Predicting Geometrical Properties of Random Packed Beds from Computer Simulation

K. Nandakumar, Yiqiang Shu, and K. T. Chuang
Dept. of Chemical and Materials Engineering, University of Alberta, Edmonton, Alberta, Canada T6G 2G6

Using advances in computational geometry and collision-detection algorithms, an algorithm was developed to analyze and predict the geometrical properties of a randomly packed structure using packing objects of arbitrary shape. A 3-D polygonal model was used to describe an arbitrarily complex packing object and a simple container object. The dynamics of the packing process is not simulated, but a search algorithm finds stable equilibrium positions for the packing objects of arbitrary shapes using a collision-detection algorithm in a 3-D space. A modified conjugate gradient optimization method determines the packing object's final packing location and orientation. Once the bed is packed, both macroscopic quantities like the overall porosity, the specific surface area, and the number of packing objects per unit volume and microscopic properties like the porosity variation in any direction could be determined. For accurate porosity calculation inside a given 3-D polygonal sample space, the Sutherland-Hodgman polygon clipper algorithm was used. Predicted results are validated against available experimental data for spheres, Raschig rings, Pall rings, and cascade minirings.

Introduction

Packed columns are used widely in chemical process industries. Typical applications involve contacting gas/liquid mixtures in a countercurrent fashion to selectively transfer a species from one phase to the other using solubility or volatility differences. Packed absorption/desorption and distillation columns are examples of this type. Several other wide-ranging applications of packed-bed equipment in chemical process industries include trickle-bed reactors (Shah, 1986), packed-bed reactors (Wakao and Kaguei, 1982), bioreactors with immobilized cells in packed beds. From the point of studying the transport phenomena (such as pressure drop, heat and mass transfer) in packed beds, the bed is treated as a porous medium with an effective resistance, an effective mass transfer area, and so on. In fact, packing objects like Raschig rings, Pall rings, and so forth, are designed with the objective of minimizing the resistance and maximizing the effective area. The packing objects are then dumped in a random manner into a container (typically a cylindrical vessel) to make a packed column, and the mean properties of the packed bed (such as number of pieces per m^3, overall porosity) are measured. These properties are then used as inputs to transport models such as the Ergun equation, Darcy's law, and its variants.

This is an important research subject in particulate systems also, as evidenced by numerous publications in this area. Much of the effort is focused on (a) experimental measurements of geometrical properties (Roblee et al., 1958; Scott, 1960; Benenati and Brosilow, 1962; Mueller, 1992; Goodling et al., 1983; (b) construction of theoretical and/or empirical models (Gyenis et al., 1982; Dodds, 1980; Ouchiyama and Tanaka, 1980, 1981, 1984, 1986, 1988, 1989; Suzuki and Oshima, 1985; Suzuki et al., 1986; Yu and Standish, 1987, 1988, 1991); and (c) more recently on computer simulation of the packing process (Nolan and Kavanagh, 1992, 1993; Powell, 1980) to predict geometrical properties of particulate systems. Works on theoretical model development and computer simulation have been restricted to simple packing objects with a high degree of symmetry such as spheres, cylinders, and disks, since the analysis required is relatively simple. They are reviewed in Haughey and Beveridge (1969). With the advances in computational geometry and the power

Correspondence concerning this article should be addressed to K. T. Chuang. Present address of Y. Shu: Prostart Technical Consultant Edmonton, Alberta, Canada T6H 3K1.

of computers, it is now possible to develop theoretical and numerical models to predict the geometric characteristics of a randomly packed bed, made of an *arbitrarily shaped packing material.*

The main objective of this work is to develop an effective mathematical algorithm and, based on that, develop a computer software tool to simulate the process of packing objects of *arbitrary shapes* into a cylindrical container. The container should be much bigger in size than the packing objects. The objective of the simulation is to find a final stable equilibrium position for each packing object in the three-dimensional space of a container. For each object, its geometrical structure is defined with respect to a local coordinate system with its centroid as the origin, while its centroid is stored in a global coordinate system of the container. These data are then used for further analysis such as porosity and effective area calculation.

This is not the first attempt to develop such a tool for studying randomly packed objects. Chan and Ng (1986) developer a similar tool to extract such information as the coordination number and porosity variation. They cite six other previous attempts to study properties of packed beds using computer-generated sphere packs (Adams and Matheson, 1972; Bennet, 1972; Visscher and Bolsterli, 1972; Tory et al., 1973; Matheson, 1974; Powel, 1980). But they share a common limitation of being restricted to a spherical geometry for the packing object. Since the spherical geometry has a high degree of symmetry, collision detection can be achieved with relative ease. Such is not the case for an object with an arbitrary shape. This important generalization to such an object is the main contribution of the present work. It can help in designing and testing new shapes of packing objects for the desirable macroproperties of the bed without having to manufacture particles of such complicated shapes during the conceptual design/testing stage.

Random Packing Process

In the real world, the dynamics of a *packing element* falling into a container with many other packing objects already in place, is a very complicated process. It is, however, not necessary to capture the dynamics in a realistic manner, for we are interested only in the *final stable equilibrium positions* of the objects. The simulation may also involve a large number of *packing elements* (normally several thousands). So we structure the algorithm in such a way that the object's location and orientation in the three-dimensional space are determined by a search algorithm that (a) avoids collision between the object and a wall that is present in its neighborhood and yet be as close as possible to its neighbors, and (b) finds the lowest available stable position within the container.

From mechanical considerations, an object's stable condition is one that has its center of gravity at the lowest possible position within the constraints found around it. To find each *packing element*'s final stable position and orientation in a three-dimensional space we assume the following:

- For each packing object, an algorithm will be designed to place it into the container with a pool of existing packing objects in such a way that the packing object will always find itself at the lowest possible place for its center of gravity. Once it is in the final place, it will never move again. To make the simulation be close to reality, the initial position and orientation of the packing object will be selected in a random manner.
- To ensure that the packing objects are close to each other in position, the algorithm will resolve a layer of packing object's position at one time. The set of packing objects in the same layer are as close as possible to each other and the average positions of the *packing objects set* is at the lowest possible position.
- No collision between any two packing objects in the container and no collision between any packing object and the container wall is allowed. The collision-detection algorithm developed by Ponamgi, Mancha, and Lin (1997) and implemented in the software package VCOLLIDE is used to place the packing elements close to each other without intersecting each other.

Features of the Algorithm

Third polygonal modeling for the packing object and container

The complex, smooth surface of the *packing object* is converted into a three-dimensional polygonal object presented by a set of triangular surfaces. Figures 1a−1d show this process for a sphere, a Pall ring, a Raschig ring, and a cascade miniring, respectively. The basic unit of the three-dimensional model is a triangle, which will be passed to the collision-detection module. This discretization can be made as accurate as possible by subdividing a complex surface into smaller triangles, but at the expense of increased computational demand. Given the fact that there are a few thousand packing objects for collision detection and porosity calculation, a moderate number of triangles have to be chosen to represent the packing element. This approach permits the algorithm to process virtually any shape of *packing object* as long as the packing object can be converted into the polygonal model. While any smooth surface can be discretized by other planar objects like quadrilaterals, we have chosen the triangulation, as it is the simplest planar object. Furthermore the collision-detection algorithms that we have used require the surface to be represented by triangular elements. For all of the objects studied, we have used sufficiently fine discretization to produce deviations of less than 1% in the predicted properties (like the number of elements per unit volume) upon further refinement of the surface by doubling the number of triangles.

