the column is not necessarily equal between all of these steps. It is evident that the small pressure pulses created by the pump are very well damped.

Aqueous suspensions of particles of kaolin clay in a buffer solution were prepared for the filtration experiments. The constituents were kaolin powder, (Mallinkrodt

33

Chemical Co., St. Louis, MO, food grade) which is hydrated aluminum silicate having the approximate formula  $H_2Al_2Si_2O_8-X$   $H_2O$ . Aluminum sulfate,  $Al_2(SO4)_3-X$   $H_2O$ , or "alum" (Mallinkrodt Chemical Co., St. Louis, MO, reagent grade) was used as the coagulant for the suspended clay particles. Sodium bicarbonate and potassium chloride were added to buffer the suspension to near a neutral pH and to provide ionic strength. The standard suspension was prepared in deionized water to produce final concentrations of 30 mg/l kaolin, 10 mg/l alum, 50 mg/l KCl and 42 mg/l NaHCO<sub>3</sub> [45]. Typically, concentrated solutions or suspensions of the individual constituents were gradually added to deionized water before an experiment and stirred strongly by mechanical or ultrasonic agitation for several minutes. The suspension in the reservoir was continuously stirred during the flow tests, however, the suspended particles required several hours to settle out when stagnant. Several filtration experiments were conducted using suspensions in which the concentration of the constituents was doubled.

Flow experiments were initiated using clean water, pumped at the same flow rate intended for the suspension. This enabled us to collect data for the hydraulic gradient of the clean packing and to verify that all the data channels were functioning properly. Flow of suspension into the column was started at the beginning of the switching cycle of the multi-position valve. The test was continued, replenishing the suspension reservoir as needed, for up to 9 hours.

#### EXPERIMENTAL RESULTS

Hydraulic gradients were determined using columns loaded with the various packing materials in order to verify that our measurements corresponded to those reported by other workers and to ensure that the experimental system was functioning properly. Flow rates were varied over a wide range to ensure the consistency of the measured permeabilities. The data for hydraulic gradients measured for flow of water in clean columns is presented in Table 2. The hydraulic gradient is given in units of hydrostatic head (inches of water) per unit length of packing (inches) and is nominally dimensionless [48]. The filtration, or Darcy, velocity was calculated based on the volumetric flow rate and the void fraction of a given packed column.

These data are also plotted in Fig. 16, in which the measurements done in this work, represented by solid symbols, are compared with data from the literature, which are shown as open symbols. A direct comparison can be made of our data for a large packing material (1400-1700  $\mu$ m) and the data of Darby, et al [47] using the same glass beads. These data agree very well. Similarly, Hunt's data [45] for a slightly smaller packing (1190-1400  $\mu$ m) display a somewhat greater hydraulic gradient at a given filtration velocity compared to the largest packing, as expected. Extrapolating Hunt's data for a smaller packing (495-589  $\mu$ m) and the data from this work for Ottawa sand (600-850  $\mu$ m) also yields good agreement. Fig. 16 illustrates

| Data Source  | Diameter ( $\mu$ m) | Velocity (cm/s) | Gradient |
|--------------|---------------------|-----------------|----------|
| P-0170 Beads | 300-450             | 0.017           | 0.195    |
| "            | 300-450             | 0.026           | 0.281    |
| >>           | 300-450             | 0.034           | 0.381    |
| >>           | 300-450             | 0.043           | 0.490    |
| Ottawa Sand  | 600-850             | 0.034           | 0.110    |
| "            | 600-850             | 0.051           | 0.161    |
| "            | 600-850             | 0.068           | 0.202    |
| A-150 Beads  | 1400-1700           | 0.034           | 0.026    |
| "            | 1400-1700           | 0.051           | 0.037    |
| "            | 1400-1700           | 0.068           | 0.047    |
| "            | 1400-1700           | 0.085           | 0.057    |
| Darby [47]   | 1400-1700           | 0.180           | 0.1      |
| Darby [46]   | 500-600             | 0.170           | 0.57     |
| Hunt [45]    | 495-589             | 0.145           | 0.38     |
| "            | 495-589             | 0.279           | 0.73     |
| >>           | 1190-1400           | 0.145           | 0.12     |
| "            | 1190-1400           | 0.279           | 0.23     |
| Hunt [45]    | 1190-1400           | 0.554           | 0.46     |
|              |                     |                 |          |

Table 2. Hydraulic gradients measured for flow of water in clean packed columns.

that the hydraulic gradients determined in this work were considerably smaller than those reported in the literature. The experimental apparatus used in our work is much more sensitive to small differential pressure changes during flow than the prior studies.

Several experiments were conducted to verify the ability of the experimental apparatus to measure the increases in local hydraulic gradients that result as fine particles deposited within the column. The parameters varied were the size of the packing material, flowrate, and concentration of the suspended particles. The smallest packing material, P-0170 glass beads (see Table 1), collected the suspended fines very effectively, resulting in relatively large pressure drops between the inlet region and first (1-cm) monitoring port. The standard suspension concentration tended to clog this packing material quite readily, even at flowrates of only 40 ml/min. The suspension was not distributed over a sufficient length of the column to permit meaningful determinations of the hydraulic gradient in this case. Conversely, the largest packing material, A-150 glass beads, did not collect enough suspended




fines during experiments lasting up to 10 hours to permit accurate measurements of the increases in hydraulic gradients. Even at the highest practical flowrate of 200 ml/min using a double concentration of fines did not deposit sufficient particulate for useful measurements.

Flow experiments with A-055 glass beads and Ottawa sand provided good examples of the utility of the experimental apparatus to measure hydraulic gradients during deposition of fines. The plot in Fig. 17 shows the increases in hydraulic gradient resulting during flow of suspended fines in the A-055 material. The units of hydraulic gradient are cm of water (head) per cm of column length in this plot. The ordinate indicates the axial position above the screen that supports the packing material, in centimeters. The filtration velocity was 0.085 cm/sec, which corresponds to a volumetric flowrate of 100 ml/min, and the concentration of the suspension was twice that specified in procedural section above. The dotted line indicates the mean hydraulic gradient for flow of water in a clean packed bed of A-055 glass beads at this filtration velocity. The values of the hydraulic gradient obtained for the upper part of the column (more than about 18 cm from the support screen) are essentially those of a clean column regardless of time. This result is expected as visual inspection confirmed that very few, if any, suspended particles appeared at the exit end of the packed bed. This result is also consistent with published reports that indicate that fines collect preferentially in the inlet section of a packed





bed, under somewhat similar experimental conditions, rather than being distributed throughout the bed [45,46,47].

The data points and associated lines in Fig. 17 indicate that the hydraulic gradient increased as the experiment progressed and material collected preferentially in the entry region of the bed. The hydraulic gradient decreased with distance further from the inlet area, as expected. The anomaly in the data at positions 4 and 5 cm from the inlet could have been caused by clogging of the monitoring ports. A localized nonuniformity in the packing of the column may also have contributed to these anomalous results. Regardless, the expected trends in the data with regard to time and position in the bed were observed. When related to a mass balance of the total flux of fine particles into the column, these data are also consistent with literature reports, e.g. [45], that relatively small amounts of deposited particles can increase local hydraulic gradients by factors of 5 or more compared to the permeability of a clean column.

## SUMMARY

The subsurface transport of nutrients and microorganisms is a key factor in determining the applicability of in situ bioremediation to waste site decontamination. One of the most important and least understood aspects of this transport process is the dramatic reduction in formation permeability resulting from the growth and accumulation of biomass. These processes, which may reduce permeabilities by three to four orders of magnitude, are strongly dependent on microstructural features of the host medium. Such features, including the pore size distribution and the degree of pore connectivity, cannot be explicitly accounted for in traditional site-scale continuum models. Instead, these processes must be investigated on scales comparable to those of the interstitial pore diameters.

To better understand the processes of transport and permeability reduction, we have constructed mathematical and laboratory models applicable to intermediate scales. These scales are large enough to encompass hundreds or thousands of pores, but still small compared to field operations. The mathematical model depicts flow and transport through a network of interconnected passages representing the pore volume of a granular material. This network model is sufficiently general to accommodate a broad range of transport processes in fully disordered materials, yet simple enough to permit derivation of closed-form relationships describing the permeability and effective diffusivity of regular particle arrays.

In contrast to most previous network models that simply interconnect randomlyplaced tubes, our computational domain is generated by packing spheres or cylinders of a specified size distribution. This approach provides a more realistic representation of granular materials, soils, and sedimentary geologic media. Such attention to microstructural detail is especially important in modeling deposition processes, since reductions in permeability are very sensitive to pore geometry. The packing may be regular or may be constructed by one of several methods using statistical means to obtain varying degrees of disorder and varying packing densities. After fixing the size and location of all particles, a Voronoi tessellation is used to define the centerlines and junctions of the channels forming the network. The aperture of each channel varies with axial position along its length in accordance with the geometry of the bounding solid surfaces. To speed numerical solutions, analytical integration along each channel axis is used to define effective apertures used in calculating the pore volume, fluid speed, transit time, and the cross-sectional area for diffusion.

Although the network model is mainly intended for numerical simulation of transport in random packings, it can also be used to derive analytical expressions relating effective transport properties to the pore geometry. Closed-form expressions were derived for the effective permeability and effective diffusivity for transport through a hexagonal array of circular cylinders. These analytical results are in good agreement with published solutions to the multidimensional equations describing Stokes flow and binary diffusion through a unit cell of the medium. Similar agreement has been obtained in applying this simplified version of the network model to more complex two and three-dimensional ordered media. In all cases, the effective transport properties are related to the fundamental geometric parameters used to characterize all permeable materials, including particle size, porosity, tortuosity, and the percolation threshold. In addressing random media and the more complex processes like dispersion and plugging, it is generally necessary to perform numerical simulations using the full capabilities of the model.

An important and powerful feature of this network model is its capability to simulate particle transport. Fictitious tracer particles can be used to compute effective diffusivities and dispersivities of disordered materials. In addition, particles of finite size can be used to simulate the transport and deposition of fines and the effect of such deposition on permeability. To compute dispersivities, particles are injected on the inlet boundary and transported through the network at the local fluid speed. During each time step of this deterministic motion, the tracer particles are also displaced by a random motion, simulating the effects of diffusion between adjacent streamlines within the pore volume. After a specified interval, the spatial distribution of the particle positions is fit with an error function or Gaussian profile to obtain the mean and variance of the particle positions. From these values, both the longitudinal and lateral dispersivity can be computed. In contrast to most previous work in which streamlines are assumed well mixed at each branching and confluence of channels, the present algorithm maps each streamline through these junctions. In the absence of diffusion it is therefore possible to trace each streamline within the network between the inlet and outlet boundaries. This avoids the artificial dispersion that can sometimes arise in network models.

Using this network model, permeability reductions during bioremediation can be simulated by computing a sequence of solutions for a fixed particle array in which particles of the array are coated by a layer of increasing thickness. In sample calculations presented here, we found that a reduction in porosity from 60% to 40% as a result of layer growth led to more than an order of magnitude reduction in permeability. Further reduction in the porosity to 20% led to a network below the percolation threshold. Such a dramatic reduction in permeability is not a direct consequence of narrowing in the larger channels, but rather results from a loss of interconnection within the pore network by the occlusion of many smaller passages.

Other sample calculations of flow in random granular materials revealed several interesting phenomena. In a rectangular domain having an imposed one-dimensional pressure gradient, we found regions in which the local pressure gradient was reversed and the direction of local flow opposed that of the mean pressure gradient. We also saw that the magnitude of the associated lateral pressure gradients were nearly 20% of that in the direction of mean flow. These deviations from the one-dimensional

39

continuum behavior are a consequence of disorder and heterogeneity. Such large variations in the pressure and velocity fields contribute substantially to macro-scale dispersion.

Experimental studies were conducted to investigate permeability reductions by the accumulation of fines. The apparatus developed for this purpose demonstrated the capability to make measurements of small, local pressure variations within a packed column at relatively low flow rates. This apparatus also enabled improved spatial resolution of permeability reductions within a column, as compared with previous systems described in the literature. Experiments with flow of clean water verified that our results agreed well with published data. Similarly, an experimental capability to measure reductions in permeability due to the deposition of fines during the flow of suspensions was demonstrated. In this series of experiments, permeability reductions were measured over a range of conditions in which the flow rate, properties of the fines suspension, and size and size distribution of the granular packing materials were varied.

One important goal in this program was to compare the measured and calculated permeability reductions for both fines deposition and biofilm growth. Out of this comparison, we intended to benchmark the simulations and develop constitutive relations describing permeability reductions. This goal was not achieved. Due to problems in writing SOPs for biologically active agents in the wake of the DOE Tiger Team visit and due to the relocation of personnel and the apparatus to the new IMTL building, the schedule for experimental work slipped significantly. This did not allow time, within the span of the program, to achieve these last goals. We have attempted to complete this portion of the work since that time, but this also has not been successful.

Despite these problems, we have already made use of several of the techniques and capabilities developed here in areas unrelated to bioremediation. The permeability and effective diffusivity of ordered and nearly-ordered carbon fiber arrays have been computed in support of an ARPA-funded program on the rapid densification of carbon-carbon composites. This manufacturing process shares with bioremediation the problem of formation blocking, leading to long processing times and very high cost. The Voronoi tessellation algorithm developed for generating our tube network has also been used to generate a structural skeleton for modeling the solid mechanics of propellant fragmentation in an MOU-funded study of propellant recovery by temperature cycling and for modeling the solid mechanics of foams in a Sandia CRADA with Dow Chemical Corporation. Finally, the network model is now being considered as a platform for modeling the transport and adsorption in micro-porous materials used as gas separation and storage devices.

### ACKNOWLEDGMENT

The authors wish to thank R. S. Larson for his assistance in integrating several of the expressions used in computing effective transport properties. The assistance of B. C. Long in constructing and operating the experimental apparatus is also gratefully acknowledged.

### REFERENCES

1. E. L. Madsen, "Determining In Situ Biodegradation: ES&T Critical Review," *Environ. Sci. Technol.*, 25 (10), 1663-1673-931, 1991.

2. S. Fiorenza, K. L. Duston, and C. H. Ward, "Decision Making: Is Bioremediation a Viable Option," J. Haz. Mat., 28, 171-183, 1991.

3. W. T. Frankenberger, K. D. Emerson, and D. W. Turner, "In Situ Bioremediation of an Underground Diesel Fuel Spill: A Case Study," *Environ. Manag.*, **13** (3), 325-332, 1989.

4. A. M. Thayer, "Bioremediation: Innovative Technology for Cleaning Up Hazardous Waste," *C&EN*, 23-37, August 26, 1991.

5. D. R. Cullimore, *Practical Manual of Groundwater Microbiology*, Lewis Publishers, Boca Raton, 1993.

6. M. E. Miller and M. Alexander, "Kinetics of Bacterial Degradation of Benzylamine in Montmorillonite Suspension," *Environ. Sci. Technol.*, 25, 240-245, 1991.

7. J. M. Thomas and C. H. Ward, "In Situ Biorestoration of Organic Contaminants in the Subsurface," *Environ. Sci. Technol.*, 23 (7), 760-766, 1989.

8. H. H. M. Rijnaarts, A. Bachmann, J. C. Jumelet and A. J. B. Zehnder, "Effect of Desorption and Intraparticle Mass Transfer on the Aerobic Biomineralization of  $\alpha$ -Hexachlorocyclohexane in a Contaminated Calcareous Soil," *Environ. Sci. Technol.*, 24 (9), 1349-1354, 1990.

9. P. N. Cheremisinoff ed., Encyclopedia of Environmental Control Technology: Hazardous Waste Containment and Treatment, Gulf Publishing Co., Houston, 1990.

10. J. R. Mihelcic and R. G. Luthy, "Sorption and Microbial Degradation of Naphthalene in Soil-Water Suspensions under Denitrification Conditions," *Environ. Sci. Technol.*, 25, 169-177, 1991.

11. S. R. Hutchins, G. W. Sewell, D. A. Kovacs and G. A. Smith, "Biodegradation of Aromatic Hydrocarbons by Aquifer Microorganisms Under Denitrifying Conditions," *Environ. Sci. Technol.*, 25, 68-76, 1991. 12. S. W. Taylor and P. R. Jaffe, "Biofilm Growth and the Related Changes in Physical Properties of a Porous Medium: 1. Experimental Investigation," *Water Resour. Res.*, **26** (9), 2153-2159, 1990.

13. S. W. Taylor and P. R. Jaffe, "Biofilm Growth and the Related Changes in Physical Properties of a Porous Medium: 2. Permeability," *Water Resour. Res.*, 26 (9), 2161-2169, 1990.

14. Y. M. Chen, L. M. Abriola, P. J. J. Alvarez, P. J. Anid and T. M. Vogel, "Modeling Transport and Biodegradation of Benzene and Toluene in Sandy Aquifer Material: Comparisons with Experimental Measurements," *Water Resour. Res.*, 28 (7), 1833-1847, 1992.

15. C. Y. Chiang, C. N. Dawson and M. F. Wheeler, "Modeling of In-Situ Biorestoration of Organic Compounds in Groundwater," *Transport in Porous Media*, 6, 667-702, 1991.

16. D. C. Mangold and C. F. Tsang, "A Summary of Subsurface Hydrological and Hydrochemical Models," *Rev. Geophys.*, **29** (1), 51-79, 1991.

17. L. Semprini and P. L. McCarthy, "Comparison Between Model Simulations and Field Results for In Situ Biorestoration of Chlorinated Aliphatics," *Ground Water*, **29** (3), 365-373, 1991.

18. E. O. Frind, "Modeling of Multicomponent Transport with Microbial Transformation in Groundwater: The Fuhrberg Case," *Water Resour. Res.*, **26** (8), 1707-1719, 1990.

19. G. M. Hornberger, A. L. Mills and J. S. Herman, "Bacterial Transport in Porous Media: Evaluation of a Model Using Laboratory Observations," *Water Resour.* Res., 28 (3), 915-938, 1992.

20. J. T. Angley, M. L. Brusseau, W. L. Miller, J. J. Delfino, "Nonequilibrium Sorption and Aerobic Biodegradation of Disolved Alkylbenzenes During Transport in Aquifer Material: Column Experiments," *Environ. Sci. Technol.*, **26** (7), 1404-1410, 1992.

21. M. Sahimi snd A. O. Imdakm, "Hydrodynamics of Particle Motion in Porous Media," *Phys. Rev. Lett.*, 66 (9), 1169-1172, 1991.

22. S. V. Sotirchos and V. N. Burganos, "Analysis of Multicomponent Diffusion on Pore Networks," *AIChE J.*, **34** (7), 1106-1118, 1988.

23. M. M. Tomadakis and S. V. Sotirchos, "Ordinary and Transition Regime Diffusion in Random Fiber Structures," *AIChE J.*, **39** (3), 397-412, 1993.

24. S. L. Bryant, D. W. Mellor and C. A. Cade, "Physically Representative Network Models of Transport in Porous Media," *AIChE J.*, **39**, 387-396, 1993.

25. S. L. Bryant, P. B. King and D. W. Mellor, "Network Model Evalutaion of Permeability and Spatial Correlation in a Real Random Sphere Packing," *Transport in Porous Media*, 11, 53-70, 1993.

26. K. A. Akanni and J. W. Evans, "Effective Transport Coefficients in Heterogeneous Media," *Chem. Eng. Sci.*, 42 (8), 1945-1954, 1987.

27. L. M. Schwartz and J. R. Banavar, "Transport Properties of Disordered Continuum Systems," *Phys. Rev.* B, **39** (16), 11965-11970, 1989.

28. A. Okabe, B. Boots and K. Sugihara, Spatial Tessellations: Concepts and Applications of Voronoi Diagrams, John Wiley & Sons, New York, 1992.

29. S. Bryant and M. Blunt, "Prediction of Relative Permeability in Simple Porous Media," *Phys. Rev.* A, 46 (4), 2004-2011, 1992.

30. R. E. Larson and J. J. L. Higdon, "A Periodic Grain Consolidation Model of Porous Media," *Phys. Fluids* A, 1 (1), 38-46, 1988.

31. J. J. L. Higdon and G. D. Ford, "Permeability of Three Dimensional Models of Fibrous Porous Media," J. Fluid Mech., 308, 341-361, 1996.

32. A. S. Sangani and A. Acrivos, "Slow Flow Past Periodic Arrays of Cylinders With Application to Heat Transfer," Int. J. Multiphase Flow, 8 (3), 193-206, 1982.

33. W. T. Perrins, D. R. McKenzie and R. C. McPhedran, "Transport Properties of Regular Arrays of Cylinders," *Proc. Roy. Soc. Lond.* A, 369, 207-225, 1979.

34. D. C. Hong, H. E. Stanley, and A. Coniglio, "Randon-Walk Approach to the Two-Component Random-Conductor Mixture," *Phys. Rev.* B, **33** (7), 4564-4573, 1986.

35. S. Reyes and K. F. Jensen, "Percolation Concepts in Modelling of Gas-Solid Reactions: 1. Application to Char Gasification in the Kinetic Regime," *Chem. Eng. Sci.*, 41 (2), 333-343, 1986.

36. S. Reyes, E. Iglesia, and K. F. Jensen, "Application of Percolation Theory Concepts to the Analysis of Gas-Solid Reactors," *Solid State Ionics*, **32**, 833-842, 1989.

37. L. M. McDowell-Boyer, "Chemical Mobilization of Micron-Sized Particles in Saturated Porous Media," *Environ. Sci. Technol.*, **26** (3), 586-593, 1992.

38. S. Torquato, "Trapping of Finite-Sized Brownian Particle in Porous Media," J. Chem. Phys., 95 (4), 2838-2841, 1991.

39. R. E. Martin, E. J. Bouwer, and L. M. Hanna, "Application of Clean-Bed Filtration Theory to Bacterial Deposition in Porous Media," *Environ. Sci. Technol.*, 26 (5), 1053-1056, 1992.

40. P. G. Saffman, "A Theory of Dispersion in a Porous Medium," J. Fluid Mech., 6, 321-349, 1959.

41. D. L. Koch and J. F. Brady, "Dispersion in Fixed Beds" J. Fluid Mech. 154, 399-427, 1985.

42. S. Hill, "Channeling in Packed Columns," Chem. Eng. Sci., 1 (6), 247-253, 1952.

43. R. L. Chuoke and P. van Meurs, "The Instability of Slow Immiscible Viscous Liquid-Liquid Displacements in Permeable Media," Pet. Trans., **216**, 188-194, 1958.

44. P. G. Saffman and G. I. Taylor, "The Penetration of a Fluid into a Porous Medium or Hele-Shaw Cell Containing a More Viscous Liquid," *Proc. Roy. Soc. Lond.* A, 245, 312-329, 1958.

45. J. B. Hunt, B-C Hwang and L. M. McDowell-Boyer, "Solids Accumulation during Deep Bed Filtration," *Environ. Sci. Technol.*, 27, 1099-1107, 1993.

46. J. L. Darby and D. F. Lawler, "Ripening in Depth Filtration: Effect of Particle Size on Removal and Head Loss," *Environ. Sci. Technol.*, 24 (7), 1069-1079, 1990.

47. J. L. Darby, D. F. Lawler and T. P. Wilshusen, "Depth Filtration of Particle Size and Ripening," J. Water Pollut. Control Fed., 63, 228-238, 1991.

48. A. B. Cunningham, W. G. Characklis, F. Abedeen and D. Crawford, "Influence of Biofilm Accumulation on Porous Media Hydrodynamics," *Environ. Sci. Technol.*, **25**, 1305-1311, 1991.

# APPENDIX A - SOURCE LISTING OF PROGRAM BIOREM

### BIOREM\$MAIN

| 00001 | с | DIMENSION $X0(500), Y0(500), SI(500)$                                  |
|-------|---|--|
| 00003 |   | DIMENSION $XX(131)$ , $YY(131)$ , $DDX(500)$ , $DDY(500)$              |
| 00004 |   | DIMENSION XS(500), YS(500), IPRM(12,500), LPRM(12,500)                 |
| 00005 |   | DIMENSION RADI (500), DIAM (500), HITE (500), DIAMO (500), HITEO (500) |
| 00006 |   | REAL LBON(12,500), LSID(12,500), AREA(500)                             |
| 00007 |   | DIMENSION XSTD $(12, 500)$ , YSTD $(12, 500)$ , DONE (500)             |
| 00008 |   | DIMENSION PERI $(500)$ , APRX $(500)$ , APRY $(500)$                   |
| 00009 |   | REAL LOADR, LOADL, LOADT, LOADB, LEOUN (200), LOADX, LOADY             |
| 00010 |   | INTEGER FSID(20), JB(200), NSID(500), NORM(4)                          |
| 00011 |   | DIMENSION WORK $(4, 500)$ , IDEC $(500)$ , LEC $(500)$                 |
| 00012 | С |  |
| 00013 |   | INTEGER IDNOD(12,500), NODAR(11,1500), IDTUB(12,500), TUBNOD(4,1500)   |
| 00014 |   | INTEGER KBLOC(1500), NNEC(4), NBID(4,200), IPIKT(3), TBID(4,200)       |
| 00015 |   | REAL XNOD (1500), YNOD (1500), TUBAR (15, 1500), FLO (3), APORE (1500) |
| 00016 |   | REAL XBA1(2), YBA1(2), XBA2(2), YBA2(2), XB(500), YB(500)              |
| 00017 |   | REAL FREQ(500), XYNPAR(4,1500)   |
| 00018 |   | REAL DNEIGH(18,500), DBOX(500)   |
| 00019 |   | REAL DPER (1500), DDIF (1500), DVOL (1500), ARRTIM (1500)              |
| 00020 |   | REAL XAA(3,3), RHS(3), XTU(1500), YTU(1500), RTU(1500), OHS(3)         |
| 00021 |   | REAL XHS(3)  |
| 00022 |   | REAL DETA, DETX, DETY, DETR  |
| 00023 |   | INTEGER INEIGH(18,500), JBLOK(1500)                                    |
| 00024 | С | <i>,</i>   |
| 00025 |   | REAL*8 PRES (1500), PRESO (1500), DPRES (1500), DOTM, PSUM             |
| 00026 |   | DIMENSION DX(1), DY(1)   |
| 00027 |   | DIMENSION $FX(1)$ , $FY(1)$ , $SMAX(1,1)$                              |
| 00028 |   | DIMENSION TEM(1), DTEM(1), SSMX(1), FIDL(1), SIDL(1)                   |
| 00029 |   | DIMENSION XSOLD(1), YSOLD(1), DXOLD(1), DYOLD(1)                       |
| 00030 |   | DIMENSION UX(1),UY(1),UXOLD(1),UYOLD(1)                                |
| 00031 |   | DIMENSION $AA(1,1), BB(1), RES(3), WX(1), WY(1)$                       |
| 00032 |   | INTEGER IDS $(1)$ , IPOP $(1,1)$ , SED, SED $(1,1)$ , IWORK $(1)$      |
| 00033 | _ | DIMENSION VTRACE(1500)   |
| 00034 | С |  |
| 00035 |   | REAL MAS, HGT, KMOD, KBLK  |
| 00036 |   | COMMON XXX   |
| 00037 | ~ | COMMON /BLKQ/ ALOGMU, ALOGSIG, ZZZ                                     |
| 00038 | C |  |
| 00039 | ~ | DATA (NORM(1), $1=1,4$ ) /-1, 1, 1, -1/                                |
| 00040 | C | אחרא (ע דראז)  |
| 00041 | č | 1. NUMBER OF ADJACENT MODER AND OD CONNECTED DUDER                     |
| 00042 | č | 2. NONDER OF ADACENT NODES AND/OR CONNECTED TOBES                      |
| 00043 | č | 2. NODE NUMBER OF ADJ NODE 2   |
| 00044 | č | 4. NODE NUMBER OF ADJ NODE 3   |
| 00046 | č | 5. TUBE NUMBER JOINING NODE 1  |
| 00047 | č | 6: TUBE NUMBER JOINING NODE 2  |
| 00048 | č | 7: TIBE NIMBER JOINING NODE 3  |
| 00049 | č | 8: BOUNDARY NUMBER OF NOD TON  |
| 00050 | ē | 9: PARTICLE NUMBER 1   |
| 00051 | С | 10: PARTICLE NUMBER 2  |
| 00052 | С | 11: PARTICLE NUMBER 3  |
| 00053 | С |  |
| 00054 | С | TUBAR (K, IDT)   |
| 00055 | С | 1: TUBE LENGTH   |
| 00056 | С | 2: TUBE DIAMETER   |
| 00057 | С | 3: X LOC OF TUBE END 1   |

A1

12

1.5<sup>m</sup> - 1117A 147

| 00058 | С      | 4: Y LOC OF TUBE END 1                     |
|-------|--------|--|
| 00059 | Ċ      | 5: X LOC OF TUBE END 2                     |
| 00060 | Ċ      | 6: Y LOC OF TUBE END 2                     |
| 00061 | С      | 7: X SHIFT IN TUBE POSITION                |
| 00062 | С      | 8: Y SHIFT IN TUBE POSITION                |
| 00063 | С      | 9: BOUNDARY NUMBER OF TUBE IDT             |
| 00064 | C 1    | LO: FLUID VELOCITY                         |
| 00065 | C 1    | 1: MASS FLOW RATE                          |
| 00066 | C 1    | L2: TUBE VOLUME                            |
| 00067 | С      |  |
| 00068 | C XI   | VOD (IDN) AND YNOD (IDN)                   |
| 00069 | С      | 1: X AND Y COORDINATES OF NODE IDN         |
| 00070 | С      |  |
| 00071 | C II   | DNOD (L, ID)                               |
| 00072 | С      | 1: NODE NUMBER ASST WITH SIDE L OF SEED ID |
| 00073 | С      |  |
| 00074 | C II   | DTUB (L,ID)                                |
| 00075 | С      | 1: TUBE NUMBER OF SIDE L OF SEED ID        |
| 00076 | С      |  |
| 00077 | с т    | JENOD (K, IDT)                             |
| 00078 | С      | 1: NODE NUMBER AT TUBE END 1               |
| 00079 | С      | 2: NODE NUMBER AT TUBE END 2               |
| 08000 | С      | 3: SEED NUMBER OF 1ST PARTICLE             |
| 00081 | С      | 4: SEED NUMBER OF 2ND PARTICLE             |
| 00082 | С      |  |
| 00083 |        | PI=4.*ATAN(1.)                             |
| 00084 |        | TWOPI = 2. *PI                             |
| 00085 |        | RT3 = SQRT(3.)                             |
| 00086 |        | RT2 = SQRT(2.)                             |
| 00087 |        | ISED = 111311                              |
| 00088 |        | PRINT*,4.*ATAN(1.),4.*ATAN(-1.)            |
| 00089 | С      |  |
| 00090 |        | CALL TK4014 (960,1)                        |
| 00091 |        | CALL PAGE(11.,8.5)                         |
| 00092 |        | CALL BLOWUP (5./4.)                        |
| 00093 |        | CALL NOBRDR                                |
| 00094 |        | CALL YAXANG(0.)                            |
| 00095 |        | CALL HEIGHT(0.10)                          |
| 00096 |        | CALL DUPLX                                 |
| 00097 |        | CALL XREVIK                                |
| 00098 |        | CALL YREVIK                                |
| 00099 |        | CALL GAPWID(.002)                          |
| 00100 |        | CALL THEFRM(.030)                          |
| 00101 |        | CALL GRACE (2.0)                           |
| 00102 | 0      | CALL NOCHER                                |
| 00103 | 1000   |  |
| 00104 | 1000   |  |
| 00105 | ~      | 1 CONF = 0                                 |
| 00107 | C      | $P_{0} = 1480 \text{ TD} - 1 = 500$        |
| 00100 |        | 1400 I = 1,000                             |
| 00100 | 1400   | $\frac{1}{100} \frac{1}{10} = 0$           |
| 00110 | C 1400 |  |
| 00111 | C      | 1/10/1 TDN-1 1500                          |
| 00112 | 1404   | NODAR (1 TDN) $-0$                         |
| 00112 | C 7404 |  |
| 00114 | C      | CMTN=0.                                    |
|       |        |  |

| 00115 |             | GMAX=0.  |
|-------|-------------|--|
| 00116 |             | GAVE=0.  |
| 00117 |             | CMIN=1.  |
| 00118 |             | CMAX=0.  |
| 00119 | С           |  |
| 00120 | 1519        | CONTINUE   |
| 00121 |             | TPROB=1  |
| 00122 |             | TF (TPROBED 4) (THEN)  |
| 00123 |             | RBORF-0 1  |
| 00123 |             | בטרעש-ט.ו<br>מסרמס קסרם קסרם הקמוגוי איזאורמס  |
| 00124 |             | PRINT, INFOLDAR RADIOS, ROCKE  |
| 00125 |             |  |
| 00120 |             | INDIF<br>TMETEL _ A  |
| 00127 |             |  |
| 00120 |             |  |
| 00129 |             | PRIMI", INPOT IMEIR, IFILL, IMEIR, IFILL   |
| 00130 |             | KEAD", IMEIR, IFILL<br>VARV-1  |
| 00132 |             |  |
| 00132 |             |  |
| 00133 |             | PRINT', 'INPUT XMAX, YMAX', XMAX, YMAX   |
| 00134 |             | KEAD^, XMAX, YMAX  |
| 00135 |             | 1DBIN = 4  |
| 00136 |             | PRINT*, INPUT INFLOW BOUNDARY ID., IDBIN   |
| 00137 |             | READ*, IDBIN   |
| 00138 |             | 1DBEV = (1DB1N / 2) * 2  |
| 00139 |             | 1DBOU = MOD (1DB1N+1, 4) + 1   |
| 00140 |             | IPLTS = 0  |
| 00141 |             | 1PLTB = 0  |
| 00142 |             |  |
| 00143 |             | PRINT", INPUT IPINC, IPINS, IPINS', IPINC, IPINS, IPINB  |
| 00144 |             |  |
| 00145 |             | $\frac{1}{1} \frac{1}{1} \frac{1}$ |
| 00140 |             | $\frac{1}{1} \frac{1}{1} \frac{1}{1} = 0$  |
| 00147 |             | $\frac{1}{2} = 0$  |
| 00140 |             | ערתכב שיינים השיינים השיינים ומשיינים אייניים אייניים שיינים אייניים שיינים אייניים שיינים אייניים שיינים אייני<br>מנוידים איינים אייני   |
| 00149 |             | PRIMI", INPOLIPEIP, IPEII, IPEIN, ISHAD', IPEIP, IPEIT, IPEIN, ISHAD   |
| 00150 |             | TDOTA = 0  |
| 00151 |             | IROIA = 0  |
| 00152 |             | PRINT", ROTATE SEEDS $P1/2$ : $U = NO$ I = YES', IROTA   |
| 00153 |             | NEAD", INDIA   |
| 00154 |             | באראשת שוות שוות שניים איינים  |
| 00155 |             | PRINI", INFOI DEMAAU , DEMAAU<br>DEMAX   |
| 00157 |             | NEAD , DEMAKO  |
| 00158 | C           | NDOX-4   |
| 00150 | C           | TNTTTAL TZE SEED   |
| 00155 | $\tilde{c}$ | INTITADIZE SEED  |
| 00160 | C           | דיבי (דאתביתהבו דיתי ה) ידיבודבאז  |
| 00162 | c           |  |
| 00102 | C           |  |
| 00164 |             | READ (-TMETH *) NSEED ISEED (YMAY VMAY DODO CDAC CDACM TOT   |
| 00165 |             | TO 413 T = 1 NISEED  |
| 00166 | 413         | READ(-TMETH *) XS(T) VS(T) HITTE(T) DIAM(T) DIA(T)   |
| 00167 | 110         | READ(-TMETH *)  TMOD   |
| 00168 |             | DO 441 TDN = 1 .TNOD   |
| 00169 |             | READ(-TMETH, *) PRES(TDM)  |
| 00170 | 441         | PRESO(TDN) = PRES(TDN)   |
| 00171 | طر لل م     | DTAMF = DTAM(1)  |
|       |             |  |

BIOREMŞMAIN

T.A. 1.