The most commonly used shape of the *container object* is a cylinder. The geometrical shape of the container is also approximated by the same method of triangulation. The main difference is that the closed bottom end of the container is modeled as a rectangle of at least the same width as the diameter of the container, as shown in Figure 2c. Assuming the support plate to be flat bottomed, we can ignore the circular cross section of this plate. This is done to keep the number of triangles to model the bottom plate to the minimum, which will make the collision-detection algorithm more efficient. Because the packing objects are always inside the container, the outside part of the rectangular bottom of the container will never be touched by any *packing object*, and hence this

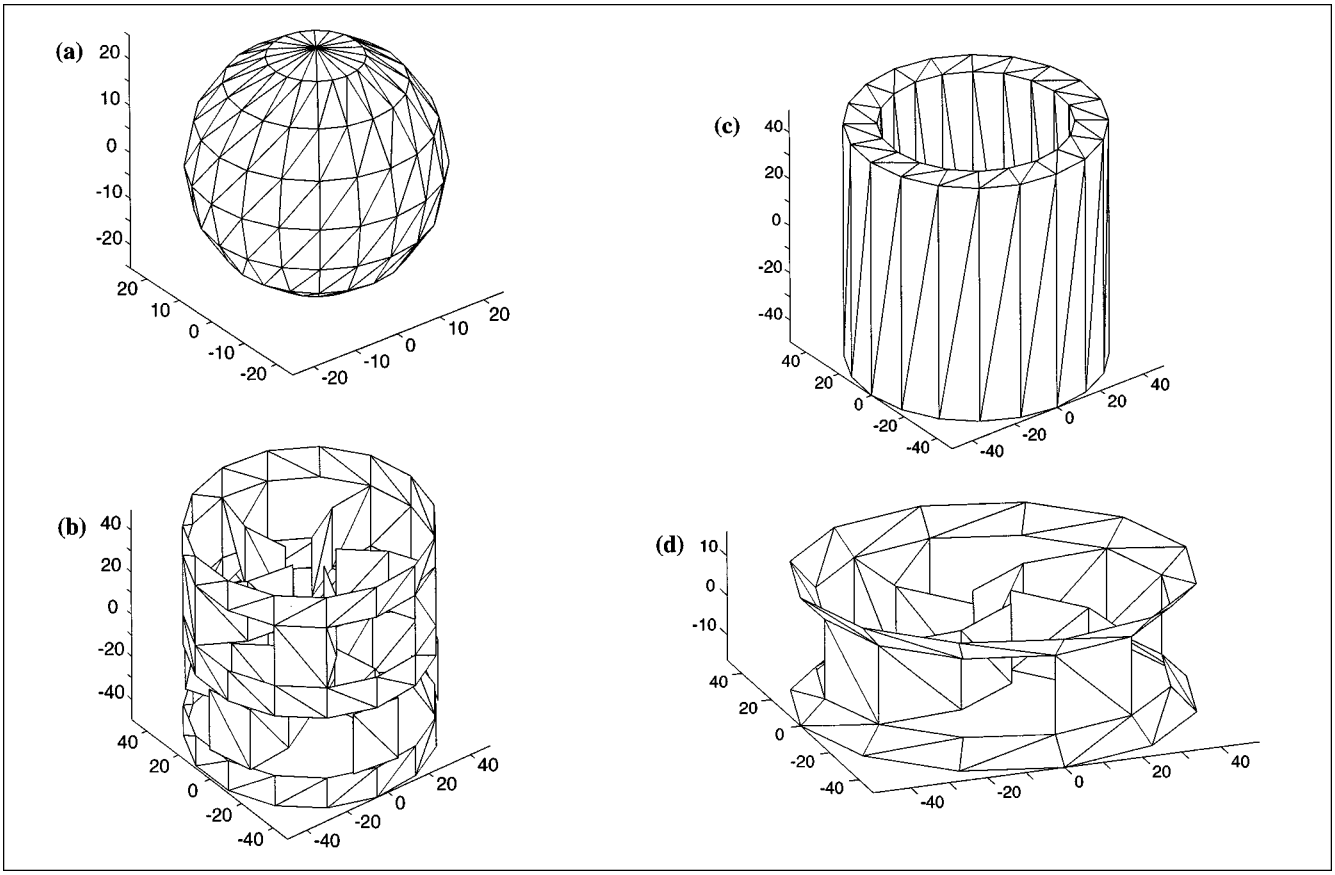

Figure 1. Examples of triangulation of (a) solid sphere, (b) Pall ring, (c) ceramic Raschig ring, (d) mini ring.

approximation does not introduce any error. This rectangular bottom can be triangulated with two pieces only. The thickness of a *container* will not be considered in the model, as it is irrelevant from the point of packing.

Binary search algorithm to find out the lowest position of a packing object

The global coordinate system (x,y,z) of the container that is used to locate each object is shown in Figure 2b. The origin of the *global coordinate system* is at the center of the bottom of the container and the (x,y) plane is coincident with the bottom of the container. The origin of the *local coordinate system* of the packing object is at its center of gravity. The direction of the local coordinate system is the same as the global coordinate system.

The orientation of the packing object is defined in Figure 2a. Consider a vector γ that lies in the $(x'\text{-}y')$ plane and let α be the angle between γ and the x-axis. The orientation of packing object is defined by using the vector γ as the axis of rotation, with the angle β being the degree of rotation in the counterclockwise direction about the rotation axis vector γ, as it is pointing from the inside to the outside. Hence α and β are the two angles that determine the orientation of the packing object.

A packing object's location and status in the global coordinate system will be determined by the set (x,y,z,α,β), where (x,y,z) is the packing object's center of gravity and (α,β) is the orientation as described above and as shown in Figure 2b.

The binary search algorithm is a search method that determines the *lowest place*, z, of a packing object such that it does not conflict with any existing pool of packing objects or container wall. For a given pool of packing objects in a container and a given starting location (x,y) and orientation (α, β) of the packing object to be placed in the container, this algorithm moves the packing object vertically down to its lowest possible position without colliding with existing packing objects. During this process, the (x,y) location and the orientation remain fixed. A binary search procedure is used, which is a process of repeatedly ruling out half the remaining search area until the element is placed in its final position. This routine will be used numerous times, as it is embedded in the next higher level of search, where the best possible final location for a single packing object is found by changing the (x,y) location and the orientation (α, β) of the packing many times. Given the fact that there are thousands of packing objects to be placed in the container, the efficiency of the search algorithm is an important factor in the whole process. The most time-consuming step in the search procedure is the one that determines if a packing object conflicts with any other existing objects or the container because it involves three-dimensional graphics operations. It is therefore important to develop an efficient data structure to represent the packing object's geometry, which is discussed next.

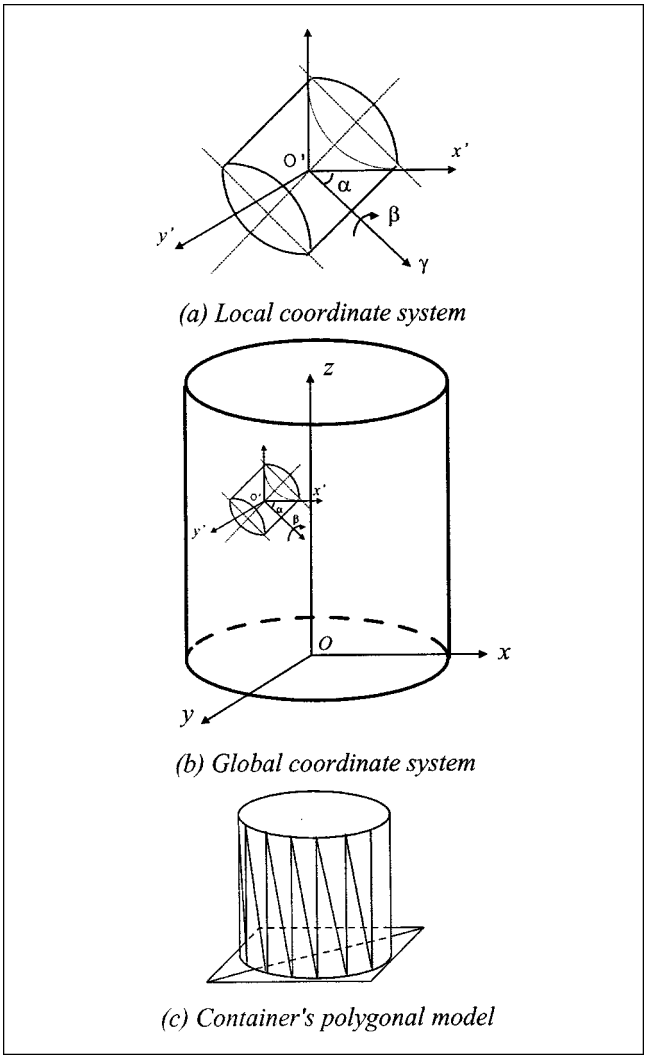

Figure 2. Coordinate system and search procedure illustrated.

Efficient data structure to represent the effective surface of the top layer

A new packing object is to be placed on top of a *set of existing objects* in the container by using the collision-detection algorithm. In this search procedure, it is not necessary to check for collisions between the new object and every object that has been already placed in the container. We can eliminate the objects existing well below the topmost surface. However, the data structure that represents the effective surface of packing objects on the top layer has to be carefully defined and maintained. It must keep a minimum number of packing objects already placed in the container to construct an effective real surface that will not allow any hole in the surface that can result in a packing object's possible breakthrough and give a wrong result. This is done by maintaining a list of candidate objects that lie within a range of $z \in (z_{\min}, z_{\max})$, where $z_{\max}$ is the highest centroid position of existing packing elements, and $z_{\min}$ is selected to include the top one or two layers. This approach cuts down the combinatorial explosion of the collision-detection problem. The problem scales quadratically with the diameter of the column and remains constant with the increasing height of the packing.

Determining candidate packing objects within possible collision distance

Maintaining an effective data structure of packing objects for the top layers alone is not sufficient to minimize the number of collision-detection evaluations. To improve the efficiency of the algorithm further, the data structure has a fast index function by the coordinate value that allows the algorithm to select candidate packing objects within possible collision distance from the object to be placed. The packing objects on the top layer that are outside the possible collision distance will be ignored. At each binary search step, with the packing object's changing coordinate value, the set of candidate packing objects will be created dynamically.

The same method will be applied to the collision testing between the packing object and the container, that is, check for possible collision only when the distance between a packing object and container bottom or container wall is in possible collision distance.

Optimization algorithm to determine the lowest position of a packing object

Since the dynamic trajectory of a packing object toward its final equilibrium position is not of interest in the present simulation, we only seek its final stable equilibrium position in the container. One of the attributes of a stable equilibrium position is that the center of mass of the packing object be at its lowest position. Thus for each packing object, the *lowest position in the z-direction* of the global coordinate system is found and this becomes the objective function. Let there be a function $z(x, y, \alpha, \beta)$. The objective of finding the lowest position of z for a given initial position and orientation (x, y, α, β) is accomplished by the binary search algorithm, while finding the lowest overall position possible (x, y, α, β) is accomplished with the modified conjugate-gradient method.

Conjugate gradient algorithms can be derived as extensions of the conjugate-gradient algorithm or as specialization of limited-memory quasi-Newton methods. Given an iterate x_k and a direction d_k, a line search is performed along d_k to produce $x_{k+1} = x_k + a_k d_k$. The Fletcher-Reeves variant of the conjugate algorithm generates d_{k+1} from the simple recursion:

$$d_{k+1} = -\nabla f(x_{k+1}) - \beta_k d_k$$

$$\beta_k = \left(\frac{\| \nabla f(x_{k+1}) \|_2}{\| \nabla f(x_k) \|_2} \right)^2.$$

The performance of the algorithm is sometimes enhanced by restarting, that is, periodically setting β_k as

$$\beta_k = \frac{[\nabla f(x_{k+1}) - \nabla f(x_k)]^T \nabla f(x_{k+1})}{\nabla f(x_k)^T \nabla f(x_k)}.$$

The original conjugate-gradient method works well with the smooth objective function. However, because there are many

packing objects in the container, the packing object's optimizing objective function, $z(x, y, \alpha, \beta)$, has many sharp peaks and valleys in the three-dimensional graph of the function. There are two potential problems, namely (a) finding a globally optimal solution is difficult because the search direction of optimization easily leads to a local optimization result, and (b) the search near a sharp valley location often leads to an oscillation around the lowest point. These oscillations reduce the efficiency of the algorithm. To resolve the problem, an ad hoc change to the procedure is used whenever an oscillation is detected, namely, let x_{k+1} equal

$$x_{k+1} = x_k + \alpha_k\left(\frac{d_k}{2}\right).$$

This is a kind of binary search just for one step, which will push the search out of oscillation and make sure the new point is near the lowest point. At the new start point, the optimization will switch back to the conjugate-gradient method.

Procedure for close packing of a layer of packing objects

The whole methodology and algorithm is based on the assumption that once a packing object's final location is discovered, it will never move again. Since only z is varied in this search step for randomly selected (x,y,α,β), it does not guarantee closeness of packing objects in the (x,y) plane. In other words, the optimization with a single objective function $z(x,y,\alpha,\beta)$ cannot lead to a closest *packing object set* inside a container. This is a case where a series of local optimizations (in this case, the lowest position of each packing object) do not necessarily lead to a global optimization result. This problem is addressed by using a multiple-objective optimization algorithm. To accomplish this objective, a layer of packing objects will be placed on the surface of the top *packing object set*, which is already inside the container. The packing elements should not be in conflict with each other in a layer, yet be as close as possible to each other. This is accomplished by starting with N packing elements with random (but conflict-free) initial positions and orientations (x,y,α,β) and finding the lowest z for each using the search algorithm outlined earlier. After this set of packing elements is created, we keep only the point with the greatest value r, where $r^2 = x^2 + y^2$. This is a multiple optimization selection where the *z value is lowest* and the packing element is *closest to the container wall*. Repeat this to construct a layer of packing objects. This procedure packs the packing elements at a given layer from the container wall toward the center.

Avoiding conflict between the packing element and the container wall

After the initial points are set up for a layer of packing objects, the conjugate-gradient algorithm is used to search (x, y, α, β) and find out its stable locations. At each step, the container's three-dimensional object acts as a constraint on the optimization procedure, enforcing the condition that the packing element must be inside the container. When a collision condition between a packing element and the container wall is detected, the z-value is increased and the search direction is guided to point toward the inside of the container. After the packing element is back inside the container, the normal calculations for the z-value are resumed. If a packing element is in conflict with the bottom of the container, the packing element will be moved up to remove the collision condition and the binary search procedure is capable of handling this condition.

Summary of the packing algorithm

The steps involved in the algorithm are summarized below:

1. Select a layer of packing elements' initial positions (x,y,α,β) such that the packing elements are close to each other (see the subsection on the procedure for close packing).
2. Find the individual packing element's stablest or lowest position by applying the modified conjugate-gradient method for each single packing element as outlined in the subsection on the optimization algorithm.
3. From this set pick the element that is closest to the container wall (see the subsection on the procedure for close packing).
4. Add the newly located packing element to the *packing element set* surface data structure and also delete the packing element in the surface data structure with the condition that the packing element no longer represents a top one, as outlined in the subsection on efficient data structure.

This process is repeated until a specified number of packing elements are placed in the container. The outcome of the simulation is a set of packing elements' final position and orientation (x,y,z,α,β) in the global coordinate system. They will be used in subsequent analyses such as porosity and surface-area calculations. Figure 3 shows a graphical visualization of the results from a simulation for Pall rings using a small number of packing elements.