71

5, 82

32

| 00172 |      | PRIMI*, INPUT EPSU, DIAMF , EPSU, DIAMF   |
|-------|------|---|
| 00173 |      | READ*, EPS0, DIAMF  |
| 00174 | C    |   |
| 00175 | ×117 |   |
| 00175 | 41/  | DEDIA = 0   |
| 00176 |      | TKETO = T   |
| 00177 |      | PRINT*, 'INPUT IREDU, DELDIA', IREDU, DELDIA  |
| 00178 |      | READ*, IREDU, DELDIA  |
| 00179 |      | SWELL = 0.  |
| 00180 |      | PRINT*. 'INPUT SWELLING FRACTION', SWELL  |
| 00101 |      | PEADA CHIEFI.   |
| 00101 |      |   |
| 00182 |      | IUN-U   |
| 00183 |      | $PRINT^*, INPUT OK: U = NO I = IES', IOK$   |
| 00184 |      | READ*, IOK  |
| 00185 |      | IF (IOK .EQ. 0) GO TO 1519  |
| 00186 | С    |   |
| 00187 | •    | ELSETE (IMETH ED. 99) THEN  |
| 00100 |      | DIAME (11111) (120 VAX YMAX FPS() DIAME   |
| 00188 |      | (0) $(0)$ $(1)$ $(0)$ $(1)$   |
| 00189 |      | $10.693 \ 1D = 1, NPAR$   |
| 00190 |      | READ(99,*) XS(ID), YS(ID), HITE(ID), DIAM(ID), RADI(ID)   |
| 00191 | 693  | CONTINUE  |
| 00192 |      |   |
| 00193 |      | FISETE (IMETH .EO. 1) THEN  |
| 00104 |      |   |
| 00194 |      | NOTED 150   |
| 00195 |      | INSEED=100  |
| 00196 |      | PRINT*, 'INPUT NSEED, ISED', NSEED, ISED  |
| 00197 |      | READ*, NSEED, ISED  |
| 00198 |      | ISEED0=ISED   |
| 00199 |      | DSAT=1.   |
| 00200 |      | PRTNER* INPLER DSATI DSAT   |
| 00200 |      |   |
| 00201 |      | $\operatorname{READ}^{*}, \operatorname{DGAI}^{*} \to \operatorname{COP}(1, \operatorname{AUCEED})$   |
| 00202 |      | DSMIN=ABS (DSAT) *0.80/ SQRT(I. "NSEED)   |
| 00203 |      | DREF = DSMIN  |
| 00204 |      | POROIN=.4   |
| 00205 |      | ALOGSIG = 1.E-8   |
| 00206 |      | PRINT*, 'INPUT PORO, SIG', POROIN, ALOGSIG  |
| 00207 |      | PEAD* POROTN ALOGSTG  |
| 00207 |      |   |
| 00208 |      | DEDDIA = 0.   |
| 00209 |      | 1REDU = 1   |
| 00210 |      | PRINT*, 'INPUT IREDU, DELDIA', IREDU, DELDIA  |
| 00211 |      | READ*, IREDU, DELDIA  |
| 00212 |      | SWELL = 0.  |
| 00213 |      | PRINT* 'INPUT SWELLING FRACTION' SWELL  |
| 00214 |      |   |
| 00214 |      |   |
| 00215 |      | TOK=0   |
| 00216 |      | PRINT*, 'INPUT OK: $U = NO$ $I = YES', IOK$   |
| 00217 |      | READ*, IOK  |
| 00218 |      | IF (IOK.EQ.0) GO TO 1519  |
| 00219 | C    | · ~ ·   |
| 00220 | •    | KCEED-0   |
| 00220 |      | CTIMD-0   |
| 00221 |      |   |
| 00222 |      |   |
| 00223 |      | IF (DSAT.EQ.1) NSEED=50*NSEED   |
| 00224 |      | DO 1010 I=1,NSEED   |
| 00225 |      | IDBC(I)=0   |
| 00226 |      | TTRY=0  |
| 00227 |      | PONE(T) = 0   |
| 00227 |      | $\sum_{i=1}^{n} \sum_{j=0}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{i=1}^{n} \sum_{i$ |
| 00228 |      | ALAAMU=ALAAA (DOMLAN/2.)  |

| 00229 |        | ZZZ=RAN(ISED)   |
|-------|--------|---|
| 00230 |        | CALL ERRINV (ALOGR)   |
| 00231 |        | DIAM(I) = 2. * EXP(ALOGR)   |
| 00232 | С      |   |
| 00233 | 1013   | XS(I)=XMAX*RAN(ISED)  |
| 00234 |        | YS(T) = YMAX * RAN(TSED)  |
| 00235 |        | HTWF(T) = DTAM(T)/2 + (1 - 2 + DAM(T))                            |
| 00233 |        | HMINI = MAY (HIMP(I) - DOMINI/FO)                                 |
| 00230 |        | $\frac{11111111}{11111111} = 11111111111111111111111111111111111$ |
| 00237 |        | IF (HITE(1) .LT. U.) THEN   |
| 00238 |        | HITE(1) = MIN (HITE(1), -HMINN)                                   |
| 00239 |        | ELSE  |
| 00240 |        | HITE(I) = MAX (HITE(I), HMINN)                                    |
| 00241 |        | ENDIF   |
| 00242 |        | RADI(I) = SQRT((DIAM(I)/2.)**2 - HITE(I)**2)                      |
| 00243 | С      |   |
| 00244 |        | IF (DSAT.GT.1.E-6) THEN   |
| 00245 |        | $\frac{1}{RPART} = RADT(T)$                                       |
| 00246 |        | DFLPO-RPART-XS(T)   |
| 00240 |        | DEI DO-MAX (DEI DO XC(T), DDADE XA(AX)                            |
| 00247 |        | DELEOUPINA (DELEO, AS $(1) + RPARI - AMAA)$                       |
| 00240 |        | DELBO=MAX (DELBO, RPART-YS(1))                                    |
| 00249 |        | DELBO=MAX (DELBO, YS (1) + RPART-YMAX)                            |
| 00250 |        | IF (DELBO.GT.0.0) THEN  |
| 00251 |        | ITRY=ITRY+1   |
| 00252 |        | IF (ITRY.GT.MAX(100,200*KSEED)) GO TO 1014                        |
| 00253 |        | GO TO 1013  |
| 00254 |        | ENDIF   |
| 00255 |        | EPSR2 = (DSMIN / 100.) **2  |
| 00256 |        | DO 1012 J=1.I-1   |
| 00257 |        | DELX2 = (XS(T) - XS(T)) * * 2                                     |
| 00258 |        | DELY2 = (YS(T) - YS(T)) **2                                       |
| 00259 |        | DFIH2 = (HT rrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrr                   |
| 00255 |        | DET P2 = ((DT3M(T), DT3M(T))) = 2                                 |
| 00200 |        |   |
| 00201 |        | IF ( DELKZ+DELYZ+DELHZ.LT.DELKZ+EPSZ ) THEN                       |
| 00202 |        |   |
| 00263 |        | IF (ITRY.GT.MAX(100,200*KSEED)) GO TO 1014                        |
| 00264 |        | GO TO 1013  |
| 00265 |        | ENDIF   |
| 00266 | 1012   | CONTINUE  |
| 00267 |        | ENDIF   |
| 00268 | С      |   |
| 00269 |        | SUMP=SUMP+3.14*RADI(I)**2   |
| 00270 |        | PORO=1SUMP/(XMAX*YMAX)  |
| 00271 |        | IF (PORO.LT.POROIN) GO TO 1014                                    |
| 00272 |        |   |
| 00273 | 1016   | KSEED=KSEED+1   |
| 00274 | 1010   | CONTINUE  |
| 00275 | 1014   | NSEED_KSEED   |
| 00276 | C      |   |
| 00270 | с<br>т |   |
| 00277 | r      | MCEED = 150   |
| 00270 |        |   |
| 00279 |        | PRIME', INPUT NSEED, ISED', NSEED, ISED                           |
| 00280 |        | READ*, NSEED, ISED  |
| 00281 |        | ISEED0 = ISED   |
| 00282 |        | DSMIN = SQRT(XMAX*YMAX) *0.80/SQRT(1.*NSEED)                      |
| 00283 |        | DREF = DSMIN  |
| 00284 |        | ALOGSIG = 1.E-8   |
| 00285 |        | PRINT*, 'INPUT SIG', ALOGSIG                                      |
|       |        |   |

00286

5 m.m.

-12<sup>10</sup> (\*

READ\*, ALOGSIG

| 00287  |        | DELDIA = 0.  |
|--------|--------|--|
| 00288  |        | IREDU = 1  |
| 00289  |        | PRINT*, 'INPUT' IREDU, DELDIA', IREDU, DELDIA  |
| 00290  |        | READ*, IREDU, DELDIA   |
| 00200  |        | SWELL = 0.   |
| 00201  |        | DRINT * 'INDIT' SWELLING FRACTION', SWELL  |
| 00292  |        | PRINT, INFOT DALIBING INCOLOR, COMMEND   |
| 00293  |        | READ", SWELLS  |
| 00294  |        | 10k=0  |
| 00295  |        | PRINT*, INPUT OK: $U = NO$ I = IES, IOK  |
| 00296  |        | READ*, IOK   |
| 00297  |        | IF (IOK.EQ.0) GO TO 1519   |
| 00298  | С      |  |
| 00299  | -      | KSEED=0  |
| 00200  |        | SIMP=0   |
| 00300  |        | NDCIM-0  |
| 00202  |        | NCEED $-10$ *NCEED   |
| 00302  |        | NSEED = 10  NSEED  |
| 00303  |        | NDIVX = 20 AMAA / DOMIN  |
| 00304  |        | NDIVY = 20 *YMAX /DSM1N  |
| 00305  |        | NDIV = MAX (NDIVX, NDIVY)  |
| 00306  |        | NACROS = XMAX /DSMIN   |
| 00307  |        | $EPS = 1.1 \times MAX / (NDIVX+1.)$  |
| 00308  |        | PRINT*, NDIVX, NDIVY, EPS  |
| 00200  |        | VTOP = 0.  |
| 00303  |        | $r_{0} = 0.1$  |
| 00310  |        | IDDO(T) = 0  |
| 00311  |        | 1DDC(1) = 0  |
| 00312  |        | DONE(1) = 0  |
| 00313  | С      |  |
| 00314  |        | ALOGMU=ALOG (DSMIN/2.)   |
| 00315  |        | ZZZ=RAN(ISED)  |
| 00316  |        | CALL ERRINV (ALOGR)  |
| 00317  |        | DIAM(I) = 2. * EXP(ALOGR)  |
| 00318  | C      |  |
| 00310  | C      | $\mu_{TTTE}(T) = \pi_{TTT}(T) / 2 + (1 - 2 + RAN(TSED))$   |
| 00319  |        | HIE(I) = DIAH(I)/2, (I. 2. $HIHE(I)/2$ )   |
| 00320  |        | RADI(1) = SQRT((DIAM(1)/2.) = MIII(1)/2.)  |
| 00321  |        | BOTF = 2.* RAN(ISED)   |
| 00322  | С      |  |
| 00323  |        | YMIN = YMAX  |
| 00324  |        | RPART = IFILL * RADI(I)  |
| 00325  |        | DO 2022 II = $1, NDIVX$  |
| 00326  |        | XST = (XMAX *II) / (NDIVX+1.)  |
| 00320  |        | DET BO = XST - RPART   |
| 00327  |        | DELEO - MUL (DELEO YMAY-YST-RDART)   |
| 00328  |        | $\frac{DELEO}{DELEO} = MIN (DELEO, XIAA ADI ICARCI)$   |
| 00329  |        | IF (DELEO .LT. EPS) GO TO 2022   |
| 00330  |        | DO 2023 $JJ = 1, NDIVY$  |
| 00331  |        | YST = YMAX * (1 (1.*JJ)/(NDIVY+1.))  |
| 00332  |        | IF (YST .GT. YTOP+DEMAX0*DSMIN) GO TO 2023   |
| 00333  |        | DELBO = YST - RPART  |
| 00334  |        | IF (IFILL .EO. 0) DELBO = YST - RADI(I) *BOTF  |
| 00335  |        | TE (DELBO LT, EPS) THEN  |
| 00333  |        | VMTN - VCT   |
| 00336  |        | $\frac{1}{1} \frac{1}{1} = \frac{1}{1} \frac{1}{1} \frac{1}{1} = \frac{1}{1} \frac{1}{1} \frac{1}{1} \frac{1}{1} = \frac{1}{1} \frac{1}{1} \frac{1}{1} \frac{1}{1} \frac{1}{1} = \frac{1}{1} $ |
| 00337  |        | $XS(1) = XST + 1.E^2  \text{(RADI(1)}$   |
| 00338  |        | YS(I) = YST + I.E-2  RADI(I)   |
| 00339  |        | GO TO 2022   |
| 00340  |        | ENDIF  |
| 00341  |        | JJJJMX = MIN (I-1, 4 * NACROS)   |
| 00342  |        | DO 2012 JJJJ = 1,JJJJMX  |
| BIOREM | \$MAIN |  |
| 000.40 |        | T _ T_1 _ TTTT / 1   |
| 00343  |        | J = 1 - 1 - JJJJ + 1   |
| 00344  |        | $D = \nabla (0) = \nabla (0)$  |

| )4) | 0 - 1 1 | 0000           |
|-----|---------|----------------|
| 344 | DELX2 = | (XST-XS(J))**2 |

| 00345  |       | DELY2 = (YST-YS(J)) **2                                     |
|--------|-------|---|
| 00346  |       | DELH2 = (HITE(I) - HITE(J)) * *2                            |
| 00347  |       | DELR2 = ((DIAM(I)+DIAM(J))/2.)**2                           |
| 00348  |       | DIST = SQRT (DELX2+DELY2+DELH2) - SQRT (DELR2)              |
| 00349  |       | IF (DIST .LT. EPS) THEN                                     |
| 00350  |       | IF (YST .LT. YMIN) THEN                                     |
| 00351  |       | YMIN = YST  |
| 00352  |       | XS(I) = XST + 1.E-2 *RADI(I)                                |
| 00353  |       | YS(I) = YST + 1.E-2 *RADI(I)                                |
| 00354  |       | ENDIF   |
| 00355  |       | GO TO 2022  |
| 00356  |       | ENDIF   |
| 00357  | 2012  | CONTINUE  |
| 00358  | 2023  | CONTINUE  |
| 00359  | 2022  | CONTINUE  |
| 00360  |       | YTOP = YS(I)  |
| 00361  | С     |   |
| 00362  |       | PRINT1901, I, XS(I), YS(I), RADI(I)                         |
| 00363  |       | DELBO = YMAX-YS(I)-RPART                                    |
| 00364  |       | IF (DELBO .LT. EPS) GO TO 2014                              |
| 00365  |       | KSEED = KSEED + 1   |
| 00366  |       | SUMP = SUMP + PI *RADI(I)**2                                |
| 00367  |       | PORO = 1SUMP/(XMAX*YMAX)                                    |
| 00368  | С     |   |
| 00369  | 2010  | CONTINUE  |
| 00370  | 2014  | NSEED = KSEED   |
| 00371  |       | PRINT*, NSEED   |
| 00372  |       | PRINT*, PORO  |
| 00373  |       | READ*, DDXX   |
| 00374  | С     |   |
| 00375  | ]     | ELSEIF (IMETH.EQ.5) THEN                                    |
| 00376  |       | NSEED = 150   |
| 00377  |       | PRINT*, 'INPUT ISED', ISED                                  |
| 00378  |       | READ*, ISED   |
| 00379  |       | ISEEDO = ISED   |
| 00380  |       | PRINT*, 'INPUT FIBER DIAM AND POROISTY', DIAMF, EPSO        |
| 00381  |       | READ*, DIAMF, EPSO  |
| 00382  |       | IRAG = 1  |
| 00383  |       | PRINT*, 'DO YOU WANT A RAGGED BOTTOM: 0 = NO 1 = YES', IRAG |
| 00384  |       | READ*, IRAG   |
| 00385  |       | DSMIN = DIAMF   |
| 00386  |       | DREF = DSMIN  |
| 00387  |       | DELDIA = 0.   |
| 00388  |       | IREDU = 1   |
| 00389  |       | PRINT*, 'INPUT IREDU, DELDIA', IREDU, DELDIA                |
| 00390  |       | READ*, IREDU, DELDIA  |
| 00391  |       | SWELL = 0.  |
| 00392  |       | PRINT*, 'INPUT SWELLING FRACTION', SWELL                    |
| 00393  |       | READ*, SWELL  |
| 00394  |       | IOK=0   |
| 00395  |       | ACON = 1  |
| 00396  |       | PRINT*, 'INPUT RANDOM VARIATION IN SPACING (0 TO 1) ', ACON |
| 00397  |       | READ*, ACON   |
| 00398  |       | PRINT*, 'INPUT OK: $0 = NO$ $1 = YES', IOK$                 |
| 00399  |       | READ*, IOK  |
| BIORFM | ŚMATN |   |
|        |       |   |
| 00400  |       | IF (IOK.EQ.0) GO TO 1519                                    |
| 00401  | C     |   |

00401 C KSEED=0

| 00403     | SUMP=0.  |
|-----------|--|
| 00404     | NBSUM=0  |
| 00405     | NSEED = $2 \times XMAX \times YMAX / (PI/4. \times DIAMF \times 2) \times (1EPSU)$   |
| 00406     | NDIVX = 20 *XMAX /DIAMF  |
| 00407     | NDIVY = 20 *YMAX /DIAMF  |
| 00408     | NDIV = MAX (NDIVX, NDIVY)  |
| 00409     | NACROS = XMAX /DSMIN   |
| 00400     | $T_{\rm T} = 1.1 \times MAX / (NDTVX+1.)$  |
| 00410     | $\frac{100}{100} = 1 - FPS0$   |
| 00411     | POWTN = DIAME (COPT(2 * RT3/PI*RHO))   |
| 00412     | DEMIN = DIAM / SQAI(2. AIS/FI MO)  |
| 00413     | PRIMI <sup>®</sup> , NDIVA, NDIVI, 10D   |
| 00414     | YIOP = 0.  |
| 00415     | DO 5010 I = 1, NSEED   |
| 00416     | IDBC(I) = 0  |
| 00417     | DONE(I) = 0  |
| 00418 C   |  |
| 00419     | DIAM(I) = DIAMF  |
| 00420 C   |  |
| 00420 0   | HTTE(T) = 0  |
| 00421     | PADT(I) = SORT((DIAM(I)/2) **2 - HITE(I) **2)  |
| 00422     | POTE = 2 * PAN(TSED)   |
| 00423     | BOIT = 2. INT(IDED) $BOIT = 2. INT(IDED)$ $BOIT = 2. INT(IDED)$ $CD3C = (1 + 3CONT(1 - BOITE)) + (DSMIN - DIAME)$  |
| 00424     | SPAC = (1. + ACON(1BOIL)) (DOMIN DIAM)   |
| 00425     | SPAC = MAX (0., SPAC)  |
| 00426     | SPACM = SPAC + TOL   |
| 00427     | RANX = (1 BOTF) * SPACM / 100.   |
| 00428     | RANY = (1 2.*RAN(ISED)) *SPACM / 100.  |
| 00429 C   |  |
| 00430     | YMIN = YMAX  |
| 00431     | JMIN1 = 0  |
| 00432     | RPART = IFILL * RADI(I)  |
| 00433     | TO 5022 TT = $1.NDTVX$   |
| 00433     | XST = (XMAX *TT) / (NDTVX+1.)  |
| 00434     | DET PO - Y GT - PDART  |
| 00435     | DELEO - MIN / DELEO YMAY_YCT_BDART)  |
| 00436     | DELEO = MIN (DELEO, MINA-ADI-ICHAR)  |
| 00437     | IF (DELBO .LT. TOL) GO TO 5022   |
| 00438     | DO 5023 JJ = $1, ND1VY$  |
| 00439     | YST1 = YMAX * (1 (1.*JJ) / (ND1VY+1.))   |
| 00440     | NUMY = (YTOP+DSMIN+SPACM) * (NDIVY+1.) / YMAX + 1  |
| 00441     | YST2 = YMAX * (NUMY-JJ+1.) / (NDIVY+1.)  |
| 00442     | YST = MIN (YST1, YST2)   |
| 00443     | DELBO = YST - RPART  |
| 00444     | IF (IFILL .EQ. 0) DELBO = YST - RADI(I) *IRAG *BOTH  |
| 00445     | IF ( DELBO .LT. TOL) THEN  |
| 00446     | JJY = JJ   |
| 00440     | YMTN = YST   |
| 00447     | $y_{C}(T) = y_{CT} + RANY$   |
| 00448     | VC(T) = VCT + DNV  |
| 00449     | 13(1) = 131 + 10101  |
| 00450     | GU 10 5022   |
| 00451     | ENDIF  |
| 00452     | JJJJMX = MIN (1-1, 2 * NACROS)   |
| 00453     | DO 5012 JJJJ = 1, JJJJMX   |
| 00454     | J = I - 1 - JJJJ + 1   |
| 00455     | DELX2 = (XST - XS(J)) * *2   |
| 00456     | DELY2 = (YST-YS(J)) **2  |
| BIOREM\$M | AIN  |
| 00457     | רעזייט - (עדיייט (ד)-עדייט (ד) איי   |
| 00457     | $DELEC = (TITE(T)^{-TITE}(T))^{2}$   |
| 00458     | ערט זייברו (עראינע בייר ער זייברו ער) – פרט זייברו ער דייברו (ער זייברו בייר) – פרט זייברו ער דייברו ער די |
| 00459     | DIST = SQKT (DELIX2+DELIX2+DELIX2) - SQKT (DELIK2)   |
| 00460     | IF (DIST. LT. SPACM) THEN  |

00461 IF (YST .LT. YMIN) THEN 00462 JJY = JJ00463 YMIN = YST 00464 XS(I) = XST + RANX00465 YS(I) = YST + RANY00466 SPAC1 = DIST00467 JMIN1 = J00468 ENDIF 00469 GO TO 5022 00470 ENDIF 00471 5012 CONTINUE 00472 5023 CONTINUE 00473 5022 CONTINUE 00474 С 00475 C--- FIND SECOND CLOSEST PARTICLE BELOW ----00476 С 00477 SMIN = YMAX 00478 JMIN2 = 0DO 5044 J = 1, I-1 00479 00480 IF (J .EQ. JMIN1) GO TO 5044 IF (YS(I)-YS(J) .LT. 3.\*TOL .AND. JMIN1.GT.0) GO TO 5044 00481 00482 DELX2 = (XS(I) - XS(J)) \*\*200483 DELY2 = (YS(I) - YS(J)) \*\*200484 DELH2 = (HITE(I) - HITE(J)) \*\*200485 DELR2 = ((DIAM(I)+DIAM(J))/2.)\*\*200486 DIST = SQRT (DELX2+DELY2+DELH2) - SQRT (DELR2) 00487 IF (DIST .GT. DIAM(I)/3.) GO TO 5044 00488 IF (DIST .LT. SMIN) THEN 00489 SMIN = DIST 00490 JMIN2 = J00491 ENDIF 00492 5044 CONTINUE 00493 SPAC2 = SMIN 00494 С 00495 C--- MOVE FIBER TO EQUILIBRIUM POSITION 00496 С 00497 BCON = TOL / 10.00498 C 00499 IF (JMIN1.EQ.0 .AND. JMIN2.EQ.0) THEN 00500 CC1 = 0.00501 CC2 = 0.00502 AA1 = 1.00503 AA2 = 0.00504 BB1 = 0.00505 BB2 = 1.00506 00507 ELSEIF (JMIN1.EQ.0 .AND. JMIN2.GT.0) THEN 00508 CC1 = (DIAM(I)/2. + DIAM(JMIN2)/2. + SPAC + BCON)\*\*200509 1 -(XS(I)-XS(JMIN2))\*\*2 - (YS(I)-YS(JMIN2))\*\*2CC2 = 0.00510 00511 AA1 = 2. \* (XS(I) - XS(JMIN2))00512 AA2 = YS(I) - YS(JMIN2)00513 BB1 = 2. \* (YS(I) - YS(JMIN2))BIOREM\$MAIN 00514 BB2 = - (XS(I) - XS(JMIN2))00515 С 00516 ELSEIF (JMIN1.GT.0 .AND. JMIN2.EQ.0) THEN 00517 CC1 = (DIAM(I)/2. + DIAM(JMIN1)/2. + SPAC + BCON)\*\*200518 1 - (XS(I)-XS(JMIN1))\*\*2 - (YS(I)-YS(JMIN1))\*\*2

-----

| 00519    |      | CC2 = 0.  |
|----------|------|---|
| 00520    |      | AA1 = 2. * (XS(I) - XS(JMIN1))  |
| 00521    |      | AA2 = YS(T) - YS(TMTN1)   |
| 00521    |      | $122 = 2 \times (V_{C}(T) - V_{C}(T))$  |
| 00522    |      | DDI = 2.  (15(1) - 15(0))   |
| 00523    | _    | BBZ = - (XS(1) - XS(0) - XS(0))   |
| 00524    | С    |   |
| 00525    |      | ELSEIF (JMIN1.GT.0 .AND. JMIN2.GT.0) THEN   |
| 00526    |      | CC1 = (DIAM(I)/2. + DIAM(JMIN1)/2. + SPAC + BCON)**2  |
| 00527    |      | - (XS(I) - XS(J) - XS(J) + *2 - (YS(I) - YS(J) + *2)  |
| 00528    | -    | CC2 = (DTAM(T)/2 + DTAM(JMTN2)/2 + SPAC + BCON)**2  |
| 00520    |      | = (YS(T) - YS(TMTN2)) **2 = (YS(T) - YS(TMTN2)) **2   |
| 00529    | -    | I = (XO(I) - XO(IIIIIZ)) Z (IO(I) IO(OIIIIZ)) Z   |
| 00530    |      | AAI = 2. (XS(I) - XS(OMINI))  |
| 00531    |      | AA2 = 2. * (XS(1)-XS(JMIN2))  |
| 00532    |      | BB1 = 2. * (YS(I) - YS(JMINI))  |
| 00533    |      | BB2 = 2. * (YS(I) - YS(JMIN2))  |
| 00534    |      | ENDIF   |
| 00535    |      | DLTX = (CC1*BB2-CC2*BB1) / (AA1*BB2-AA2*BB1)  |
| 00536    |      | $DI(TY = (CC2 = \Delta 2 \times DI(TX)) / BB2$  |
| 00530    |      | DIT = (2 + m) + 2   |
| 00537    |      | $\mathbf{RLM} = (2.101)^{-1/2}$   |
| 00538    |      | IF (DEIX ~ 2+DEIY ~ 2 .GI. REIM) THEN   |
| 00539    |      | DLTX = 0.   |
| 00540    |      | DLTY = 0.   |
| 00541    |      | ENDIF   |
| 00542    |      | XS(I) = XS(I) + DLTX  |
| 00543    |      | YS(T) = YS(T) + DITY  |
| 00544    | C    |   |
| 00544    | C    | $\mathbf{v}_{\mathbf{C}}(\mathbf{T}) = \mathbf{M} \mathbf{v}_{\mathbf{C}}(\mathbf{T})  \mathbf{m} \mathbf{O} \mathbf{T} \cdot \mathbf{D} \mathbf{N} \mathbf{W}$ |
| 00545    |      | $M_{\text{AD}}(1) = M_{\text{AA}} \left( M_{\text{AD}}(1), 10 \text{ DATATA} \right)$   |
| 00546    |      | XS(1) = MIN (XS(1), XMAX-TOL-RAINA)   |
| 00547    |      | YS(I) = MAX (YS(I), 'IOL+RANY)  |
| 00548    |      | YS(I) = MIN (YS(I), YMAX-TOL-RANY)  |
| 00549    | С    |   |
| 00550    |      | YTOP = MAX (YS(I), YTOP)  |
| 00551    | C    |   |
| 00552    | •    | DET.BO - YMAX-YS(T)-RPART   |
| 00552    |      | TE (DELEO LU UDI) CO UD 5014  |
| 00553    |      | II (IIII = 0.1) CO = 5014   |
| 00554    |      |   |
| 00555    |      | PRINT1903, 1, JMIN1, JMIN2, XS(1), YS(1), RADI(1), SPAC, DDIX, DDI1   |
| 00556    |      | KSEED = KSEED + 1   |
| 00557    |      | SUMP = SUMP + PI * RADI(I) * 2  |
| 00558    |      | PORO = 1SUMP/(XMAX*YMAX)  |
| 00559    | С    |   |
| 00560    | 5010 | CONTINUE  |
| 00561    | 5014 | NSEED - KSEED   |
| 00562    | JOIT |   |
| 00562    |      | PRINT", NOLED   |
| 00563    |      |   |
| 00564    |      | READ*, DDXX   |
| 00565    | С    |   |
| 00566    |      | ELSEIF (IMETH.EQ.3) THEN  |
| 00567    |      | NSEED = 150   |
| 00568    |      | PRINT*, 'INPUT NSEED, ISED', NSEED, ISED  |
| 00569    |      | READ* .NSEED. TSED  |
| 00570    |      | ISEFDO - ISED   |
| 00570    |      |   |
| BIOREM\$ | MAIN |   |
| 00571    |      | DSMTN = SORT(XMAX*YMAX) *0.80/SORT(1.*NSEED)  |
| 00572    |      | DREF = DSMIN  |
| 00572    |      | $\Delta I COSTG = 1 E-8$  |
| 00574    |      | PRINT 'INDEF STG' ALOSIG  |
| 00574    |      | DENDA NI CORTO  |
| 00575    |      | ח אז מדער שריי אראיד איז  |
| 00370    |      | $\mathcal{D}$   |

- ----

| 00577<br>00578<br>00579 | IREDU = 1<br>PRINT*, 'INPUT IREDU, DELDIA', IREDU, DELDIA<br>READ*, IREDU, DELDIA |
|-------------------------|---|
| 00580                   | SWELL = 0.  |
| 00581                   | PRINT*, 'INPUT SWELLING FRACTION', SWELL  |
| 00582                   | READ*, SWELL  |
| 00583                   | IOK=0   |
| 00584                   | PRINT*, 'INPUT OK: $0 = NO$ $1 = YES', IOK$                                       |
| 00585                   | READ*, IOK  |
| 00586                   | IF (IOK.EQ.0) GO TO 1519  |
| 00587 C                 |   |
| 00588                   | KSEED=0   |
| 00589                   | SUMP=0.   |
| 00590                   | NBSUM=U   |
| 00591                   | NSELD = 10  NSELD   |
| 00592                   | NDIVX = 10 "AMAX / DSMIN  |
| 00593                   | NDIVI = 10 "IMAX / DSMIN  |
| 00595                   |   |
| 00596                   | NACROS = $XMAX / DSMTN$   |
| 00597                   | EPS = 1.1 * XMAX / (NDIVX+1.)   |
| 00598                   | PRINT*, NDIVX, NDIVY, EPS   |
| 00599                   | YTOP = 0.   |
| 00600                   | DO $3010 I = 1$ , NSEED   |
| 00601                   | IDBC(I) = 0   |
| 00602                   | DONE(I) = 0   |
| 00603 C                 |   |
| 00604                   | ALOGMU=ALOG (DSMIN/2.)  |
| 00605                   | ZZZ=RAN(ISED)   |
| 00606                   | CALL ERRINV (ALOGR)   |
| 00607                   | DIAM(I) = 2. *EXP(ALOGR)  |
| 00608                   | BOTF = $2.*$ RAN(ISED)  |
| 00609 C                 | MATRI - MANY  |
| 00610                   | $IM_{LN} = IMAA$  |
| 00612                   | XST = (XMAX * TT) / (NDTVX + 1)   |
| 00613                   | DELBO = XST - RPART   |
| 00614                   | DELBO = MIN (DELBO, XMAX-XST-RPART)   |
| 00615                   | IF (DELEO .LT. EPS) GO TO 3022  |
| 00616                   | DO 3021 KK = $1, NDIVZ$   |
| 00617                   | HIT = DIAM(I)/2. *(1 2.*(1.*KK)/(NDIVZ+1.))                                       |
| 00618                   | IF (I.LT.NACROS) HIT = $DIAM(I)/2$ . *(12.*RAN(ISED))                             |
| 00619                   | RAD = SQRT( (DIAM(I)/2.)**2 - HIT**2 )  |
| 00620                   | RPART = IFILL *RAD  |
| 00621                   | DO $3023 \text{ JJ} = 1, \text{NDIVY}$  |
| 00622                   | YSI' = YMAX * (1 (1.*JJ) / (NDIVY+1.))  |
| 00623                   | IF (IST .GT. TIOP+DEMAXU*DSMIN) GO TO 3023<br>DELPO - YOU - DDADU                 |
| 00625                   | TE (TETLL EQ 0) = TO - VOT - VOT - TO + DOTE                                      |
| 00626                   | IF (IFILD .EQ: 0) DELEO = ISI - RAD BOIF $IF (DELEO LT EPS) THEN$                 |
| 00627                   | YMIN = YST  |
| BIOREM\$MAIN            |   |
| 00628                   | XS(I) = XST + 1.E-2 *RAD  |
| 00629                   | YS(I) = YST + 1.E-2 *RAD  |
| 00630                   | HITE(I) = HIT   |
| 00631                   | RADI(I) = RAD   |
| 00632                   | GO TO 3022  |
| 00633                   | ENDIF   |
| 00034                   | JJJJMX = MIN (1-1, 3 * NACROS)  |

| 00635    |      | DO 3012 JJJJ = $1, JJJJMX$   |
|----------|------|--|
| 00636    |      | J = I - 1 - JJJJ + 1   |
| 00637    |      | DELX2 = (XST-XS(J)) **2  |
| 00638    |      | DELY2 = (YST-YS(J)) **2  |
| 000000   |      | DELH2 = (HIT-HITE(J)) **2  |
| 00033    |      | DELR2 = ((DTAM(T) + DTAM(J))/2) **2  |
| 00040    |      | DIST - SORT (DELX2+DELY2+DELH2) - SORT (DELR2)   |
| 00641    |      | TE DICE IN EDC) THEN   |
| 00642    |      | TT (VOT IT VMTN) THEN  |
| 00643    |      | IF (YST .LT. MILN) THEN  |
| 00644    |      | YMIN = YST   |
| 00645    |      | RADI(I) = RAD  |
| 00646    |      | XS(I) = XST + 1.E-2 *RADI(I)   |
| 00647    |      | YS(I) = YST + 1.E-2 *RADI(I)   |
| 00648    |      | HITE(I) = HIT  |
| 00649    |      | ENDIF  |
| 00650    |      | GO TO 3021   |
| 00651    |      | ENDIF  |
| 00652    | 3012 | CONTINUE   |
| 00052    | 3023 | CONTINUE   |
| 00055    | 3023 | CONTENT  |
| 00004    | 2021 | CONTINUE   |
| 00655    | 3022 | MOD = MC(T)  |
| 00656    | ~    | 10P = 15(1)  |
| 00657    | C    | $P_{T}$  |
| 00658    |      | PRINTIPUL, I, XS(I), IS(I), 2. "HILE(I)/DIAM(I), NADI(I)   |
| 00659    |      | DELBO = YMAX - YS(1) - RPART   |
| 00660    |      | IF (DELBO .LT. EPS) GO TO 3014   |
| 00661    |      | KSEED = KSEED + 1  |
| 00662    |      | SUMP = SUMP + PI * RADI(I) * 2   |
| 00663    |      | PORO = 1SUMP/(XMAX*YMAX)   |
| 00664    | С    |  |
| 00665    | 3010 | CONTINUE   |
| 00666    | 3014 | NSEED = KSEED  |
| 00667    |      | PRINT*.NSEED   |
| 00668    |      | PRTNT*, PORO   |
| 000000   |      | BEAD* DDXX   |
| 00000    | C    |  |
| 00070    | C    | FIGHTE (THETHER A) THEN  |
| 00671    |      | $\frac{1}{100} = 1$  |
| 00672    |      | T = T  |
| 00673    |      | $\frac{1}{100} = \frac{1}{100} = \frac{1}$ |
| 00674    |      | READ', IPAR  |
| 00675    |      | EPSO = 0.3   |
| 00676    |      | DIAMF = 0.1  |
| 00677    |      | PRINT*, INPUT FIBER DIAMETER AND POROISTY, DIAMF, EPSO   |
| 00678    |      | READ*, DIAMF, EPS0   |
| 00679    |      | DREF = DIAMF   |
| 00680    |      | DELDIA = 0.  |
| 00681    |      | IREDU = 1  |
| 00682    |      | PRINT*, 'INPUT IREDU, DELDIA', IREDU, DELDIA   |
| 00683    |      | READ*, IREDU, DELDIA   |
| 00684    |      | ACON = 0   |
| 00001    |      |  |
| BIOREM\$ | MAIN |  |
| 00005    |      | אריזאנט חיז איזיארפט אריאנט איינערע אויינערע אוייערע אוייערט אייזערט אוייערטע אוייערט אוייערט אוייערט אוייערט א  |
| 00685    |      | PENDA ACON   |
| 00686    |      | $T_{0} = 10244221$   |
| 00687    |      | $T_{CEM} = T_{CEMENTM}$  |
| 00688    |      | PRINT, INPUT SEED FOR RANDOW DISPLACEMENT , ISED   |
| 00689    |      | READ*, ISED  |
| 00690    | С    |  |
| 00691    |      | RHOS = 1 EPS0  |
| 00692    |      | DXXX = DIAMF *SQRT(PI /2. /RT3 /RHOS)  |

•

| 00693   |      | IF (IPAK .EQ. 2) DXXX = DIAMF *SQRT(PI /2. /RHOS)  |
|---------|------|--|
| 00694   |      | NX = NINT (XMAX /DXXX) + 1   |
| 00695   |      | XMAX = (NX-1) *DXXX  |
| 00696   |      |  |
| 000000  |      | $\frac{D D D A}{D A} = \frac{1}{2} \frac{D D A}{D A} = \frac{1}{2} $   |
| 00697   |      | IF (IPAK .EQ. 2) DIDX = 1.   |
| 00698   |      | FAC = DYDX /2.   |
| 00699   |      | DYYY = FAC * DXXX  |
| 00700   |      | NY = NINT (YMAX /DYYY) + 1   |
| 00701   |      | YMAX - (NY-1) *DYYY  |
| 00702   |      | CDAC = DYYY = DIAME  |
| 00702   |      | SFAC = DAAA - DIAMF  |
| 00703   |      | IF (IPAK .EQ. 2) SPAC = $DXXX /RI2 - DIAMF$  |
| 00704   |      | SPAC0 = SPAC   |
| 00705   |      | DISD = 0.000 *50.  |
| 00706   |      | PRINT*, 'INPUT FACTOR FOR RANDOMIZATION', DISD   |
| 00707   |      | RFAD* DTSD   |
| 00709   |      | TDTTTTT = 0  |
| 00708   |      | IFLIWA = 0   |
| 00709   |      | IF (DISD .GT. U.) THEN   |
| 00710   |      | PRINT*, 'PLOT RANDOM MOTION ? $0 = NO$ $1 = YES', IPLTWK$  |
| 00711   |      | READ*, IPLTWK  |
| 00712   |      | ENDIF  |
| 00713   |      | SWELL = 0.   |
| 00714   |      |  |
| 00715   |      | DENDY CHELI  |
| 00715   |      | READ , SWELL   |
| 00716   |      | TOK=0  |
| 00717   |      | PRINT*, 'INPUT OK: $0 = NO$ $1 = YES', IOK$  |
| 00718   |      | READ*, IOK   |
| 00719   |      | IF (IOK.EO.0) GO TO 1519   |
| 00720   | С    |  |
| 00721   | -    | KSEED - 0  |
| 00722   |      | $EDC = 1 E_3 * CODT (YMAY*VMAY)$   |
| 00722   |      | $DPG = 1.E^{-3} - 5QAT (AMAA^{-1MAA})$   |
| 00723   |      | EPS = MIN (EPS, SPAC/100.)   |
| 00724   |      | ID = 0   |
| 00725   |      | DO 1025 I=1,NY   |
| 00726   |      | JD=1-MOD(I,2)  |
| 00727   |      | DO 1015 J=1.NX-JD  |
| 00728   |      | TD = TD + 1  |
| 00729   |      | T = T = T  |
| 00720   |      | $DET C = CDAC + (0 \in DAM(TOPD))$   |
| 00730   |      | DELS = SPAC $(0.5-RAN(1SED))$  |
| 00731   |      | XS(ID) = (J-1) *DXXX+JD*DXXX/2. + ACON *DELS *COS(THET)  |
| 00732   |      | YS(ID) = DYYY*(I-1) + ACON * DELS * SIN(THET)  |
| 00733   |      | HITE(ID) = 0.  |
| 00734   |      | DIAM(ID) = DIAMF   |
| 00735   |      | RADI(ID) = DTAMF /2  |
| 00736   |      | DBOX(ID) = EPS   |
| 00737   |      | $\frac{1}{1} \frac{1}{1} = \frac{1}{1} $ |
| 00730   |      | WOR((1, D) = XG(D)   |
| 00730   | 0    | WORK(2, ID) = IS(ID)   |
| 00739   | C    |  |
| 00740   | 1015 | CONTINUE   |
| 00741   | 1025 | CONTINUE   |
|         |      |  |
| BIOREMS | MAIN |  |
| 00740   |      |  |
| 00742   |      | KSEED = ID   |
| 00743   |      | NSEED = KSEED  |
| 00744   |      | NPAR = NSEED   |
| 00745   |      | ENDIF  |
| 00746   | С    |  |
| 00747   | C    | ROUGH POROSTITY SPAC FTC   |
| 00749   | č    | NOON TONDETT, BERC, BIC  |
| 00740   | C    |  |
| 00749   |      | IVEAR = INSEED   |
| 00750   |      | VSUM = U   |

| 00751  |        | RBAR = 0.   |
|--------|--------|---|
| 00752  |        | DBAR = 0.   |
| 00753  |        | DO 887 ID = 1, NPAR   |
| 00754  |        | RBAR = RBAR + RADI(ID)  |
| 00755  |        | DBAR = DBAR + DIAM(ID)  |
| 00756  |        | ARID = PI * RADI(ID)**2   |
| 00757  |        | VSIM = VSIM + ABTD  |
| 00757  | 007    | CONTINUE  |
| 00750  | 007    | $\Delta \alpha \alpha \alpha = \alpha \alpha \alpha$              |
| 00759  |        | RBAR = RBAR / NPAR  |
| 00760  |        | DBAR = DBAR / NPAR  |
| 00761  |        | PORO = 1 VSUM / (XMAX*YMAX)                                       |
| 00762  |        | RHOS = 1 PORO   |
| 00763  |        | HLEG = SQRT (PI /2. /RT3 /RHOS) *DBAR                             |
| 00764  |        | SPAC = HLEG - DBAR  |
| 00765  |        | SPAC = MAX (SPAC, DBAR /100.)                                     |
| 00766  |        | DIAMF = DBAR  |
| 00767  |        | DREF = DBAR   |
| 00768  | C      |   |
| 00700  | ĉ      |   |
| 00709  | 0      | KANDOMIZE SEED LOOKIIONS  |
| 00770  | C      |   |
| 00771  |        | $DISD = 0.000$ ^50.   |
| 00772  |        | PRINT*, 'INPUT FACTOR FOR RANDOMIZATION', DISD                    |
| 00773  |        | READ*, DISD   |
| 00774  |        | IPLTWK = 0  |
| 00775  |        | IF (DISD .GT. 0.) THEN  |
| 00776  |        | PRINT*, 'PLOT RANDOM MOTION ? 0 = NO 1 = YES', IPLTWK             |
| 00777  |        | READ*, IPLIWK   |
| 00778  |        | ਤੇ ਸੋਹਿਲਤ   |
| 00779  | 1784   | MBORS = 6   |
| 007700 | 1104   | NDAD = 18   |
| 00700  |        | ANTI NERVICE AND NOODE NEED VE VE VO VO DNETCH INFTERN            |
| 18/00  |        | CALL NERVEB (NDARR, NDORS, NSEED, AS, 15, A0, 10, DNEIGH, INDIGH) |
| 00782  |        | DELS = SPAC / 10.   |
| 00783  |        | DELSO = DELS  |
| 00784  |        | KCYMX = DISD *DIAMF /DELS   |
| 00785  |        | IF (KCYMX .GT. 0) THEN  |
| 00786  |        | IF (IPLTWK .EQ. 1) THEN   |
| 00787  |        | XORIG = 0.  |
| 00788  |        | YORIG = $0$ .   |
| 00789  |        | BATXY = XMAX /YMAX  |
| 00700  |        | TE (RATIXY CT 1 0) THEN   |
| 00790  |        | $\frac{11}{1000000000000000000000000000000000$                    |
| 00791  |        | $\frac{1}{1}$   |
| 00792  |        | TALIS=XALS "0.95775 /RAILI  |
| 00793  |        | ELSE States States  |
| 00794  |        | XAXIS = 7.5 * RATXY   |
| 00795  |        | YAXIS=XAXIS *0.95775 /RATXY                                       |
| 00796  |        | ENDIF   |
| 00797  |        | XSTP = (XMAX - XORIG) /1.   |
| 00798  |        | YSTP = (YMAX - YORIG) /1.   |
| BIOREM | \$MAIN |   |
| 00700  |        | CALL APPRODIXAXIS VAXIS)  |
| 00000  |        | CALL GOAR (YOPTC YOPT YMAY VODIC VOTO VMAY)                       |
| 00000  |        | ONT DAME (AULTO, ADIF, ATMA, IURIG, IDIF, IMAA)                   |
| 00801  |        |   |
| 00802  |        | CALL MARKER (15)  |
| 00803  |        | CALL CURVE(XS, YS, NSEED, -1)                                     |
| 00804  |        | ENDIF   |
| 00805  | С      |   |
| 00806  |        | SLMAX = 0.  |
| 00807  |        | ITRYMX = 100  |
| 80800  |        | DO 1091 KCY = 1, KCYMX  |