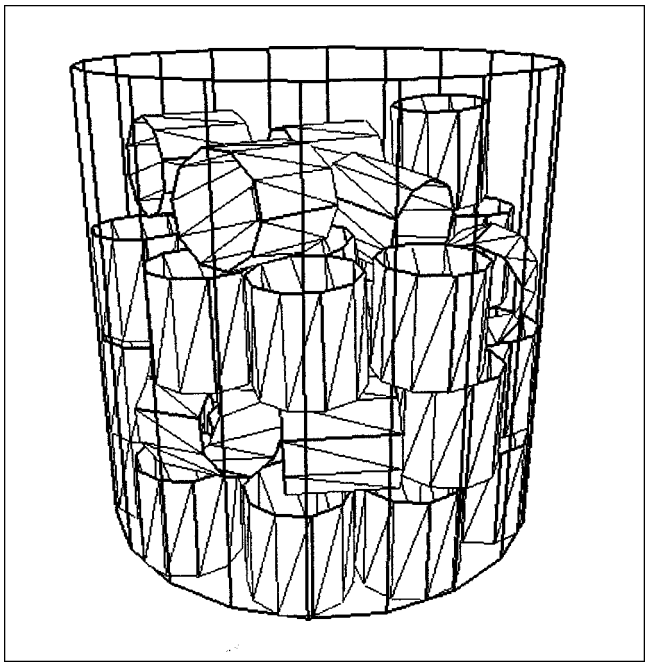

Figure 3. Sample visualization of computer-generated random packing.

Calculation of Geometrical Quantities

Definition of sample volume

The porosity is defined as void volume inside a three-dimensional measurement space divided by the volume of measurement space. If the measurement space is taken as the entire volume of the packed bed, we get the mean porosity of the bed, $\langle \varphi \rangle$, which is typically reported in tables characterizing geometrical properties of packing materials. The other properties that are reported on the scale of the bed are the number of pieces per m^3, N, and the specific surface area per m^3, a, of the bed. These are reported by the manufacturer for any new packing and are reported in standard references for well-known packing objects like Pall rings and Raschig rings.

To get an idea of any structural changes in these properties within the bed on a scale comparable to the pore scale, measurement volumes of the same order of magnitude as the packing-object size should be taken. Experimentally such detailed data have been taken only for spherical objects (see Roblee et al., 1958; Benenati and Brosilow, 1962; Mueller, 1992). Since we have complete information on the location and orientation of each packing element from the simulation, we can reconstruct such data by suitably choosing the measurement volume as shown in Figure 4. This requires (a) defining the location (r,θ,z) and the size $\delta r,\delta\theta,\delta z$ of the measurement space, (b) locating and clipping the packing elements that lie within the measurement space, and (c) measuring the solid volume and surface area of the packing elements that lie within the measurement subspace.

Although a simple *cube* as a three-dimensional measure space is easy for programming of Boolean operation of three-dimensional models, it is not well suited for porosity measurement in a cylindrical container. Near the outer edge of the container, it will introduce error, the magnitude of which depends on the size of the measurement space. Also, in experimental measurements, the sample space is typically circular rings, which provides circumferentially averaged values of geometrical properties. The measurement space that we take is a small circular segment as shown in Figure 4. The center point location of this sample volume (r,θ,z) is varied over $r \in [r_i, r_f], \theta \in [\theta_i, \theta_f], z \in [z_i, z_f]$. The measurement subspace is still approximated by six bounding planes that form a cuboid; hence, it is essential to have fine resolution in the θ-direction.

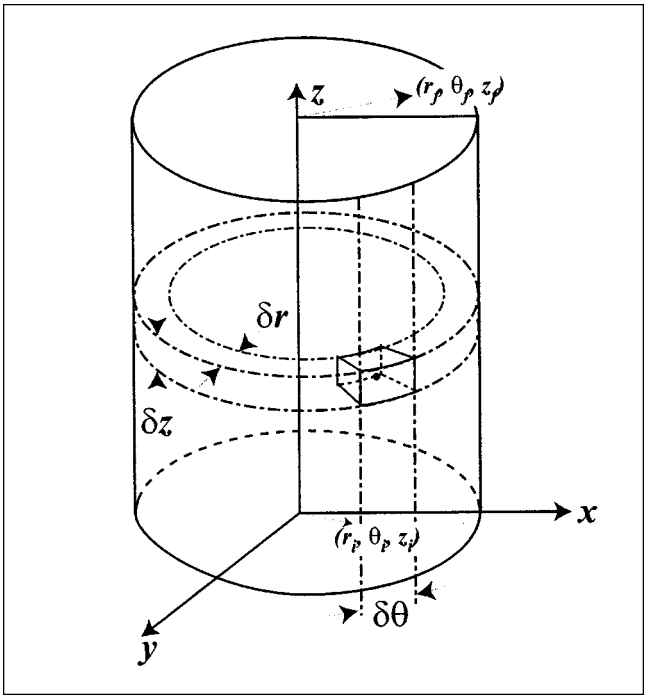

Figure 4. Definition of sample volume for porosity calculation.

The next task is to identify the packing elements that are within the measurement subspace and clip parts of the elements that lie outside the measurement subspace. Since the general geometrical problem of clipping curved surfaces in three-dimensional space to identify bounding curves is very difficult, we work with the triangulated data structure of the packing elements together with the Sutherland-Hodgman polygon clipper algorithm to construct parts of polygonal-shaped planes of packing elements that lie within the measurement subspace. The result of the clipping operation between a triangle and the three-dimensional measurement subspace is a general three-dimensional polygon, as sketched in Figure 5. The resulting polygons are triangulated again to yield a group of derived triangles that are used in computing quantities such as the total surface area.

Sutherland-Hodgman polygon clipper algorithm

The Sutherland-Hodgman Polygon-Clipper Algorithm is described in many references (Plastock and Kalley, 1990). It is a true three-dimensional algorithm that clips a general convex polgyon with any number of planes. In the case of the packing object's porosity calculation, the three-dimensional-measurement subspace consists of six planes. Within this algorithm, the term INSIDE means that the clipping plane's normal vector points toward the point, as defined by

$$(r + r_0) \cdot n \geq 0,$$

where r is the point in question, r_0 is a point on the plane, and n is normal to the plane. OUTSIDE is implied by NOT INSIDE. The intersection is the intersection point of the edge in question with the clipping plane.

The Sutherland-Hodgman Polygon-Clipper Algorithm can be represented as follows. For each edge of a plane, S is the starting point of the edge and P is the end point of the edge.

- If S is inside and P is inside, append the point P into the *result point set* (Figure 5b).
- If S is inside and P is outside, compute and append the intersection point of the edge SP with the clipping plane into the *result point set* (Figure 5c).
- If S is outside and P is outside, no operation.
- If S is outside and P is inside, compute and append the intersection point of the edge SP with the clipping plane into the *result point set*, then append the second node P into the *result point set* (Figure 5d).
- Construct a new polygon from the *result point set*.

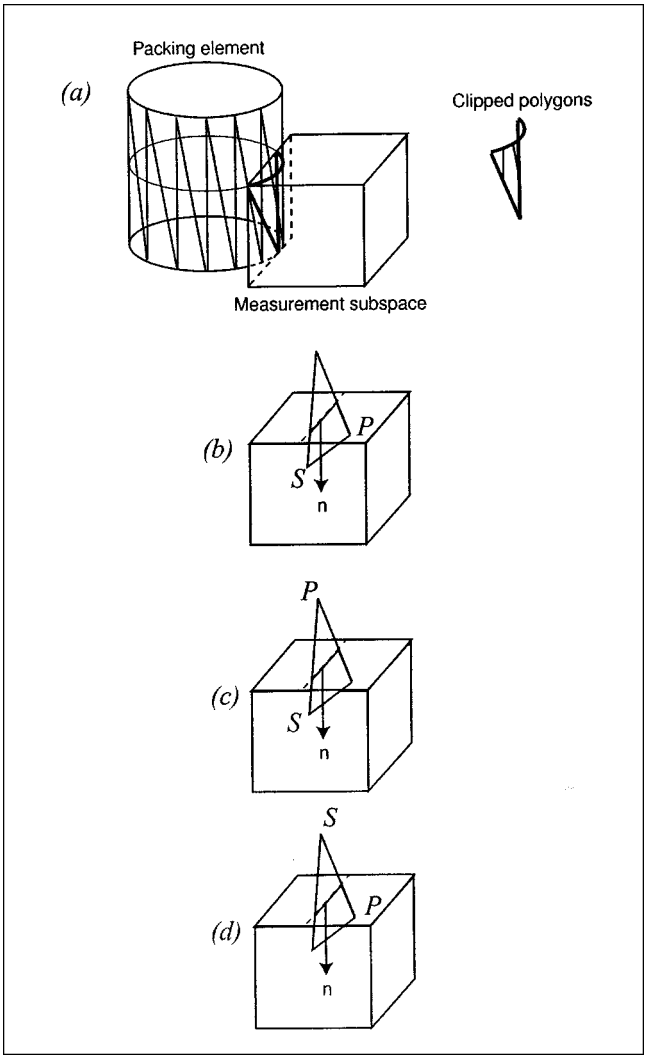

Figure 5. Clipping procedure is illustrated using a Raschig ring as packing element and a cuboid as measuring subspace.

- Repeat the preceding procedure for other clipping planes.
- The result of the clipped polygons will be decomposed to a set of three-dimensional triangles.

The algorithm of porosity calculation

The algorithm of porosity calculation will depend on both the packing element's 3-D polygonal model data and the packing result data that defines the packing element's locations and orientations. The most important step of porosity calculation is to determine the total solid space of packing elements inside the porosity measurement subspace. The algorithms for porosity calculation in a certain measurement space is described below:

- Construct the three-dimensional polygonal model for the porosity measurement subspace.
- Check the collision between the three-dimensional-measurement object and every packing object in the packing simulation result data.
- If a packing object is totally inside the three-dimensional-measurement object, keep the packing object's data in the *candidate packing* object set.
- If a packing object conflicts with the three-dimensional-measurement object, keep the packing object's data in the *candidate packing* object set.
- For each packing object in the *candidate set*, use the Sutherland-Hodgman Polygon-Clipper Algorithm, to create the clipped polygons that lie entirely within the three-dimensional-measurement subspace.
- Triangulate the clipped polygons and compute the surface area. For open packing elements like Pall rings and Raschig rings, the thickness is used to estimate the solid volume. For solid objects like spheres, cylinders, in addition to the triangulated surface data structure for each packing element, we maintain a volume data structure in the form of thin disks, which is used to calculate the solid volume inside the measurement subspace.

This algorithm is general enough to handle any packing element shape.

Results and Discussion

The coding of the algorithm is done using C++ and is portable to both Windows and Unix platforms. To pack 3000 elements into a container, the computing time required is about 90 minutes on a dual Pentium 450-Mhz processor running Windows NT 4.0. The result is a packing geometry file that contains the centroid and orientation of each element in the container. It is then processed for calculation of geometrical properties like porosity and surface area. The time required for this part of the calculation is a function of the sample size. At a given height z, using 36×72 unit cells in the radial and angular directions, it takes about 10 minutes of CPU time to calculate the porosity distribution over a plane.

Packing simulation of uniform spherical particles

Recall that the packing algorithm does not capture the dynamics of the packing process; it places the packing elements in a horizontal layer radially inward and packs layer by layer vertically upward. Hence it is difficult to distinguish between dense and loose packing as is commonly done in experiments, particularly with uniform spheres. Scott (1960) established the limiting values of porosities as 0.36 for dense packing and 0.40 for loose packing of equal spheres. These figures were obtained from a series of experiments conducted with containers of different heights and diameters and extrapolating to infinite height and diameter. Hence they are representative values of uniform packing of an *infinite extent*. In finite-size containers these values tend to be larger. In addition, a clear oscillatory variation of porosity has been observed by several workers (see Roblee et al., 1958; Benenati and Brosilow, 1962; Mueller, 1992) near bounding cylindrical surfaces. We check whether these characteristics are captured by the packing simulation.

A series of four packing simulations were done with particle (d) and container (D) diameters of ($d = 0.5$, $D = 4$), ($d = 0.5$, $D = 8$), ($d = 0.5$, $D = 12$), and ($d = 1$, $D = 12$). The results are shown in Table 1. The mean porosity ranges from 0.43 to 0.47. But a clear trend emerges if they are plotted as

Table 1. Macroscopic Simulation Results for Spheres

(d,D) (in.)	N (particles per m^3)	$\langle\varphi\rangle$ (mean porosity)
(0.5,4.0)	536,593	0.4773
(0.5,8.0)	582,846	0.4390
(0.5,12.0)	590,977	0.4312
(1.0,12.0)	70,970	0.4535

a function of (d/D) as shown in Figure 6. In the limit of $(d/D) \to 0$ we get a mean porosity of 0.406, which is close to the *loose* packing limit.

The third simulation with $(d = 0.5,\ D = 12)$ was done with 5,000 particles. For this case, the porosity variations in the radial and angular directions are shown in Figure 7 using a sample sizes of $\delta r = 0.02$ in., $\delta\theta = 10$, $\delta z = 4$ in. There is very little variation in the angular direction (as observed by Mueller, 1993) and a regular periodic variation in the radial direction (as observed by Mueller, 1992). A quantitative measure of this radial variation is shown in Figure 8 using two different sampling sizes of $\delta r = 0.02$ in. (0.51 mm), $\delta\theta = 10$, $\delta z = 4$ in. (102 mm), and $\delta r = 0.25$ in. (6.35 mm), $\delta\theta = 10$, $\delta z = 4$ in. (102 mm). The averaging is done over 8 diameters in the axial direction. The radial oscillations are quite evident near the wall and it decays over a distance of 6–7 diameters from the wall. The mean porosity averaged over this plane at

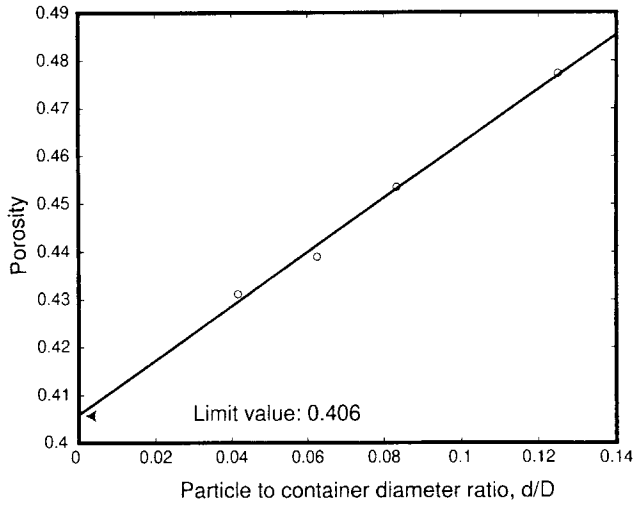

Figure 6. Extrapolated value of the mean porosity of spherical packing.

$z = 2.5$ in. (63.5 mm) is 0.438. The empirical equation developed by Mueller (1992) is also shown in the figure. The equation is,