| 00809   |            | JMOV = 0  |
|---------|------------|---|
| 00810   |            | DO $1092 I = 1$ , NSEED   |
| 00811   |            | TTRY = 0  |
| 00812   |            |   |
| 00813   | 651        | $\pi H E \pi - 2 + D T + D A M (T C E D)$                           |
| 00011   | 0.51       | $\frac{11151 - 2}{1000}$  |
| 00014   |            | DELX = DELS (COS(THET))   |
| 00812   |            | DELY = DELS *SIN(THET)  |
| 00816   |            | XNEW = XS(I) + DELX   |
| 00817   |            | YNEW = YS(I) + DELY   |
| 00818   |            | TOL = DBOX(I)   |
| 00819   |            | IF (XNEW .LT. TOL) XNEW = 2.*TOL - XNEW                             |
| 00820   |            | IF (XNEW .GT. XMAX-TOL) XNEW = 2.*(XMAX-TOL) - XNEW                 |
| 00821   |            | TE (YNEW $I_{T}$ , $TOI_{t}$ ) YNEW = 2 * $TOI_{t}$ - YNEW          |
| 00822   |            | TE (VNEW CT VMAX-TOL) VNEW - 2 * (VMAX-TOL) - VNEW                  |
| 00823   |            | PO(1003.TT = 1) NBODS (MEM = 2. (MAX 100) = MEM                     |
| 00023   |            | I = I = I = I   |
| 00024   |            | 0 = 11VEIGH(00, 1)  |
| 00825   |            | DIST = SQRT((XNEW-XS(J)) * *2 + (YNEW-YS(J)) * *2)                  |
| 00826   |            | IF (DIST .LT. RADI(I)+RADI(J)) THEN                                 |
| 00827   |            | IF (ITRY .GT. ITRYMX) GO TO 1092                                    |
| 00828   |            | ITRY = ITRY + 1   |
| 00829   |            | GO TO 651   |
| 00830   |            | ENDIF   |
| 00831   | 1093       | CONTINUE  |
| 00832   |            | JMOV = JMOV + 1   |
| 00833   |            | XS(T) = XNFW  |
| 00834   |            | VC(T) = VNEW  |
| 00034   |            | IJ(I) = INCW $O[(I) = ODE ((XO(I) YO(I)) + 2) + (XO(I) YO(I)) + 2)$ |
| 00035   |            | $SL(1) = SQRT ((AS(1) - AU(1))^{2} + (YS(1) - YU(1))^{2})$          |
| 00836   | 1000       | SLMAX = MAX (SL(1), SLMAX)  |
| 00837   | 1092       | CONTINUE  |
| 00838   | С          |   |
| 00839   |            | IF (MOD(KCY-1, MAX(1,KCYMX/500)) .EQ. 0) THEN                       |
| 00840   |            | CALL SPACEF (57, NSEED, XS, YS, XMAX, YMAX, SDEVX, SDEVY)           |
| 00841   |            | WRITE(16,900) KCY*DELS/DIAMF, SDEVX, SDEVY                          |
| 00842   |            | ENDIF   |
| 00843   | С          |   |
| 00844   | _          | TE (SIMAX OF DIAME/3) THEN  |
| 00845   |            | DTSMX = 0   |
| 00846   |            | DISPR - 0   |
| 00040   |            | DISDR = 0   |
| 00047   |            | DISA = 0.   |
| 00848   |            | DISY = 0.   |
| 00849   |            | 101855  1D = 1, NSEED   |
| 00850   |            | DSID = (XS(ID) - WORK(1, ID)) **2 + (YS(ID) - WORK(2, ID)) **2      |
| 00851   |            | DSID = SQRT (DSID)  |
| 00852   |            | DISMX = MAX (DISMX, DSID)   |
| 00853   |            | DISBR = DISBR + DSID  |
| 00854   |            | DISX = MAX (DISX, ABS(XS(ID)-WORK(1, ID)))                          |
| 00855   |            | DISY = MAX (DISY, ABS(YS(ID)-WORK(2, ID)))                          |
| DTODEN  | 1) (7) TNT |   |
| BIOREMS | MATIN      |   |
| 00856   |            | IF (IPLTWK .EO. 1) THEN   |
| 00857   |            | XB(1) = XO(TD)  |
| 00858   |            | XB(2) = XS(1D)  |
| 00859   |            | VB(1) - VO(TD)  |
| 00860   |            | VB(2) = VC(TD)  |
| 00000   |            | D(2) = D(1D)  |
| 00001   |            | CALL CURVE(AB, IB, Z, U)  |
| 00862   | 1055       |   |
| 00863   | T822       | CONTINUE  |
| 00864   |            | DISBR = DISBR /NSEED  |
| 00865   |            | CALL SPACEF (87, NSEED, XS, YS, XMAX, YMAX, SDEVX, SDEVY)           |
| 00866   |            | WRITE(17,900) KCY*DELS/DIAMF, DISMX/DIAMF, DISBR/DIAMF.             |

| 00867   | 1    | SDEVX, SDEVY, DISMX/SPAC, DISBR/SPAC   |
|---------|------|--|
| 00868   | С    |  |
| 00869   |      | SLMAX = 0.   |
| 00870   |      | CALL NERNEB (NDARR, NBORS, NSEED, XS, YS, X0, Y0, DNEIGH, INEIGH)  |
| 00871   | С    |  |
| 00872   | -    | IF (IPLTWK .EO.0) THEN   |
| 00072   |      | CBAR = 0   |
| 00075   |      | NSIM = 0   |
| 00074   |      | $r_{\rm DOI} = 0$  |
| 00075   |      | DSY = MTN (XS(TD) XMAX-XS(TD)) /DTAMF  |
| 00070   |      | DEXX = MIN (XG(ID), MAX IG(ID)) / DIAMF  |
| 00877   |      | $\frac{10}{10} = \frac{10}{10} = 10$   |
| 00878   |      | $\frac{1101}{1} = 1  \text{NEOPS}$   |
| 00879   |      | NCIM = NCIM + 1  |
| 00880   |      | $\frac{1}{1} = \frac{1}{1} $ |
| 00881   | 1101 | CBAR = CDAR + DNEIGH(0, 1D)  |
| 00882   | TIUT | CONTINUE   |
| 00883   |      | CBAR = CBAR / NSUM   |
| 00884   |      | SDEV = 0.  |
| 00885   |      | DO 1102 D = 1, WSEED   |
| 00886   |      | $\frac{DBXX}{DBXX} = MIN (XS(ID), XMAX-XS(ID)) / DIAMP$  |
| 00887   |      | DBXY = MIN (YS(ID), YMAX-YS(ID)) / DLAMF   |
| 00888   |      | IF (DBXX.L1.2.0 .OR. DBXY.L1.2.0) GO 10 1102   |
| 00889   |      | DO 1102 J = 1, NBORS   |
| 00890   |      | SDEV = SDEV + (DNEIGH(J, ID) - CBAR) * 2   |
| 00891   | 1102 | CONTINUE   |
| 00892   |      | SDEV = SQRT (SDEV / (NSUM-1.))   |
| 00893   |      | PRINT901, JMOV, CBAR, SDEV, SDEVX, SDEVY, DISX, DISY   |
| 00894   |      | ENDIF  |
| 00895   |      | ENDIF  |
| 00896   | 1091 | CONTINUE   |
| 00897   | С    |  |
| 00898   |      | IF (IPLTWK .EQ. 1) CALL ENDPL(0)   |
| 00899   |      | ICRAN = 1  |
| 00900   |      | PRINT*, 'MORE RANDOMIZING ? $0 = NO 1 = YES', ICRAN$   |
| 00901   |      | READ*, ICRAN   |
| 00902   |      | IF (ICRAN .EQ. 1) THEN   |
| 00903   |      | CALL SPACEF (87, NSEED, XS, YS, XMAX, YMAX, SDEVX, SDEVY)  |
| 00904   |      | PRINT900, SDEX, SDEVY  |
| 00905   |      | PRINT*, 'INPUT FACTOR FOR RANDOMIZATION', DISD   |
| 00906   |      | READ*, DISD  |
| 00907   |      | PRINT*, 'PLOT RANDOM MOTION ? 0 = NO 1 = YES', IPLIWK  |
| 00908   |      | READ*, IPLTWK  |
| 00909   |      | GO TO 1784   |
| 00910   |      | ENDIF  |
| 00911   |      | ENDIF  |
| 00912   | С    |  |
|         | -    |  |
| BTOREMS | MATN |  |
|         |      |  |
| 00913   | C    | TAKE SUBSET OF PARTICLES   |
| 00914   | č    |  |
| 00915   | •    | TSUB = 0   |
| 00916   |      | PRINT*, 'TAKE SUBSET OF PARTICLES ? 0 = NO 1 = YES', ISUB  |
| 00917   |      | READ*. ISUB  |
| 00918   |      | TE (TSUB ED. 1) THEN   |
| 00919   |      | XONEW = 0.   |
| 00920   |      | VONEW = 0  |
| 00920   |      | XMAXN = XMAX   |
| 00921   |      | VMAXN = YMAX   |
| 00922   |      | PRINT - INPIT XORIG. XMAX. YORIG. YMAX'.   |
| 00924   | -    | XONEW, XMAXN, YONEW, YMAXN   |

READ\*, XONEW, XMAXN, YONEW, YMAXN 00925 XMAX = XMAXN - XONEW 00926 YMAX = YMAXN - YONEW 00927 DO 519 ID = 1, NSEED 00928 XS(ID) = XS(ID) - XONEW 00929 YS(ID) = YS(ID) - YONEW 00930 519 CONTINUE 00931 ENDIF 00932 00933 С C--- DROP PARTICLES OUTSIDE BOX FROM SEED SET ---00934 С 00935 NSUM = 000936 NOUT = 000937 SMEAS = SQRT (XMAX\*YMAX) 00938 EPS = 1.E-3 \* SMEAS00939 DO 1188 ID = 1, NSEED 00940 DBXX = MIN (XS(ID), XMAX-XS(ID)) 00941 DBXY = MIN (YS(ID), YMAX-YS(ID)) 00942 DBZZ = MIN (DBXX, DBXY) 00943 IF (DEXX.LT.0.0 .AND. DEXY.LT.0.0) DEZZ = -SQRT(DEXX\*\*2+DEXY\*\*2) 00944 IF (DBZZ .GT. 0.0) THEN 00945 NSUM = NSUM + 100946 X0(NSUM) = XS(ID) 00947 YO (NSUM) = YS(ID) 00948 DIAMO (NSUM) = DIAM (ID) 00949 HITEO (NSUM) = HITE (ID) 00950 ELSEIF (DBZZ .GT. EPS-RADI(ID)) THEN 00951 С ELSEIF (DBZZ .GT. -0.5 \*RADI(ID)) THEN 00952 NOUT = NOUT + 100953 WORK(1, NOUT) = XS(ID)00954 WORK(2, NOUT) = YS(ID)00955 WORK(3, NOUT) = DIAM(ID) 00956 WORK(4, NOUT) = HITE(ID) 00957 ENDIF 00958 1188 CONTINUE 00959 00960 DO 1189 ID = 1, NSUM XS(ID) = XO(ID)00961 YS(ID) = YO(ID)00962 DIAM(ID) = DIAMO(ID) 00963 HITE(ID) = HITEO(ID)00964 00965 1189 CONTINUE DO 1190 ID = 1, NOUT 00966 XS(ID+NSUM) = WORK(1, ID) 00967 YS(ID+NSUM) = WORK(2, ID) 00968 DIAM(ID+NSUM) = WORK(3, ID)00969 BIOREM\$MAIN HITE (ID+NSUM) = WORK (4, ID) 00970 1190 CONTINUE 00971 NSEED = NSUM + NOUT 00972 NPAR = NSEED00973 00974 С ASSIGN RADII AND SWELL PARTICLES ---C---00975 00976 С DIAMFO = DIAMF 00977 00978 DIAMF = 0.DO 1163 ID = 1, NPAR 00979 DIAMO(ID) = DIAM(ID) 00980 DIAM(ID) = DIAM(ID) \*(1.+SWELL) 00981 RAD2 = MAX (1.E-20\*XMAX, (DIAM(ID)/2.)\*\*2-HITE(ID)\*\*2) 00982

A17

15150010 11 11 157

.....

13

| 00983   |             | RADI(ID) = SQRT (RAD2)   |
|---------|-------------|--|
| 00984   |             | DIAMF = DIAMF + DIAM(ID)   |
| 00985   | 1163        | CONTINUE   |
| 00986   |             | DIAMF = DIAMF /NPAR  |
| 00987   | С           |  |
| 00988   | C           | MONTE CARLO POROSITY   |
| 00989   | č           |  |
| 00990   | <u>4</u> 91 | ראסד = 11554321  |
| 00990   | 171         | NSHOT $= 5000$   |
| 000001  |             | NVOTD = 0  |
| 000002  |             | 10010 = 0  |
| 00993   |             | VVV = VMVV *DVV(TCVD)  |
| 00994   |             | AAA = AMAA "AAN(IOAD)  |
| 00995   |             | III = IMAA "RAN(ISAD)  |
| 00996   |             | DU 495 ID = 1, INSEED  |
| 00997   |             | DELS2 = (XS(1D) - XXX) * 2 + (YS(1D) - YYY) * 2  |
| 00998   |             | 1F (DELS2 .LT. RADI(1D)**2) GO TO 492  |
| 00999   | 493         | CONTINUE   |
| 01000   |             | NVOID = NVOID + 1  |
| 01001   | 492         | CONTINUE   |
| 01002   |             | EPSMC = (1. *NVOID) / NSHOT  |
| 01003   | С           |  |
| 01004   | C           | SUM POROSITY   |
| 01005   | С           |  |
| 01006   |             | SUMV = 0.  |
| 01007   |             | DIAMF = 0.   |
| 01008   |             | DO $677 \text{ ID} = 1$ , NSEED  |
| 01009   |             | DIAMO(ID) = DIAM(ID)   |
| 01010   |             | DIAMF = DIAMF + DIAM(ID)   |
| 01011   |             | DELBX = MIN (XS(ID), XMAX-XS(ID))  |
| 01012   |             | DELBY = MIN (YS(ID), YMAX-YS(ID))  |
| 01013   |             | DELB = MIN (DELBX, DELBY)  |
| 01014   |             | ABSD = MIN (ABS(DELB), RADI(ID))   |
| 01015   |             | VTD = PT * RADT (TD) * 2   |
| 01016   |             | TE (ABSD LE. BADT (TD)) THEN   |
| 01017   |             | $\frac{11}{1000} = \frac{100}{1000} (ABSD (BADT(TD))$  |
| 01018   |             | $\frac{1}{1000} = \frac{1}{1000} + 1$   |
| 01010   |             | HCT = SORT (RADI(ID) **2 = ABSD**2)  |
| 01020   |             | $\frac{1}{10} = \frac{1}{10} + \frac{1}{10} = \frac{1}{10} + \frac{1}{10} = \frac{1}{10} $ |
| 01020   |             |  |
| 01021   |             | ארביע  |
| 01022   |             | $\frac{1}{10} - \frac{1}{10} = \frac{1}{10} $ |
| 01023   |             |  |
| 01024   |             |  |
| 01025   |             | VID = DELV   |
| 01026   |             | ENDIF  |
| BTOREMS | MATN        |  |
|         |             |  |
| 01027   |             | FNDIF  |
| 01028   |             | SIMV = SIMV + VID  |
| 01029   | 677         | CONTINUE   |
| 01030   | 07.         | PORO - 1 - SIMV / YMAX / YMAX  |
| 01031   |             | DIAME - DIAME /NSEED   |
| 01032   |             | DIRINE - DIRA I ARDED  |
| 01032   | C           |  |
| 01033   | C           | משמיוונים סד לא דר אם אחשיים   |
| 01025   | C           | WINTE SEEDS DI LI'S IL KENDESIED   |
| 01035   | C           |  |
| 01020   |             | 1044 TD $= 1$ NEVED  |
| 01037   |             | 101044 1D = 1, NSEED   |
| 01038   |             | IIIP = XMAX - XS(ID)   |
| 01039   |             | VO(TD) = IO(TD)  |
| 01040   |             | 12(TD) = TIME  |

| 01041   | 1044 | CONTINUE   |
|---------|------|--|
| 01042   |      | XTMP = YMAX  |
| 01043   |      | YMAX = XMAX  |
| 01044   |      | XMAX = XTMP  |
| 01045   |      | ENDIF  |
| 01046   | С    |  |
| 01047   | Č    | INTERMEDIATE DUMP  |
| 01048   | č    |  |
| 01049   | Ũ    | WRTTE (99 *) NEAR YMAY YMAY FECO DIAME   |
| 01050   |      | (50, 50, 50, 50, 50, 50, 50, 50, 50, 50,   |
| 01050   |      | $\frac{1}{1000} \frac{1}{1000} = \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{10000} \frac$ |
| 01051   | 602  | WALLE(J), / AS(L), IS(L), MILE(L), DIAM(L), ADI(L)   |
| 01052   | 092  | CONTINUE   |
| 01053   | C    |  |
| 01054   | 0    | INITIALIZE PLOT  |
| 01055   | C    | 10070  |
| 01056   |      | XORIG = 0.   |
| 01057   |      | YORIG = $0.$   |
| 01058   |      | RATXY = XMAX /YMAX   |
| 01059   |      | IF (RATXY.GT.1.0) THEN   |
| 01060   |      | XAXIS = 7.5  |
| 01061   |      | YAXIS=XAXIS *0.95775 /RATXY  |
| 01062   |      | ELSE   |
| 01063   |      | XAXIS = 7.5 *RATXY   |
| 01064   |      | YAXIS=XAXIS *0.95775 /RATXY  |
| 01065   |      | ENDIF  |
| 01066   |      | XSTP=(XMAX-XORIG)/1.   |
| 01067   |      | YSTP=(YMAX-YORIG)/1.   |
| 01068   | С    |  |
| 01069   |      | CALL AREA2D(XAXIS, YAXIS)  |
| 01070   |      | CALL GRAF (XORIG, XSTP, XMAX, YORIG, YSTP, YMAX)   |
| 01071   |      | CALL FRAME   |
| 01072   | С    |  |
| 01073   |      | PRINT*, ' '  |
| 01074   |      | PRINT*, ' '  |
| 01075   |      | PRINT*, ' '  |
| 01076   |      | PRINT*, ' '  |
| 01077   |      | PRINT981, 'NPAR', NPAR   |
| 01078   |      | PRINT980, 'XMAX', XMAX   |
| 01079   |      | PRINT980, 'YMAX', YMAX   |
| 01080   |      | PRINT980, 'EPS0', EPS0   |
| 01081   |      | PRINT980, 'PORO', PORO   |
| 01082   |      | PRINT980, 'EPSM', EPSMC  |
| 01083   |      | PRINT980. 'SPAC'. SPAC   |
|         |      |  |
| BIOREMS | MAIN |  |
| 01004   |      | מתרכת ומתרכו 0.000 מתרכת   |
| 01084   | ~    | PRIMI980, DREF, DREF   |
| 01085   | C    |  |
| 01086   | C    | PLOTT SEEDS  |
| 01087   | C    |  |
| 01088   |      | IF (IPLITS.EQ.1) THEN  |
| 01089   |      | CALL SCLPIC (0.5)  |
| 01030   |      | CALL MARKER (18)   |
| 01091   |      | CALL CURVE (XS, YS, NSEED, -1)   |
| 01092   |      | CALL RESET('MARKER')   |
| 01093   |      | CALL SCLPIC(1.)  |
| 01094   |      | ENDIF  |
| 01095   | С    |  |
| 01096   |      | RJBAR=0.   |
| 01097   |      | ABAR=0.  |
| 01098   |      | NSUM=0   |

01099 NSUMOLD=0 JIMAX=0 01100 01101 JIMIN=200 SUMLB=0. 01102 NBSUM=0 01103 DO 1621 K=1,20 01104 01105 1621 FSID(K)=0 01106 С DO 1100 I=1,NSEED 01107 01108 С 01109 C--- COMPUTE LINE CONSTANTS FOR Ith SEED ---01110 С ரட=0 01111 DEMX=DEMAX0\*1.6/SQRT(1.\*NSEED) 01112 DO 1030 J=1,NSEED 01113 IF (I.EQ.J) GO TO 1030 01114 DELS = SQRT((XS(I) - XS(J)) \* 2 + (YS(I) - YS(J)) \* 2)01115 IF (DELS.GT.DEMX) GO TO 1030 01116 01117 JL = JL + 1XO(JL) = (XS(I) + XS(J)) /2.01118 YO(JL) = (YS(I)+YS(J)) /2.01119 DELX = XS(J) - XS(I)01120 DELY = YS(J) - YS(I)01121 DELY = SIGN (MAX (ABS (DELY), 1.E-8), DELY) 01122 SL(JL) = -DELX /DELY 01123 JB(JL) = J01124 1030 CONTINUE 01125 01126 NLIN = JL01127 С C--- ADD BOUNDARY DOMAIN LINES ---01128 01129 С EPS = 0.01130 01131 RRR = 0.5 + EPSDO 1040 J = 1, NBOX 01132 01133 С THET = 2. \*PI \* (J-1) /NBOX - PI /2. 01134 X000 = (0.5 + RRR \* COS(THET)) \* XMAX01135 Y000 = (0.5 + RRR \*SIN(THET)) \*YMAX 01136 SL00 = -1. /SIGN (MAX (ABS(TAN(THET)), 1.E-10), TAN(THET)) 01137 01138 С C--- REF: PURCEL PAGE 50 01139 01140 С BIOREMSMAIN 01141 BBB = 1.AAA = - SL0001142 01143 CCC = SL00 \* X000 - Y000DELS = ABS(AAA\*XS(I)+BBB\*YS(I)+CCC)/SQRT(AAA\*\*2+BBB\*\*2)01144 01145 IF (DELS.GT.DEMX) GO TO 1040 JL = JL + 101146 01147 JB(JL) = -JXO(JL) = X00001148 YO(JL) = Y00001149 01150 SL(JL) = SL001040 01151 CONTINUE JIMAX = MAX (JL, JIMAX) 01152 01153 JIMIN = MIN (JIMIN, JL) 01154 С C--- LOCATE NEAREST (PARTICLE) BOUNDING LINE ---01155 01156 С

| 01157   |          | DMIN = 1.E8  |
|---------|----------|--|
| 01158   |          | EPS = 1.E-4 *SMEAS                                     |
| 01159   |          | DO 1050 $J = 1$ , NLIN                                 |
| 01160   | C        | ,  |
| 01161   | Č        |  |
| 01101   | 2        | NEW DIOTI  |
| 01162   | C        |  |
| 01163   |          | IF (X0(J).GT.XMAX-EPS .OR. X0(J).LT.EPS) GO TO 1050    |
| 01164   |          | IF (Y0(J).GT.YMAX-EPS .OR. Y0(J).LT.EPS) GO TO 1050    |
| 01165   |          | IF (JB(J) .GT. NSEED-NOUT) GO TO 1050                  |
| 01166   | С        |  |
| 01167   |          | DELS=SORT((XS(T) - XO(T)) * *2 + (YS(T) - YO(T)) * *2) |
| 01168   |          | TE (DELS IN DWIN) = (15(2) 10(0)) 2)                   |
| 01160   |          |  |
| 01170   |          |  |
| 01170   |          |  |
| 01171   |          | ENDIF.   |
| 01172   | 1050     | CONTINUE   |
| 01173   | С        |  |
| 01174   |          | JC=-1  |
| 01175   |          | JN=JMN   |
| 01176   |          | XX(1) = XO(0.TN)                                       |
| 01177   |          | VV(1) - VO(1)  |
| 01170   | C        | 11(1)210(00)   |
| 01170   |          |  |
| 01100   | C        | START WALK AROUND CELL BOUNDARY                        |
| 01180   | C        |  |
| 01181   |          | KSAV=1   |
| 01182   |          | NSYD=0   |
| 01183   |          | MSID=0   |
| 01184   |          | RAVE=0.  |
| 01185   |          | RMIN=1.  |
| 01186   |          | RMAX=0.  |
| 01187   |          | ASIM=0.  |
| 01188   |          | $EPS - 1 E-2 \times CMEDS$                             |
| 01190   |          | $D_{10} = 1.02$ $D_{110} = 1.20$                       |
| 01100   | 0        | DO 1110 R=1,20   |
| 01101   | C        | 70.70  |
| 01191   |          |  |
| 01192   |          | JC=JN  |
| 01193   | С        |  |
| 01194   |          | DMIN=SQRT(2.)  |
| 01195   |          | XVJO=XX (K)  |
| 01196   |          | YVJO=YY (K)  |
| 01197   |          | DO 1120 J=1,JL   |
|         |          |  |
| BIOREMS | MAIN     |  |
|         |          |  |
| 01198   | C        |  |
| 01100   | C        | TE (TED.TC) CO TO 1120                                 |
| 01200   |          | IF (0.EQ.0C) GO IO II20<br>IF (I FO TO) CO TO 1120     |
| 01200   | <u>^</u> | IF (0.EQ.00) GO TO II20                                |
| 01201   | C        |  |
| 01202   |          | DSLO = SL(JC) - SL(J)                                  |
| 01203   |          | DSLO = SIGN (MAX(ABS(DSLO), 1.E-6), DSLO)              |
| 01204   |          | XVJ = ( (Y0(J) - SL(J) * X0(J))                        |
| 01205   |          | 1 - (Y0(JC) - SL(JC) * X0(JC)) ) / DSLO                |
| 01206   |          | YVJ=Y0(JC)+SL(JC)*(XVJ-X0(JC))                         |
| 01207   |          | IF $(ABS(SL(J)).LT.ABS(SL(JC)))$ THEN                  |
| 01208   |          | YVJ=Y0(J)+SL(J)*(XVJ-X0(J))                            |
| 01209   |          | ENDIF  |
| 01210   | С        |  |
| 01211   | C        | NEW STILLE   |
| 01212   | c        |  |
| 01213   | C        |  |
| 01213   |          | TE (VUI OT VMAY EDG OF VUI IT EDG) OF 1120             |
| 01214   |          | 11 (100.01.112ATERS .OK. 100.01EPS) GO 10 1120         |

| 01215     | C    |   |
|-----------|------|---|
| 01216     | C    | CHECK FOR COUNTER-CLOCKWISE PATH                                  |
| 01217     | С    | NOTE: PUT THIS FIX IN BONZO                                       |
| 01218     | С    |   |
| 01219     |      | IF (JC .LE. NLIN) THEN  |
| 01220     |      | DIREC1 = (XO(JC) - XS(I)) * (YVJ - YVJO)                          |
| 01221     |      | DIREC2 = (YS(I) - YO(JC)) * (XVJ - XVJO)                          |
| 01222     |      | FLSE  |
| 01222     |      | DIREC1 = XVI - XVIO   |
| 01223     |      | $\frac{1}{100} = \frac{1}{100} = \frac{1}{100}$                   |
| 01224     |      | TE (TO) TE 2) TE 2)   |
| 01225     |      |   |
| 01226     |      | DIRECT = - DIRECT   |
| 01227     |      | DIREC2 = - DIREC2   |
| 01228     |      | ENDIF   |
| 01229     |      | ENDIF   |
| 01230     |      | DIREC = DIREC1  |
| 01231     |      | IF $(ABS(DIREC2) .GT. ABS(DIREC1))$ DIREC = DIREC2                |
| 01232     |      | IF (DIREC. LT. 0.0) GO TO 1120                                    |
| 01233     | C    |   |
| 01233     | C    | DW = SORT((XW TO - XW T) * 2 + (YV TO - YV T) * 2)                |
| 01234     |      |   |
| 01235     |      |   |
| 01230     |      |   |
| 01237     |      |   |
| 01238     |      | YY(K+1)=YVJ   |
| 01239     |      | XSID(K, 1) = XVJ  |
| 01240     |      | YSID(K,I)=YVJ   |
| 01241     |      | DMIN=DVV  |
| 01242     |      | DSXMIN=DSX  |
| 01243     |      | DVXMIN=DVX  |
| 01244     |      | DSVMIN=DSV  |
| 01245     |      | DSLOMTN=DSLO  |
| 01245     |      |   |
| 01240     | 1120 |   |
| 01247     | 1120 | CONTINOE<br>KONI KONI 1   |
| 01248     | _    | KSAV=KSAV+1   |
| 01249     | С    |   |
| 01250     | C    | COMPUTE BOND AND SIDE LENGTH                                      |
| 01251     | С    |   |
| 01252     |      | IF (K.GT.1) THEN  |
| 01253     |      | LSID(K-1,I)=SQRT((XX(K+1)-XX(K))**2+(YY(K+1)-YY(K))**2)           |
| 01254     |      | NBSUM=NBSUM+1   |
|           |      |   |
| BIOREM\$  | MAIN |   |
| 01255     |      | IF (JC. GP. NLIN) THEN  |
| 01256     |      | LBON(K-1, T) = 1, E10   |
| 01250     |      |   |
| 01257     |      |   |
| 01258     |      |   |
| 01259     |      |   |
| 01260     |      | $LBON(K-1, 1) = 5QKT((XS(1) - XS(L))^{2} + (YS(1) - YS(L))^{2})$  |
| 01261     |      | SUMLB=SUMLB+LBON(K-1,1)   |
| 01262     |      | ENDIF   |
| 01263     |      | ENDIF   |
| 01264     | С    |   |
| 01265     | С    | COMPUTE AREA OF TRIANGULAR SEGMENT                                |
| 01266     | С    |   |
| 01267     |      | IF (K.GT.1) THEN  |
| 01268     |      | DELH=SQRT((XS(I)-XO(JC))**2+(YS(I)-YO(JC))**2)                    |
| 01269     |      | DELB=SORT ( $(XX(K+1) - XX(K)) * * 2 + (YY(K+1) - YY(K)) * * 2$ ) |
| 01270     |      | TF (JC.GT.NLIN) THEN  |
| 01271     |      | BBB=1.  |
| 01272     |      | AAA=-SL(JC)   |
| U I G I G |      |   |

.

| 01273   |      | CCC=SL(JC) *X0(JC) -Y0(JC)   |
|---------|------|--|
| 01274   |      | DELH=ABS $(AAA*XS(I)+BBB*YS(I)+CCC)/SQRT(AAA**2+BBB**2)$   |
| 01275   |      | ENDIF  |
| 01276   |      | ASUM=ASUM+0.5*DELH*DELB  |
| 01277   |      | ENDIF  |
| 01278   | С    |  |
| 01279   | •    | TE (K.CT.3, AND, JC.ED.JMN) CO TO 1130   |
| 01280   | C    |  |
| 01200   | C    |  |
| 01201   | C .  | CONFOLE CONTESCENDENCE AND THAS BOONDANT SEEDS   |
| 01202   | C    | דוד (דאז ראד דאז) תוודאז   |
| 01203   |      | IF (JN.GI.NLIN) THEN   |
| 01284   |      | IPRM(K, I) = JB(JN)  |
| 01285   |      | IF (IDBC(1).EQ.U) THEN   |
| 01286   |      | 1DBC(1) = 1PRM(K, 1)   |
| 01287   |      | LBC(I) = K   |
| 01288   |      | ELSEIF (IDBC(I).EQ1 .OR. IDBC(I).EQ3) THEN   |
| 01289   |      | IDBC(I) = IPRM(K, I)   |
| 01290   |      | LBC(I)=K   |
| 01291   |      | ENDIF  |
| 01292   | С    |  |
| 01293   |      | ELSE   |
| 01294   |      |  |
| 01295   |      | TPRM(K, T) = TDP   |
| 01296   |      | TF (TOP I T T) THEM  |
| 01290   |      | $\frac{107.01.1}{100.01}$  |
| 01297   |      | $\frac{1}{1744} = \frac{1}{1700} = \frac{1}{100} = $ |
| 01298   |      | IF (IFRM(L, IDF).EQ.I) THEN  |
| 01299   |      | LPRM(K, 1) = L   |
| 01300   |      | LPRM(L, IDP) = K   |
| 01301   |      | ENDIF'   |
| 01302   | 1744 | CONTINUE   |
| 01303   |      | ENDIF  |
| 01304   |      | ENDIF  |
| 01305   | С    |  |
| 01306   |      | IF (JN.LE.NLIN) THEN   |
| 01307   |      | XSJ=XS(JB(JN))   |
| 01308   |      | YSJ=YS(JB(JN))   |
| 01309   |      | DDSJ=SORT((XS(I)-XSJ)**2+(YS(I)-YSJ)**2)   |
| 01310   |      | RAVE=RAVE+DDSJ   |
| 01311   |      | RMTN=MTN(RMTN, DDST)   |
|         |      |  |
| BIOREMS | MAIN |  |
|         |      |  |
| 01312   |      | RMAX=MAX (RMAX, DDSJ)  |
| 01313   |      | (MTN=MTN (CMTN, DDSJ)  |
| 01314   |      | (MAX=MAX ((MAX, DDST)  |
| 01315   |      | MSTD-MSTD+1  |
| 01316   |      |  |
| 01317   | C    |  |
| 01210   | L.   | N(%)77-N(%)77-1  |
| 01310   | ~    | NSID=NSID+1  |
| 01319   |      |  |
| 01320   | C    | PLOTI. BONDS   |
| 01321   | C    |  |
| 01322   |      | IF (IPLIB.EQ.I .AND. JN.LE.NLIN) THEN  |
| 01323   |      | XB(1) = XS(1)  |
| 01324   |      | YB(1) = YS(1)  |
| 01325   |      | XB(2) = XS(JB(JN))   |
| 01326   |      | YB(2) = YS(JB(JN))   |
| 01327   |      | CALL CURVE(XB,YB,2,1)  |
| 01328   |      | ENDIF  |
| 01329   | С    |  |
| 01330   | 1110 | CONTINUE   |

| 01331    | 1130 | CONTINUE                                       |
|----------|------|--|
| 01332    |      | RAVE=RAVE/MAX(MSID,1)                          |
| 01333    | С    |  |
| 01334    | C    | COMPUTE SIDE AND AREA DIAGNOSTICS              |
| 01335    | С    |  |
| 01336    |      | NSUM=NSUM+1                                    |
| 01337    |      | RJBAR=RJBAR+NSYD                               |
| 01338    |      | FSID(NSYD)=FSID(NSYD)+1                        |
| 01339    |      | GAVE=GAVE+RAVE                                 |
| 01340    |      | GMIN=GMIN+RMIN                                 |
| 01341    |      | GMAX=GMAX+RMAX                                 |
| 01342    |      | ABAR=ABAR+ASUM                                 |
| 01343    | С    |  |
| 01344    | C    | PLOTT CELL BOUNDARIES                          |
| 01345    | С    |  |
| 01346    |      | IF (IPLTC.EQ.1) THEN                           |
| 01347    |      | IF $(YS(I) . LT. 0.5)$ THEN                    |
| 01348    |      | CALL CURVE(XX, YY, KSAV, U)                    |
| 01349    |      | CALL RESET ('DASH')                            |
| 01350    |      | ENDIF  |
| 01351    | ~    | ENDIF  |
| 01352    | С    |  |
| 01353    |      | AREA(I)=ASOM                                   |
| 01354    | ~    | INSID(1)=NSID                                  |
| 01355    | 1100 |  |
| 01257    | 1100 | CONTINOE                                       |
| 01357    | C    |  |
| 01250    | C    | FIAC POINDARY MODES                            |
| 01360    | ĉ    | FIAS BOONDART NODES                            |
| 01361    | C    |  |
| 01362    |      | D = 1400 TD=1. NSEED                           |
| 01363    |      | DO 1400  L=1 NSID(TD)                          |
| 01364    |      | TF (TONOD(I, TD), FO, 0)  THEN                 |
| 01365    | C    |  |
| 01366    | C    | TDN = TDN + 1                                  |
| 01367    |      | XNOD(IDN) = XSID(L, ID)                        |
| 01368    |      | YNOD(IDN) = YSID(L, ID)                        |
|          |      |  |
| BIOREMŞI | MAIN |  |
| 01260    |      | TON(T) - TON                                   |
| 01370    |      | IDR - IPRM(I, ID)                              |
| 01370    |      | TF (TDP CT 0) THEN                             |
| 01372    |      | I.P - I.PRM(I.TD)                              |
| 01372    |      | I.PP1 = MOD(I.P.NSTD(TDP)) + 1                 |
| 01374    |      | TDNOD(I.PP1, TDP) = TDN                        |
| 01375    |      | FNDTF  |
| 01376    |      | $I_{M1} = MOD(I_{+}NSID(ID) - 2.NSID(ID)) + 1$ |
| 01377    |      | TDP = IPRM(IM1, ID)                            |
| 01378    |      | IF (IDP.GT.0) THEN                             |
| 01379    |      | LP = LPRM(LM1, ID)                             |
| 01380    |      | IDNOD(LP, IDP) = IDN                           |
| 01381    |      | ENDIF  |
| 01382    | С    |  |
| 01383    |      | NODAR(8, IDN) = 0                              |
| 01384    |      | IDP = IPRM(L, ID)                              |
| 01385    |      | IDPM1 = IPRM(LM1, ID)                          |
| 01386    |      | IF (IDP.LT.0) NODAR(8, IDN) = IDP              |
| 01387    |      | IF (IDPM1.LT.0) NODAR(8, IDN) = IDPM1          |
| 01388    |      | IF (IDP.EQIDBIN .OR. IDPM1.EQIDBIN )           |

| 01389    | 1 NODAR(8, IDN) = $-IDBIN$   |
|----------|--|
| 01390    | IF (IDP.EQIDBOU .OR. IDPM1.EQIDBOU )   |
| 01391    | 1 $NODAR(8, IDN) = -IDBOU$   |
| 01392    | с  |
| 01393    | FNOTE  |
| 01304    |  |
| 01394    |  |
| 01395    | NWOD = 1LM   |
| 01396    |  |
| 01397    | C NUMBER TUBES, IDENTIFY TUBE NODES, LENGTHS AND DIAMETERS   |
| 01398    | C ASSIGN TUBE ENDPOINTS AND XY SHIFTS  |
| 01399    | C  |
| 01400    | 1610  IDT = 0  |
| 01401    | DO 1402 ID=1,NSEED   |
| 01402    | DO 1402 L=1,NSID(ID)   |
| 01403    | IDP = IPRM(L, ID)  |
| 01404    | IF (IDP.GT.ID .OR. IDP.LT.0) THEN  |
| 01405    | С  |
| 01406    | IDT = IDT + 1  |
| 01407    | $TUENOD(1, TDT) = TDNOD(T_1, TD)$  |
| 01408    | TUBNOD(3, TDT) = TD  |
| 01400    | $\frac{100}{1000} \frac{100}{4} = 100$   |
| 01410    | I D = MOD(I NGTD(TD)) + 1  |
| 01410    |  |
| 01411    | $\frac{1000002,101}{0} = \frac{100000011,10}{0}$   |
| 01412    | $TUDAR(1, IDI) = MAR (LSID(L, ID), I.E-0^{AMAR})$  |
| 01413    | $TOBAR(9, IDP) = 1.^{IDP}$   |
| 01414    | 1DTOB(L, 1D) = 1DT   |
| 01415    | IF (IDP.GT.0) THEN   |
| 01416    | LP = LPRM(L, LD)   |
| 01417    | IDTUB(LP, IDP) = IDT   |
| 01418    | DEL=SQRT((XS(ID)-XS(IDP))**2 + (YS(ID)-YS(IDP))**2   |
| 01419    | IF (IREDU.EQ.1) THEN   |
| 01420    | TUBAR(2, IDT) = DEL - RADI(ID) - RADI(IDP) - DELDIA  |
| 01421    | ELSEIF (IREDU.EQ.2) THEN   |
| 01422    | TUBDI = DEL - RADI(ID) - RADI(IDP)   |
| 01423    | TUBAR(2, IDT) = TUBDI * (1 DELDIA)   |
| 01424    | ELSEIF (IREDU.EQ.3) THEN   |
| 01425    | . TUBDI = DEL - RADI(ID) - RADI(IDP)   |
|          |  |
| BIOREMŞI | MAIN   |
| 01426    | TUBAR(2,IDT) = TUBDI /(1. + DELDIA*TUBDI)  |
| 01427    | ENDIF  |
| 01428    | ELSE   |
| 01429    | IF (IDP .EQ. $-1$ ) DEL = YS(ID)   |
| 01430    | IF (IDP .EO. $-2$ ) DEL = XMAX - XS(ID)  |
| 01431    | IF (IDP .EO, $-3$ ) DEL = YMAX - YS(ID)  |
| 01432    | IF (IDP, EO, $-4$ ) DEL = XS(ID)   |
| 01433    | IF (IREDU. EO.1) THEN  |
| 01434    | TUBAR(2, TDP) = DEL - RADT(TD) - DELDTA/2  |
| 01435    | ELSELF (TREDULED 2) THEN   |
| 01436    | $\frac{1}{1} \frac{1}{1} \frac{1}$   |
| 01437    | $\frac{1}{1000} = \frac{1}{1000} = \frac{1}{1000} + 1$   |
| 01438    | FISEIF (TRENIER) 3) THEN   |
| 01430    | $\pi\pi\pi\pi\pi^{-} = \pi\pi^{-} = \pi\pi^{-} \pi^{-}$  |
| 01440    | עדר זיסריאדרפרזיין / (1 איזר זיסראדרפרזיי) איזר דעראי דעראי דערטין (1 איזר זיסראדרפרזיי)   |
| 01440    | TODAT(2, LUI) - TODAT / (I. + TODAT DELUIA)  |
| 01441    |  |
| 01442    | EXPLC (TDT) = 1  |
| 01443    | $\mathbf{X} = \mathbf{X} = $ |
| 01444    | $\frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{10000} \frac{1}{10000} \frac{1}{10000000000000000000000000000000000$   |
| 01445    | NBLOC(1DT) = 0   |
| 01446    | TUBAR(2, IDI) = 1.E-8 * XMAX   |