$$\epsilon = \epsilon_b + (1 - \epsilon_b) J_0(ar^*) e^{-br^*},$$

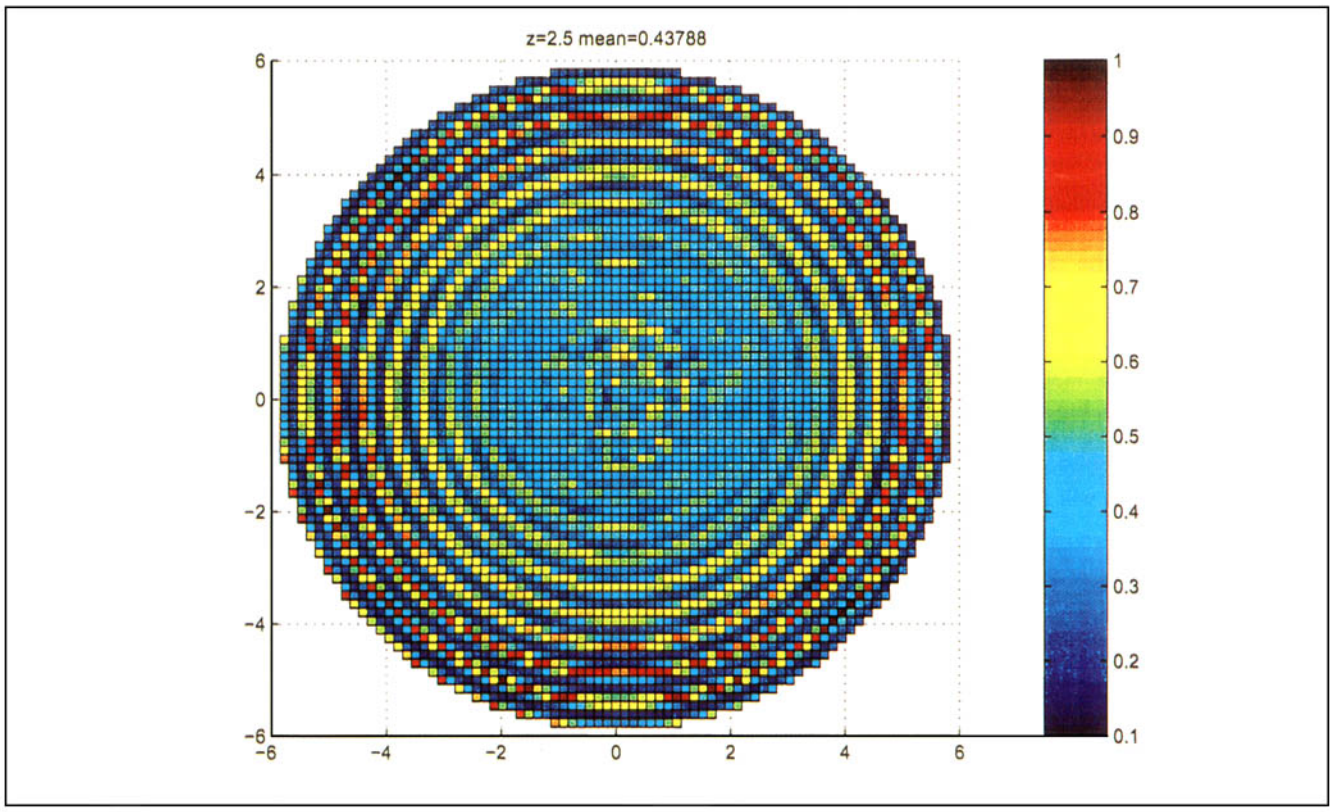

Figure 7. Solid sphere of particle size 12.7 mm is packed in a container with a 12 in. diameter.

Radial and circumferential porosity variation is shown at an axial position of $z = 2.5$. The mean porosity value at this plane is 0.438.

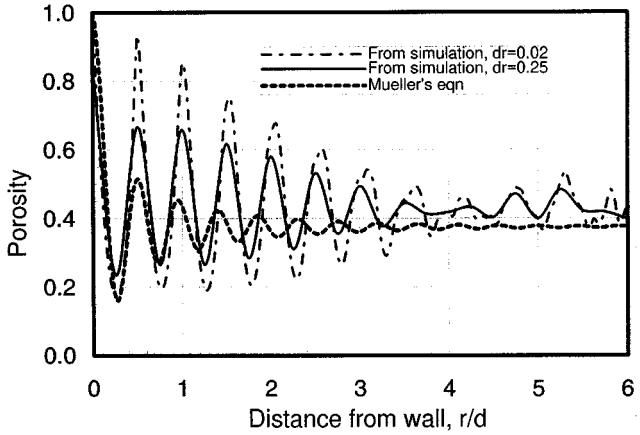

Figure 8. Solid sphere of particle size 12.7 mm is packed in a container with a 12 in. diameter.

Circumferentially averaged porosity variation in the radial direction is shown at an axial position of $z = 2.5$. Also shown is the empirical equation of Mueller (1992).

where

$$a = 7.45 - \frac{3.15}{D/d} \qquad \text{for } 2.02 \le D/d \le 13.0$$

$$a = 7.45 - \frac{11.25}{D/d} \qquad \text{for } 13.0 \ge D/d$$

$$b = 0.315 - \frac{0.725}{D/d}, \qquad r^* = r/d$$

$$\epsilon_b = 0.365 + \frac{0.220}{D/d}.$$

While the period of oscillation is the same in all three cases in Figure 8, the rate of decay of the amplitude depends on the sampling size in the radial direction. It shows the progressive loss of information with increasing sample size in the radial direction, a phenomenon that is to be expected. Thadani and Peebles (1966), who measured such variations using a nondestructive method, find similar oscillations on the scale of the sphere, but with the amplitude reduced more significantly than all other workers. It merely points out the fact that different researchers, using a different sample size in the radial direction, will find varying results for the amplitude variation, although the variance on the period will be much less.

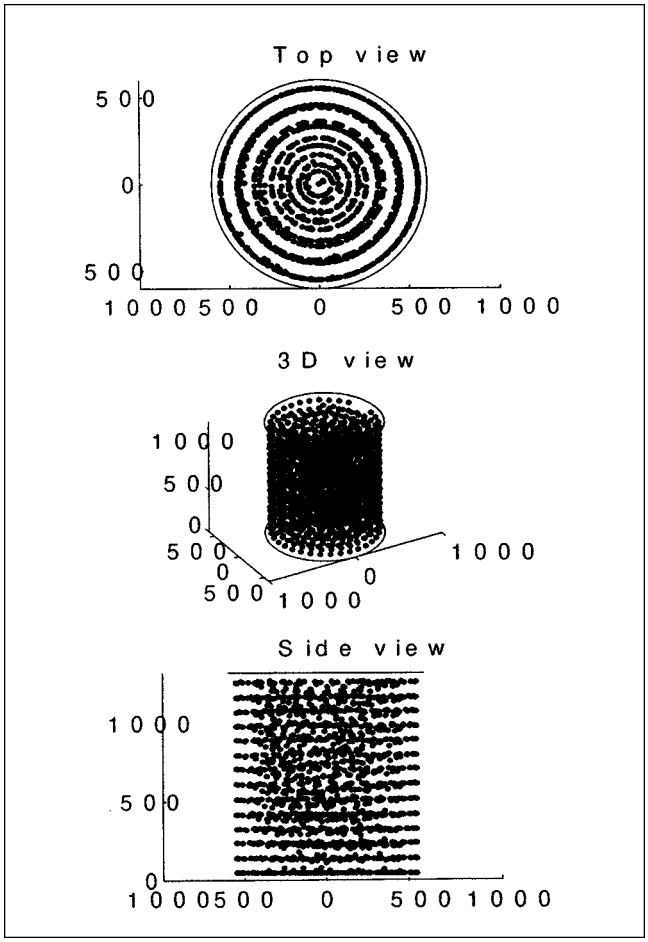

Figure 9. Point cloud view that shows the location of the center of each sphere for the case of ($d = 1.0$ in., $D = 12$ in.) with 1500 particles.

Clear structures from the bottom upward direction and radially inward direction are evident. True random packing is observed in the inner core region.

While this observation might appear self-evident, it does leave an unanswered question concerning the appropriate length scale for the averaging process. The usefulness of such detailed calculations or measurements on porosity variation is that they could be used as inputs in volume-averaged trans-

Table 2. Literature (Eckert, 1979) vs. Computed Macroproperties (Using PACK Program)

Packing Type	Dia. (mm)	Thick. (mm)	No. Pieces per m^3			Porosity			Surface Area m^2 per m^3		
			Eckert	Simul.	Error	Eckert	Simul.	Error	Eckert	Simul.	Error
Raschig rings (metal)	12.7	1.59	384,930	365,524	6.0%	0.73	0.74	1.4%	364	365	0.3%
	25.4	0.79	49,794	47,588	4.4%	0.92	0.93	1.1%	203	193	4.9%
Raschig rings (ceramic)	12.7	2.38	377,867	381,288	0.9%	0.64	0.63	1.6%	367	372	1.4%
	25.4	3.18	47,675	47,782	0.2%	0.74	0.73	1.4%	190	191	0.5%
Pall rings	25.4	0.61	49,441	46,509	5.9%	0.94	0.943	0.3%	207	186	10.1%
Minirings	25.4	0.61	—	152,149	—	—	0.94	—	—	205	—

port equations to model flow and mass-transport phenomena in packed columns. But the unambiguity observed here needs to be resolved. The magnitude of this problem is less severe for other packing objects like metal Pall rings and Raschig rings, as the fraction of the solid volume is significantly lower.