)

| С    |  |
|------|--|
|      | IF (IDP.GT.0) THEN   |
|      | DEL=SQRT((XS(ID)-XS(IDP))**2 + (YS(ID)-YS(IDP))**2)  |
|      | DELTA = (RADI(ID) - RADI(IDP))/2.  |
|      | DXDL=(XS(IDP)-XS(ID))/DEL  |
|      | DYDL = (YS(TDP) - YS(TD)) / DEL  |
|      | DET TAY = DET TA *DYDI.  |
|      |  |
|      | DEDIAI = DEDIA "DIDD   |
|      | ELSE   |
|      | DELTAX = 0.  |
|      | DELTAY = 0.  |
|      | IF (IDP .EQ. $-1$ ) DELTAY = TUBAR(2, IDT) /2.   |
|      | IF (IDP .EO. $-2$ ) DELTAX = $-$ TUBAR(2, IDT) /2.   |
|      | TF (TDP, FO, $-3$ ) DELTAY = $-$ TUBAR(2, IDT) /2.   |
|      | TE (TDP EQ -4) DELEVAX = $(\text{TIBAR}(2, \text{TDP}))/2$ .   |
|      | II (IDI : LQ: 4) DEDING = IOME((2)(DI) / 2).   |
|      |  |
|      | TOBAR(3, IDT) = XSID(L, ID) + DEDITAX  |
|      | TUBAR(4, IDT) = YSID(L, ID) + DELTAY   |
|      | TUBAR(5, IDT) = XSID(L+1, ID) + DELTAX   |
|      | TUBAR(6, IDT) = YSID(L+1, ID) + DELTAY   |
|      | TUBAR(7, IDT) = DELTAX   |
|      | TUBAR(8, IDT) = DELTAY   |
| C    |  |
| C    | ENITE:   |
| 1400 |  |
| 1402 | CONTINUE   |
|      | NIUB = IDT   |
| С    |  |
| C    | IDENTIFY NODE TRIOS AND ASSOCIATED TUBE NUMBERS  |
| С    | SEQUENCE OF TRIOS IS IN ANTI-CLOCKWISE DIRECTION   |
| С    |  |
|      | DO 1406  ID=1.NSEED  |
|      | DO 1406 I = 1.NSTD(TD)   |
|      | $\frac{1}{100} = \frac{1}{100} \frac{1}{100} \frac{1}{100}$  |
|      | TE (TE, TE) = TEM(E, TE)   |
|      | IF (IDP.GT.ID .OK. IDP.BI.O) INEN  |
|      | 1DN = 1DNOD(L, 1D)   |
| MAIN |  |
|      |  |
|      | IF (NODAR(1, IDN).EQ.0) THEN   |
|      | LP1 = MOD(L, NSID(ID)) + 1   |
|      | LM1 = MOD(L+NSID(ID)-2, NSID(ID))+1  |
|      | NODAR(2, IDN) = IDNOD(LP1, ID)   |
|      | NODAR(3, TDN) = TDNOD(I, M1, TD)   |
|      | NODAR(5, IDN) = IDNOD(211, ID)   |
|      | NODAR(G, IDN) = IDND(I, ID)  |
|      | NUARTO, UNU = UUUDULMU, UU   |
|      |  |
|      | NODAR(9, IDN) = ID   |
|      | NODAR(9, IDN) = ID<br>NODAR(10, IDN) = IPRM(L, ID)   |
|      | NODAR $(9, IDN) = ID$<br>NODAR $(10, IDN) = IPRM(L, ID)$<br>NODAR $(11, IDN) = IPRM(LM1, ID)$  |
|      | NODAR $(9, IDN) = ID$<br>NODAR $(10, IDN) = IPRM(L, ID)$<br>NODAR $(11, IDN) = IPRM(LM1, ID)$<br>NODAR $(1, IDN) = 2$  |
|      | NODAR $(9, ILN) = ID$<br>NODAR $(10, ILN) = IPRM(L, ID)$<br>NODAR $(11, IDN) = IPRM(LM1, ID)$<br>NODAR $(1, IDN) = 2$<br>IDP = IPRM(L, ID)   |
|      | NODAR $(9, IDN) = ID$<br>NODAR $(10, IDN) = IPRM (L, ID)$<br>NODAR $(11, IDN) = IPRM (LM1, ID)$<br>NODAR $(1, IDN) = 2$<br>IDP = IPRM $(L, ID)$<br>IF (IDP.GT.0) THEN  |
|      | NODAR $(9, IDN) = ID$<br>NODAR $(10, IDN) = IPRM(L, ID)$<br>NODAR $(11, IDN) = IPRM(LM1, ID)$<br>NODAR $(1, IDN) = 2$<br>IDP = IPRM(L, ID)<br>IF (IDP.GT.0) THEN<br>LP = LPRM(L, ID)   |
|      | NODAR $(9, IDN) = ID$<br>NODAR $(10, IDN) = IPRM(L, ID)$<br>NODAR $(11, IDN) = IPRM(LM1, ID)$<br>NODAR $(1, IDN) = 2$<br>IDP = IPRM(L, ID)<br>IF (IDP.GT.0) THEN<br>LP = LPRM(L, ID)<br>ID1 = NOD(ID NOTD(IDD)):1  |
|      | NODAR $(9, IDN) = ID$<br>NODAR $(10, IDN) = IPRM(L, ID)$<br>NODAR $(11, IDN) = IPRM(LM1, ID)$<br>NODAR $(1, IDN) = 2$<br>IDP = IPRM(L, ID)<br>IF (IDP.GT.0) THEN<br>LP = LPRM(L, ID)<br>LPP1 = MOD(LP,NSID(IDP))+1<br>IP (IDPOP(I DD) DO IDDOD(IDD)) (TUP)   |
|      | NODAR (9, IDN) = ID<br>NODAR (10, IDN) = IPRM(L, ID)<br>NODAR (11, IDN) = IPRM(LM1, ID)<br>NODAR (1, IDN) = 2<br>IDP = IPRM(L, ID)<br>IF (IDP.GT.0) THEN<br>LP = LPRM(L, ID)<br>LPP1 = MOD(LP,NSID(IDP))+1<br>IF (IDNOD(L, ID).EQ.IDNOD(LPP1, IDP)) THEN   |
|      | NODAR (9, IDN) = ID<br>NODAR (10, IDN) = IPRM(L, ID)<br>NODAR (11, IDN) = IPRM(LM1, ID)<br>NODAR (1, IDN) = 2<br>IDP = IPRM(L, ID)<br>IF (IDP.GT.0) THEN<br>LP = LPRM(L, ID)<br>LPP1 = MOD(LP,NSID(IDP))+1<br>IF (IDNOD(L, ID).EQ.IDNOD(LPP1, IDP)) THEN<br>LPP2 = MOD(LPP1,NSID(IDP))+1   |
|      | NODAR $(9, ILN) = ID$<br>NODAR $(10, ILN) = IPRM(L, ID)$<br>NODAR $(11, IDN) = IPRM(L, ID)$<br>NODAR $(1, IDN) = IPRM(IM1, ID)$<br>NODAR $(1, IDN) = 2$<br>IDP = IPRM(L, ID)<br>IF (IDP.GT.0) THEN<br>LP = LPRM(L, ID)<br>LPP1 = MOD(LP,NSID(IDP))+1<br>IF (IDNOD(L, ID).EQ.IDNOD(LPP1, IDP)) THEN<br>LPP2 = MOD(LPP1,NSID(IDP))+1<br>NODAR $(4, IDN) = IDNOD(LPP2, IDP)$  |
|      | NODAR $(9, IDN) = ID$<br>NODAR $(10, IDN) = IPRM(L, ID)$<br>NODAR $(11, IDN) = IPRM(L, ID)$<br>NODAR $(1, IDN) = IPRM(IM1, ID)$<br>NODAR $(1, IDN) = 2$<br>IDP = IPRM(L, ID)<br>IF (IDP.GT.0) THEN<br>LP = LPRM(L, ID)<br>LPP1 = MOD(LP,NSID(IDP))+1<br>IF (IDNOD(L, ID).EQ.IDNOD(LPP1, IDP)) THEN<br>LPP2 = MOD(LPP1,NSID(IDP))+1<br>NODAR $(4, IDN) = IDNOD(LPP2, IDP)$<br>NODAR $(7, IDN) = IDTUB(LPP1, IDP)$                                       |
|      | NODAR $(9, IDN) = ID$<br>NODAR $(10, IDN) = IPRM(L, ID)$<br>NODAR $(11, IDN) = IPRM(L, ID)$<br>NODAR $(1, IDN) = IPRM(IM1, ID)$<br>NODAR $(1, IDN) = 2$<br>IDP = IPRM(L, ID)<br>IF (IDP.GT.0) THEN<br>LPP = LPRM(L, ID)<br>LPP1 = MOD(LP,NSID(IDP))+1<br>IF (IDNOD(L, ID) .EQ.IDNOD(LPP1, IDP)) THEN<br>LPP2 = MOD(LPP1,NSID(IDP))+1<br>NODAR $(4, IDN) = IDNOD(LPP2, IDP)$<br>NODAR $(7, IDN) = IDTUB(LPP1, IDP)$<br>NODAR $(1, IDN) = 3$             |
|      | NODAR $(9, IDN) = ID$<br>NODAR $(10, IDN) = IPRM(L, ID)$<br>NODAR $(11, IDN) = IPRM(L, ID)$<br>NODAR $(1, IDN) = IPRM(LM1, ID)$<br>NODAR $(1, IDN) = 2$<br>IDP = IPRM(L, ID)<br>IF (IDP.GT.0) THEN<br>LP = LPRM(L, ID)<br>LPP1 = MOD(LP,NSID(IDP))+1<br>IF (IDNOD(L, ID).EQ.IDNOD(LPP1, IDP)) THEN<br>LPP2 = MOD(LPP1,NSID(IDP))+1<br>NODAR $(4, IDN) = IDNOD(LPP2, IDP)$<br>NODAR $(7, IDN) = IDTUB(LPP1, IDP)$<br>NODAR $(1, IDN) = 3$<br>GO TO 1406 |
|      | C<br>1402<br>C<br>C<br>C   |

| 01505        |      | ENDIF  |
|--------------|------|--|
| 01506        |      | IDP = IPRM(LM1, ID)  |
| 01507        |      | NODAR(10, IDN) = IDP   |
| 01508        |      | IF (IDP.GT.0) THEN   |
| 01509        |      | LP = LPRM(LM1, ID)   |
| 01510        |      | TF $(TDNOD(I, TD), FO, TDNOD(I, P, TDP))$ THEN   |
| 01511        |      | LPM1 = MOD(LP+NSTD(TDP) - 2, NSTD(TDP)) + 1  |
| 01512        |      | $\frac{1}{100} = \frac{100}{100} (1000) (100$   |
| 01512        |      | MODAR(4, IDA) = IDAOD(IDAI)  |
| 01513        |      | MODAR(7, IDN) = 3  |
| 01514        |      | NODAR(1, 1DN) = 5  |
| 01515        |      |  |
| 01516        |      |  |
| 01517        |      | ENDIF  |
| 01518        |      | ENDIF  |
| 01519        | 1406 | CONTINUE   |
| 01520        | С    |  |
| 01521        | C    | TABULATE AND STORE BOUNDARY NODES AND CROSSING TUBES   |
| 01522        | С    | NNEC(J) IS THE NUMBER OF NODES/TUBES ON BOUNDARY J; NBID(I,J) IS   |
| 01523        | С    | THE NODE NODE NUMBER OF THE JTH NODE ON BOUNDARY I; TBID(I,J) IS   |
| 01524        | С    | THE TUBE NUMBER OF THE JTH TUBE CROSSING BOUNDARY I.   |
| 01525        | Ċ    |  |
| 01526        | •    | NNBC(1) = 0  |
| 01527        |      | NNBC(2) = 0  |
| 01528        |      | NNBC(3) = 0  |
| 01520        |      | $\operatorname{NNEC}(A) = 0$   |
| 01529        |      | $\frac{1447}{12} = 0$  |
| 01530        |      | $\frac{1}{100} = \frac{1}{1000} \frac{1}{1000}$  |
| 01531        |      | IDB = - NODAR(0, IDR)  |
| 01532        |      | $\frac{1}{100} \cdot \frac{1}{100} = \frac{1}{100} \cdot \frac{1}$ |
| 01533        |      | MMBC(IDB) = MMBC(IDB) + I  |
| 01534        |      | NBTD(TDR'NNBC(TDR)) = TDN  |
| 01535        |      | LO 1448 J = 1, NODAR(1, IDN)   |
| 01536        |      | 1DNJ = NODAR(1+J, 1DN)   |
| 01537        |      | IF (NODAR(8, IDN) .NE. NODAR(8, IDNJ)) THEN  |
| 01538        |      | TBID(IDB, NNBC(IDB)) = NODAR(4+J, IDN)   |
| 01539        |      | GO TO 1447   |
| BIOREM\$MAIN |      |  |
| 01540        |      |  |
| 01540        | 1    | ENDIF  |
| 01541        | 1448 | CONTINUE   |
| 01542        |      | ENDIF  |
| 01543        | 1447 | CONTINUE   |
| 01544        | C    |  |
| 01545        | C    | PLOTT PARTICLES  |
| 01546        | С    |  |
| 01547        |      | IF (IPLTP.EQ.1) THEN   |
| 01548        |      | DO 1811 ID = 1, NSEED  |
| 01549        |      | XCEN = XS(ID) *XAXIS /XMAX   |
| 01550        |      | YCEN = YS(ID) *YAXIS /YMAX   |
| 01551        |      | RADIN = 0.95 *RADI(ID) *XAXIS /XMAX  |
| 01552        |      | CALL BLCIR (XCEN, YCEN, RADIN, 0)  |
| 01553        | 1811 | CONTINUE   |
| 01554        |      | ENDIF  |
| 01555        | С    |  |
| 01556        |      | IF (IPLTP.NE.0) THEN   |
| 01557        |      | IF (IPLTP .EQ. 1) THEN   |
| 01558        |      | CALL MARKER (16)   |
| 01559        | CXXX | CALL BLSYM   |
| 01560        |      | ELSEIF (IPLTP .EQ1) THEN   |
| 01561        |      | CALL MARKER (15)   |
| 01562        |      | ENDIF  |
|              |      |  |
| 01563<br>01564<br>01565<br>01566<br>01567<br>01568<br>01569<br>01570<br>01571 |      | CALL BLREC(-0.2*XAXIS,-0.2*YAXIS,0.2*XAXIS,1.4*YAXIS,0)<br>CALL BLREC(-0.2*XAXIS,-0.2*YAXIS,1.4*XAXIS,0.2*YAXIS,0)<br>CALL BLREC(1.0*XAXIS,-0.2*YAXIS,0.2*XAXIS,1.4*YAXIS,0)<br>CALL BLREC(-0.2*XAXIS, 1.0*YAXIS,1.4*XAXIS,0.2*YAXIS,0)<br>DO 1801 ID=1,NSEED<br>XB(1)=XS(ID)<br>YB(1)=YS(ID)<br>FAC= 2.* RADI(ID)/XMAX *XAXIS/0.082<br>CALL SCLPIC(FAC)   |
|---|------|--|
| 01572   |      | CALL CURVE(XB, YB, 1, 1)   |
| 01573   | 1801 | CONTINUE   |
| 01574   |      | CALL SCLPIC(1.)  |
| 01575   |      | CALL RESET('BLSYM')  |
| 01576   |      | ENDIF  |
| 01577   | С    |  |
| 01578   | C    | COMPUTE MAX NODE DIAMETERS   |
| 01579   | С    |  |
| 01580   |      | DO 3018 IDN = 1, NNOD  |
| 01581   |      | XTU(TDN) = XNOD(IDN)   |
| 01582   |      | YTTI(TDN) = YNOD(TDN)  |
| 01583   |      | RTI(TDN) = XMAX /100   |
| 01594   |      | PMTNI – YMAY   |
| 01505   | C    | TE (NODER (1 TON) TT 3) CO TO 3018   |
| 01505   | C    | $\frac{11}{100} = \frac{1}{100} = 1$   |
| 01500   |      | $\frac{1}{100} = 1, \frac{1}{100}$   |
| 01507   |      | IDI = IVUDAR (4+0, IDIV)   |
| 01566   |      | ID = IUDI(OD (3, IDI))   |
| 01589   |      | IDP = IODIOD (4, IDI)  |
| 01230   |      | $\frac{11}{10.01.0} \cdot \frac{100.01.0}{100.01.0} \cdot \frac{100.01.0}{100.000} \cdot \frac{100.000}{100.000} \cdot 10$   |
| 01201   |      | XAA(U, 1) = -2.  (AS(1D) - AS(1DP))  |
| 01592   |      | $XAA(J, Z) = -Z \cdot * (YS(ID) - YS(IDP))$  |
| 01593   |      | XAA(J,3) = -2. * (RADI(1D) - RADI(1DP))  |
| 01594   |      | RHS(J) = RADI(ID) **2 - RADI(IDP) **2  |
| 01595   | -    | $\frac{1}{2} - XS(ID) **2 + XS(IDP) **2 - YS(ID) **2 + YS(IDP) **2$  |
| 01596   |      | QHS(J) = RADI(ID) - RADI(IDP)  |
| BIOREMŞI  | MAIN |  |
| 01507   |      |  |
| 01500   |      | DIS = SQNI ((NS(ID) - NS(IDI)) = VIII)   |
| 01500   |      | AIAI = DIS = IADI(ID) = IADI(IDI)  |
| 01599   |      | $\mathbf{F} = \mathbf{F} + $ |
| 01000   |      | ELDE $(T, 1) = 0$  |
| 01601   |      | AAA(0,1) = 0.  |
| 01602   |      | XAA(0,2) = 0.  |
| 01603   |      | XAA(J, 3) = 0.   |
| 01604   |      | $\operatorname{RHS}(J) = 0.$   |
| 01605   |      | QHS(J) = 0.  |
| 01606   |      | 1DSAV = 1D   |
| 01607   |      | ID = MAX (ID, IDP)   |
| 01608   |      | IDP = MIN (IDSAV, IDP)   |
| 01609   |      | IF (IDP.EQ2 .OR. IDP.EQ4) THEN   |
| 01610   |      | XAA(J,1) = 1.  |
| 01611   |      | $\operatorname{RHS}(J) = 0.$   |
| 01612   |      | IF (IDP .EQ. $-2$ ) RHS(J) = XMAX  |
| 01613   |      | ELSE   |
| 01614   |      | XAA(J,2) = 1.  |
| 01615   |      | RHS(J) = 0.  |
| 01616   |      | IF (IDP .EQ. $-3$ ) RHS(J) = YMAX  |
| 01617   |      | ENDIF  |
| 01618   |      | ENDIF  |
| 01619   | 3019 | CONTINUE   |
| 01620   | C    |  |

| 01621          |      | WT = 1.  |
|----------------|------|--|
| 01622          |      | WTR = 0.3  |
| 01623          |      | RTU(IDN) = 1.1 * RMIN  |
| 01624          |      | RNEW = RTU(TDN)  |
| 01625          |      | TTSK = 0   |
| 01626          |      | $r_{0} = 0$  |
| 01020          |      | $100 \ 5077 \ 113 \ = \ 1, \ 50$   |
| 01627          | ~    | 1TSK = 1TSK + 1  |
| 01628          | С    |  |
| 01629          |      | RTU(IDN) = (1WTR) * RTU(IDN) + WTR * RNEW  |
| 01630          |      | DETA = XAA(1,1) * XAA(2,2) - XAA(1,2) * XAA(2,1)   |
| 01631          |      | XHS(1) = RHS(1) + 2.* QHS(1) * RTU(IDN)  |
| 01632          |      | XHS(2) = RHS(2) + 2.* QHS(2) *RTU(IDN)   |
| 01633          |      | DETX = $XHS(1)$ * $XAA(2,2) - XAA(1,2)$ * $XHS(2)$   |
| 01634          |      | DETY = XAA(1,1) * XHS(2) - XHS(1) * XAA(2,1)   |
| 01635          |      | $\frac{2}{2} = \frac{1}{2} = \frac{1}$ |
| 01636          |      | VNEW - DEIN / DEIN   |
| 01030          |      | INEW = DEII / DEIA   |
| 01637          |      | $XTO(1DN) = (1 - WT)^{XTO(1DN)} + WT^{XNEW}$   |
| 01638          |      | YTU(IDN) = (1WT) * YTU(IDN) + WT * YNEW  |
| 01639          | С    |  |
| 01640          |      | DELM = 0.  |
| 01641          |      | ERR = 0.   |
| 01642          |      | DO $3076 J = 1$ , NODAR (1, IDN)   |
| 01643          |      | TD = NODAR(J+8, TDN)   |
| 01644          |      |  |
| 01645          |      | PTTE = (YC(TD) - YTT(TDN)) **2 + (VC(TD) - VTT(TDN)) **2   |
| 01645          |      | $RIOr = (RS(ID) - RIO(IDN))^{**2} + (IS(ID) - IIO(IDN))^{**2}$   |
| 01040          |      | RAA = SQRI (RIOF) - RADI(ID)   |
| 01647          |      | $\operatorname{RES}(J) = \operatorname{SQRT}(\operatorname{RTOF}) - \operatorname{RADI}(\operatorname{ID}) - \operatorname{RTO}(\operatorname{IDN})$   |
| 01648          |      | ERR = MAX (ERR, ABS(RES(J)))   |
| 01649          |      | IF (ABS(RXX-RTU(IDN)) .GT. DELM) THEN  |
| 01650          |      | DELM = ABS (RXX-RTU(IDN))  |
| 01651          |      | RNEW = RXX   |
| 01652          |      | ENDIF  |
| 01653          | 3076 | CONTINUE   |
|                |      |  |
| BIOREMŞI       | MAIN |  |
| 01654          |      | IF (ERR .LT. 1.E-6*XMAX) GO TO 3078  |
| 01655          | С    |  |
| 01656          | 3077 | CONTRACT   |
| 01657          | 3078 | CONTINUE   |
| 01650          | 5070 | $D_{\text{TITINI}} = Max (D_{\text{TITII}} + D_{\text{TITII}}) = 1 = 20$   |
| 01050          | ~    | KIO(IDA) = MAX (KIO(IDA), I.E-30)  |
| 01659          | 0    |  |
| 01660          | 3018 | CONTINUE   |
| 01661          | C    |  |
| 01662          | C    | COMPUTE EFFECTIVE PORE RADII   |
| 01663          | С    |  |
| 01664          |      | PORSUM = 0.  |
| 01665          |      | DO $3017 \text{ IDN} = 1$ , NNOD   |
| 01666          |      | APORE(IDN) = 0.  |
| 01667          |      | $DO_{3016,T} = 1$ , NODAR (1, TDN)   |
| 01668          |      | TDT - NODAR (1+T TDN)  |
| 01660          |      | IDI = MODAR (470, IDR)   |
| 01670          |      | $T = \frac{1}{100} $     |
| 01070          |      | ID = 100 IV (4, ID1)   |
| 010/1          |      | Ir (ID.GT.U .AND. IDP.GT.U ) THEN  |
| 01672          |      | HASE2 = (XS(ID) - XS(IDP)) * 2 + (YS(ID) - YS(IDP)) * 2  |
| 01673          |      | BASE = SQRT (BASE2)  |
| 01674          |      | RLEG = BASE /2.  |
| 01675          |      | XMID = (XS(ID) + XS(IDP)) /2.  |
|                |      |  |
| 01676          |      | YMID = (YS(ID) + YS(IDP)) /2.  |
| 01676<br>01677 |      | YMID = (YS(ID) + YS(IDP)) /2.<br>HIGHT2 = (XNOD(IDN)-XMID)**2 + (YNOD(IDN)-YMID)**2  |

| 01679   |         | DTHETA = ATAN (HIGHT /RLEG)  |
|---------|---------|--|
| 01680   |         | IF (RADI(ID)+RADI(IDP) .LE. BASE) THEN   |
| 01681   |         | APART = DTHETA $/2$ . *(RADI(ID)**2 + RADI(IDP)**2)  |
| 01682   |         | ELSE   |
| 01683   |         | RLAP = (RADI(ID) + RADI(IDP) - BASE) /2.   |
| 01684   |         | PHT1 = ACOS ((RADI(TD) - RLAP) / RADI(TD))   |
| 01695   |         | $H_{AB} = RADI(TD) *SIN(PH11)$   |
| 01605   |         | $\frac{1}{1} = \frac{1}{1} = \frac{1}$   |
| 01686   |         | FIIZ = ACOS ((RAJI(IDF) - IURF) / MADI(IDF))   |
| 01687   |         | $ATRI = (RADI(1D) - RLAP) \cap RLAP / 2.$  |
| 01688   |         | APART = (DTHETA-PHII) / 2. ~RADI(1D) ~ 2 + ATRI  |
| 01689   |         | ATRI = (RADI(IDP) - RLAP) * HLAP /2.   |
| 01690   |         | APART = APART + (DTHETA-PHI2)/2. *RADI(IDP)**2 + ATRI  |
| 01691   |         | ENDIF  |
| 01692   |         | DAREA = MAX (0., HIGHT *RLEG - APART)  |
| 01693   |         | ELSE   |
| 01694   |         | IDSAV = ID   |
| 01695   |         | ID = MAX (ID, IDP)   |
| 01696   |         | TDP = MTN (TDSAV, TDP)   |
| 01607   |         | HYPOTP = (XS(TD) - XNOD(TDN)) **2 + (YS(TD) - YNOD(TDN)) **2   |
| 016097  |         | $\frac{1}{100} = \frac{1}{100} $   |
| 01090   |         | $\frac{1}{10} = \frac{1}{10} \frac{1}{10}$   |
| 01699   |         | $\frac{1}{10} \frac{10}{2} \frac{2}{2} \frac{2}{2} \frac{10}{10} $   |
| 01700   |         | RLEG = ABS (YS(ID) - YNOD(IDN))  |
| 01701   |         |  |
| 01702   |         | RLEG = ABS (XS(ID) - XNOD(IDN))  |
| 01703   |         | ENDIF  |
| 01704   |         | HIGHT = SQRT (HYPOT2 - RLEG**2)  |
| 01705   |         | DTHETA = PI $/2$ ACOS (RLEG /HYPOT)  |
| 01706   |         | IF (RADI(ID) .LE. HIGHT) THEN  |
| 01707   |         | APART = DTHETA /2. *RADI(ID)**2  |
| 01708   |         | ELSE   |
| 01709   |         | RLAP = RADI(ID) - HIGHT  |
| 01710   |         | PHT = ACOS ((RADT(TD) - RLAP) / RADT(TD))  |
| 01710   |         |  |
| DTODEWC | MATN    |  |
| BIOMIN  | 1.ILLIN |  |
| 01711   |         | עד אם – מאסד (דם) אפיזאז (מעד)   |
| 01711   |         | $\frac{1}{1} \frac{1}{1} \frac{1}$ |
| 01/12   |         | $ATRI = (RADI(1D) - RLAP) \cap RLAP / 2.$  |
| 01713   |         | APART = (DTHETA-PHI) / 2. ARADI(ID) A 2 + ATRI   |
| 01714   |         | ENDIF  |
| 01715   |         | DAREA = MAX (0., HIGHT *RLEG /2 APART)   |
| 01716   |         | ENDIF  |
| 01717   |         | APORE(IDN) = APORE(IDN) + DAREA  |
| 01718   | 3016    | CONTINUE   |
| 01719   |         | PORSUM = PORSUM + APORE(IDN)   |
| 01720   | 3017    | CONTINUE   |
| 01721   |         | EPSVOID = PORSUM / (XMAX*YMAX)   |
| 01722   | C       |  |
| 01723   | C       |  |
| 01724   | ĉ       | HOIT NOLD  |
| 01724   | C       | דבי (דוז אדב () אדבי ()  |
| 01725   |         | $\frac{11}{1000} \frac{1000}{1000} = 1$  |
| 01720   |         | VOITE = 0  |
| 01727   |         | ADRIF = U.   |
| 01728   |         | SHIF = 0.  |
| 01729   |         | FAC = 1.   |
| 01730   |         | IF (NODAR(8, IDN).EQ.0) THEN   |
| 01731   |         | SIZ = 0.   |
| 01732   |         | DSHIF = 0.   |
| 01733   |         | DO 1808 J=1,NODAR(1,IDN)   |
| 01734   |         | IDT = NODAR(4+J, IDN)  |
| 01735   |         | XSHIF = XSHIF + TUBAR(7, IDT)  |
|         |         |  |

| 01737<br>01738<br>01739 | 1808 | DSHIF = DSHIF + SQRT(TUBAR(7, IDT)**2 + TUBAR(8, IDT)**2)<br>SIZ = SIZ + TUBAR(2, IDT)<br>CONTINIE |
|-------------------------|------|--|
| 01740                   | 1000 | XSHTE - XSHTE /NODAR(1 TDN)  |
| 01741                   |      | $V_{\text{SHIF}} = V_{\text{SHIF}} / NODAR(1, 1DN)$  |
| 01741                   |      | $\frac{15}{111} = \frac{15}{111} \frac{1000}{1000} \frac{1}{1000}$                                 |
| 01742                   |      | SIZ = SIZ / NODAR(1, 1DN)  |
| 01745                   |      | FAC = 0.0 "MIN(DSHIF, SIZ) /AMAA "AAAIS /0.082   |
| 01/44                   |      |  |
| 01745                   |      | FAC = MAX (FAC, .01)   |
| 01746                   |      | XB(1) = XNOD(1DN) + XSHIF  |
| 01747                   |      | YB(1) = YNOD(IDN) + YSHIF  |
| 01748                   |      | IF (IPLTN.NE.1) $FAC = 1$ .  |
| 01749                   | С    |  |
| 01750                   |      | CALL SCLPIC (FAC)  |
| 01751                   | С    |  |
| 01752                   |      | XB(1) = XTU(IDN)   |
| 01753                   |      | YB(1) = YTU(IDN)   |
| 01754                   |      | FAC = 2. *RTU(IDN) /XMAX *XAXIS /0.082   |
| 01755                   |      | FAC = MAX (FAC, 0.01)  |
| 01756                   |      | FAC = MIN (FAC, 0.2 *XMAX /XMAX *XAXIS /0.082 )  |
| 01757                   |      | CALL SCLPIC (FAC)  |
| 01758                   |      | CALL MARKER (15)   |
| 01759                   |      |  |
| 01760                   | С    | CALL SCLPIC(0.5)   |
| 01761                   |      | CALL CURVE (XB, YB, 1, 1)  |
| 01762                   | 1809 | CONTINUE   |
| 01763                   |      | CALL SCLPIC(1.)  |
| 01764                   |      | ENDIF  |
| 01765                   | С    |  |
| 01766                   | C PI | LOTT TUBES   |
| 01767                   | С    |  |
| BIOREM\$                | MAIN |  |
| 01760                   |      |  |
| 01768                   |      | IF (IPLIT.NE.U) THEN   |
| 01769                   |      | CALL BLREC (-0.2*XAXIS, -0.2*YAXIS, 0.2*XAXIS, 1.4*YAXIS, 0)                                       |
| 01770                   |      | CALL BLREC(-0.2*XAXIS,-0.2*YAXIS,1.4*XAXIS,0.2*YAXIS,0)  |
| 01771                   |      | CALL BLREC( 1.0*XAXIS, -0.2*YAXIS, 0.2*XAXIS, 1.4*YAXIS, 0)  |
| 01772                   |      | CALL BLREC(-0.2*XAXIS, 1.0*YAXIS,1.4*XAXIS,0.2*YAXIS,0)  |
| 01773                   |      | DO 1803 IDT=1,NIUB   |
| 01774                   |      | XB(1) = TUBAR(3, IDT)  |
| 01775                   |      | YB(1) = TUBAR(4, IDT)  |
| 01776                   |      | XB(2) = TUBAR(5, IDT)  |
| 01777                   |      | YB(2) = TUBAR(6, IDT)  |
| 01778                   |      | WID = MIN (TUBAR(2, IDT), 3.*TUBAR(1, IDT))  |
| 01779                   |      | IF (TUBAR(2, IDT) .LT. $1.1E-2 \text{ *SPAC}$ ) WID = 0.   |
| 01780                   |      | WID = 0.98 * MAX (WID, 0.)   |
| 01781                   |      | WID = WID /XMAX *XAXIS   |
| 01782                   |      | IF (IPLTT.EQ.1) CALL THKCRV(WID)   |
| 01783                   |      | CALL CURVE(XB,YB,2,0)  |
| 01784                   | 1803 | CONTINUE   |
| 01785                   |      | CALL RESET ('THKCRV')  |
| 01786                   |      | ENDIF  |
| 01787                   | С    |  |
| 01788                   | C    | PLOTT TUBE/NODE TRIOS  |
| 01789                   | С    |  |
| 01790                   |      | IPLTTRI = 0  |
| 01791                   |      |  |
|                         |      | IF (IFEIIRI.NE.U) THEN   |
| 01792                   |      | DO 1793 $IDN = 1$ , NNOD   |
| 01792<br>01793          |      | DO 1793 IDN = 1, NNOD<br>XB(1) = XNOD(IDN)   |

| 01795   | ICH = NODAR(1, IDN)                               |
|---------|---|
| 01796   | DO 1793 $J = 1$ , ICH                             |
| 01797   | ID = NODAR(J+1, IDN)                              |
| 01798   | XB(2) = XNOD(ID)                                  |
| 01799   | YB(2) = YNOD(ID)                                  |
| 01800   | CALL CURVE $(XB, YB, 2, 0)$                       |
| 01801   | 1793 CONTINUE                                     |
| 01802   | CALL RESET('DASH')                                |
| 01803   | ENDIF   |
| 01005   | C .   |
| 01004   |   |
| 01805   | C SHADE DACKGROUND                                |
| 01806   |   |
| 01807   | $\frac{11}{12} (12) \frac{11}{12} = 0 $           |
| 01808   | IF (XBA1(2), DI.0.9) IHEN                         |
| 01809   | XBAI(1) = 0.                                      |
| 01810   | YBAI(1) = 0.                                      |
| 01811   | XBAI(2) = XMAX                                    |
| 01812   | YBA1(2) = 0.                                      |
| 01813   | XBA2(1) = 0.                                      |
| 01814   | YBA2(1) = YMAX                                    |
| 01815   | XBA2(2) = XMAX                                    |
| 01816   | YBA2(2) = YMAX                                    |
| 01817   | ENDIF   |
| 01818   | CALL SHDPAT (17)                                  |
| 01819   | IF (ISHAD .LT. 0) CALL SHDPAT(16)                 |
| 01820   | CALL SHDCRV(XBA1,YBA1,2,XBA2,YBA2,2)              |
| 01821   | ENDIF   |
| 01822   | С   |
| 01823   | CALL ENDPL(0)                                     |
| 01824   | PRINT*, ISEEDO                                    |
| BIOREMŞ | MAIN  |
| 01825   | PRINT913, NPAR                                    |
| 01826   | PRINT913, NNOD                                    |
| 01827   | PRINT913.NTUB                                     |
| 01828   | PRINT900, PORO                                    |
| 01829   | PRINT980, 'EPSV', EPSVOID                         |
| 01830   | PRINT*.''   |
| 01831   | с — — ,   |
| 01832   | C CHECK CONSTSTANCY                               |
| 01833   |   |
| 01834   | TPAS = 1  |
| 01835   | DO = 502 TON = 1. NNOD                            |
| 01836   | DO 503 I 1 NODAR (1 TDN)                          |
| 01837   | NIM = NODAR (L+1, TDN)                            |
| 01037   | TDT = NODAR (L+A TDN)                             |
| 01020   | 100 = 0   |
| 01033   | MOK = 0   |
| 01040   | PO = 0  |
| 01041   | TE (NODAR(1, NOM) = TEN) TEN                      |
| 01042   | $\frac{12}{120} = 12$                             |
| 01043   | MOK = MOK + 1                                     |
| 01044   | NOR - NOR + 1                                     |
| 01040   |   |
| 01040   |   |
| 01040   | $\frac{1}{1000} = 0$                              |
| 01040   | 1 HAO = V   |
| 01050   | PRIMI 100 TON NUME CORRESPONDENCE.                |
| 01051   | PKINTIJU4, IUN, NUM, L, IUT, XNOD(IUN), INOD(IUN) |
| 01050   | LO 400 IKL = I, NODAR(I, NOM)                     |
| 01852   | NZ = NODAR(IRL+I, NOM)                            |

| 01853  | 465  | PRINT1903, IKL, N2, NODAR (IKL+4, N2), XNOD (N2), YNOD (N2)  |
|--|--|--|
| 01854  |  | ELSEIF (NOK .EQ. 1) THEN   |
| 01855  |  | IDT2 = NODAR (L20K+4, NUM)   |
| 01856  |  | IF (IDT .NE. IDT2) THEN  |
| 01857  |  | IPAS = 0   |
| 01858  |  | PRINT*, 'NO TUBE CORRESPONDANCE'   |
| 01859  |  | PRINT1910, IDN, NODAR (1, IDN), L, IDT,  |
| 01860  |  | 1 NUM, NODAR (1, NUM), L20K, IDT2  |
| 01861  |  | ENDIF  |
| 01862  |  | ELSE   |
| 01863  |  | TPAS = 0   |
| 01864  |  | PRINT / MILPIPLE NODE CORRESPONDANCE!  |
| 01865  |  | FNDTF  |
| 01866  | 503  | CONTENTIE  |
| 01867  | 502  | CONTINUE   |
| 01868  | 502  | TF (TDAS FO 1) DRTNTT* CORRESPONDANCE TS OK!   |
| 01060  |  | $11^{11}$ (11AD .1.2. 1) INCINI , CONTESTONDANCE IS ON   |
| 01070  |  | PEAD* TONT   |
| 01070  |  | $\frac{1}{100} \frac{1}{100} \frac{1}$ |
| 01071  | C  | IF (ICONI.EQ.2) GO IO 1000   |
| 01072  | ĉ  |  |
| 01074  | 0  | COMPUTE TUBE DIAGNOSTICS   |
| 01075  | C  | and the second sec   |
| 01875  |  | SOMT = 0.  |
| 01876  |  | NSUM = 0   |
| 01877  |  | DMAX = U.  |
| 01878  |  | DO 1//1 DD' = 1, NTOB  |
| 01879  |  | IDBOUN = MIN (TUBNOD(3, IDI), TUBNOD(4, IDI))  |
| 01880  |  | IF (IDBOUN .GF. 0) THEN  |
| 01881  |  | NSUM = NSUM + 1  |
|  |  |  |
| BIOREMŞ  | MAIN   |  |
| BIOREMS  | MAIN   |  |
| BIOREMS  | MAIN   | SUMT = SUMT + TUBAR(2, IDT)  |
| BIOREMS<br>01882<br>01883<br>01883   | MAIN   | SUMT = SUMT + TUBAR(2, IDT)<br>DMAX = MAX (DMAX, TUBAR(2, IDT))  |
| BIOREMS<br>01882<br>01883<br>01884<br>01884  | MAIN   | SUMT = SUMT + TUBAR(2, IDT)<br>DMAX = MAX (DMAX, TUBAR(2, IDT))<br>ENDIF   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885  | MAIN<br>1771                                 | SUMT = SUMT + TUBAR(2, IDT)<br>DMAX = MAX (DMAX, TUBAR(2, IDT))<br>ENDIF<br>CONTINUE   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886   | MAIN<br>1771                                 | SUMT = SUMT + TUBAR(2, IDT)<br>DMAX = MAX (DMAX, TUBAR(2, IDT))<br>ENDIF<br>CONTINUE<br>DTBAR = SUMT /NSUM   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887  | MAIN<br>1771<br>C                            | SUMT = SUMT + TUBAR(2, IDT)<br>DMAX = MAX (DMAX, TUBAR(2, IDT))<br>ENDIF<br>CONTINUE<br>DTBAR = SUMT /NSUM   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01888  | MAIN<br>1771<br>C                            | SUMT = SUMT + TUBAR(2, IDT)<br>DMAX = MAX (DMAX, TUBAR(2, IDT))<br>ENDIF<br>CONTINUE<br>DTBAR = SUMT /NSUM<br>SDEVT = 0.   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01885<br>01886<br>01887<br>01888<br>01889   | MAIN<br>1771<br>C                            | SUMT = SUMT + TUBAR(2, IDT)<br>DMAX = MAX (DMAX, TUBAR(2, IDT))<br>ENDIF<br>CONTINUE<br>DTBAR = SUMT /NSUM<br>SDEVT = 0.<br>IFRE = NTUB /20  |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01885<br>01886<br>01887<br>01888<br>01889<br>01890  | MAIN<br>1771<br>C                            | SUMT = SUMT + TUBAR(2, IDT)<br>DMAX = MAX (DMAX, TUBAR(2, IDT))<br>ENDIF<br>CONTINUE<br>DTBAR = SUMT /NSUM<br>SDEVT = 0.<br>IFRE = NTUB /20<br>DO 1773 IBIN = 1, IFRE  |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01885<br>01886<br>01887<br>01888<br>01889<br>01890<br>01891   | 5MAIN<br>1771<br>C<br>1773                   | SUMT = SUMT + TUBAR(2, IDT)<br>DMAX = MAX (DMAX, TUBAR(2, IDT))<br>ENDIF<br>CONTINUE<br>DTBAR = SUMT /NSUM<br>SDEVT = 0.<br>IFRE = NTUB /20<br>DO 1773 IBIN = 1, IFRE<br>FREQ(IBIN) = 0.   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01889<br>01889<br>01890<br>01891<br>01892  | MAIN<br>1771<br>C<br>1773<br>C               | SUMT = SUMT + TUBAR(2, IDT)<br>DMAX = MAX (DMAX, TUBAR(2, IDT))<br>ENDIF<br>CONTINUE<br>DTBAR = SUMT /NSUM<br>SDEVT = 0.<br>IFRE = NTUB /20<br>DO 1773 IBIN = 1, IFRE<br>FREQ(IBIN) = 0.   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01889<br>01890<br>01891<br>01892<br>01893  | MAIN<br>1771<br>C<br>1773<br>C               | SUMT = SUMT + TUBAR(2, IDT)<br>DMAX = MAX (DMAX, TUBAR(2, IDT))<br>ENDIF<br>CONTINUE<br>DTBAR = SUMT /NSUM<br>SDEVT = 0.<br>IFRE = NTUB /20<br>DO 1773 IBIN = 1, IFRE<br>FREQ(IBIN) = 0.<br>DIMIN = 0.   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01889<br>01890<br>01891<br>01892<br>01893<br>01894   | 1771<br>C<br>1773<br>C                       | SUMT = SUMT + TUBAR(2, IDT)<br>DMAX = MAX (DMAX, TUBAR(2, IDT))<br>ENDIF<br>CONTINUE<br>DTBAR = SUMT /NSUM<br>SDEVT = 0.<br>IFRE = NTUB /20<br>DO 1773 IBIN = 1, IFRE<br>FREQ(IBIN) = 0.<br>DIMIN = 0.<br>DIMAX = 2.*DTEAR   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01889<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895  | 1771<br>C<br>1773<br>C                       | SUMT = SUMT + TUBAR(2, IDT)<br>DMAX = MAX (DMAX, TUBAR(2, IDT))<br>ENDIF<br>CONTINUE<br>DTBAR = SUMT /NSUM<br>SDEVT = 0.<br>IFRE = NTUB /20<br>DO 1773 IBIN = 1, IFRE<br>FREQ(IBIN) = 0.<br>DIMIN = 0.<br>DIMIN = 0.<br>DIMAX = 2.*DTBAR<br>DO 1772 IDT = 1,NTUB   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01889<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01896   | 1771<br>C<br>1773<br>C                       | <pre>SUMT = SUMT + TUBAR(2, IDT)<br/>DMAX = MAX (DMAX, TUBAR(2, IDT))<br/>ENDIF<br/>CONTINUE<br/>DTBAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NTUB /20<br/>DO 1773 IBIN = 1, IFRE<br/>FREQ(IBIN) = 0.<br/>DIMIN = 0.<br/>DIMIN = 0.<br/>DIMAX = 2.*DTBAR<br/>DO 1772 IDT = 1,NTUB<br/>IDBOUN = MIN (TUBNOD(3, IDT), TUBNOD(4, IDT))</pre>  |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01889<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01896<br>01897  | 1771<br>C<br>1773<br>C                       | <pre>SUMT = SUMT + TUBAR(2, IDT)<br/>DMAX = MAX (DMAX, TUBAR(2, IDT))<br/>ENDIF<br/>CONTINUE<br/>DTBAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NTUB /20<br/>DO 1773 IBIN = 1,IFRE<br/>FREQ(IBIN) = 0.<br/>DIMIN = 0.<br/>DIMIN = 0.<br/>DIMAX = 2.*DTBAR<br/>DO 1772 IDT = 1,NTUB<br/>IDEOUN = MIN (TUBNOD(3, IDT), TUBNOD(4, IDT))<br/>IF (IDEOUN .GT. 0) THEN</pre>   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01889<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01895<br>01896<br>01897<br>01898  | 1771<br>C<br>1773<br>C                       | <pre>SUMT = SUMT + TUBAR(2, IDT)<br/>DMAX = MAX (DMAX, TUBAR(2, IDT))<br/>ENDIF<br/>CONTINUE<br/>DTBAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NTUB /20<br/>DO 1773 IBIN = 1, IFRE<br/>FREQ(IBIN) = 0.<br/>DIMIN = 0.<br/>DIMIN = 0.<br/>DIMAX = 2.*DTBAR<br/>DO 1772 IDT = 1,NTUB<br/>IDBOUN = MIN (TUBNOD(3, IDT), TUBNOD(4, IDT))<br/>IF (IDBOUN .GT. 0) THEN<br/>SDEVT = SDEVT + (TUBAR(2, IDT) - DTEAR)**2</pre>   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01889<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01895<br>01896<br>01897<br>01898<br>01899   | 1771<br>C<br>1773<br>C                       | <pre>SUMT = SUMT + TUBAR(2, IDT)<br/>DMAX = MAX (DMAX, TUBAR(2, IDT))<br/>ENDIF<br/>CONTINUE<br/>DTBAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NTUB /20<br/>DO 1773 IBIN = 1,IFRE<br/>FREQ(IBIN) = 0.<br/>DIMIN = 0.<br/>DIMIN = 0.<br/>DIMAX = 2.*DTBAR<br/>DO 1772 IDT = 1,NTUB<br/>IDEOUN = MIN (TUENOD(3,IDT), TUENOD(4,IDT))<br/>IF (IDEOUN .GT. 0) THEN<br/>SDEVT = SDEVT + (TUBAR(2,IDT)-DTEAR)**2<br/>IBIN = NINT((IFRE-1.)*(TUBAR(2,IDT)-DTMIN)/(DTMAX-DTMIN)) + 1</pre>   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01889<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01896<br>01897<br>01898<br>01899<br>01900   | 1771<br>C<br>1773<br>C                       | <pre>SUMT = SUMT + TUBAR(2, IDT)<br/>DMAX = MAX (DMAX, TUBAR(2, IDT))<br/>ENDIF<br/>CONTINUE<br/>DTBAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NTUB /20<br/>DO 1773 IBIN = 1,IFRE<br/>FREQ(IBIN) = 0.<br/>DIMIN = 0.<br/>DIMIN = 0.<br/>DIMAX = 2.*DTBAR<br/>DO 1772 IDT = 1,NTUB<br/>IDBOUN = MIN (TUBNOD(3, IDT), TUBNOD(4, IDT))<br/>IF (IDBOUN .GT. 0) THEN<br/>SDEVT = SDEVT + (TUBAR(2, IDT) -DTBAR)**2<br/>IBIN = NINT((IFRE-1.)*(TUBAR(2, IDT) -DTMIN)/(DIMAX-DIMIN)) + 1<br/>FREQ(IBIN) = FREQ(IBIN) + 1.</pre>  |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01889<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01896<br>01897<br>01898<br>01899<br>01900<br>01901  | 1771<br>C<br>1773<br>C                       | <pre>SUMT = SUMT + TUBAR(2, IDT)<br/>DMAX = MAX (DMAX, TUBAR(2, IDT))<br/>ENDIF<br/>CONTINUE<br/>DTBAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NTUB /20<br/>DO 1773 IBIN = 1, IFRE<br/>FREQ(IBIN) = 0.<br/>DTMIN = 0.<br/>DTMIN = 0.<br/>DIMAX = 2.*DTBAR<br/>DO 1772 IDT = 1,NTUB<br/>IDEOUN = MIN (TUENOD(3, IDT), TUENOD(4, IDT))<br/>IF (IDBOUN .GT. 0) THEN<br/>SDEVT = SDEVT + (TUBAR(2, IDT) -DTEAR)**2<br/>IBIN = NINT((IFRE-1.)*(TUBAR(2, IDT) -DIMIN)/(DTMAX-DTMIN)) + 1<br/>FREQ(IBIN) = FREQ(IBIN) + 1.<br/>ENDIF</pre>   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01899<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01896<br>01897<br>01898<br>01899<br>01900<br>01901<br>01902   | 5MAIN<br>1771<br>C<br>1773<br>C              | <pre>SUMT = SUMT + TUBAR(2, IDT)<br/>DMAX = MAX (DMAX, TUBAR(2, IDT))<br/>ENDIF<br/>CONTINUE<br/>DTBAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NTUB /20<br/>DO 1773 IBIN = 1, IFRE<br/>FREQ(IBIN) = 0.<br/>DIMIN = 0.<br/>DIMIN = 0.<br/>DIMAX = 2.*DTBAR<br/>DO 1772 IDT = 1,NTUB<br/>IDBOUN = MIN (TUBNOD(3, IDT), TUBNOD(4, IDT))<br/>IF (IDBOUN .GT. 0) THEN<br/>SDEVT = SDEVT + (TUBAR(2, IDT) -DTBAR)**2<br/>IBIN = NINT((IFRE-1.)*(TUBAR(2, IDT) -DTMIN)/(DTMAX-DTMIN)) + 1<br/>FREQ(IBIN) = FREQ(IBIN) + 1.<br/>ENDIF<br/>CONTINUE</pre>  |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01896<br>01897<br>01898<br>01897<br>01898<br>01899<br>01900<br>01901<br>01902<br>01903   | MAIN<br>1771<br>C<br>1773<br>C               | <pre>SUMT = SUMT + TUBAR(2, IDT)<br/>DMAX = MAX (DMAX, TUBAR(2, IDT))<br/>ENDIF<br/>CONTINUE<br/>DTBAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NTUB /20<br/>DO 1773 IBIN = 1, IFRE<br/>FREQ(IBIN) = 0.<br/>DTMIN = 0.<br/>DTMIN = 0.<br/>DTMAX = 2.*DTEAR<br/>DO 1772 IDT = 1,NTUB<br/>IDEOUN = MIN (TUENOD(3, IDT), TUENOD(4, IDT))<br/>IF (IDBOUN .GT. 0) THEN<br/>SDEVT = SDEVT + (TUBAR(2, IDT) -DTBAR)**2<br/>IBIN = NINT((IFRE-1.)*(TUBAR(2, IDT) -DTMIN)/(DIMAX-DTMIN)) + 1<br/>FREQ(IBIN) = FREQ(IBIN) + 1.<br/>ENDIF<br/>CONTINUE<br/>SDEVT = SQRT(SDEVT /NSUM)</pre>  |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01896<br>01897<br>01898<br>01897<br>01898<br>01899<br>01900<br>01901<br>01902<br>01903<br>01904  | 5MAIN<br>1771<br>C<br>1773<br>C              | <pre>SUMT = SUMT + TUBAR(2, IDT)<br/>DMAX = MAX (DMAX, TUBAR(2, IDT))<br/>ENDIF<br/>CONTINUE<br/>DTBAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NTUB /20<br/>DO 1773 IBIN = 1, IFRE<br/>FREQ(IBIN) = 0.<br/>DTMIN = 0.<br/>DTMAX = 2.*DTBAR<br/>DO 1772 IDT = 1,NTUB<br/>IDEOUN = MIN (TUENOD(3, IDT), TUENOD(4, IDT))<br/>IF (IDEOUN .GT. 0) THEN<br/>SDEVT = SDEVT + (TUBAR(2, IDT) -DTBAR)**2<br/>IBIN = NINT((IFRE-1.)*(TUBAR(2, IDT) -DTMIN)/(DTMAX-DTMIN)) + 1<br/>FREQ(IBIN) = FREQ(IBIN) + 1.<br/>ENDIF<br/>CONTINUE<br/>SDEVT = SQRT(SDEVT /NSUM)<br/>PRINT981, 'NSUM', NSUM</pre>  |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01896<br>01897<br>01898<br>01899<br>01900<br>01901<br>01902<br>01903<br>01904<br>01905   | 5MAIN<br>1771<br>C<br>1773<br>C              | <pre>SUMT = SUMT + TUBAR(2, IDT)<br/>DMAX = MAX (DMAX, TUBAR(2, IDT))<br/>ENDIF<br/>CONTINUE<br/>DTBAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NIUB /20<br/>DO 1773 IBIN = 1, IFRE<br/>FREQ(IBIN) = 0.<br/>DTMIN = 0.<br/>DTMIN = 0.<br/>DTMAX = 2.*DTBAR<br/>DO 1772 IDT = 1,NTUB<br/>IDBOUN = MIN (TUBNOD(3, IDT), TUBNOD(4, IDT))<br/>IF (IDBOUN .GT. 0) THEN<br/>SDEVT = SDEVT + (TUBAR(2, IDT) -DTBAR)**2<br/>IBIN = NINT((IFRE-1.)*(TUBAR(2, IDT) -DIMIN)/(DIMAX-DTMIN)) + 1<br/>FREQ(IBIN) = FREQ(IBIN) + 1.<br/>ENDIF<br/>CONTINUE<br/>SDEVT = SQRT(SDEVT /NSUM)<br/>PRINT981, 'NEOU', NTUB-NSUM</pre>  |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01896<br>01897<br>01898<br>01897<br>01898<br>01899<br>01900<br>01901<br>01902<br>01903<br>01904<br>01905<br>01906                            | MAIN<br>1771<br>C<br>1773<br>C               | <pre>SUMT = SUMT + TUEAR(2,IDT)<br/>DMAX = MAX (DMAX, TUBAR(2,IDT))<br/>ENDIF<br/>CONTINUE<br/>DTBAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NTUB /20<br/>DO 1773 IBIN = 1,IFRE<br/>FREQ(IBIN) = 0.<br/>DIMAX = 2.*DTEAR<br/>DO 1772 IDT = 1,NTUB<br/>IDEOUN = MIN (TUENOD(3,IDT), TUENOD(4,IDT))<br/>IF (IDEOUN .GT. 0) THEN<br/>SDEVT = SDEVT + (TUBAR(2,IDT)-DTEAR)**2<br/>IBIN = NINT((IFRE-1.)*(TUEAR(2,IDT)-DTMIN)/(DIMAX-DTMIN)) + 1<br/>FREQ(IBIN) = FREQ(IBIN) + 1.<br/>ENDIF<br/>CONTINUE<br/>SDEVT = SQRT(SDEVT /NSUM)<br/>PRINT981, 'NSUM',NSUM<br/>PRINT981, 'NSUM',NSUM<br/>PRINT981, 'NEOU',NTUE-NSUM<br/>PRINT981, 'NEOU',DTEAR</pre>   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01889<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01896<br>01897<br>01898<br>01899<br>01900<br>01901<br>01902<br>01903<br>01904<br>01905<br>01906<br>01907                            | MAIN<br>1771<br>C<br>1773<br>C               | <pre>SUMT = SUMT + TUBAR(2,IDT)<br/>DMAX = MAX (DMAX, TUBAR(2,IDT))<br/>ENDIF<br/>CONTINUE<br/>DTHAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NTUB /20<br/>DO 1773 IBIN = 1,IFRE<br/>FREQ(IBIN) = 0.<br/>DTMAX = 2.*DTEAR<br/>DO 1772 IDT = 1,NTUB<br/>IDEOUN = MIN (TUENOD(3,IDT), TUENOD(4,IDT))<br/>IF (IDEOUN .GT. 0) THEN<br/>SDEVT = SDEVT + (TUBAR(2,IDT)-DTEAR)**2<br/>IBIN = NINT((IFRE-1.)*(TUBAR(2,IDT)-DTMIN)/(DIMAX-DTMIN)) + 1<br/>FREQ(IBIN) = FREQ(IBIN) + 1.<br/>ENDIF<br/>CONTINUE<br/>SDEVT = SQET (SDEVT /NSUM)<br/>PRINT981, 'NSUM',NSUM<br/>PRINT981, 'NSUM',NSUM<br/>PRINT981, 'NEOU', NTUB-NSUM<br/>PRINT980, 'DEAR',DTEAR<br/>PRINT980, 'SDEV', SDEVT</pre>   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01889<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01896<br>01897<br>01898<br>01899<br>01900<br>01901<br>01902<br>01903<br>01904<br>01905<br>01906<br>01907<br>01908                   | 5MAIN<br>1771<br>C<br>1773<br>C<br>1772<br>C | <pre>SUMT = SUMT + TUBAR(2, IDT)<br/>DMAX = MAX (DMAX, TUBAR(2, IDT))<br/>ENDIF<br/>CONTINUE<br/>DTBAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NTUB /20<br/>DO 1773 IBIN = 1,IFRE<br/>FREQ(IBIN) = 0.<br/>DTMIN = 0.<br/>DTMIN = 0.<br/>DIMAX = 2.*DTBAR<br/>DO 1772 IDT = 1,NTUB<br/>IDBOUN = MIN (TUBNDD(3,IDT), TUBNOD(4,IDT))<br/>IF (IDBOUN .GT. 0) THEN<br/>SDEVT = SDEVT + (TUBAR(2,IDT)-DTBAR)**2<br/>IBIN = NINT((IFRE-1.)*(TUBAR(2,IDT)-DTMIN)/(DTMAX-DTMIN)) + 1<br/>FREQ(IBIN) = FREQ(IBIN) + 1.<br/>ENDIF<br/>CONTINUE<br/>SDEVT = SQRT(SDEVT /NSUM)<br/>FRINT981, 'NSUM',NSUM<br/>PRINT981, 'NSUM', DEAR<br/>PRINT980, 'DBAR', DTBAR<br/>PRINT980, 'SDEV', SDEVT</pre>  |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01899<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01896<br>01897<br>01898<br>01899<br>01900<br>01901<br>01902<br>01903<br>01904<br>01905<br>01906<br>01907<br>01908<br>01909          | 5MAIN<br>1771<br>C<br>1773<br>C<br>1772<br>C | <pre>SUMT = SUMT + TUBAR(2, IDT)<br/>IMAX = MAX (DMAX, TUBAR(2, IDT))<br/>ENDIF<br/>CONTINUE<br/>DTEAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NTUB /20<br/>DO 1773 IBIN = 1,IFRE<br/>FREQ(IBIN) = 0.<br/>DTMIN = 0.<br/>DTMAX = 2.*DTEAR<br/>DO 1772 IDT = 1,NTUB<br/>IDBOUN = MIN (TUBNOD(3,IDT), TUBNOD(4,IDT))<br/>IF (IDBOUN .GT. 0) THEN<br/>SDEVT = SDEVT + (TUBAR(2,IDT)-DTEAR)**2<br/>IBIN = NINT((IFRE-1.)*(TUBAR(2,IDT)-DIMIN)/(DTMAX-DTMIN)) + 1<br/>FREQ(IBIN) = FREQ(IBIN) + 1.<br/>ENDIF<br/>CONTINUE<br/>SDEVT = SQRT(SDEVT /NSUM)<br/>PRINT981, 'NSUM',NSUM<br/>PRINT981, 'NSUM', DTEAR<br/>PRINT980, 'DEAR', DTEAR<br/>PRINT980, 'SDEV', SDEVT<br/>PRINT900, DMAX</pre>   |
| BIOREMS<br>01882<br>01883<br>01884<br>01885<br>01886<br>01887<br>01888<br>01899<br>01890<br>01891<br>01892<br>01893<br>01894<br>01895<br>01896<br>01897<br>01898<br>01899<br>01900<br>01901<br>01902<br>01903<br>01904<br>01905<br>01906<br>01907<br>01908<br>01909<br>01910 | 5MAIN<br>1771<br>C<br>1773<br>C<br>1772<br>C | <pre>SUMT = SUMT + TUEAR(2, IDT)<br/>DMAX = MAX (DMAX, TUEAR(2, IDT))<br/>ENDIF<br/>CONTINUE<br/>DTEAR = SUMT /NSUM<br/>SDEVT = 0.<br/>IFRE = NTUE /20<br/>DO 1773 IBIN = 1, IFRE<br/>FREQ(IBIN) = 0.<br/>DTMIN = 0.<br/>DTMIN = 0.<br/>DTMAX = 2.*DTEAR<br/>DO 1772 IDT = 1,NTUE<br/>IDEOUN = MIN (TUENOD(3, IDT), TUENOD(4, IDT))<br/>IF (IDEOUN .GT. 0) THEN<br/>SDEVT = SDEVT + (TUEAR(2, IDT)-DTEAR)**2<br/>IEIN = NINT((IFRE-1.)*(TUEAR(2, IDT)-DTMIN)/(DIMAX-DIMIN)) + 1<br/>FREQ(IBIN) = FREQ(IBIN) + 1.<br/>ENDIF<br/>CONTINUE<br/>SDEVT = SQRT(SDEVT /NSUM)<br/>FRINT981, 'NSUM',NSUM<br/>PRINT981, 'NSUM',NSUM<br/>PRINT981, 'NEOU',NTUE-NSUM<br/>PRINT981, 'NEOU',SDEVT<br/>PRINT980, 'SDEV',SDEVT<br/>PRINT900,DMAX<br/>WRITE(15,901) IFRE</pre>  |

| 01911   |                                | DO 1774 IBIN = 1, IFRE   |
|---|--------------------------------|--|
| 01912   |                                | FSUM = 0.  |
| 01913   |                                | DO 1788 II = 1, IBIN   |
| 01914   | 1788                           | FSUM = FSUM + FREO(II)   |
| 01915   |                                | FSUM = FSUM /NSUM  |
| 01916   |                                | FOIS = 1. *FREQ(IBIN) /NSUM / (1./IFRE)  |
| 01017   |                                | WRITE $(15, 900)$ (TRIN-0.5) / TERE, FOIS, FSUM  |
| 01010   | 1771                           | DDTNTO(11 TETNI FILLS FOLM   |
| 01910   | ~ 1//4                         | PRIMIPOL, IDIN, IDIO, IDIN   |
| 01919   | C                              |  |
| 01920   | C                              | COMPUTE BOUNDARY LENGTHS   |
| 01921   | C                              |  |
| 01922   |                                | DO 1776 J=1, NBOX  |
| 01923   | 1776                           | LBOUN(J) = 0.  |
| 01924   |                                | DO 1777 I=1,NSEED  |
| 01925   |                                | IBOU=-IDBC(I)  |
| 01926   |                                | IF (IBOU.GT.0) THEN  |
| 01927   |                                | LB=LBC(I)  |
| 01928   |                                | LBOUN(IBOU) = LBOUN(IBOU) + LSID(LB, I)  |
| 01929   |                                | ENDIF  |
| 01930   | 1777                           | CONTINUE   |
| 01931   | С                              |  |
| 01932   | C                              | COMPUTE PERIMETER AND PROJECTED AREAS  |
| 01933   | č                              |  |
| 01034   | C                              | TO 1778 I-1 NISEED   |
| 01025   |                                |  |
| 01935   |                                | PEX(I) = 0   |
| 01930   |                                | APRA(1) = 0  |
| 01937   |                                | APKI(1)=0.   |
| 01938   |                                | 1/18 L=1, NSID(1)  |
|   |                                |  |
| BIOREMS   | MAIN                           |  |
|   |                                |  |
| 01020   |                                |  |
| 01939   |                                | PERI(I) = PERI(I) + LSID(L, I)   |
| 01939<br>01940  |                                | PERI(I)=PERI(I)+LSID(L,I)<br>IDP=IPRM(L,I)   |
| 01939<br>01940<br>01941   |                                | PERI(I)=PERI(I)+LSID(L,I)<br>IDP=IPRM(L,I)<br>IF (IDP.LT.0) THEN   |
| 01939<br>01940<br>01941<br>01942  |                                | PERI(I)=PERI(I)+LSID(L,I)<br>IDP=IPRM(L,I)<br>IF (IDP.LT.0) THEN<br>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX  |
| 01939<br>01940<br>01941<br>01942<br>01943   |                                | PERI(I)=PERI(I)+LSID(L,I)<br>IDP=IPRM(L,I)<br>IF (IDP.LT.0) THEN<br>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br>DXDL=COS(THET)  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944  |                                | PERI(I)=PERI(I)+LSID(L,I)<br>IDP=IPRM(L,I)<br>IF (IDP.LT.0) THEN<br>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br>DXDL=COS(THET)<br>DYDL=SIN(THET)  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945   |                                | PERI(I)=PERI(I)+LSID(L,I)<br>IDP=IPRM(L,I)<br>IF (IDP.LT.0) THEN<br>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br>DXDL=COS(THET)<br>DYDL=SIN(THET)<br>ELSE  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946  |                                | PERI(I)=PERI(I)+LSID(L,I)<br>IDP=IPRM(L,I)<br>IF (IDP.LT.0) THEN<br>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br>DXDL=COS(THET)<br>DYDL=SIN(THET)<br>ELSE<br>DXXX=XS(IDP)-XS(I)  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947   |                                | PERI(I)=PERI(I)+LSID(L,I)<br>IDP=IPRM(L,I)<br>IF (IDP.LT.0) THEN<br>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br>DXDL=COS(THET)<br>DYDL=SIN(THET)<br>ELSE<br>DXXX=XS(IDP)-XS(I)<br>DYYY=YS(IDP)-YS(I)  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948  |                                | PERI(I)=PERI(I)+LSID(L,I)<br>IDP=IPRM(L,I)<br>IF (IDP.LT.0) THEN<br>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br>DXDL=COS(THET)<br>DYDL=SIN(THET)<br>ELSE<br>DXXX=XS(IDP)-XS(I)<br>DYYY=YS(IDP)-YS(I)<br>DLLL=SQRT(DXXX**2+DYYY**2)  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949   |                                | PERI(I)=PERI(I)+LSID(L,I)<br>IDP=IPRM(L,I)<br>IF (IDP.LT.0) THEN<br>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br>DXDL=COS(THET)<br>DYDL=SIN(THET)<br>ELSE<br>DXXX=XS(IDP)-XS(I)<br>DYYY=YS(IDP)-YS(I)<br>DLLL=SQRT(DXXX*2+DYYY**2)<br>DXDL=DXXX/DLLL   |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950  |                                | PERI(I)=PERI(I)+LSID(L,I)<br>IDP=IPRM(L,I)<br>IF (IDP.LT.0) THEN<br>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br>DXDL=COS(THET)<br>DYDL=SIN(THET)<br>ELSE<br>DXXX=XS(IDP)-XS(I)<br>DYYY=YS(IDP)-YS(I)<br>DLLL=SQRT(DXXX**2+DYYY**2)<br>DXDL=DXXX/DLLL<br>DYDL=DYYY/DLLL  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951   |                                | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYYY=YS(IDP)-YS(I)<br/>DLLL=SQRT(DXXX**2+DYYY**2)<br/>DXDL=DXXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF</pre>  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952  |                                | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYYY=YS(IDP)-YS(I)<br/>DLLL=SQRT(DXXX**2+DYYY**2)<br/>DXDL=DXXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.</pre>   |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953   |                                | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYYY=YS(IDP)-YS(I)<br/>DLLL=SQRT(DXXX**2+DYYY**2)<br/>DXDL=DXXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRY(I)+LSID(L,I)*ABS(DYDL)/2.</pre>  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954  | 1778                           | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYYY=YS(IDP)-YS(I)<br/>DLLL=SQRT(DXXX*2+DYYY**2)<br/>DXDL=DXXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRY(I)+LSID(L,I)*ABS(DYDL)/2.</pre>   |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954<br>01955   | 1778<br>C                      | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYYY=YS(IDP)-YS(I)<br/>DLLL=SQRT(DXXX*2+DYYY*2)<br/>DXDL=DXXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRY(I)+LSID(L,I)*ABS(DYDL)/2.<br/>CONTINUE</pre>   |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954<br>01955<br>01956  | 1778<br>C                      | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYYY=YS(IDP)-YS(I)<br/>DLLL=SQRT(DXXX*2+DYYY*2)<br/>DXDL=DXXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRY(I)+LSID(L,I)*ABS(DYDL)/2.<br/>CONTINUE</pre>   |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954<br>01955<br>01956<br>01957   | 1778<br>C                      | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYYY=YS(IDP)-YS(I)<br/>DLLL=SQRT(DXXX*2+DYYY*2)<br/>DXDL=DXXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRX(I)+LSID(L,I)*ABS(DYDL)/2.<br/>CONTINUE<br/>D0 1785 I=1,NSEED<br/>IE (IDPC(I) LT =NEOX) PRIMT902 I IDPC(I).XS(I).YS(I)</pre>  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954<br>01955<br>01956<br>01957<br>01958  | 1778<br>C<br>1785              | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYYY=YS(IDP)-YS(I)<br/>DLLL=SQRT(DXXX*2+DYYY**2)<br/>DXDL=DXXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRX(I)+LSID(L,I)*ABS(DYDL)/2.<br/>CONTINUE<br/>D0 1785 I=1,NSEED<br/>IF (IDBC(I).LTNEOX) PRINT902,I,IDEC(I),XS(I),YS(I)</pre>   |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954<br>01955<br>01956<br>01957<br>01958<br>01958   | 1778<br>C<br>1785<br>C         | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYYY=YS(IDP)-YS(I)<br/>DLLL=SQRT(DXXX*2+DYYY**2)<br/>DXDL=DXXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRY(I)+LSID(L,I)*ABS(DYDL)/2.<br/>CONTINUE<br/>D0 1785 I=1,NSEED<br/>IF (IDEC(I).LTNEOX) PRINT902,I,IDEC(I),XS(I),YS(I)</pre>   |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954<br>01955<br>01956<br>01957<br>01958<br>01959<br>01959  | 1778<br>C<br>1785<br>C         | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYYY=YS(IDP)-YS(I)<br/>DLLL=SQRT(DXXX*2+DYYY**2)<br/>DXDL=DXXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRY(I)+LSID(L,I)*ABS(DYDL)/2.<br/>CONTINUE<br/>DO 1785 I=1,NSEED<br/>IF (IDEC(I).LTNEOX) PRINT902,I,IDEC(I),XS(I),YS(I)<br/>CKSUM=0.<br/>DDINUE</pre>   |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954<br>01955<br>01956<br>01957<br>01958<br>01959<br>01960  | 1778<br>C<br>1785<br>C         | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DLLL=SQRT(DXXX*2+DYYY**2)<br/>DXDL=DXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRY(I)+LSID(L,I)*ABS(DYDL)/2.<br/>CONTINUE<br/>D0 1785 I=1,NSEED<br/>IF (IDEC(I).LTNEOX) PRINT902,I,IDEC(I),XS(I),YS(I)<br/>CKSUM=0.<br/>PRINT*,NSEED</pre>   |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954<br>01955<br>01956<br>01957<br>01958<br>01959<br>01960<br>01961   | 1778<br>C<br>1785<br>C         | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DLLL=SQRT(DXXX*2+DYYY**2)<br/>DXDL=DXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRY(I)+LSID(L,I)*ABS(DYDL)/2.<br/>CONTINUE<br/>D0 1785 I=1,NSEED<br/>IF (IDEC(I).LTNEOX) PRINT902,I,IDEC(I),XS(I),YS(I)<br/>CKSUM=0.<br/>PRINT*,NSEED<br/>D0 1631 K=1,20</pre>  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954<br>01955<br>01956<br>01957<br>01958<br>01959<br>01960<br>01961<br>01962  | 1778<br>C<br>1785<br>C         | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYYY=YS(IDP)-YS(I)<br/>DLLL=SQRT(DXXX*2+DYYY**2)<br/>DXDL=DXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRY(I)+LSID(L,I)*ABS(DYDL)/2.<br/>CONTINUE<br/>DO 1785 I=1,NSEED<br/>IF (IDEC(I).LTNEOX) PRINT902,I,IDEC(I),XS(I),YS(I)<br/>CKSUM=0.<br/>PRINT*,NSEED<br/>DO 1631 K=1,20<br/>CKSUM=CKSUM+(1.*K)*(1.*FSID(K))/NSUM</pre>  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954<br>01955<br>01956<br>01957<br>01958<br>01959<br>01960<br>01961<br>01962<br>01963                                     | 1778<br>C<br>1785<br>C         | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYYY=YS(IDP)-YS(I)<br/>DLLL=SQRT(DXXX*2+DYYY**2)<br/>DXDL=DXXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRY(I)+LSID(L,I)*ABS(DYDL)/2.<br/>CONTINUE<br/>DO 1785 I=1,NSEED<br/>IF (IDBC(I).LTNBOX) PRINT902,I,IDBC(I),XS(I),YS(I)<br/>CKSUM=0.<br/>PRINT*,NSEED<br/>DO 1631 K=1,20<br/>CKSUM=CKSUM+(1.*K)*(1.*FSID(K))/NSUM<br/>PRINT902,K,FSID(K),(1.*FSID(K))/NSUM</pre>  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954<br>01955<br>01956<br>01957<br>01958<br>01959<br>01960<br>01961<br>01962<br>01963<br>01964                            | 1778<br>C<br>1785<br>C<br>1631 | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYYY=YS(IDP)-YS(I)<br/>DLLL=SQRT(DXXX**2+DYYY**2)<br/>DXDL=DXXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRY(I)+LSID(L,I)*ABS(DXDL)/2.<br/>CONTINUE<br/>D0 1785 I=1,NSEED<br/>IF (IDEC(I).LTNEOX) PRINT902,I,IDEC(I),XS(I),YS(I)<br/>CKSUM=0.<br/>PRINT*,NSEED<br/>D0 1631 K=1,20<br/>CKSUM=CKSUM+(1.*K)*(1.*FSID(K))/NSUM<br/>PRINT902,K,FSID(K),(1.*FSID(K))/NSUM<br/>PRINT902,JIMIN,JIMAX</pre>  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954<br>01955<br>01956<br>01957<br>01958<br>01959<br>01960<br>01961<br>01962<br>01963<br>01964<br>01965                   | 1778<br>C<br>1785<br>C<br>1631 | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYY=YS(IDP)-YS(I)<br/>DLL=SQRT(DXXX*2+DYYY*2)<br/>DXDL=DXXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRY(I)+LSID(L,I)*ABS(DYDL)/2.<br/>CONTINUE<br/>DO 1785 I=1,NSEED<br/>IF (IDBC(I).LTNBOX) PRINT902,I,IDBC(I),XS(I),YS(I)<br/>CKSUM=0.<br/>PRINT*,NSEED<br/>DO 1631 K=1,20<br/>CKSUM=CKSUM+(1.*K)*(1.*FSID(K))/NSUM<br/>PRINT902,K,FSID(K),(1.*FSID(K))/NSUM<br/>PRINT902,JIMIN,JIMAX<br/>GAVE=GAVE/NSUM*SQRT(1.*NSEED)</pre>  |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954<br>01955<br>01956<br>01957<br>01958<br>01959<br>01960<br>01961<br>01962<br>01963<br>01964<br>01965<br>01966          | 1778<br>C<br>1785<br>C<br>1631 | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DYY=YS(IDP)-YS(I)<br/>DLL=SQRT(DXXX*2+DYYY*2)<br/>DXDL=DXYX/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRY(I)+LSID(L,I)*ABS(DYDL)/2.<br/>CONTINUE<br/>DO 1785 I=1,NSEED<br/>IF (IDEC(I).LTNEOX) PRINT902,I,IDEC(I),XS(I),YS(I)<br/>CKSUM=0.<br/>PRINT*,NSEED<br/>DO 1631 K=1,20<br/>CKSUM=CKSUM+(1.*K)*(1.*FSID(K))/NSUM<br/>PRINT902,K,FSID(K),(1.*FSID(K))/NSUM<br/>PRINT902,JIMIN,JIMAX<br/>GAVE=GAVE/NSUM*SQRT(1.*NSEED)<br/>GMIN=GMIN/NSUM*SQRT(1.*NSEED)</pre>                                 |
| 01939<br>01940<br>01941<br>01942<br>01943<br>01944<br>01945<br>01946<br>01947<br>01948<br>01949<br>01950<br>01951<br>01952<br>01953<br>01954<br>01955<br>01956<br>01957<br>01958<br>01959<br>01960<br>01961<br>01962<br>01963<br>01964<br>01965<br>01966<br>01967 | 1778<br>C<br>1785<br>C<br>1631 | <pre>PERI(I)=PERI(I)+LSID(L,I)<br/>IDP=IPRM(L,I)<br/>IF (IDP.LT.0) THEN<br/>THET=-PI/2.+(-IDP-1)*2.*PI/NBOX<br/>DXDL=COS(THET)<br/>DYDL=SIN(THET)<br/>ELSE<br/>DXXX=XS(IDP)-XS(I)<br/>DLL=SQRT(DXXX**2+DYYY**2)<br/>DXDL=DXX/DLLL<br/>DYDL=DYYY/DLLL<br/>ENDIF<br/>APRX(I)=APRX(I)+LSID(L,I)*ABS(DXDL)/2.<br/>APRY(I)=APRY(I)+LSID(L,I)*ABS(DXDL)/2.<br/>CONTINUE<br/>D0 1785 I=1,NSEED<br/>IF (IDEC(I).LTNBOX) PRINT902,I,IDEC(I),XS(I),YS(I)<br/>CKSUM=0.<br/>PRINT*,NSEED<br/>D0 1631 K=1,20<br/>CKSUM=CKSUM+(1.*K)*(1.*FSID(K))/NSUM<br/>PRINT902,K,FSID(K),(1.*FSID(K))/NSUM<br/>PRINT902,JLMIN,JIMAX<br/>GAVE=GAVE/NSUM*SQRT(1.*NSEED)<br/>GMIN=GMIN/NSUM*SQRT(1.*NSEED)<br/>GMAX=GMAX/NSUM*SQRT(1.*NSEED)</pre> |