Finally, the centers of the spheres for ($d = 1.0$, $D = 12$) with 1,500 particles are shown in Figure 9 as a point cloud view. This figure shows merely a qualitative view of the distribution of spheres in the three-dimensional space. From the top and side views, a periodic variation on the scale of the particles is observed near the wall region and a true random packing state is achieved only in the inner core of the packing.

Packing simulation of uniform nonspherical particles

Results from several simulations are shown in Table 2 for nonspherical particles. The agreement of macroproperties such as number of pieces per m^3, specific surface area, and porosity for metal (thin) and ceramic (thick) Raschig rings is excellent. For example, a ceramic Raschig ring of size 12.7 mm is predicted to have a porosity of 0.63, which compares well with a mean value of 0.64 reported by Eckert (1979). For the same case, the specific surface area is predicted to be 367 m^2/m^3, which compares well with a value of 372 m^2/m^3 reported by Eckert (1979). For metal Raschig rings of size 25.4 mm, the predicted porosity is 0.93, while the value reported by Eckert (1979) is 0.92.

Results for the cascade minirings are also shown in Table 2, although comparison with literature values are not available. The errors are somewhat higher for Pall rings, but within an acceptable range of about 10%.

Figures 10 and 12 show the porosity variation for metal Raschig rings of diameters 0.5 in. (12.7 mm) and 1.0 in. (25.4 mm), respectively, at three different axial locations of $z = 0.5$ in., 1.5 in., and 2.5 in. Figures 11 and 13 show a three-dimensional-perspective view of the porosity variation in the angular and radial direction at a plane of $z = 1.5$ in. These correspond to different container diameters of 12 in. and 36 in., respectively. In both the containers, the porosity near the wall

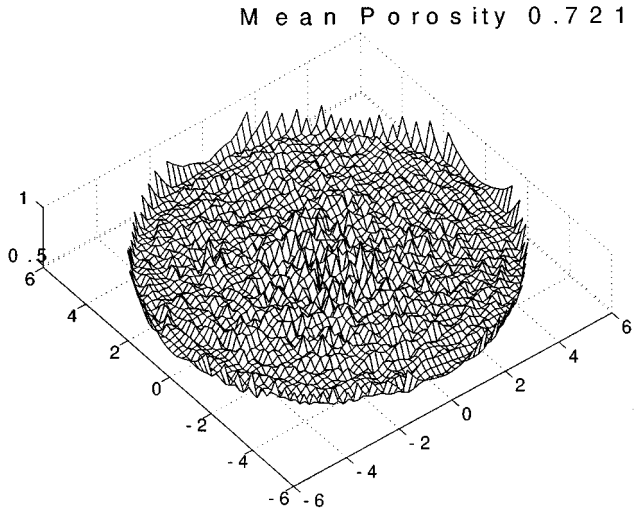

Figure 11. Metal Raschig rings.

Particle size is 12.7 mm. Container diameter is 12 in. Three-dimensional perspective plot of porosity variation is shown at an axial position of $z = 1.5$. The mean porosity value at this plane is 0.721.

increases toward unity, as documented experimentally by Roblee et al. (1958). Both cases show oscillations on the scale of the particle diameter in the radial direction. Unlike the spherical particles, however, the amplitude of the variation is quite small. In the smaller container (Figure 10) the porosity increases gradually toward the center of the container, as seen with the 4-point moving average curve that detects the trend. A similar trend curve in Figure 12 shows the mean porosity to be more uniform in the radial direction away from the wall.

The simulation results for metal Pall rings of size 25.4 mm (1.0 in.) in a small container of diameter of 12 in. are shown in Figures 14 and 15. Circumferentially averaged porosity

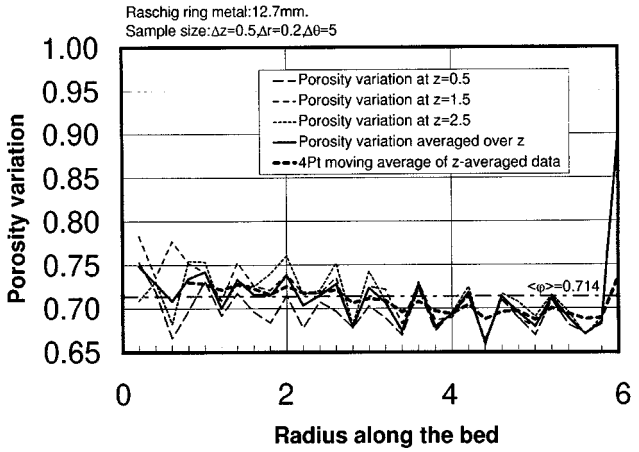

Figure 10. Metal Raschig rings.

Particle size is 12.7 mm. Container diameter is 12 in. Circumferentially averaged porosity variation is shown as a function of radial position at three different axial positions of $z = 0.5$, 1.5, 2.5.

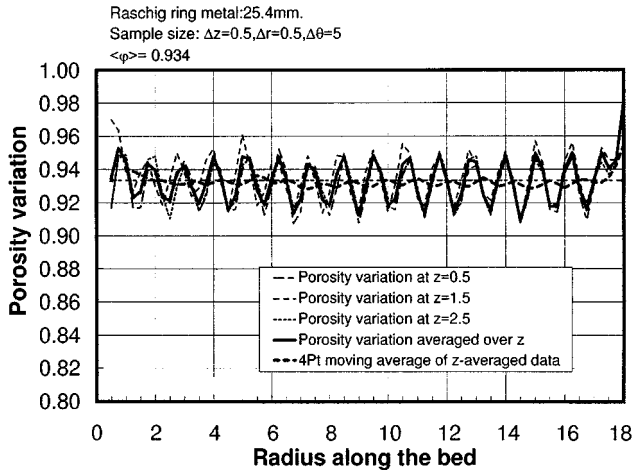

Figure 12. Metal Raschig rings.

Particle size is 25.4 mm. Container diameter is 36 in. Circumferentially averaged porosity variation is shown as a function of radial position at three different axial positions of $z = 0.5$, 1.5, 2.5.

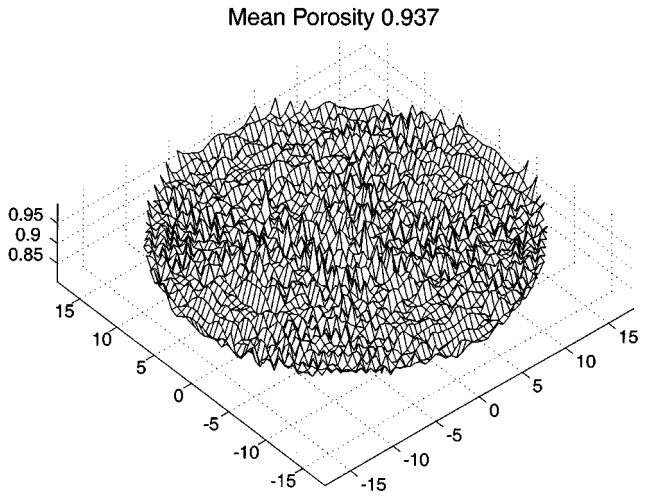

Figure 13. Metal Raschig rings.

Particle size is 25.4 mm. Container diameter is 36 in. Three-dimensional perspective plot of porosity variation is shown at an axial position of $z = 1.5$. The mean porosity value at this plane is 0.937.