| 01969    |            | CMAX=CMAX*SQRT(1.*NSEED)   |
|----------|------------|--|
| 01970    | С          |  |
| 01971    |            | PRINT900, CMIN   |
| 01972    |            | PRINT900, GMIN   |
| 01973    |            | PRINT900, GAVE   |
| 01974    |            | PRINT900, GMAX   |
| 01975    |            | PRINT900, CMAX   |
| 01976    |            | READ*  |
| 01977    | 2000       | CONTINUE   |
| 01978    | с          |  |
| 01979    | C          | PRINT NODE/TUBE CORRESPONDANCE STUFF   |
| 01980    | Ċ          |  |
| 01981    | -          | DO 1733 IDN = 1. 0 *NNOD   |
| 01982    |            | ICH = NODAR(1, IDN)  |
| 01983    |            | DO 1733 J = 1, ICH   |
| 01984    |            | IDT = NODAR (4+J, IDN)   |
| 01985    |            | PRTNT904, TDN, NODAR(1+1, TDN), (TERNOD(1, TDT), (TERNOD(2, TDT))            |
| 01986    | 1733       | CONTINUE:  |
| 01987    | C          |  |
| 01988    | C          | BEGIN CALCULATIONS OF PRESSURE FIELD   |
| 01989    | č          |  |
| 01990    | 1620       | PTN = 1000   |
| 01991    | 1000       | PEX = 1  |
| 01992    |            | TMP = 298  |
| 01993    |            | $R_{GAS} = 287$  |
| 01994    |            | ממצט - 2011.<br>מיזאדיא דער די דער דער די איזאראר ארא ארא איזאראריא איזאראיי |
| 01995    |            | READ*, PTN, PEX  |
|          |            |  |
| BIOREMŞN | <b>AIN</b> |  |
|          |            |  |
| 01996    |            | NSOR = 1000  |
| 01997    |            | WGT = 0.5  |
| 01998    | •          | PRINT*, 'INPUT NSOR, WGT', NSOR, WGT   |
| 01999    |            | READ*, NSOR, WGT   |
| 02000    |            | ABSERR = 1.E-20  |
| 02001    |            | RELERR = 1.E-8   |
| 02002    |            | PRINT*, 'INPUT ABSERR, RELERR', ABSERR, RELERR                               |
| 02003    |            | READ*, ABSERR, RELERR  |
| 02004    |            | IPL/TPR = 0  |
| 02005    |            | IPLTPF = 0   |
| 02006    |            | IPLTPFY = 0  |
| 02007    |            | IF (IDBIN .NE. IDBEV) THEN   |
| 02008    |            | IPLTPF = 0   |
| 02009    |            | IPLTPFY = 0  |
| 02010    |            | ENDIF  |
| 02011    |            | PRINT*, 'INPUT IPLTPR, IPLTPF, IPLTFY', IPLTPR, IPLTPF, IPLTPFY              |
| 02012    |            | READ*, IPLTPR, IPLTPF, IPLTPFY   |
| 02013    |            | ICOMP = 0  |
| 02014    |            | PRINT*, 'INPUT ICOMP: 0 = INCOMP 1 = COMP', ICOMP                            |
| 02015    |            | READ*, ICOMP   |
| 02016    |            | KPAR = 300   |
| 02017    |            | DTSTR = 0.5  |
| 02018    |            | PRINT*, 'INPUT NUMBER OF TRACER PARTICLES AND TIME', KPAR, DTSTR             |
| 02019    |            | READ*, KPAR, DTSTR   |
| 02020    |            | PECL = 1.E10   |
| 02021    |            | 1SED = 12344321  |
| 02022    |            | LF (KPAR .GI. U) THEN  |
| 02023    |            | PKINI'', INPUT PECLET NUMBER', PECL  |
| 02024    |            | KEAD^, PECL  |
| 02025    |            | $\mathbf{M}_{\mathbf{M}} = \mathbf{U}$                                       |
| 02020    |            | PRIME, SPATIAL DIST OR MICRO-FINGERS ? $U = DIST = MF', IMICF$               |

| 02027    |      | READ*, IMICF   |
|----------|------|--|
| 02028    |      | IPARA = 1  |
| 02029    |      | PRINT*, 'PARABOLIC VEL PROFILE ? 0 = NO 1 = YES', IPARA              |
| 02030    |      | READ*, IPARA   |
| 02031    |      | IMIX = 0   |
| 02032    |      | PRINT*, 'MIX NODE STREAMLINES ? 0 = NO 1 = YES', IMIX                |
| 02032    |      | READ*. TMTX  |
| 02033    |      | TINT = 0   |
| 02034    |      | DRINT* 'INTFORM PARTICLE DIST IN INLET TUBE ? 0 = NO 1 = YES',       |
| 02035    | 1    |  |
| 02030    | 1    |  |
| 02037    |      | TDMC = 0   |
| 02038    |      | TRANS = 0<br>POINTOMTZE CEDERALINE AT EVERY STEP 2. 0 = NO 1 = YES'. |
| 02039    | -    | PRINT, RANDOMIZE SIREADINE AI EVENI DIEL ., 0 - NO 2 - 120,          |
| 02040    | 1    | IKANO  |
| 02041    |      | READ*, IRANS   |
| 02042    |      | 1222 = 0   |
| 02043    |      | PRINT*, 'EXIT TUBE ON ENTRANCE STREAMLINE ? 0 = NO 1 = 125 ,         |
| 02044    | 1    |  |
| 02045    |      | READ*, IZZZ  |
| 02046    |      | IPLTTR = 0   |
| 02047    |      | IF (IMICF .NE. 1) THEN   |
| 02048    |      | PRINT*, 'PARTICLE TRACES ? $0 = NO$ $1 = YES', IPLI'IR$              |
| 02049    |      | READ*, IPLTTR  |
| 02050    |      | ENDIF  |
| 02051    |      | PRINT*, 'INPUT SEED FOR TRACER INJECTION', ISED                      |
| 02052    |      | READ*, ISED  |
| BIOREMŞI | MAIN |  |
| 02053    |      | ENDIF  |
| 02054    |      | IPRINI = 1   |
| 02055    |      | IF (IMETH .LT. 0) $IPRINI = 0$                                       |
| 02056    |      | IF (ICONT .NE. 0) IPRINI = 0   |
| 02057    |      | PRINT*, 'DO YOU WANT PRESSURES (RE) INITIALIZED', IPRINI             |
| 02058    |      | READ*, IPRINI  |
| 02059    |      | XXX0 = 0.5   |
| 02060    |      | EDGE = 0.  |
| 02061    |      | PRINT*, 'INPUT XXX0, EDGE', XXX0, EDGE                               |
| 02062    |      | READ*, XXX0, EDGE  |
| 02063    |      | KMETH = 1  |
| 02064    |      | PRINT*, 'AVERAGES OR LOCAL APERTURE: 1 = AVE 2 = LOC ?', KMETH       |
| 02065    |      | READ*, KMEIH   |
| 02066    |      | IGEO = 2   |
| 02067    |      | PRINT*, 'INPUT IGEO', IGEO   |
| 02068    |      | READ*, IGEO  |
| 02069    |      | IOK = 0  |
| 02070    |      | PRINT*, 'INPUT OK', IOK  |
| 02071    |      | RFAD*.IOK  |
| 02072    |      | TF (TOK.EO.0) GO TO 1620   |
| 02073    | C    |  |
| 02074    | C    | TNTTTALTZE PRESSURES   |
| 02074    | c    |  |
| 02076    | C    | TF (IPRINI .NE. 0) THEN  |
| 02077    |      | DO 440 TDN = 1.NNOD  |
| 02078    |      | THET = $XNOD(IDN) / XMAX$  |
| 02070    |      | TE (TOBIN NE. IDBEV) THET = YNOD(IDN) /YMAX                          |
| 02079    |      | TE ((TDBIN, EO.2), OR, (TDBIN, EO.3)) THET = 1 THET                  |
| 02000    |      | DDEC(TIN) - DIN  |
| 02081    |      | TE (TCOMP EO (1) THEN  |
| 02002    |      | DPES(TIN) - PIN - (PIN-PEX)*THET                                     |
| 02083    |      | TE (TERINT LT. 0.) PRES(TEN) = PEX                                   |
| 02084    |      |  |

02085 FLSE 02086 PRES(IDN) = SQRT (PIN\*\*2 - (PIN\*\*2-PEX\*\*2)\*THET)02087 IF (IPRINI .LT. 0.) PRES(IDN) = PEX 02088 ENDIF 02089 PRESO(IDN) = PRES(IDN) 02090 440 CONTINUE 02091 ENDIF 02092 С 02093 TIM = 0.02094 DTIM = 0.102095 ISTOP = 002096 KMAX = 102097 TOUT = 0С 02098 02099 C--- UPDATE CURRENT VALUES OF PRESSURE ---02100 С 02101 DO 710 K = 1, KMAX 02102 IF (ISTOP.EQ.1) GO TO 830 02103 TIM = TIM + DTIM02104 IF (K.GT.1) THEN 02105 DO 321 IDN = 1, NNOD 02106 PRINT\*, IDN, PRESO(IDN) 02107 321 PRES(IDN) = PRESO(IDN)02108 ENDIF 02109 С BIOREMSMAIN 02110 1630 IGO = 002111 С 02112 C--- COMPUTE SPECIFIC SURFACE AREA ---02113 C 02114 SUMA = 0.02115 SUMV = 0. DO 312 ID = 1, NPAR 02116 02117 IF (IGEO .EQ. 2) THEN 02118 SUMA = SUMA + PI \*DIAM(ID) SUMV = SUMV + PI \*DIAM(ID)\*\*2 /4. 02119 02120 ELSE 02121 SUMA = SUMA + PI \*DIAM(ID) \*\*2 02122 SUMV = SUMV + PI \* DIAM(ID) \*\*3 /6.02123 ENDIF 02124 312 CONTINUE 02125 SV000 = SUMA /SUMV 02126 С 02127 C--- COMPUTE EFFECTIVE PORE RADII ---02128 С 02129 PORSUM = 0. DO 317 IDN = 1, NNOD 02130 02131 APORE(IDN) = 0.02132 DO 316 J = 1, NODAR(1, IDN) 02133 IDT = NODAR (4+J, IDN)ID = TUBNOD (3, IDT) 02134 IDP = TUBNOD (4, IDT) 02135 02136 IF (ID.GT.O .AND. IDP.GT.O ) THEN 02137 BASE2 = (XS(ID) - XS(IDP)) \*\*2 + (YS(ID) - YS(IDP)) \*\*202138 BASE = SQRT (BASE2) 02139 RLEG = BASE /2. 02140 XMID = (XS(ID) + XS(IDP)) /2.02141 YMID = (YS(ID) + YS(IDP)) /2.02142 HIGHT2 = (XNOD(IDN)-XMID)\*\*2 + (YNOD(IDN)-YMID)\*\*2

| 02143    |      | HIGHT = SQRT (HIGHT2)  |
|----------|------|--|
| 02144    |      | DTHETA = ATAN (HIGHT /RLEG)  |
| 02145    |      | IF (RADI(ID)+RADI(IDP) .LE. BASE) THEN   |
| 02146    |      | APART = DIHETA $/2$ . * (RADI(ID)**2 + RADI(IDP)**2)   |
| 02147    |      | ELSE   |
| 02148    |      | RLAP = (RADI(ID) + RADI(IDP) - BASE) /2.   |
| 02140    |      | PHT1 = ACOS ((RADT(TD) - RLAP) / RADI(ID))   |
| 02140    |      | HIAP = RADT(TD) + STN (PHT1)   |
| 02150    |      | $\frac{1}{1} = \frac{1}{1} = \frac{1}$   |
| 02151    |      | $\frac{1}{2} = \frac{1}{2} $ |
| 02152    |      | AIRI = (RADI(ID) - REAR) IEER / Z.   |
| 02153    |      | APART = (DIAETA-PHII) /2. "RADI(ID) "2 + AIRI  |
| 02154    |      | $ATRI = (RADI(IDP) - RLAP) \wedge RLAP / 2.$   |
| 02155    |      | $APART = APART + (DTHETA-PHIZ)/2. AADI(IDP)^2 + ATRI$  |
| 02156    |      | ENDIF'   |
| 02157    |      | DAREA = MAX (0., HIGHT * RLEG - APART)   |
| 02158    |      | ELSE   |
| 02159    |      | IDSAV = ID   |
| 02160    |      | ID = MAX (ID, IDP)   |
| 02161    |      | IDP = MIN (IDSAV, IDP)   |
| 02162    |      | HYPOT2 = (XS(ID) - XNOD(IDN)) **2 + (YS(ID) - YNOD(IDN)) **2   |
| 02163    |      | HYPOT = MAX (1.E-10*XMAX, SQRT (HYPOT2))   |
| 02164    |      | IF (IDP/2*2 .EO. IDP) THEN   |
| 02165    |      | RLFG = ABS (YS(ID) - YNOD(IDN))  |
| 02166    |      | EI SE  |
| 02100    |      |  |
| BIOREM\$ | MAIN |  |
|          |      |  |
| 02167    |      | RLEG = ABS (XS(1D) - XNOD(1DN))  |
| 02168    |      | ENDIF  |
| 02169    |      | $HIGHT = SQRT (HYPOT2 - RLEG^{**2})$   |
| 02170    |      | DTHETA = PI $/2$ ACOS (RLEG /HYPOT)  |
| 02171    |      | IF (RADI(ID) .LE. HIGHT) THEN  |
| 02172    |      | APART = DTHETA $/2$ . *RADI(ID)**2   |
| 02173    |      | ELSE   |
| 02174    |      | RLAP = RADI(ID) - HIGHT  |
| 02175    |      | PHI = ACOS ((RADI(ID) - RLAP) / RADI(ID))  |
| 02176    |      | HLAP = RADI(ID) * SIN(PHI)   |
| 02177    |      | ATRI = (RADI(ID) - RLAP) * HLAP /2.  |
| 02178    |      | APART = $(DTHET-PHI) / 2$ , *RADI(ID)**2 + ATRI  |
| 02170    |      | FNDTF  |
| 02179    |      | DAREA - MAX (0 HIGHT * RLFG /2 - APART)  |
| 02100    |      | ENDIF  |
| 02101    |      |  |
| 02182    | 210  | APORE(IDN) = APORE(IDN) + DAREA  |
| 02183    | 310  |  |
| 02184    |      | PORSUM = PORSUM + APORE(ILIN)  |
| 02185    | 317  |  |
| 02186    |      | EPSVOID = PORSUM / (XMAX*YMAX)   |
| 02187    | С    |  |
| 02188    | C    | COMPUTE MAX NODE DIAMETERS   |
| 02189    | С    |  |
| 02190    |      | DO 318 IDN = 1, NNOD   |
| 02191    |      | XTU(IDN) = XNOD(IDN)   |
| 02192    |      | YTU(IDN) = YNOD(IDN)   |
| 02193    |      | RTU(IDN) = XMAX / 100.   |
| 02194    |      | RMIN = XMAX  |
| 02195    |      | DO 319 J = 1, NODAR(1, IDN)  |
| 02196    |      | IDT = NODAR (4+J, IDN)   |
| 02197    |      | TD = TUBNOD (3. IDT)   |
| 02100    |      | TDP = TTIBNOD (4, TDT)   |
| 02190    |      | TF (TD, GT, 0, AND, TDP, GT, 0) THEN   |
| 02199    |      | XAA(T, 1) = -2 * (XS(TD) - XS(TDP))  |
| 07.7.00  |      |  |

| 02201    |      | XAA(J,2) = -2. * (YS(ID) - YS(IDP))  |
|----------|------|--|
| 02202    |      | XAA(J,3) = -2. * (RADI(ID) - RADI(IDP))  |
| 02203    | _    | RHS(J) = RADI(ID) **2 - RADI(IDP) **2  |
| 02204    | 1    | - XS(ID) **2 + XS(IDP) **2 - YS(ID) **2 + YS(IDP) **2  |
| 02205    |      | QHS(J) = RADI(ID) - RADI(IDP)  |
| 02206    |      | DIS = SQRT ((XS(ID) - XS(IDP)) **2 + (YS(ID) - YS(IDP)) **2)   |
| 02207    |      | RTRY = DIS - RADI(ID) - RADI(IDP)  |
| 02208    |      | RMIN = MIN (RMIN, RTRY)  |
| 02209    |      | ELSE   |
| 02210    |      | XAA(J,1) = 0.  |
| 02211    |      | XAA(J,2) = 0.  |
| 02212    |      | XAA(J,3) = 0.  |
| 02213    |      | RHS(J) = 0.  |
| 02214    |      | QHS(J) = 0.  |
| 02215    |      | IDSAV = ID   |
| 02216    |      | ID = MAX (ID, IDP)   |
| 02217    |      | IDP = MIN (IDSAV, IDP)   |
| 02218    |      | IF (IDP .EQ2 .OR. IDP .EQ4) THEN   |
| 02219    |      | XAA(J,1) = 1.  |
| 02220    |      | RHS(J) = 0.  |
| 02221    |      | IF (IDP .EQ. $-2$ ) RHS(J) = XMAX  |
| 02222    |      | ELSE   |
| 02223    |      | XAA(J,2) = 1.  |
| BIOREM\$ | MAIN |  |
| 02224    |      | $\operatorname{DUC}(\mathbf{T}) = 0$   |
| 02224    |      | TE (TDP EO -3) PUC(T) - VMAY   |
| 02225    |      | $\frac{11}{100} \frac{100}{100} = 1$ |
| 02220    |      | FNIDTF   |
| 022227   | 319  |  |
| 02220    | C ST | CONTINOL   |
| 02230    | C    | WT - 1   |
| 02231    |      | WTR = 0.3  |
| 02232    |      | BTI(TDN) = 1.1 * RMTN  |
| 02233    |      | RNEW = RTI(TDW)  |
| 02234    |      | TTSK = 0   |
| 02235    |      | DO 377 TTS = 1.50  |
| 02236    |      | TTSK = TTSK + 1  |
| 02237    | C    |  |
| 02238    | •    | RTI(TON) = (1 - WTR) * RTI(TON) + WTR * RNEW   |
| 02239    |      | DETA = XAA(1,1) * XAA(2,2) - XAA(1,2) * XAA(2,1)   |
| 02240    |      | XHS(1) = RHS(1) + 2.* OHS(1) *RTI(TDN)   |
| 02241    |      | XHS(2) = RHS(2) + 2.* OHS(2) * RTI(TDN)  |
| 02242    |      | DETX = XHS(1) + XAA(2,2) - XAA(1,2) + XHS(2)   |
| 02243    |      | DETY = XAA(1,1) * XHS(2) - XHS(1) * XAA(2,1)   |
| 02244    |      | XNEW = DETX / DETA   |
| 02245    |      | YNEW = DETY / DETA   |
| 02246    |      | XTU(IDN) = (1, -WT) * XTU(IDN) + WT * XNFW   |
| 02247    |      | YTU(IDN) = (1WT) * YTU(IDN) + WT * YNEW  |
| 02248    | С    |  |
| 02249    |      | DELM = 0.  |
| 02250    |      | ERR = 0.   |
| 02251    |      | DO 376 J = 1, NODAR(1, IDN)  |
| 02252    |      | ID = NODAR(J+8, IDN)   |
| 02253    |      | IF (ID .LT. 1) GO TO 376   |
| 02254    |      | RTUF = (XS(ID) - XTU(IDN)) * 2 + (YS(ID) - YTU(IDN)) * 2   |
| 02255    |      | RXX = SQRT (RTUF) - RADI(ID)   |
| 02256    |      | RES(J) = SQRT(RTUF) - RADI(ID) - RTU(IDN)  |
| 02257    |      | ERR = MAX (ERR, ABS(RES(J)))   |
| 02258    |      | IF (ABS(RXX-RTU(IDN)) .GT. DELM) THEN  |

DELM = ABS (RXX-RTU(IDN)) 02259 RNEW = RXX 02260 ENDIF 02261 376 CONTINUE 02262 IF (ERR .LT. 1.E-6\*XMAX) GO TO 378 02263 02264 С 377 CONTINUE 02265 CONTINUE 02266 378 RTU(IDN) = MAX (RTU(IDN), 1.E-30)02267 02268 С 318 CONTINUE 02269 02270 С C--- COMPUTE EFFECTIVE TUBE DIAMETERS FOR 3-D GEOMETRY 02271 02272 С IF (IGEO .EQ. 3) THEN 02273 BBAR = 0. 02274 MSUM = 002275 ELBAR = 0.02276 DO 344 IDT = 1, NTUB 02277 IDN = TUBNOD(1, IDT)02278 IDNP = TUBNOD(2, IDT)02279 AR1 = PI \* RTU(IDN) \* 202280 BIOREM\$MAIN AR2 = PI \* RTU(IDNP) \* \* 202281 AR1 = MAX (AR1, 1.E-20) 02282 AR2 = MAX (AR2, 1.E-20)02283 TUL = TUBAR(1, IDT)02284 TUL1 = TUL /2.02285 TUL2 = TUL /2.02286 AEFF = (TUL1 \*AR1 + TUL2 \*AR2) /TUL 02287 RAD1 = SORT (AR1 /PI) 02288 RAD2 = SQRT (AR2 /PI) 02289 RK1 = MAX (RAD1\*\*2 /8., 1.E-20) 02290 RK2 = MAX (RAD2\*\*2 / 8., 1.E-20)02291 IF (RK1 .LT. RK2) THEN 02292 BOT = AEFF /TUL \* (TUL1 /AR1 + TUL2 \*RK1 /RK2 /AR2) 02293 RKEFF = RK1 /BOT 02294 ELSE 02295 BOT = AEFF /TUL \* (TUL1 \*RK2 /RK1 /AR1 + TUL2 /AR2) 02296 RKEFF = RK2 / BOT02297 ENDIF 02298 RKEFF = MAX (RKEFF, 1.E-30) 02299 DEFF = SORT (32. \*RKEFF) 02300 DPER(IDT) = SQRT (12. \*RKEFF)02301 02302 DVOL(IDT) = AEFFDDIF(IDT) = TUL /(TUL1 /AR1 + TUL2 /AR2) 02303 TUBAR(14, IDT) = 2. \*SQRT (AEFF /PI) 02304 TUBAR(15, IDT) = DDIF(IDT)02305 IF (TUBAR(9, IDT) .LT. 0) THEN 02306 DPER(IDT) = 0.02307 DVOL(IDT) = 0.02308 DDIF(IDT) = 0.02309 TUBAR(14, IDT) = 0.02310 TUBAR(15, IDT) = 0.02311 02312 ENDIF IF (TUBAR(9, IDT) .GT. 0) THEN 02313 MSUM = MSUM + 102314 BBAR = BBAR + TUBAR(14, IDT)02315 ELBAR = ELBAR + TUBAR(1, IDT)02316

~

| 02317   |      | ENDIF   |
|---------|------|---|
| 02318   | 344  | CONTINUE  |
| 02319   |      | BBAR = BBAR /MSUM   |
| 02320   |      | ELBAR = ELBAR /MSUM   |
| 02321   |      | EI SE   |
| 02322   | C    |   |
| 02322   | c    |   |
| 02323   | 0    | COMPUTE CHARACTERISTIC 2-D PASSAGE DIMENSIONS               |
| 02324   | C    |   |
| 02325   |      | 1P1K = 1  |
| 02326   |      | BBAR = 0.   |
| 02327   |      | MSUM = 0  |
| 02328   |      | ELBAR = 0.  |
| 02329   |      | DO $371 \text{ IDT} = 1$ , NTUB                             |
| 02330   |      | ID = TUBNOD(3, IDT)   |
| 02331   |      | TDP = TUBNOD(4, TDT)  |
| 02332   |      | TF (TD CT 0 AND TDP CT 0) THEN                              |
| 02322   |      | P1 - PADT(TD)   |
| 02333   |      | $\frac{1}{100}$   |
| 02334   |      | $R_2 = RADI(IDP)$   |
| 02335   |      | ELSE  |
| 02336   |      | ID = MAX (ID, IDP)  |
| 02337   |      | R1 = RADI(ID)   |
| BIOREMŞ | MAIN |   |
| 00000   |      | D) D1   |
| 02330   |      |   |
| 02339   |      |   |
| 02340   |      | DNEK = TOBAR(2, IDT)  |
| 02341   |      | RLEN = MIN (MIN (TUBAR(1, IDT), 1.999*R1), 1.999*R2)        |
| 02342   |      | CALL DELTAS (IPIK, R1, R2, DNEK, RLEN, DDIF(IDT),           |
| 02343   |      | 1 DPER(IDT), DVOL(IDT))                                     |
| 02344   |      | TUBAR(14, IDT) = DVOL(IDT)                                  |
| 02345   |      | TUBAR(15, IDT) = DDIF(IDT)                                  |
| 02346   |      | IF (TUBAR (9, IDT) .GT. 0) THEN                             |
| 02347   |      | MSUM = MSUM + 1   |
| 02348   |      | BBAR = BBAR + DVOL(TDT)                                     |
| 02349   |      | ELBAR = ELBAR + TUBAR $(1, TDT)$                            |
| 02350   |      | FNDTF   |
| 02351   | 371  |   |
| 02351   | 571  | CONTINUE /MOUN  |
| 02332   |      | ELDAR - DEAR / MOUN   |
| 02355   |      | ELBAR = ELBAR /MSUM   |
| 02354   | -    | ENDIF   |
| 02355   | C    |   |
| 02356   | C    | CYCLE OVER RELAXATION STEPS                                 |
| 02357   | С    |   |
| 02358   |      | ICONV = 1   |
| 02359   |      | DO 210 ICY = 1, NSOR  |
| 02360   |      | IGO = IGO + 1   |
| 02361   |      | ERR = 0.  |
| 02362   |      | ERRMDT = 0.   |
| 02363   | С    |   |
| 02364   | C    | COMPUTE MASS FLOW RATES AND WEIGHTS FOR EACH NODE.          |
| 02365   | Ċ    | SIGN CONVENTION FOR FLOW IS POSITIVE FOR POSITIVE FLOW FROM |
| 02366   | č    | NODE N1 TO NODE N2: PRESSURE GRADIENT IS (P2-P1)/L          |
| 02367   | č    |   |
| 02368   | C    | 1 + 1 = 1 + 1 = 1   |
| 02360   |      | VOI = 0   |
| 02303   |      | $V \cup I = U$ .  |
| 02370   |      | LOIM = 0.   |
| 02371   |      | $W_{\text{U}} = U.$   |
| 02372   |      | UPUKHU = 1.   |
| 02373   |      | IF (ICOMP.EQ.1) DPDRHO = RGAS *TMP                          |
| 02374   |      | LO 230 L = 1, NODAR(1, IDN)                                 |

| NUM = NODAR(L+1, IDN)                         |
|---|
| IDT = NODAR(L+4, IDN)                         |
| N1 = TUBNOD(1, IDT)                           |
| TUL = TUBAR(1, IDT)                           |
| TUD = TUBAR(2, IDT)                           |
| ALF = DPER(IDT) **2 / 12.                     |
| VOL = VOL + DVOL(IDT) *TUL /2.                |
| PBAR = (PRES(IDN) + PRES(NUM))/2.             |
| RHO = 1.                                      |
| IF (ICOMP.EQ.1) RHO = PBAR /RGAS /TMP         |
| RMU = 1.                                      |
| DPDL = (PRES(IDN) - PRES(NUM)) / TUL          |
| VEL = - ALF /RMU *DPDL                        |
| DOIM = DOIM + VEL *RHO *DVOL(IDT)             |
| WGTSUM = WGTSUM + ALF/RMU/TUL *RHO *DVOL(IDT) |
| ISGN = 1                                      |
| IF (NUM .NE. N1) ISGN $= -1$                  |
| TUBAR(10, IDT) = ISGN * VEL                   |
| TUBAR(11, IDT) = ISGN *VEL *RHO *DVOL(IDT)    |
| TUBAR(12, IDT) = DVOL(IDT) *TUL               |
|   |

7

۰,

BIOREM\$MAIN

| 02395 |     | TUBAR(13, IDT) = ISGN * VEL * DVOL(IDT)                    |
|-------|-----|--|
| 02396 | 230 | CONTINUE   |
| 02397 | С   |  |
| 02398 | C   | COMPUTE RESIDUAL OF GOVERNING EQUATION FOR NODE IDN        |
| 02399 | С   |  |
| 02400 |     | IF (NODAR(8, IDN).EQIDBIN .OR. NODAR(8, IDN).EQIDBOU) THEN |
| 02401 |     | DPRES(IDN) = 0.  |
| 02402 |     | ELSE   |
| 02403 |     | PDOT = (PRES(IDN) - PRESO(IDN)) / DTIM                     |
| 02404 |     | RESP = 0.0000000 *PDOT - DOIM /VOL *DPDRHO                 |
| 02405 |     | WEIGHT = WGT *VOL /DPDRHO /WGTSUM                          |
| 02406 |     | DPRES(IDN) = - WEIGHT * RESP                               |
| 02407 |     | ENDIF  |
| 02408 | С   |  |
| 02409 | C   | UPDATE PRESSURES AND NODE IDN                              |
| 02410 | С   |  |
| 02411 |     | PRES(IDN) = PRES(IDN) + DPRES(IDN)                         |
| 02412 |     | IF (NODAR $(8, IDN)$ .EQ IDBIN) PRES $(IDN) = PIN$         |
| 02413 |     | IF $(NODAR(8, IDN) . EQ IDBOU)$ PRES $(IDN) = PEX$         |
| 02414 | С   |  |
| 02415 | C   | COMPUTE LOCAL ERROR ESTIMATE AT NODE IDN                   |
| 02416 | С   |  |
| 02417 |     | TOL = RELERR * ABS(PIN-PEX) * WGT                          |
| 02418 |     | ERRIDN = ABS(DPRES(IDN)) / TOL                             |
| 02419 |     | IF (ERRIDN .GT. ERR) THEN                                  |
| 02420 |     | ERR = ERRIDN   |
| 02421 |     | IDNERR = IDN   |
| 02422 |     | ENDIF  |
| 02423 | С   |  |
| 02424 | C   | NOTE MAXIMUM NET FLOW RATE AT INTERIOR POINTS              |
| 02425 | С   |  |
| 02426 |     | IF ((NODAR(8, IDN).NEIDBIN) .AND.                          |
| 02427 |     | 1 (NODAR (8, IDN) .NE IDBOU) ) THEN                        |
| 02428 |     | IF (ABS(DOTM) .GT. ERRMDT) THEN                            |
| 02429 |     | ERRMDT = ABS(DOTM)   |
| 02430 |     | IDNMDT = IDN   |
| 02431 |     | ENDIF  |
| 02432 |     | ENDIF  |

02433 С 02434 220 CONTINUE 02435 IF (ERR.LT.1.0) GO TO 280 02436 С C--- PRINT CURRENT MAXIMUM ERROR, NODE IDENTIFIERS AND PRESSURES ---02437 02438 С 02439 IF (MOD(IGO,100).EQ.0) THEN 02440 PRINT903, IGO, IDNERR, NODAR (8, IDNERR), ERRMDT, 02441 1 ERR, PRES (IDNERR), XNOD (IDNERR), YNOD (IDNERR) 02442 ENDIF 02443 С 02444 210 CONTINUE 02445 ICONV = 0280 02446 CONTINUE 02447 С 02448 C--- SUM FLOW RATES AND POROSITY 02449 С 02450 DOTMIN = 0. 02451 DOTMOUT = 0. BIOREMȘMAIN 02452 SUMDOT = 0. 02453 DO 892 IDT = 1, NTUB 02454 С 02455 N1 = TUBNOD(1, IDT)02456 N2 = TUBNOD(2, IDT)02457 IF (NODAR(8,N1) .EQ. NODAR(8,N2)) GO TO 892 02458 С 02459 C--- GLOBAL MASS BALANCE FOR ALL INTERIOR NODES TUBE MUST HAVE ONE BOUNDARY ( < 0 ) AND ONE INTERIOR ( = 0 ) NODE 02460 С 02461 С 02462 IF (NODAR(8,N1)\*NODAR(8,N2) .EQ. 0) THEN 02463 NN = MIN (NODAR(8, N1), NODAR(8, N2))02464 NEVEN = (NN / 2) \* 202465 ISNN = 102466 IF ((XNOD(N1).GT.XNOD(N2)).AND. (NN.EQ.NEVEN)) ISNN = -1 02467 IF ((YNOD(N1).GT.YNOD(N2)) .AND. (NN.NE.NEVEN)) ISNN = -1 02468 SUMDOT = SUMDOT + ISNN \*NORM(-NN) \*TUBAR(11, IDT) 02469 ENDIF 02470 С 02471 C--- SUM MASS FLOW RATES INTO AND OUT OF X = 0 AND X = L 02472 С 02473 ISNX = 102474 IF (IDBIN .EQ. IDBEV) THEN 02475 IF (XNOD(N1) .GT. XNOD(N2)) ISNX = -1 02476 ELSE 02477 IF (YNOD(N1) .GT. YNOD(N2)) ISNX = -102478 ENDIF 02479 С 02480 IF (NODAR(8,N1).EQ.-IDBOU .OR. NODAR(8,N2).EQ.-IDBOU) THEN 02481 DOTMOUT = DOTMOUT + ISNX \*TUBAR(11, IDT) 02482 ELSEIF (NODAR(8,N1).EQ.-IDBIN .OR. NODAR(8,N2).EQ.-IDBIN) THEN 02483 DOTMIN = DOTMIN + ISNX \*TUBAR(11, IDT) 02484 ENDIF 02485 С 02486 892 CONTINUE 02487 С 02488 PBAR = (PIN + PEX) /2.02489 RHO = 1.02490 IF (ICOMP.EQ.1) RHO = PBAR /RGAS /TMP

A43

سسريوا بالدا ستسر مولد إسارم الأدار

| 02491    |     | RMU = 1.   |
|----------|-----|--|
| 02492    |     | UBAR = (ABS (DOTMIN) + ABS (DOTMOUT))/2. /YMAX /RHO  |
| 02493    |     | IF (IGEO .EQ. 3) UBAR = UBAR /ELBAR  |
| 02494    |     | VBAR = UBAR / PORO   |
| 02495    | C   | OLD UREF WAS UBAR  |
| 02496    |     | UREF = VBAR  |
| 02497    |     | PERM = UBAR *RMU *XMAX /ABS(PIN-PEX)   |
| 02498    |     | SV0 = 4. /DREF   |
| 02499    |     | TF (IGEO .EO. 3) SV0 = 6. $/DREF$  |
| 02500    |     | PFRM0 = FPSVOID**3 / (1, -FPSVOID)**2 /SV0**2  |
| 02500    |     | PERMO = EPSMC**3 / (1, -EPSMC) **2 /SV0**2   |
| 02501    |     | PKOZE - PERMO / PERM   |
| 02502    |     | $\frac{1}{10} = \frac{1}{10} $   |
| 02505    |     | $\frac{1}{100} = \frac{1}{2} \frac{1}{10} \frac{1}{2} \frac{1}{10} $                                     |
| 02504    |     |  |
| 02505    |     | $\frac{P(1)}{100} = \frac{P(1)}{100} = P($   |
| 02506    |     | $\frac{P_{\text{MM}}}{P_{\text{MM}}} = \frac{P_{\text{MM}}}{P_{\text{MM}}} = \frac{P_{\text{MM}}}{P_{$ |
| 02507    | •   | PRIN1900, (10DAR(10, 11+N10D/2), 11-1, 10)   |
| 02508    | C   |  |
| BIOREMSN | AIN |  |
| 02509    |     | SUMV = 0.  |
| 02510    |     | SUMW = 0.  |
| 02511    |     | DO 819 IDT = 1, NTUB   |
| 02512    |     | VOLT = TUBAR(1, IDT) * DVOL(IDT)   |
| 02513    |     | SUMV = SUMV + VOLT   |
| 02514    |     | SUMW = SUMW + ABS(TUBAR(10, IDT)) * VOLT   |
| 02515    | 819 | CONTINUE   |
| 02516    |     | VAVE = SUMW /SUMV  |
| 02517    |     | RHOS = 1 SUMV /XMAX /YMAX  |
| 02518    |     | PRINT900, VAVE, RHOS   |
| 02519    | С   |  |
| 02520    | -   | PRINT900, SUMDOT, DOTMIN, DOTMOUT, PERM, RKOZE   |
| 02521    | С   |  |
| 02522    | C   | SOLICIT INPUT FOR CONTINUATION   |
| 02523    | č   |  |
| 02524    | Ŭ.  | PRINT *. 'INPUT ICONT 1 = MORE CYCL 2 = NEW PRESS  |
| 02525    |     | 1  3 = NEW TUBES  4 = RESTART'   |
| 02525    |     | READ* TOOM   |
| 02520    |     | TE $(TCONTEC 1)$ THEN  |
| 02527    |     | DELIVITY TUDITI NGOR WOT' NGOR WOT   |
| 02520    |     | PRINT, INCOLUCIO, NOT PRODUCTION   |
| 02529    |     | CO = TO = 1620   |
| 02530    |     |  |
| 02531    |     | $\frac{1}{20} = \frac{1}{20}$  |
| 02532    |     | $\frac{1}{10} \frac{1020}{10} \frac{1020}{10} \frac{100}{10} \frac{100}{10$   |
| 02533    |     | ELSEIF (ICONT.EQ.4) THEN   |
| 02534    |     | GO 10 1600   |
| 02535    |     | ENDIF  |
| 02536    | С   |  |
| 02537    | С   | ADJUST TIME STEP   |
| 02538    | С   |  |
| 02539    |     | DO 832 IDT = 1, 0 *NTUB  |
| 02540    |     | N1 = TUBNOD(1, IDT)  |
| 02541    |     | N2 = TUBNOD(2, IDT)  |
| 02542    |     | TUL = TUBAR(1, IDT)  |
| 02543    |     | TUD = TUBAR(2, IDT)  |
| 02544    |     | DOTM = TUBAR(11, IDT)  |
| 02545    |     | IF (NODAR(8,N1) .EQ. NODAR(8,N2)) GO TO 832  |
| 02546    |     | IF (NODAR(8,N1).LT.0 .OR. NODAR(8,N2).LT.0) THEN   |
| 02547    |     | <pre>PRINT903, IDT, NODAR(8, N1), NODAR(8, N2), DOTM, TUBAR(3, IDT),</pre>   |
| 02548    |     | 1 TUBAR(4, IDT), TUBAR(5, IDT), TUBAR(6, IDT), PRES(N1), PRES(N2)  |
|          |     |  |

| C<br>832<br>833<br>C<br>C | ENDIF<br>CONTINUE<br>DO 833 IDT = 1, 0 *NTUB<br>N1 = TUENOD(1, IDT)<br>N2 = TUENOD(2, IDT)<br>TUL = TUBAR(1, IDT)<br>TUD = TUBAR(2, IDT)<br>DOIM = TUBAR(2, IDT)<br>IF (NODAR(8,N1) .NE. NODAR(8,N2)) GO TO 833<br>IF (NODAR(8,N1) .NE. NODAR(8,N2).LT.0) THEN<br>PRINT903, IDT, NODAR(8,N1), NODAR(8,N2), LDT.0) THEN<br>PRINT903, IDT, NODAR(8,N1), NODAR(8,N2), DOIM, TUBAR(3, IDT),<br>1 TUBAR(4, IDT), TUBAR(5, IDT), TUBAR(6, IDT), PRES(N1), PRES(N2)<br>ENDIF<br>CONTINUE<br>PLOTT PRESSURE POINTS   |
|---------------------------|--|
| \$MAIN                    |  |
| С<br>827<br>С             | <pre>IF (IPLTPR.EQ.1) THEN<br/>CALL AREA2D(XAXIS, VAXIS)<br/>CALL GRAF(XORIG, XSTP, XMAX, YORIG, YSTP, YMAX)<br/>CALL THKFRM(.030)<br/>CALL FRAME<br/>PRINT*, ISEED0<br/>PRINT*, ISEED0<br/>PRINT913, NFAR<br/>PRINT913, NNOD<br/>PRINT913, NNOD<br/>PRINT913, NNOD<br/>PRINT913, NNOD<br/>PRINT900, PIN<br/>PRINT900, PIN<br/>PRINT900, PEX<br/>PRINT900, ABSERR<br/>PRINT900, ABSERR<br/>PRINT900, ABSERR<br/>PRINT900, NCT<br/>PRINT*, '<br/>IF (ICONV.EQ.0) PRINT*, 'NO SOR CONV'<br/>PRINT*, IGO<br/>PRINT900, DERR<br/>PRINT900, DOTMOUT<br/>PRINT900, DOTMOUT<br/>PRINT900,</pre> |
| С                         | XORIG = 0.   |
|                           | 833<br>С<br>с<br>Змали<br>с<br>с   |

Sec. 7.

ž,

| 02607  |      | YORIG = $0$ .  |
|--------|------|--|
| 02608  |      | YMAX2 = 1.   |
| 02609  |      | CALL SCLPIC(1.)                                      |
| 02610  |      | IF (IPLTPF.EQ.1) THEN                                |
| 02611  |      | IF (RATXY.GT.1.0) THEN                               |
| 02612  |      | XAXIS = 7.5  |
| 02613  |      | YAXIS = 0.95775 *7.5                                 |
| 02614  |      | ELSE   |
| 02615  |      | XAXIS = 7.5 * RATXY                                  |
| 02616  |      | YAXIS = 0.95775 *7.5                                 |
| 02617  |      | FNDTF  |
| 02618  |      | XSTP = (XMAX - XORTG) / 1                            |
| 02619  |      | VSTP = (YMAX2 - YORTG) / 1.                          |
| 02620  |      | CALL AREADD (XAXTS, YAXTS)                           |
| 02620  |      | CALL GRAF (XORTG, XSTP, XMAX, YORTG, YSTP, YMAX2)    |
| 02621  |      | CALL FRAME   |
| 02022  |      |  |
| BIOREM | MAIN |  |
| 02623  |      | CALL MARKER(15)                                      |
| 02624  |      | PRINT*, ISEEDO                                       |
| 02625  |      | PRINT913, NPAR                                       |
| 02626  |      | PRINT913, NNOD                                       |
| 02627  |      | PRINT913.NTUB  |
| 02628  |      | PRINT913, NSOR                                       |
| 02629  |      | PRINT900.PTN   |
| 02630  |      | PRINT900. PEX  |
| 02631  |      | PRINT900, PORO                                       |
| 02632  |      | PRINT900. ABSERR                                     |
| 02633  |      | PRINT900, RELERB                                     |
| 02634  |      | PRINT900 WIT   |
| 02635  |      | DRINT ''   |
| 02635  |      | TE (TOONT EO 0) DRIVET IND SOR CONVI                 |
| 02030  |      | DRIMTONI TOO   |
| 02037  |      |  |
| 02030  |      |  |
| 02039  |      |  |
| 02640  |      |  |
| 02641  |      | PRIMI900, SUMDOF                                     |
| 02642  |      | 10 826 10N = 1, NNOD                                 |
| 02643  |      | XB(1) = XIVOD(1DN)                                   |
| 02644  |      | IB(I) = (PRES(IDN) - PEA) / (PIN - PEA)              |
| 02645  | 000  | CALL CORVE(XB, YB, 1, 1)                             |
| 02646  | 826  | CONTINUE   |
| 02647  |      | IF (IDBIN .EQ. IDBEV) THEN                           |
| 02648  |      | JMX = 200  |
| 02649  |      | 100 825 J = 1, JMX                                   |
| 02650  |      | $XB(J) = XMAX \wedge (J-I.) / (JMX-I.)$              |
| 02651  |      | IF (IDBIN .EQ. 2) XB(J) = I - XB(J)                  |
| 02652  |      | IF (ICOMP.EQ.0) THEN                                 |
| 02653  |      | YB(J) = 1 XB(J) / XMAX                               |
| 02654  |      | ELSE   |
| 02655  |      | YB(J) = SQRT (PIN**2 - (PIN**2 - PEX**2)*XB(J)/XMAX) |
| 02656  |      | YB(J) = (YB(J) - PEX) / (PIN - PEX)                  |
| 02657  |      | ENDIF  |
| 02658  | 825  | CONTINUE   |
| 02659  |      | CALL CURVE (XB, YB, JMX, 0)                          |
| 02660  |      | ENDIF  |
| 02661  | С    |  |
| 02662  | C    | PLOT TUBE CONNECTIONS IN X-P SPACE                   |
| 02663  | С    |  |
| 02664  |      | CALL DASH  |

| 829<br>828<br>C<br>C<br>C | DO 828 IDT = 1, NTUB<br>IF (KBLOC(IDT) .EQ. 1) THEN<br>DO 829 J = 1, 2<br>IDN = TUENOD(J,IDT)<br>XB(J) = XNOD(IDN)<br>YB(J) = (PRES(IDN)-PEX) /(PIN-PEX)<br>CALL CURVE (XB,YB,2,0)<br>ENDIF<br>CONTINUE<br>CALL RESET('DASH')<br>CALL ENDPL(0)<br>ENDIF<br>PLOTT Y-P PRESSURE PROFILE   |
|---------------------------|---|
| MAIN                      |   |
| 856                       | <pre>XORIG = 0.<br/>YORIG = 0.<br/>YMAX2 = 1.<br/>IF (IPLTPFY .EQ. 1) THEN<br/>IF (RATXY.GT.1.0) THEN<br/>XAXIS = 7.5<br/>YAXIS = 0.95775 *7.5<br/>ELSE<br/>XAXIS = 7.5 *RATXY<br/>YAXIS = 0.95775 *7.5<br/>ENDIF<br/>XSTP=(XMAX-XORIG)/1.<br/>YSTP=(YMAX2-YORIG)/1.<br/>CALL AREA2D(XAXIS,YAXIS)<br/>CALL GRAF(XORIG,XSTP,XMAX,YORIG,YSTP,YMAX2)<br/>CALL FRAME<br/>CALL MARKER(15)<br/>PRINT*, ISEED0<br/>PRINT*, ISEED0<br/>PRINT*913,NPAR<br/>PRINT913,NPAR<br/>PRINT913,NFAR<br/>PRINT913,NFAR<br/>PRINT913,NFAR<br/>PRINT913,NFAR<br/>PRINT900,PIN<br/>PRINT900,PIN<br/>PRINT900,PIN<br/>PRINT900,PIN<br/>PRINT900,RELERR<br/>PRINT900,RELERR<br/>PRINT900,COTMAUT<br/>PRINT900,COTMAUT<br/>PRINT900,COTMAUT<br/>PRINT900,SUMDOT<br/>DO 856 IDN = 1,NNOD<br/>XB(1) = YNOD(IDN)<br/>YB(1) = (PRES(IDN)-PEX) /(PIN-PEX)<br/>CALL CURVE(XE,YE,1,1)<br/>CONTINUE<br/>IF (IDBIN .NE. IDBEV) THEN</pre> |
|                           | DO 855 J = 1, JMX   |
|                           | 829<br>828<br>C<br>C<br>C<br>C<br>SMAIN   |

XB(J) = YMAX \* (J-1.) / (JMX-1.)02723 IF (IDBIN .EQ. 3) XB(J) = 1. - XB(J)/YMAX02724 IF (ICOMP.EQ.0) THEN 02725 YB(J) = 1. - XB(J)02726 02727 ELSE YB(J) = SQRT (PIN\*\*2 - (PIN\*\*2-PEX\*\*2)\*XB(J)/YMAX)02728 YB(J) = (YB(J) - PEX) / (PIN - PEX)02729 ENDIF 02730 02731 855 CONTINUE CALL CURVE (XB, YB, JMX, 0) 02732 ENDIF 02733 02734 С C--- PLOT TUBE CONNECTIONS IN Y-P SPACE ---02735 02736 С BIOREMŞMAIN CALL DASH 02737 DO 858 IDT = 1, NTUB 02738 IF (KBLOC(IDT) .EQ. 1) THEN 02739 DO 859 J = 1, 2 02740 IDN = TUBNOD(J, IDT)02741 02742 XB(J) = YNOD(IDN)YB(J) = (PRES(IDN) - PEX) / (PIN - PEX)02743 859 CALL CURVE (XB, YB, 2, 0) 02744 ENDIF 02745 CONTINUE 02746 858 02747 CALL RESET('DASH') CALL ENDPL(0) 02748 ENDIF 02749 02750 C С UPDATE VALUES 02751 02752 С DO 740 ID=1, IDMX02753 PRESO(IDN) = PRES(IDN)02754 740 02755 CONTINUE 02756 С C--- WRITE RESTART FILE ---02757 02758 С 02759 IFILE = 99PRINT\*, 'INPUT FILE NUMBER FOR WRITING RESTART FILE', IFILE 02760 02761 READ\*, IFILE IF (IFILE .GT. 0) THEN 02762 REWIND (IFILE) 02763 WRITE (IFILE, 922) NSEED, ISEED0, XMAX, YMAX, PORO, SPAC, SPACM, TOL, 02764 02765 EPSO, DIAMF, PIN, PEX 1 DO 414 I = 1, NSEED 02766 WRITE(IFILE, \*) XS(I), YS(I), HITE(I), DIAM(I), RADI(I) 02767 414 02768 WRITE(IFILE,901) NNOD 02769 DO 415 I = 1, NNOD 02770 415 WRITE(IFILE, \*) PRES(I) ENDIF 02771 С 02772 C--- TRACER PARTICLE MOTION ---02773 02774 С 02775 IF (KPAR .GT. 0) THEN XORIG=0. 02776 02777 YORIG=0. RATXY=XMAX/YMAX 02778 IF (RATXY.GT.1.0) THEN 02779 02780 XAXIS = 7.5

| 02781  |        | YAXIS=XAXIS *0.95775 /RATXY                                  |
|--------|--------|--|
| 02782  |        | ELSE   |
| 02783  |        | XAXIS = 7.5 * RATXY  |
| 02784  |        | VAXTS = XAXTS * 0.95775 / RATEY                              |
| 02785  |        | ENDIE  |
| 02705  |        | HADIF (NARW WODTO) (1  |
| 02786  |        | XSTP = (XMAX - XORIG) / 1.                                   |
| 02787  |        | YSTP=(YMAX-YORIG)/1.   |
| 02788  |        | CALL AREA2D(XAXIS, YAXIS)                                    |
| 02789  |        | CALL GRAF (XORIG, XSTP, XMAX, YORIG, YSTP, YMAX)             |
| 02790  |        | CALL FRAME   |
| 02701  | C      |  |
| 02701  | C      |  |
| 02792  |        | LCOF = ABS (UREF * DIAMF/1.000 / PECL)                       |
| 02793  |        | PECS = PECL *SDEVT /DTBAR                                    |
| BIOREM | \$MAIN |  |
|        |        |  |
| 02794  |        | DTIM1 = ABS (XMAX /VBAR *DTSTR)                              |
| 02795  |        | DTIM2 = (XMAX * DTSTR) * *2 / DCOF /2.                       |
| 02796  |        | DTIM = 1. / (1. / DTIM1 + 1. / DTIM2)                        |
| 02797  |        | DTTM = DTTM1   |
| 02700  | C      |  |
| 02790  | C      |  |
| 02/99  |        | PRINT981, METH, IMETH  |
| 02800  |        | PRINT980, 'DIAM', DIAMF                                      |
| 02801  |        | PRINT980, 'DREF', DREF                                       |
| 02802  |        | PRINT980, 'UBAR', UBAR                                       |
| 02803  |        | PRINT980, 'UREF', UREF                                       |
| 02804  |        |  |
| 02004  |        |  |
| 02005  |        | PRINI960, VAVE, VAVE   |
| 02806  |        | PRINT980, PERM', PERM  |
| 02807  |        | PRINT980, 'SV00', SV000                                      |
| 02808  |        | PRINT980, 'RKOZ', RKOZE                                      |
| 02809  |        | PRINT980, 'EDSK', EDSK                                       |
| 02810  |        | PRINT980 'EPSO' EPSO   |
| 02811  |        | DDTNTOQO IDODO DODO  |
| 02011  |        | PRINT980, FORU, FORU   |
| 02012  |        | PRINT980, EPSV, EPSVOID                                      |
| 02813  |        | PRINI980, 'EPSM', EPSMC                                      |
| 02814  |        | PRINT980, 'RHOS', RHOS                                       |
| 02815  |        | PRINT980, 'DTIM', DTIM                                       |
| 02816  |        | PRINT980, 'DTST', DTSTR                                      |
| 02817  |        | PRINT980, 'DCOF', DCOF                                       |
| 02818  |        | PRINT980 'DECT.' DECT.                                       |
| 02010  |        | DETAINIDOU, FECH, FECH                                       |
| 02019  |        | PRINT980, PECS, PECS   |
| 02820  |        | PRINT980, "TBAR", DTBAR                                      |
| 02821  |        | PRINT980, 'BBAR', BBAR                                       |
| 02822  |        | PRINT980, 'TSIG', SDEVT                                      |
| 02823  |        | TSTAR = DCOF *DTIM / DTBAR**2                                |
| 02824  |        | PRINT980. 'TSTR', TSTAR                                      |
| 02825  |        | TE (TETAR IT 1 0) DETATION INCOME TO CMATTI                  |
| 02826  | C      | II (ISIAC.DI.I.O) FAINISOU, ISIK IOO SMALL                   |
| 02020  | C      |  |
| 02827  |        | CALL MARKER (16)   |
| 02828  |        | CALL BLREC(-0.2*XAXIS,-0.2*YAXIS,0.2*XAXIS,1.4*YAXIS,0)      |
| 02829  |        | CALL BLREC(-0.2*XAXIS,-0.2*YAXIS,1.4*XAXIS,0.2*YAXIS,0)      |
| 02830  |        | CALL BLREC( 1.0*XAXIS, -0.2*YAXIS, 0.2*XAXIS, 1.4*YAXIS, 0)  |
| 02831  |        | CALL BLREC (-0.2*XAXIS, 1.0*YAXIS, 1.4*XAXIS, 0.2*VAVIS, 0.) |
| 02832  |        | DO 1381 TD = 1. NSEED  |
| 02833  |        | XB(1) - YS(TD)   |
| 02033  |        | VD(1) - NO(TD)   |
| 02034  |        |  |
| 02835  |        | FAC= 2.* RADI(ID)/XMAX *XAXIS/0.082                          |
| 02836  |        | CALL SCLPIC (FAC)  |
| 02837  |        | CALL CURVE(XB,YB,1,1)  |
| 02838  | 1381   | CONTINUE   |
|        |        |  |

.

A49

 $\gamma_{\Delta}$ 

02839 CALL SCLPIC(1.) CALL RESET('BLSYM') 02840 02841 CALL SCLPIC(1.0) 02842 С 02843 DO 1369 IFR = 1, 100 02844 1369 FREQ(IFR) = 0.02845 С 02846 DO 1707 IDT = 1, NTUB 02847 1707 JBLOK(IDT) = 002848 С ASUM = 0.02849 02850 DO 1374 I = 1, NNBC (IDBIN) BIOREM\$MAIN IDN = NBID(IDBIN, I) 02851 IDT = TBID(IDBIN, I) 02852 02853 ASUM = ASUM + KBLOC(IDT) \*TUBAR(15, IDT)02854 1374 CONTINUE 02855 С 02856 VTBAR = 0. XBAR = 0.02857 02858 KPSUM = 002859 TORSUM = 0. ARTBAR = 0.02860 02861 DO 1310 J = 1, KPAR 02862 С C--- GENERATE NEW PARTICLE AND PLACE AT ENTRANCE NODE; PARTICLES ARE 02863 RANDOMLY DISTRIBUTED BETWEEN ENTRANCE NODES IN PROPORTION TO С 02864 02865 С INLET TUBE FLOW RATES. 02866 С 02867 1315 CONTINUE DPART =  $-100 \times \text{SPACM} / 10$ . 02868 XXX = RAN(ISED)02869 02870 XTRM = (4.\*XXX-2.)/3.02871 ZZZ = XTRMDO 1382 I = 1, 100 02872  $ZZZ = ZZZ^{**3} / 3. + XTRM$ 02873 02874 RESID = ZZZ\*\*3 - 3.\*ZZZ + 3.\*XTRM 02875 IF (ABS(RESID) .LT. 1.E-4) GO TO 1383 1382 02876 CONTINUE WRITE(17,\*) 'ZZZ DID NOT CONVERGE', XXX, ZZZ, RESID 02877 02878 1383 CONTINUE 02879 С 02880 C--- OPTION FOR UNIFORM INJECTION ---02881 С 02882 IF (IUNI .EQ. 1) THEN 02883 ZZZ = (2.\*XXX - 1.)02884 ENDIF C 02885 02886 GGG = (ZZZ + 1.) /2.02887 С 02888 C--- SUM ENTRANCE FLOWS TO PICK INLET TUBE ---02889 С XXX = RAN(ISED)02890 02891 XXX = EDGE + (1.-2.\*EDGE) \*XXX02892 IF (XXX0 .LT. 0.) THEN 02893 XXX = -XXX0ENDIF 02894 02895 FLSUM = 0. DO 1364 I = 1, NNBC (IDBIN) 02896

| 02897<br>02898<br>02899<br>02900<br>02901<br>02902<br>02903<br>02904<br>02905<br>02906<br>02907   | 1364        | IDN = NBID(IDBIN,I)<br>IDT = TBID(IDBIN,I)<br>DFLUX = ABS(TUBAR(11,IDT)) /DOIMIN<br>DFLUX = DFLUX + 0.00000 *TUBAR(15,IDT) /ASUM /PECL<br>FLSUM = FLSUM + DFLUX /(1. + 0.00000 /PECL)<br>IF (FLSUM .GT. XXX) THEN<br>INOD = IDN<br>GO TO 1365<br>ENDIF<br>CONTINUE<br>WRITE(17,*) 'CANT FIND ENTRANCE NODE'  |
|---|-------------|--|
| BIOREMŞI  | MAIN        |  |
| 02908<br>02909<br>02910<br>02911<br>02912<br>02913<br>02914   | C<br>1365   | TREM = DTIM<br>IDON = 0<br>TORT = 0.<br>XYNPAR(4,J) = YNOD(INOD)<br>CALL THKCRV(.020)  |
| 02914<br>02915<br>02916   | С<br>С<br>С | STEP THROUGH TUBE SET ALONG PARTICLE PATH  |
| 02917<br>02918<br>02919   | с           | DO 1320 ISTP = 1, 10000<br>IF (IDON .EQ. 1) GO TO 1321   |
| 02919<br>02920<br>02921<br>02922<br>02923<br>02924<br>02925<br>02926<br>02927<br>02928<br>02927<br>02930<br>02931<br>02932<br>02933<br>02934<br>02935<br>02936<br>02937<br>02938<br>02939<br>02940<br>02941<br>02942<br>02943<br>02944<br>02945 | с<br>с<br>с | <pre>IDENTIFY TUBES HAVING (POSITIVE) FLOW OUT OF NODE IDNOD<br/>AND SUM TUBE CROSS SECTION AREAS<br/>IF (ISTP .EQ. 1) GO TO 1363<br/>IDTIN = IDT<br/>DIFSUM = 0.<br/>FLOSUM = 0.<br/>NCH = NODAR(1,INOD)<br/>NFIN = 0<br/>NFOUT = 0<br/>DO 1330 JNOD = 1, NCH<br/>IDT = NODAR(4+JNOD,INOD)<br/>ADIF = 1.00000 *TUBAR(15,IDT)<br/>QFLO = TUBAR(13,IDT)<br/>DIFSUM = DIFSUM + ADIF *DCOF /TUBAR(1,IDT)<br/>ISN = 1<br/>IF (TUBNOD(1,IDT) .NE. INOD) ISN = -1<br/>FLOW = ISN *QFLO<br/>IF (FLOW .GT. 0.0) THEN<br/>NFOUT = NFOUT + 1<br/>FLOSUM = FLOSUM + FLOW<br/>FLO (JNOD) = FLOW<br/>ELSE<br/>NFIN = NFIN + 1<br/>FLO (JNOD) = 0.<br/>ENDIF</pre> |
| 02946<br>02947<br>02948   | 1330        | CONTINUE<br>FLOSUM = MAX (FLOSUM, 1.E-30)<br>DIFSUM = MAX (DIFSUM, 1.E-30)   |
| 02949<br>02950<br>02951   | с<br>с<br>с | IDENDITFY TUBE/NODE SEQUENCE NUMBER OF INFLOW TUBE   |
| 02952<br>02953<br>02954   |             | QFLOIN = MIN (ABS (TUBAR(13, IDTIN)), FLOSUM) /FLOSUM<br>DO 1398 JNOD = 1, NCH<br>IF (NODAR(4+JNOD, INOD) .EQ. IDTIN) THEN   |

02955 JNODIN = JNOD 02956 GO TO 1393 02957 ENDIF 1398 CONTINUE 02958 02959 1393 CONTINUE 02960 С C--- MIX JUNCTION STREAMLINES ---02961 02962 С IF (IMIX .EQ. 1) THEN 02963 02964 ZZZ = 2. \*RAN(ISED) - 1. BIOREM\$MAIN 02965 ENDIF 02966 С 02967 C--- SELECT TUBE BASED ON RELATIVE FLUXES ---02968 С GGG = (ZZZ + 1.) /2.02969 02970 FLSUM = 0.DO 1335 JCNT = 1, NCH 02971 02972 JNOD = MOD (JNODIN+JCNT-1, NCH) + 102973 IDTJ = NODAR(4+JNOD, INOD)02974 DFLOW = FLO(JNOD) /FLOSUM 02975 FLSUM = FLSUM + DFLOW 02976 IF (FLSUM .GE. GGG) THEN 02977 IDT = IDTJ02978 JCOUT = JCNT02979 QFLOOUT = MIN (ABS (TUBAR(13, IDT)), FLOSUM) /FLOSUM FLSUM = FLSUM - DFLOW 02980 02981 GO TO 1336 02982 ENDIF 02983 1335 CONTINUE WRITE(17,\*) 'CANT FIND NEW TUBE' 02984 WRITE(17,\*) XXX, FLSUM 02985 02986 IDON = 102987 GO TO 1321 02988 1336 CONTINUE 02989 С C--- COMPUTE NEW GGG AND ZZZ POSITION FOR NO MIXING ---02990 02991 С IF (IMIX .NE. 1) THEN 02992 IF (NFIN.EQ.1 .AND. NFOUT.EQ.1) THEN 02993 02994 GGG = GGG02995 ELSEIF (NFIN.EQ.1 .AND. NFOUT.EQ.2) THEN 02996 GGG = (GGG - FLSUM) /QFLOOUT ELSEIF (NFIN.EQ.2 .AND. NFOUT.EQ.1) THEN 02997 02998 IF (JCOUT .EQ. 1) THEN 02999 GGG = QFLOIN \*GGG ELSEIF (JCOUT .EQ. 2) THEN QFLOIN2 = 1. - QFLOIN 03000 03001 GGG = QFLOIN \*GGG + QFLOIN2 03002 03003 FLSE 03004 PRINT\*, 'JCOUT IS NOT 1 OR 2', JCOUT 03005 WRITE(17,\*) 'JCOUT IS NOT 1 OR 2', JCOUT 03006 ENDIF 03007 ELSE. 03008 PRINT\*, 'NUMBER OF TUBES IS WRONG' 03009 WRITE(17,\*) 'NUMBER OF TUBES IS WRONG' 03010 ENDIF 03011 ENDIF 03012 С

| 03013<br>03014<br>03015<br>03016<br>03017<br>03018<br>03019<br>03020<br>03021  | 1<br>C<br>1363    | <pre>IF (GGG .GT. 1.0 .OR. GGG .LT. 0.) THEN PRINT*, 'GGG OUT OF RANGE', GGG WRITE(17,*) 'GGG OUT OF RANGE', GGG, NFIN, NFOUT, QFLOIN,</pre>  |
|--|-------------------|---|
| BIOREM\$   | MAIN              |   |
| 03022<br>03023<br>03024<br>03025<br>03026<br>03027<br>03028<br>03029<br>03030<br>03031   | C<br>C START<br>C | JELOK(IDT) = 1<br>JNOD = TUENOD(1,IDT)<br>IF (JNOD .EQ. INOD) JNOD = TUENOD(2,IDT)<br>ISN = 1<br>IF (TUENOD(1,IDT) .NE. INOD) ISN = - 1<br>VEL = ISN *TUEAR(10,IDT)<br>VEL = SIGN (MAX (ABS (VEL), 1.E-30), VEL)<br>MOVE PARTICLE   |
| 03032<br>03033<br>03034<br>03035<br>03036<br>03037<br>03038<br>03039<br>03040<br>03041   | C                 | <pre>ISTIK = 0<br/>IMOV = 0<br/>IEXIT = 0<br/>DYDS = (YNOD(JNOD)-YNOD(INOD)) /TUBAR(1,IDT)<br/>DXDS = (XNOD(JNOD)-XNOD(INOD)) /TUBAR(1,IDT)<br/>XB(1) = XNOD(INOD) + 1.000 *TUBAR(7,IDT)<br/>YB(1) = YNOD(INOD) + 1.000 *TUBAR(8,IDT)<br/>XB(2) = XB(1)<br/>YB(2) = YB(1)</pre>   |
| 03042<br>03043<br>03044<br>03045<br>03046<br>03047<br>03048<br>03049<br>03050<br>03051<br>03052<br>03053<br>03054<br>03055<br>03056<br>03057 | C CHECK<br>C      | <pre>FOR TUBE BLOCKAGE IF (DPART .GT. TUBAR(2,IDT)) THEN ISTIK = 1 IDON = 1 XB(2) = XNOD(INOD) + DXDS *TUBAR(1,IDT) /2. YB(2) = YNOD(INOD) + DYDS *TUBAR(1,IDT) /2. ELSE IARIV = 0 DELTIMF = TUBAR(1,IDT) /ABS (VEL) /100. DIFFL = TUBAR(14,IDT) DELTIMD = (DIFFL /(20.*PI/3.))**2 /DCOF /2. DELTIM = MIN (DELTIMF, DELTIMD) IF (PECL .GT. 9.E9) THEN DELTIM = TUBAR(1,IDT) /ABS (VEL) /0.9 ENDIF</pre> |
| 03058<br>03059<br>03060<br>03061   | c<br>c            | SSS = 0.<br>ZZZIN = ZZZ   |
| 03062<br>03063<br>03064<br>03065<br>03066<br>03067<br>03068<br>03069<br>03069<br>03070   | С ТАКЕ<br>С       | INTERMEDIATE STEPS ALONG TUBE LENGTH<br>TSUM = 0.<br>BSUM = 0.<br>DO 1340 KSTP = 1, 1000000<br>IF (IARIV .NE. 0) GO TO 1341<br>IF (IDON .EQ. 1) GO TO 1341<br>IF (DELTIM .GT. TREM) THEN<br>DELTIM = MAX (TREM, 1.E-30)   |

.:

A53

53

1.73

03071 IDON = 103072 FNDTF 03073 1343 CONTINUE 03074 C C--- COMPUTE LOCAL APERTURE ---03075 03076 С 03077 HHH = 0.03078 IF (KMETH .EQ. 2) THEN BIOREMSMAIN 03079 ID = TUBNOD(3, IDT)03080 HH1 = 0.03081 IF (ID .GE. 1) THEN TRM = RADI(ID) \*\*2 - (SSS-TUBAR(1, IDT)/2.) \*\*203082 TRM = MAX (TRM, 1.E-30)03083 03084 HH1 = RADI(ID) + TUBAR(2, IDT)/2. - SQRT(TRM)03085 ENDIF ID = TUBNOD(4, IDT)03086 03087 HH2 = 0.03088 IF (ID .GE. 1) THEN TRM = RADI(ID) \*\*2 - (SSS-TUBAR(1, IDT)/2.) \*\*203089 03090 TRM = MAX (TRM, 1.E-30) 03091 HH2 = RADI(ID) + TUBAR(2, IDT)/2. - SQRT(TRM)03092 ENDIF 03093 HHH = MAX (HH1+HH2, 1.E-30)03094 ENDIF 03095 С C--- TAKE ADVECTIVE STEP ----03096 03097 С 03098 IF (IPARA .EQ. 1) THEN 03099 VLOC = 1.5 \* VEL \* (1. - ZZZ\*\*2)03100 ELSE VLOC = VEL 03101 03102 ENDIF 03103 IF (KMETH .EQ. 2) VLOC = VLOC \*TUBAR(14, IDT) /HHH 03104 DELSADV = VLOC \*DELTIM 03105 DELS = DELSADV 03106 IF (SSS+DELS .GT. TUBAR(1, IDT)) THEN 03107 LARIV = 103108 DELS = TUBAR(1, IDT) - SSS03109 DELTIM = ABS (DELS /VLOC) IF (NODAR (8, JNOD) .EQ. - IDBOU) THEN 03110 IEXIT = 103111 03112 IDON = 103113 ENDIF 03114 ENDIF 03115 BSUM = BSUM + HHH \*DELS 03116 TSUM = TSUM + DELTIM 03117 03118 С C--- TAKE DIFFUSIVE STEP ----03119 03120 С 03121 IDIR = 103122 IF (RAN(ISED) .LT. 0.5) IDIR = -103123 DELSDIF = IDIR \*SQRT (2. \*DCOF \*DELTIM) 03124 DELZ = DELSDIF / (TUBAR(14, IDT) /2.) 03125 IF (KMETH .EQ. 2) DELZ = DELZ \*TUBAR(14, IDT) /HHH 03126 IF (ZZZ+DELZ .GT. 1.0) THEN DELZ = 2.\*(1.-ZZZ) - DELZ03127 ELSEIF (ZZZ+DELZ .LT. -1.0) THEN 03128

•

э

03129 DELZ = -2.\*(1.+ZZZ) - DELZ03130 ENDIF 03131 IF (PECL .GT. 9.E9) THEN DELZ = 0.03132 ENDIF 03133 03134 C 03135 C--- UPDATE POSITION AND TIME ---BIOREMSMAIN 03136 С SSS = SSS + DELS03137 03138 ZZZ = ZZZ + DELZ03139 IF (IRANS .EQ. 1) ZZZ = (2. \*RAN(ISED) - 1.)03140 TREM = TREM - DELTIM 03141 C 1340 CONTINUE 03142 ENDIF 03143 03144 1341 XB(2) = XB(1) + DXDS \*SSSYB(2) = YB(1) + DYDS \*SSS03145 03146 IF (KMETH .EQ. 2) THEN DELZ = ZZZ \*HHH /2. 03147 XB(2) = XB(2) - DELZ \* DYDS03148 YB(2) = YB(2) + DELZ \* DXDS03149 03150 ENDIF 03151 TORT = TORT + SSS03152 XYNPAR(1,J) = XB(2)03153 XYNPAR(2,J) = YB(2)03154 XYNPAR(3,J) = INODIF (IARIV .EQ. 1) THEN 03155 03156 INOD = JNODELSEIF (IARIV .EQ. -1) THEN 03157 03158 INOD = INOD03159 ENDIF 03160 С 03161 BSUM = BSUM /TUBAR(1, IDT) WRITE(17,900) TUBAR(2, IDT), TUBAR(14, IDT), BSUM, TSUM, 03162 03163 1 TUBAR(1, IDT)/VEL 03164 С C--- RESET STREAMLINE TO ENTRANCE VALUE ---03165 03166 С 03167 IF (IZZZ .EQ. 1) THEN 03168 ZZZ = ZZZINENDIF 03169 03170 С C--- PLOTT TRACER PARTICLE TRAJECTORY ---03171 03172 С 03173 IF (IPLTTR .EQ. 1) THEN 03174 CALL CURVE(XB, YB, 2, 0) 03175 ENDIF 03176 С 03177 1320 CONTINUE 03178 С 03179 C--- PLOTT FINAL PARTICLE POSITION ----03180 С CONTINUE 03181 1321 IF (IEXIT .EQ. 0) THEN 03182 03183 CALL MARKER(15) 03184 CALL CURVE (XB(2), YB(2), 1, -1)ELSEIF (IEXIT .EQ. 1) THEN 03185 ARRTIM(J) = DTIM - TREM 03186

A55

ú

L

| 03187<br>03188 |      | ARTBAR = ARTBAR + ARRTIM(J)<br>CALL MARKER(13)  |
|----------------|------|---|
| 03189          |      | XB(2) = 0.99 * XMAX   |
| 03190          |      | CALL CURVE $(XB(2), YB(2), 1, -1)$  |
| 03191          |      | ELSEIF (IEXIT .EQ1) THEN  |
| 03192          |      | CALL MARKER (13)  |
| BIOREMŞI       | MAIN |   |
| 03193          |      | XB(2) = 0.01 * XMAX   |
| 03194          |      | CALL CURVE(XB(2),YB(2),1,-1)  |
| 03195          | _    | ENDIF   |
| 03196          | C    |   |
| 03197          |      | 10RT = (10RT / MAX (1.E-10, XB(2))) **2   |
| 03198          | ~    | TORSUM = TORSUM + TORT  |
| 03199          | C    | CHARLE DAMAGE DAGAMETAN IN PREVIDENT DECEMBERS  |
| 03200          | C    | STORE PATICLE POSITION IN FREQUNCY DISTRIBUTION   |
| 03201          | C    | AND SUM AVERAGE POSITION  |
| 03202          | C    |   |
| 03203          |      | RPSOM = RPSOM + 1 $TED = NTNT (VD(2) - (VD(2) + 1)$   |
| 03204          |      | ITR = NINI (AD(2) / AMAA "33) + 1   |
| 03205          |      | rcQ(Irr) = rcQ(Irr) + I.  |
| 03200          |      | $\Delta DAX = \Delta DAX + \Delta D(2)$ $\Delta T = \Delta DAX + \Delta D(2)$ $\Delta T = \Delta T = \Delta T = \Delta D = \Delta T = \Delta D = \Delta T = \Delta D $ |
| 03207          |      | VIIACE(RESOM) = XD(2) / DIEN  |
| 03200          | C    | VIEAL - VIEAL + VIEACE (ALSON)  |
| 03205          | 1310 | CONTENTIF   |
| 03210          | 1910 | CALL BESET ('DASH')   |
| 03212          |      | CALL RESET ('THKCRV')   |
| 03213          |      | ARTBAR = MAX (1.E-10, ARTBAR / KPAR)  |
| 03214          |      | XBAR = XBAR / KPAR  |
| 03215          |      | VTBAR = VTBAR /KPAR   |
| 03216          |      | TORT =. TORSUM /KPAR  |
| 03217          | С    |   |
| 03218          | C    | COMPUTE STANDARD DEVIATIONS   |
| 03219          | С    |   |
| 03220          |      | SDEV = 0.   |
| 03221          |      | STIM = 0.   |
| 03222          |      | SIGV = 0.   |
| 03223          |      | SDTR = 0.   |
| 03224          |      | DO 1348 IPAR = 1, KPAR  |
| 03225          |      | SDEV = SDEV + (XBAR-XYNPAR(1, IPAR)) **2  |
| 03226          |      | SDTR = SDTR + (XYNPAR(2, IPAR) - XYNPAR(4, IPAR)) **2   |
| 03227          |      | STIM = STIM + (ARTBAR - ARRTIM(IPAR))**2  |
| 03228          | 1240 | SIGV = SIGV + (VTRACE(1PAR) - VTBAR) **2  |
| 03229          | 1348 | CONTINUE  |
| 03230          |      | SDEV = SQRI' (SDEV / KPAR)  |
| 03231          |      | SDIR = SQRT (SDIR / RPAR)   |
| 03232          |      | STIM = SQRT (STIM / RPAR)   |
| 03233          | ~    | SIGV = SQRT (SIGV / RPAR)   |
| 03234          | C    |   |
| 03235          | C    | COMPUTE CONDERTIVE DISTRIBUTION   |
| 03230          | C    | DO 1391 TER - 1 100   |
| 03237          |      | XB(TER) - (TER - 5) /100 *YMAY  |
| 03239          |      | YB(TFR) = 1.  |
| 03240          |      | DO 1392  JFR = 1.  TFR  |
| 03241          |      | YB(IFR) = YB(IFR) - FRFO(JFR) / KPAR  |
| 03242          | 1392 | CONTINUE  |
| 03243          |      | IF (YB(IFR) .GE. 0.5) $X50 = XB(IFR)$   |
| 03244          |      | YB(TFR) = YB(TFR) * YMAX  |

۰.

| 03245 139<br>03246<br>03247 C |      | CONTINUE<br>X50 = MAX (1.E-10, X50)  |  |  |  |  |
|-------------------------------|------|--|--|--|--|--|
| 03247                         | C    |  |  |  |  |  |
| 03248                         |      | CALL THKCRV(.050)  |  |  |  |  |
| 05215                         |      |  |  |  |  |  |
| BIOREM                        | MAIN |  |  |  |  |  |
| 03250                         |      | CALL CURVE (XB, YB, 100, 0)  |  |  |  |  |
| 03251                         |      | ENDIF  |  |  |  |  |
| 03252                         | С    |  |  |  |  |  |
| 03253                         | C    | COMPUTE APPARENT DIFFUSIVITY AND PLOTT CORRESPONDING ERROR FUNC              |  |  |  |  |
| 03255                         | C    | VIDAR - XBAR /DITM   |  |  |  |  |
| 03256                         |      | PEVP = PECL *VPAR /VBAR  |  |  |  |  |
| 03257                         |      | PEBB = PEVP *BBAR /DIAMF   |  |  |  |  |
| 03258                         |      | DIFF = SDEV**2 /2. /XBAR   |  |  |  |  |
| 03259                         |      | DIF2 = SDEV**2 /2. /DTIM   |  |  |  |  |
| 03260                         |      | DIFT = SDIR**2 /2. /DIM<br>(IDPT = (DOPO + OUTA) + +2 /2 / APUPAP            |  |  |  |  |
| 03261                         |      | DIF3 = $(OREF / PORO ~SIIM)^{2} / 2 \cdot / ARIDAR$                          |  |  |  |  |
| 03263                         |      | DIFF0 = MAX (1.E-20, SDEVT/DTBAR*DIAMF)                                      |  |  |  |  |
| 03264                         |      | DCON = DIFF / DIFF0  |  |  |  |  |
| 03265                         |      | CON1 = PORO *DIF2 /DCOF /(2.*PECL)   |  |  |  |  |
| 03266                         |      | CONS = PORO *DIF2 /DCOF / (2.*PECS)  |  |  |  |  |
| 03267                         |      | DO 1367 IFR = 1, 101<br>VD(TED) = (TED-1) (100 *VMN)                         |  |  |  |  |
| 03268                         |      | ETA = (XBAR-XB(IFR)) / 2. /SORT(XBAR*DIFF)                                   |  |  |  |  |
| 03270                         |      | YB(IFR) = 0.5 * (1. + ERF(ETA)) * YMAX                                       |  |  |  |  |
| 03271                         | 1367 | CONTINUE   |  |  |  |  |
| 03272                         |      | IF (IMICF .NE. 1) THEN   |  |  |  |  |
| 03273                         |      | CALL DASH  |  |  |  |  |
| 03274                         |      | CALL CORVE(AB, 16, 100, 0)<br>CALL RESET ('DASH')                            |  |  |  |  |
| 03276                         |      | ENDIF  |  |  |  |  |
| 03277                         | С    |  |  |  |  |  |
| 03278                         | C    | PLOTT USED TUBES FOR MICRO-FINGERS   |  |  |  |  |
| 03279                         | С    |  |  |  |  |  |
| 03280                         |      | DO 1703 TDT = 1. NTUB  |  |  |  |  |
| 03282                         |      | XB(1) = TUBAR(3, IDT)  |  |  |  |  |
| 03283                         |      | YB(1) = TUBAR(4, IDT)  |  |  |  |  |
| 03284                         |      | XB(2) = TUBAR(5, IDT)  |  |  |  |  |
| 03285                         |      | YB(2) = TUBAR(0, 1DT) $HTD = MTM (TTERR 2 (2, 1DT) = 3 * (TTERR 2 (1, 1DT))$ |  |  |  |  |
| 03280                         |      | WID = 0.98  MAX (WID, 0.)  |  |  |  |  |
| 03288                         |      | WID = WID /XMAX *XAXIS   |  |  |  |  |
| 03289                         |      | WID = MAX (WID, .001)  |  |  |  |  |
| 03290                         |      | CALL THKCRV (WID)  |  |  |  |  |
| 03291                         | 1703 | IF (JBLOK(IDF) .EQ. I) CALL CURVE(XB,YB,Z,U)                                 |  |  |  |  |
| 03292                         | 1/05 | ENDIF  |  |  |  |  |
| 03294                         | с    |  |  |  |  |  |
| 03295                         |      | CALL RESET ('THKCRV')  |  |  |  |  |
| 03296                         |      | CALL SCLPIC (1.0)  |  |  |  |  |
| 03297                         |      | $\frac{AB(1)}{VB(1)} = 0.$   |  |  |  |  |
| 03299                         |      | CALL SCLPIC(0.0001)  |  |  |  |  |
| 03300                         |      | CALL CURVE (XB, YB, 1, 1)  |  |  |  |  |
| 03301                         |      | CALL ENDGR(0)  |  |  |  |  |
| 03302                         |      | PRINT980, 'PECL', PECL   |  |  |  |  |

۳,

ζ

A57

~

| 03303     |      | PRINT980, 'PEVP', PEVP             |
|-----------|------|------------------------------------|
| 03304     |      | PRINT980, 'PEBB', PEBB             |
| 03305     |      | PRINT980, 'DIFL', DIF2/DCOF        |
| 03306     |      | PRINT980, 'DIFT', DIFT/DCOF        |
|           |      |                                    |
| BIOREM\$N | AIN  |                                    |
| 03307     |      | PRINT980, 'TORT', TORT             |
| 03308     |      | PRTNT980.                          |
| 03309     |      | PRINT980, 'X50 ', X50              |
| 03310     |      | PRINT'980, 'VPAR', VPAR            |
| 03311     |      | PRINT'980, 'XBAR', XBAR            |
| 03312     |      | PRINT'980, 'DCON', DCON            |
| 03313     |      | PRINT'980, 'CONS', CONS            |
| 03314     |      | PRINT980, 'CON1', CON1             |
| 03315     |      | PRINT'980, 'SDEV', SDEV            |
| 03316     |      | PRINT980, 'DIFF', DIFF             |
| 03317     |      | PRINT980, 'DIF2', DIF2             |
| 03318     |      | PRINT980, 'DIF3', DIF3             |
| 03319     |      | PRINT'980, 'DRAT', DIFF/DOOF       |
| 03320     |      | PRINT980, 'DRA4', DIF4/DOF         |
| 03321     |      | PRINT'980, 'SIGV', SIGV            |
| 03322     |      | PRINT980, 'SVRA', SIGV/VTBAR       |
| 03323     |      | PRINT 980. '                       |
| 03324     |      | CALL FNDPL(0)                      |
| 03325     |      | FNDIF                              |
| 03326     | С    |                                    |
| 03327     | 710  | CONTINUE                           |
| 03328     | 830  | CONTINUE                           |
| 03329     | С    |                                    |
| 03330     | -    | PRINT*, 'INPUT ICONT 1 = NEW PRESS |
| 03331     |      | 1  2 = NEW TUBES  3 = RESTART'     |
| 03332     |      | READ*, ICONT                       |
| 03333     |      | IF (ICONT.EO.1) THEN               |
| 03334     |      | GO TO 1620                         |
| 03335     |      | ELSEIF (ICONT.EQ.2) THEN           |
| 03336     |      | GO TO 1610                         |
| 03337     |      | ELSEIF (ICONT.EQ.3) THEN           |
| 03338     |      | GO TO 1600                         |
| 03339     |      | ENDIF                              |
| 03340     | С    |                                    |
| 03341     | 840  | CALL DONEPL                        |
| 03342     | С    |                                    |
| 03343     | 988  | FORMAT('1', A, 2X, 1PE11.3)        |
| 03344     | 980  | FORMAT(1X,A,2X,1PE11.3)            |
| 03345     | 981  | FORMAT(1X, A, 2X, I11, 1PE11.3)    |
| 03346     | 900  | FORMAT(11(1PE12.4))                |
| 03347     | 990  | FORMAT(12(1PE11.3))                |
| 03348     | 901  | FORMAT(I10,10(1PE12.4))            |
| 03349     | 902  | FORMAT(2(I10),8(1PE12.4))          |
| 03350     | 922  | FORMAT(2(I10),10(1PE11.3))         |
| 03351     | 912  | FORMAT(2(15),8(1PE12.4))           |
| 03352     | 903  | FORMAT(3(I10),8(1PE12.4))          |
| 03353     | 913  | FORMAT(3(16),8(1PE12.4))           |
| 03354     | 904  | FORMAT(4(I10),8(1PE12.4))          |
| 03355     | 905  | FORMAT(5(110),8(1PE12.4))          |
| 03356     | 1900 | FORMAT(10(1PE12.4))                |
| 03357     | 1901 | FORMAT(14,10(1PE12.4))             |
| 03358     | 1902 | FORMAT(2(14), $8(1PE12.4)$ )       |
| 03359     | 1912 | FORMAT(2(15), $8(1PE12.4))$        |
| 03360     | 1903 | FORMAT(3(14), 8(1PE12.4))          |

, <del>-</del>

Ĵ

٦,

| 00000   | 1010   | $ \begin{array}{c} FORMAT(4(14), 0(1EE12.4)) \\ FORM(20(14), 0(1EE12.4)) \\ \end{array} $   |
|---|--|---|
| 03362   | 1910   | FORMAT(20(14),8(1PE12.4))   |
| 03303   |  | SIOP  |
|   |  |   |
| 00001   | C  |   |
| 00001   | *****  | ***************************************   |
| 00002   | C  |   |
| 00003   | C  | CIEDOLIFITHE COLORE (NETH NCEED YO YOUY YMAY CDETY CDETY)   |
| 00004   |  | SUBROUTINE SFACE (NDIN, NOED), NJ, 13, ATAA, IMA, SDEVA, SDEVI)   |
| 00005   | <u>^</u>   | KEAL AS(1), IS(1), FREQA(100), FREQI(100)   |
| 00006   | C  |   |
| 00007   |  | D = 1, NEIN   |
| 00008   | 761  | $\operatorname{FREQA}(J) = 0.$  |
| 00009   | 101  | PREQ(J) = 0.  |
| 00010   |  | 10 / 02 ID = 1, NSEED   |
| 00011   |  | $IX = NINT (XS(ID) / XMAX ^ (NBIN-1)) + 1$  |
| 00012   |  | IY = NINT (YS(ID) / YMAX (NBIN-1)) + 1  |
| 00013   | 700  | FREQX(1X) = FREQX(1X) + 1.  |
| 00014   | /62  | FREQY(IY) = FREQY(IY) + 1.  |
| 00015   |  |   |
| 00010   |  | SDEVI = 0.  |
| 00017   |  | BARN = (I. 'NSEED) /NBIN  |
| 00018   |  | D / 63 J = 1, MBIN  |
| 00019   | 7.00   | SDEVX = SDEVX + (FREQX(J) - EARN) **2   |
| 00020   | 763  | $SDEVY = SDEVY + (FREQY(0) - EARN)^{2}$   |
| 00021   |  | SDEVX = SQRT(SDEVX / (NBIN-1.))   |
| 00022   |  | SDEVI = SQRI(SDEVI / (NBLN-1.))   |
| 00023   |  | RETORN  |
| 00024   |  | END   |
|   |  |   |
| 00001   | c  |   |
| 00001   | C  |   |
| (MMMAY)   | *****  | ***************************************   |
| 00002   | *****<br>C   | ***************************************   |
| 00002   | *****<br>C   |   |
| 00002<br>00003<br>00004<br>00005  | *****<br>C   | SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, XO, YO,   |
| 00002<br>00003<br>00004<br>00005<br>00006   | *****<br>C   | SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, X0, Y0,<br>DNEIGH, INEIGH)<br>EFAL DNETCH(NDARR 1) XS(1) X0(1) X0(1)  |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007  | *****<br>C   | SUBROUTTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, X0, Y0,<br>1 DNEIGH, INEIGH)<br>REAL DNEIGH(NDARR,1), XS(1), YS(1), X0(1), Y0(1)<br>INTEGER INEIGH(NDARR,1)  |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008   | *****<br>C   | SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, X0, Y0,<br>DNEIGH, INEIGH)<br>REAL DNEIGH(NDARR,1), XS(1), YS(1), X0(1), Y0(1)<br>INTEGER INEIGH(NDARR,1)   |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009  | с<br>С   | SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, XO, YO,<br>DNEIGH, INEIGH)<br>REAL DNEIGH(NDARR,1), XS(1), YS(1), XO(1), YO(1)<br>INTEGER INEIGH(NDARR,1)<br>NERNEB LOCATES NEORS NEAREST NEIGHBORS XS(1) AND YS(1) ARE SEED  |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00009   | с<br>С<br>С<br>С<br>С  | SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, XO, YO,<br>DNEIGH, INEIGH)<br>REAL DNEIGH(NDARR,1), XS(1), YS(1), XO(1), YO(1)<br>INTEGER INEIGH(NDARR,1)<br>NERNEB LOCATES NBORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br>LOCATIONS. DNEIGH(I.I.) IS THE DISTANCE TO NEIGHBOR I OF SEED J  |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011  | с<br>С<br>С<br>С<br>С  | SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, XO, YO,<br>DNEIGH, INEIGH)<br>REAL DNEIGH(NDARR,1), XS(1), YS(1), XO(1), YO(1)<br>INTEGER INEIGH(NDARR,1)<br>NERNEB LOCATES NBORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br>AND INFIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF  |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012   | с<br>С<br>С<br>С<br>С  | SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, XO, YO,<br>DNEIGH, INEIGH)<br>REAL DNEIGH (NDARR, 1), XS(1), YS(1), XO(1), YO(1)<br>INTEGER INEIGH (NDARR, 1)<br>NERNEB LOCATES NBORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br>SEED J. NEERER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING   |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013  | с<br>С<br>С<br>С<br>С<br>С<br>С  | SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, X0, Y0,<br>1 DNEIGH, INEIGH)<br>REAL DNEIGH (NDARR,1), XS(1), YS(1), X0(1), Y0(1)<br>INTEGER INEIGH (NDARR,1)<br>NERNEB LOCATES NBORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING<br>DISTANCE.  |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014   | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C | SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, X0, Y0,<br>1 DNEIGH, INEIGH)<br>REAL DNEIGH (NDARR,1), XS(1), YS(1), X0(1), Y0(1)<br>INTEGER INEIGH (NDARR,1)<br>NERNEB LOCATES NBORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING<br>DISTANCE.  |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015  | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C                               | SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, XO, YO,<br>1 DNEIGH, INEIGH)<br>REAL DNEIGH (NDARR,1), XS(1), YS(1), XO(1), YO(1)<br>INTEGER INEIGH (NDARR,1)<br>NERNEB LOCATES NBORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING<br>DISTANCE.<br>DDMX = 1 E20  |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016   | *****<br>С<br>С<br>С<br>С<br>С<br>С<br>С   | SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, XO, YO,<br>1 DNEIGH, INEIGH)<br>REAL DNEIGH (NDARR, 1), XS(1), YS(1), XO(1), YO(1)<br>INTEGER INEIGH (NDARR, 1)<br>NERNEB LOCATES NBORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING<br>DISTANCE.<br>DDMX = 1.E20<br>DO 1107 J = 1. NEORS  |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016<br>00017  | *****<br>С<br>С<br>С<br>С<br>С<br>С<br>С   | SUBROUTINE NERNEB (NDARR, NEORS, NSEED, XS, YS, XO, YO,<br>1 DNEIGH, INEIGH)<br>REAL DNEIGH (NDARR, 1), XS(1), YS(1), XO(1), YO(1)<br>INTEGER INEIGH (NDARR, 1)<br>NERNEB LOCATES NEORS NEAREST NEIGHEORS. XS(I) AND YS(I) ARE SEED<br>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHEOR I OF SEED J,<br>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHEOR OF<br>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHEORS IN INCREASING<br>DISTANCE.<br>DDMX = 1.E20<br>DD 1107 J = 1, NEORS<br>DO 1107 JD = 1, NEORS   |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016<br>00017<br>00018   | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C | <pre>SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, X0, Y0,<br/>1 DNEIGH, INEIGH)<br/>REAL DNEIGH(NDARR,1), XS(1), YS(1), X0(1), Y0(1)<br/>INTEGER INEIGH(NDARR,1)<br/>NERNEB LOCATES NBORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br/>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br/>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br/>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING<br/>DISTANCE.<br/>DDMX = 1.E20<br/>DO 1107 J = 1, NBORS<br/>DO 1107 ID = 1, NSEED<br/>DNEIGH(I, ID) = DDMX</pre>  |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016<br>00017<br>00018<br>00019  | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C | <pre>SUBROUTINE NERNEB (NDARR, NEORS, NSEED, XS, YS, X0, Y0,<br/>1 DNEIGH, INEIGH)<br/>REAL DNEIGH(NDARR,1), XS(1), YS(1), X0(1), Y0(1)<br/>INTEGER INEIGH(NDARR,1)<br/>NERNEB LOCATES NBORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br/>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br/>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br/>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING<br/>DISTANCE.<br/>DDMX = 1.E20<br/>DD 1107 J = 1, NEORS<br/>DO 1107 ID = 1, NSEED<br/>DNEIGH(J,ID) = DDMX</pre>   |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016<br>00017<br>00018<br>00019<br>00020   | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C | <pre>SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, X0, Y0,<br/>1 DNEIGH, INEIGH)<br/>REAL DNEIGH(NDARR,1), XS(1), YS(1), X0(1), Y0(1)<br/>INTEGER INEIGH(NDARR,1)<br/>NERNEB LOCATES NBORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br/>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br/>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br/>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING<br/>DISTANCE.<br/>DDMX = 1.E20<br/>DO 1107 J = 1, NBORS<br/>DO 1107 ID = 1, NSEED<br/>DNEIGH(J,ID) = DDMX<br/>DO 1108 ID = 1, NSEED</pre>   |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016<br>00017<br>00018<br>00019<br>00020<br>00021  | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C | <pre>SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, X0, Y0,<br/>1 DNEIGH, INEIGH)<br/>REAL DNEIGH(NDARR,1), XS(1), YS(1), X0(1), Y0(1)<br/>INTEGER INEIGH(NDARR,1)<br/>NERNEB LOCATES NBORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br/>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br/>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br/>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING<br/>DISTANCE.<br/>DDMX = 1.E20<br/>DO 1107 J = 1, NBORS<br/>DO 1107 ID = 1, NSEED<br/>DNEIGH(J,ID) = DDMX</pre>   |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016<br>00017<br>00018<br>00019<br>00020<br>00021<br>00022   | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C | <pre>SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, XO, YO,<br/>1 DNEIGH, INEIGH)<br/>REAL DNEIGH(NDARR,1), XS(1), YS(1), XO(1), YO(1)<br/>INTEGER INEIGH(NDARR,1)<br/>NERNEB LOCATES NBORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br/>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br/>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br/>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING<br/>DISTANCE.<br/>DDMX = 1.E20<br/>DO 1107 J = 1, NBORS<br/>DO 1107 ID = 1, NSEED<br/>DNEIGH(J,ID) = DDMX<br/>DO 1108 ID = 1, NSEED<br/>XO(ID) = XS(ID)<br/>YO(ID) = YS(ID)</pre>   |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016<br>00017<br>00018<br>00019<br>00020<br>00021<br>00022<br>00023  | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C | <pre>SUBROUTINE NERNEB (NDARR, NEORS, NSEED, XS, YS, X0, Y0,<br/>1 DNEIGH, INEIGH)<br/>REAL DNEIGH(NDARR,1), XS(1), YS(1), X0(1), Y0(1)<br/>INTEGER INEIGH(NDARR,1)<br/>NERNEB LOCATES NEORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br/>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br/>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br/>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING<br/>DISTANCE.<br/>DDMX = 1.E20<br/>DO 1107 J = 1, NEORS<br/>DO 1107 ID = 1, NSEED<br/>DNEIGH(J,ID) = DDMX<br/>DO 1108 ID = 1, NSEED<br/>X0(ID) = XS(ID)<br/>Y0(ID) = YS(ID)</pre>   |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016<br>00017<br>00018<br>00019<br>00020<br>00021<br>00022<br>00023<br>00024   | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C | <pre>SUBROUTINE NERNEB (NDARR, NEORS, NSEED, XS, YS, X0, Y0,<br/>1 DNEIGH, INEIGH)<br/>REAL DNEIGH(NDARR,1), XS(1), YS(1), X0(1), Y0(1)<br/>INTEGER INEIGH(NDARR,1)<br/>NERNEB LOCATES NEORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br/>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br/>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br/>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING<br/>DISTANCE.<br/>DDMX = 1.E20<br/>DO 1107 J = 1, NEGRS<br/>DO 1107 ID = 1, NSEED<br/>INEIGH(J,ID) = DDMX<br/>DO 1108 ID = 1, NSEED<br/>X0(ID) = XS(ID)<br/>Y0(ID) = YS(ID)<br/>DO 1101 ID = 1, NSEED</pre>   |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016<br>00017<br>00018<br>00019<br>00020<br>00021<br>00022<br>00023<br>00024<br>00025  | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C | <pre>SUBROUTINE NERNEB (NDARR, NEORS, NSEED, XS, YS, X0, Y0,</pre>  |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016<br>00017<br>00018<br>00019<br>00020<br>00021<br>00022<br>00023<br>00024<br>00025<br>00026                                     | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C | <pre>SUBROUTINE NERNEB (NDARR, NEORS, NSEED, XS, YS, X0, Y0,</pre>  |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016<br>00017<br>00018<br>00019<br>00020<br>00021<br>00022<br>00023<br>00024<br>00025<br>00026<br>00027                            | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C | <pre>SUBROUTINE NERNEB (NDARR, NEORS, NSEED, XS, YS, X0, Y0,<br/>1 DNEIGH, INEIGH)<br/>REAL DNEIGH(NDARR,1), XS(1), YS(1), X0(1), Y0(1)<br/>INTEGER INEIGH(NDARR,1)<br/>NERNEB LOCATES NEORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br/>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br/>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br/>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING<br/>DISTANCE.<br/>DDMX = 1.E20<br/>DO 1107 J = 1, NBORS<br/>DO 1107 ID = 1, NSEED<br/>DNEIGH(J,ID) = DDMX<br/>DO 1108 ID = 1, NSEED<br/>X0(ID) = XS(ID)<br/>Y0(ID) = YS(ID)<br/>DO 1101 ID = 1, NSEED<br/>IF (IDF .EQ. ID) GO TO 1102<br/>DIS = SQRT((XS(ID)-XS(IDF))**2 + (YS(ID)-YS(IDF))**2)</pre>   |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016<br>00017<br>00018<br>00019<br>00020<br>00021<br>00022<br>00023<br>00024<br>00025<br>00026<br>00027<br>00028                   | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C | <pre>SUBROUTINE NERNEB (NDARR, NBORS, NSEED, XS, YS, X0, Y0,<br/>1 DNEIGH, INEIGH)<br/>REAL DNEIGH(NDARR,1), XS(1), YS(1), X0(1), Y0(1)<br/>INTEGER INEIGH(NDARR,1)<br/>NERNEB LOCATES NBORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br/>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br/>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br/>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING<br/>DISTANCE.<br/>DDMX = 1.E20<br/>DO 1107 J = 1, NEORS<br/>DO 1107 ID = 1, NSEED<br/>DNEIGH(J,ID) = DDMX<br/>DO 1108 ID = 1, NSEED<br/>X0(ID) = XS(ID)<br/>Y0(ID) = YS(ID)<br/>DO 1101 ID = 1, NSEED<br/>IF (IDP .EQ. ID) GO TO 1102<br/>DIS = SQRT((XS(ID)-XS(IDP))**2 + (YS(ID)-YS(IDP))**2)<br/>IF (DIS .IT. DNEIGH(NEORS, ID)) THEN</pre>  |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016<br>00017<br>00018<br>00019<br>00020<br>00021<br>00022<br>00023<br>00024<br>00025<br>00026<br>00027<br>00028<br>00029          | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C | <pre>SUBROUTINE NERNEB (NDARR, NEORS, NSEED, XS, YS, X0, Y0,<br/>1 DNEIGH, INEIGH)<br/>REAL DNEIGH(NDARR,1), XS(1), YS(1), X0(1), Y0(1)<br/>INTEGER INEIGH(NDARR,1)<br/>NERNEB LOCATES NEORS NEAREST NEIGHBORS. XS(I) AND YS(I) ARE SEED<br/>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHBOR I OF SEED J,<br/>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHBOR OF<br/>SEED J. NERRER USES A BUBBLE SORT TO RANK NEIGHBORS IN INCREASING<br/>DISTANCE.<br/>DDMX = 1.E20<br/>DO 1107 J = 1, NEORS<br/>DO 1107 J = 1, NEORS<br/>DO 1107 ID = 1, NSEED<br/>DNEIGH(J,ID) = DDMX<br/>DO 1108 ID = 1, NSEED<br/>X0(ID) = XS(ID)<br/>Y0(ID) = YS(ID)<br/>DO 1101 ID = 1, NSEED<br/>ID 1101 ID = 1, NSEED<br/>ID 1102 IDF = 1, NSEED<br/>IF (IDF.EQ. ID) GO TO 1102<br/>DIS = SQRT((XS(ID)-XS(IDF))**2 + (YS(ID)-YS(IDF))**2)<br/>IF (DIS .LT. DNEIGH(NEORS, ID) THEN<br/>DNEIGH(NEORS,ID) = DIS</pre> |
| 00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014<br>00015<br>00016<br>00017<br>00018<br>00019<br>00020<br>00021<br>00022<br>00023<br>00024<br>00025<br>00026<br>00027<br>00028<br>00029<br>00030 | *****<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C<br>C | <pre>SUBROUTINE NERNEB (NDARR, NEORS, NSEED, XS, YS, X0, Y0,<br/>1 DNEIGH, INEIGH)<br/>REAL DNEIGH(NDARR,1), XS(1), YS(1), X0(1), Y0(1)<br/>INTEGER INEIGH(NDARR,1)<br/>NERNEB LOCATES NEORS NEAREST NEIGHEORS. XS(I) AND YS(I) ARE SEED<br/>LOCATIONS, DNEIGH(I,J) IS THE DISTANCE TO NEIGHEOR I OF SEED J,<br/>AND INEIGH(I,J) IS THE SORTED SEED NUMBER OF THE ITH NEIGHEOR OF<br/>SEED J. NERBER USES A BUBBLE SORT TO RANK NEIGHEORS IN INCREASING<br/>DISTANCE.<br/>DDMX = 1.E20<br/>DO 1107 J = 1, NEORS<br/>DO 1107 ID = 1, NSEED<br/>DNEIGH(J,ID) = DDMX<br/>DO 1108 ID = 1, NSEED<br/>X0(ID) = XS(ID)<br/>Y0(ID) = YS(ID)<br/>DO 1101 ID = 1, NSEED<br/>IF (IDP .EQ. ID) GO TO 1102<br/>DIS = SQRT((XS(ID)-XS(IDP))**2 + (YS(ID)-YS(IDP))**2)<br/>IF (DIS .LT. DNEIGH(NEORS,ID)) THEN<br/>DNEIGH(NEORS,ID) = DIS<br/>INEIGH(NEORS,ID) = IDP</pre>   |

-

ć,

1

Ç

| 00031 |      | DO 1103 J = 1, NEORS-1                       |
|-------|------|--|
| 00032 |      | K = NBORS - J                                |
| 00033 |      | IF (DNEIGH(K+1, ID) .LT. DNEIGH(K, ID)) THEN |
| 00034 |      | DTMP = DNEIGH(K, ID)                         |
| 00035 |      | ITMP = INEIGH(K, ID)                         |
| 00036 |      | DNEIGH(K, ID) = DNEIGH(K+1, ID)              |
| 00037 |      | INEIGH(K, ID) = INEIGH(K+1, ID)              |
| 00038 |      | DNEIGH(K+1, ID) = DTMP                       |
| 00039 |      | INEIGH(K+1, ID) = ITMP                       |
| 00040 |      | ELSE   |
| 00041 |      | GO TO 1102                                   |
| 00042 |      | ENDIF  |
| 00043 | 1103 | CONTINUE                                     |
| 00044 |      | ENDIF  |
| 00045 | 1102 | CONTINUE                                     |
| 00046 | 1101 | CONTINUE                                     |
| 00047 |      | RETURN                                       |
| 00048 |      | END  |
|       |      |  |

| 00001<br>00002<br>00003   | C<br>************************************ |  |  |
|---|---|--|--|
| 00004<br>00005<br>00006<br>00007<br>00008<br>00009<br>00010<br>00011<br>00012<br>00013<br>00014 |   | SUBROUTINE ERRINV(X)<br>EXTERNAL FUNC<br>DATA ERR, ERA /1.E-4,1.E-4/<br>COMMON /BLKQ/ RMU, SIG,Z<br>XL=RMU-20.*SIG<br>XR=RMU+20.*SIG<br>X=RMU<br>CALL FZERO(FUNC, XL, XR, X, ERR, ERA, IFLAG)<br>X=XL<br>RETURN<br>END |  |
| 00001<br>00002<br>00003<br>00004<br>00005<br>00006<br>00007<br>00008                            | C<br>C                                    | FUNCTION FUNC(X)<br>COMMON /BLKQ/ RMU,SIG,Z<br>ETA=(X-RMU)/SIG/SQRT(2.)<br>FUNC=(1.+ERF(ETA))/2Z<br>RETURN<br>END  |  |
| 00001   | C   | *****  |  |
| 00002   | C   |  |  |
| 00004   | •   | FUNCTION ERRETA (Y)  |  |
| 00005   |   | EXTERNAL GUNC<br>Date FOR FOR $(1 F - A + 1 F - A)$  |  |
| 00007   |   | COMMON /BLKZ/ VALU   |  |
| 80000   |   | VALU = Y   |  |
| 00009   |   | XL = -20   |  |
| 00010   |   | XR = 20  |  |
| 00011   |   | X = 0.<br>Call FZERO (CINC XI, XR X FRR FRA TELAG)   |  |
| 00013   |   | ERRETA = XL  |  |
| 00014   |   | RETURN   |  |
| 00015   |   | END  |  |

| 00001 | С     | FUNCTION CUNC (X)   |
|-------|-------|---|
| 00003 |       | COMMON /BLKZ/ VALU  |
| 00004 |       | GUNC = ERF(X) - VALU  |
| 00005 |       | RETURN  |
| 00006 |       | END   |
| 00001 | с     |   |
| 00002 | C**** | ***************************************                                 |
| 00003 | С     |   |
| 00004 |       | SUBROUTINE PLIEOX (XMAX, YMAX, NEOX)                                    |
| 00005 |       | DIMENSION XX(20), YY(20)  |
| 00006 |       | P1=4.*ATAN(1.)  |
| 00007 |       | $DDD=0.5^{SQRT}(1.+TAN(2.^{PI/NBOX})**2)$                               |
| 00008 |       | $\frac{1}{1000} = 1, \text{NDOA+1}$ $\frac{1}{1000} = 1, \text{NDOA+1}$ |
| 00010 |       | XX(J) = (0.5 + DDD*COS(THET)) * XMAX                                    |
| 00011 |       | YY(J) = (0.5 + DDD * SIN(THET)) * YMAX                                  |
| 00012 | 814   | CONTINUE  |
| 00013 |       | CALL THKCRV(.030)   |
| 00014 |       | CALL CURVE(XX,YY,NBOX+1,0)  |
| 00015 |       | CALL THKCRV(.010)   |
| 00010 |       | FND   |
|       |       |   |
| 00001 | C     | *****   |
| 00002 | C     |   |
| 00004 | C     | SUBROUTINE PLIENDS (IDMX, NSID, IPRM, XSTD, YSTD, TPOP)                 |
| 00005 |       | DIMENSION NSID(1), IPRM(12,1), XSID(12,1), YSID(12,1), IPOP(12,1)       |
| 00006 |       | DIMENSION XB(2), YB(2)  |
| 00007 |       | CALL THKCRV(.010)   |
| 80000 |       | DO $815 \text{ ID}=1, \text{IDMX}$                                      |
| 00009 |       |   |
| 00010 |       | TF (TDP CT TD) CO TO 815  |
| 00012 |       | IF $(IPOP(L, ID), EO.1)$ THEN   |
| 00013 |       | XB(1) = XSID(L, ID)   |
| 00014 |       | YB(1) = YSID(L, ID)   |
| 00015 |       | XB(2) = XSID(L+1, ID)   |
| 00016 |       | YB(2) = YSID(L+1, ID)   |
| 00017 |       | CALL CURVE (AB, YB, 2, 0)   |
| 00019 | 815   | CONTINUE  |
| 00020 | 010   | RETURN  |
| 00021 |       | END   |
| 00001 | С     |   |
| 00002 | c     |   |
| 00003 |       | SUBROUTINE DELTAS (IFLAG, R1, R2, DELTAO, L, DELTAD, DELTAP,            |
| 00004 | -     | L DELTAV)   |
| 00005 | С     |   |
| 00006 |       | (LAG = 1 => ANALYTICAL SOLUTION FOR R2 = R1 (ZEROTH-ORDER PERTURBATION  |
| 00007 | Ст    | $PT_{AG} = 2 \implies FTRST-ORDER DEPUTIBRATION SOLUTION$               |
| 00009 | C II  | FLAG = 3 => EXACT SOLUTION (VIA NUMERICAL INTEGRATION IF NECESSARY)     |
|       |       |   |

1.12

1997 - J.-

150

| 00010  | C  |      |
|--------|--|------|
| 00011  | IMPLICIT DOUBLE PRECISION (A - H, O - Z)   |      |
| 00012  | DOUBLE PRECISION K, I, L, INT3, INT1, LS4, KS, KI  |      |
| 00013  | DIMENSION SUM(4), ZUM(4)   |      |
| 00014  | DSTAR (TERM1, R3S, T, COST, SINT) = TERM1 - COST -   |      |
| 00015  | 1 SQRT (R3S - SINT*SIN   | 1L)  |
| 00016  | $ARG = 0.5 \times L/R1$  |      |
| 00017  | BATTO2 = DELTAO/R1   |      |
| 00017  | $P_{2} = P_{2} / P_{1}$  |      |
| 00010  | $\frac{1}{10} = \frac{1}{10}$  |      |
| 00019  | $RIS = RI^{*}RI$   |      |
| 00020  | $LS4 = L^{-}L/4$ .   |      |
| 00021  | ROOTI = SQRT(RIS - LS4)  |      |
| 00022  | THETA = $ASIN(ARG)$  |      |
| 00023  | IF (IFLAG .NE. 3) THEN   |      |
| 00024  | K = RATIO2/2. + 1.   |      |
| 00025  | K1 = K + 1.  |      |
| 00026  | KS = K*K   |      |
| 00027  | V1 = KS - 1.   |      |
| 00028  | EPS = RATIO3 - 1.  |      |
| 00020  | BASTC = ATAN(SORT( $K1/(K - 1.)$ )*TAN(THETA/2.))/SORT   | (V1) |
| 00025  | BASTOK - BASTOKK   |      |
| 00030  | $C = -\pi \mu \mu \tau \lambda + 2 * BASTCK$   |      |
| 00031  | G = -InEIR + 2. DESIGN   |      |
| 00032  | DEDIAD = L/G   |      |
| 00033  | $\frac{\text{DENOM}}{\text{DENOM}} = K - \cos(\text{THETA})$   |      |
| 00034  | VZ = SIN(THETA)/DENOM  |      |
| 00035  | $V_3 = V_2/DENOM$  |      |
| 00036  | F = (3.*BASICK + (1. + 0.5*KS)*V2)/V1 + 0.5*K*V3   |      |
| 00037  | V1F = V1/F   |      |
| 00038  | DELTAV = DELTAO + $R1*(2 THETA/ARG) - ROOT1$   |      |
| 00039  | DELTAP = $SQRT(4.*R1S*L*V1F/DELTAV)$   |      |
| 00040  | IF (IFLAG .EQ. 1) RETURN   |      |
| 00041  | CD = (0.5*V2 - BASIC) / (K1*G)   |      |
| 00042  | $DET TAD = DET TAD^* (1. + EPS^*CD)$   |      |
| 00042  | $H = (2 \times (2 \times KS + 1)) \times BASTC + 3 \times K \times V2) / V1 + V3$  |      |
| 00043  | $CE = (33 \times 71 E) = (33 \times 71 $ |      |
| 00044  | $C_{1} = (\sqrt{3} \sqrt{11}) D_{1} (0) + 20 = 100 H/1/7 (00 H/2)$   |      |
| 00045  | DVI = KI (I IIIIK/ACO)   |      |
| 00046  | $CP = 0.5^{\circ}(5.^{\circ}CE - DVI/DEDIAV)$  |      |
| 00047  | $DELTAV = DELTAV + EPS^{-}DVI$   |      |
| 00048  | $DELTAP = DELTAP^* (1. + EPS^*CP)$   |      |
| 00049  | RETURN   |      |
| 00050  | ELSE   |      |
| 00051  | TERM1 = RATIO2 + 1. + RATIO3   |      |
| 00052  | R3S = RATIO3*RATIO3  |      |
| 00053  | R2S = R2*R2  |      |
| 00054  | DT = THETA/20.   |      |
| 00055  | T = 0.   |      |
| 00056  | DELTA = DSTAR (TERM1, R3S, T, $1.D0$ , $0.D0$ )  |      |
| 00057  | ZIM(2) = 0.5/DELTA   |      |
| 00057  |  |      |
| DELTAS |  |      |
| 00058  | ZUM(4) = 0.  |      |
| 00059  | SUM(2) = ZUM(2) / (DELTA*DELTA)  |      |
| 00060  | SUM(4) = 0.  |      |
| 00061  | TFT.TP = 2   |      |
| 00061  | $r_{0} = 1$  |      |
| 00002  |  |      |
| 00003  |  |      |
| 00064  | CUST = CUS(T)  |      |
| 00065  | SINT = SIN(T)  |      |
| 00066  | DELTA = DSTAR(TERM1, R3S, T, COST, SINT)   |      |
| 00067  | GRAND2 = COST/DELTA  |      |

| 00068 | GRAND1 = GRAND2/(DELTA*DELTA)                     |
|-------|---|
| 00069 | IFLIP = 6 - IFLIP                                 |
| 00070 | SUM(IFLIP) = SUM(IFLIP) + GRAND1                  |
| 00071 | ZUM(IFLIP) = ZUM(IFLIP) + GRAND2                  |
| 00072 | 11 CONTINUE                                       |
| 00073 | SUM(2) = SUM(2) - 0.5*GRAND1                      |
| 00074 | ZUM(2) = ZUM(2) - 0.5*GRAND2                      |
| 00075 | $FACTOR = 1.5 \pm L/DT$                           |
| 00076 | DECUBE = FACTOR*R1S/(2.*SUM(2) + 4.*SUM(4))       |
| 00077 | DELTAD = FACTOR/(2.*ZUM(2) + 4.*ZUM(4))           |
| 00078 | ROOT2 = SQRT(R2S - LS4)                           |
| 00079 | DELTAV = DELTA0 + R1 + R2 - 0.5*(ROOT1 + ROOT2) - |
| 00080 | 1 $(R2S/L)*ASIN(0.5*L/R2) - (R1S/L)*THETA$        |
| 00081 | DELTAP = SQRT (DECUBE/DELTAV)                     |
| 00082 | END IF  |
| 00083 | RETURN  |
| 00084 | END   |
|       |   |
|       |   |
| 00001 | C   |
| 00002 | C   |
| 00003 | FUNCTION BOXMUL (ISED, RMU, SDEV, R2)             |
| 00004 | DATA TWOPI /6.2831853072/                         |
| 00005 | XXX1 = RAN (ISED)                                 |
| 00006 | XXX2 = RAN (ISED)                                 |
| 00007 | BOXMUL = SQRT(-2. *ALOG(XXX1)) *COS (TWOPI *XXX2) |
| 80000 | BOXMUL = SDEV *BOXMUL + RMU                       |
| 00009 | R2 = SQRT(-2. *ALOG(XXX1)) *SIN (TWOPI *XXX2)     |
| 00010 | R2 = SDEV *R2 + RMU                               |
| 00011 | RETURN  |
| 00012 | END   |

:

A63

1.1
## UNLIMITED RELEASE INITIAL DISTRIBUTION

ż

e

3

1436 C. E. Meyers, 4523 9001 T. O. Hunter, 8000 Attn: M. E. John, 8100 A. West, 8200 R. C. Wayne, 8400 D. L. Crawford, 8900 9403 J. M. Hruby, 8230 9054 W. J. McLean, 8300 Attn: W. Bauer, 8302 L. A. Rahn, 8351 F. P. Tully, 8353 G. A. Fisk, 8355 R. W. Carling, 8362 9042 G. H. Evans, 8345 9042 J. F. Grcar, 8345 9042 S. K. Griffiths, 8345 (5)9042 C. M. Hartwig, 8345 9042 R. S. Larson, 8345 9042 R. H. Nilson, 8345 (5) 9056 P. H. Paul, 8351 9052 M. D. Allendorf, 8361 9052 D. R. Hardesty, 8361 9053 S. R. Vosen, 8362 9405 T. M. Dyer, 8700 Attn: M. Perra, 8711 M. I. Baskes, 8712 G. J. Thomas, 8715 K. L. Wilson, 8716 9904 W. D. Wilson, 8700 9403 J. C. F. Wang, 8713 9403 R. W. Bradshaw, 8713 (5)9161 W. G. Wolfer, 8717 0834 M. J. Martinez, 9112 0826 M. R. Baer, 9112 9141 D. A. Nissen, 11600 9021 Technical Communications for OSTI, 8815 (10) 9021 Technical Communications, 8815 0899 Technical Library, 13414 (4)

9018 Central Technical Files, 8940 (3)