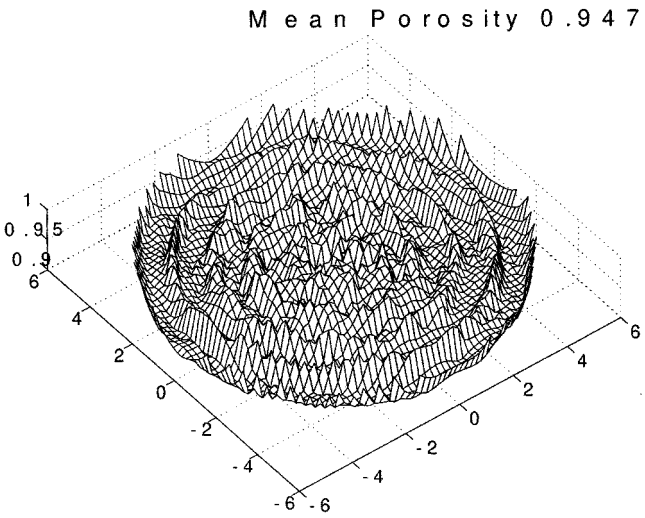

Figure 15. Metal Pall rings.

Particle size is 25.4 mm. Container diameter is 12 in. Three-dimensional perspective plot of porosity variation is shown as an axial position of $z = 2.5$. The mean porosity value at this plane is 0.947.

variation is shown as a function of radial position at three different axial positions of $z = 1.5$ in., 2.5 in., 3.5 in. in Figure 14, while both radial and angular variations are shown in Figure 15 as a three-dimensional perspective plot. The mean porosity is 0.94. All of the features observed for the Raschig rings, namely, particle-scale oscillations, approach unity near the container wall, and the gradual increase of the trend line toward the center of the container in a small container is also found for this case. The most significant difference is in the predicted number of pieces per m^3 of 45,432, which is much lower than the value of 49,441 reported by Eckert (1979).

A final set of results, shown in Figure 16, is for cascade minirings. This is similar to the structure of a Pall ring, except for the diameter-to-height ratio, which is 1:1 for Pall rings and 3:1 for minirings. Hence the number of particles per m^3 is approximately three times that for the cascade minirings, and the porosity and surface area are approximately the same as for Pall rings of the same nominal size. The mean porosity is predicted to be 0.94.

Conclusion

We have developed an algorithm and, based on that, a computer program to simulate the construction of random packing in a cylindrical container. The most significant improvement over previous such studies is the ability to handle arbitrarily complex-shaped packing objects. The algorithm uses advances in collision detection schemes to pack objects one-by-one, as close to each other as possible. The dynamics

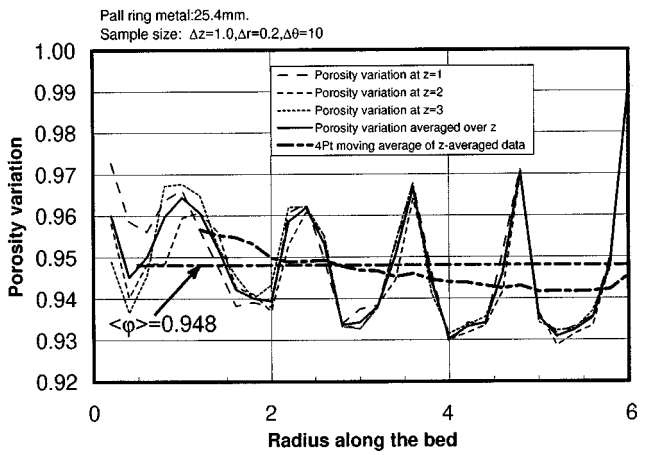

Figure 14. Metal Pall Rings.

Particle size is 25.4 mm. Container diameter is 12 in. Circumferentially averaged porosity variation is shown as a function of radial position at three different axial positions of $z = 1.5$, 2.5, 3.5.

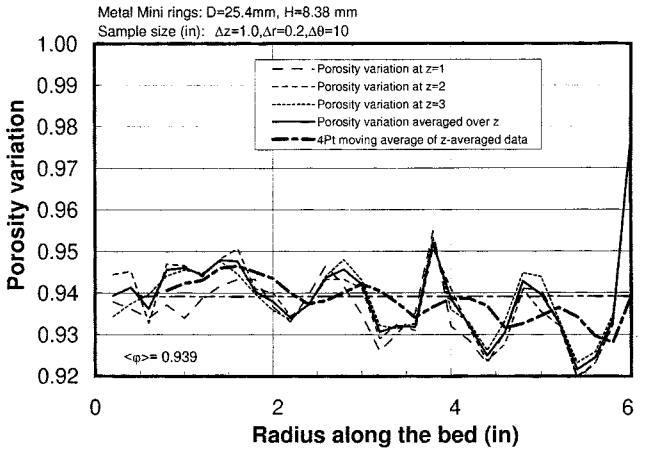

Figure 16. Metal minirings.

Particle size is 25.4 mm, height is 8.38 mm. Container diameter is 12 in. Circumferentially averaged porosity variation is shown as a function of radial position at three different axial positions of $z = 1.0$, 2.0, 3.0.

is not captured, but a stable equilibrium position for each packing object is sought. Once the packing simulation is completed, we have a complete description of every object in a global coordinate system, namely, its location and orientation are saved for further analysis. In particular we can compute mean porosity and porosity variation in space. Other characteristics such as specific surface area can also be determined.

The algorithm is validated against experimental data for simple geometries like solid-sphere packings and more complicated geometeries like Raschig and Pall rings. The real utility of this tool is in developing a conceptual design of new packing geometries and comparing the geometrical properties of such packing with existing shapes.

Literature Cited

Adams, D. J., and A. J. Matheson, ''Computation of Dense Random Packing of Hard Spheres,'' *J. Chem. Phys.*, **56**, 1989 (1972).

Benenati, R. F., and C. B. Brosilow, ''Void Fraction Distribution in Beds of Spheres,'' *AIChE J.*, **8**, 359 (1962).

Bennett, C. H., ''Serially Deposited Amorphous Aggregates of Hard Spheres,'' *J. Appl. Phys.*, **43**, 2727 (1972).

Chan, S. K., and K. N. Ng, ''Geometrical Characteristics of a Computer-Generated Three-Dimensional Packed Column of Equal and Unequal Sized Spheres—With Special Reference to Wall Effects,'' *Chem. Eng. Commun.*, **48**, 215 (1986).

Dodds, J. A., ''The Porosity and Contact Points in Multicomponent Random Sphere Packings Calculated by a Simple Statistical Geometric Model,'' *J. Colloid Interface Sci.*, **77**, 317 (1980).

Eckert, J. S., ''Design of Packed Columns,'' *Handbook of Separation Techniques for Chemical Engineers*, P. A. Schweitzer, ed., McGraw-Hill, New York (1979).

Goodling, J. S., R. I. Vachon, W. S. Stelpelug, and S. J. Ying, ''Radial Porosity Distribution in Cylindrical Beds Packed with Spheres,'' *Powder Technol.*, **35**, 23 (1983).

Gyenis, J., T. Brickle, and R. Hajdu, ''Mathematical Model for the Calculation of Internal Granule Porosity,'' *Powder Technol.*, **33**, 257 (1982).

Haughey, D. P., and G. S. G. Beveridge, ''Structural Properties of Packed Beds—A Review,'' *CJChE*, **47**, 130 (1969).

Matheson, A. J., ''Computation of Random Packing of Hard Spheres,'' *J. Phys. C. (Solid State Phys.*, **7** 2569 (1974).

Mueller, G. E., ''Radial Void Fraction Distributions in Randomly Packed Fixed Beds of Uniformly Sized Spheres in Cylindrical Containers,'' *Powder Technol.*, **72**, 269 (1992).

Mueller, G. E., ''Angular Void Fraction Distributions in Randomly Packed Fixed Beds of Uniformly Sized Spheres in Cylindrical Containers,'' *Powder Technol.*, **77**, 313 (1993).

Nolan, G. T., and P. E. Kavanagh, ''Computer Simulation of Random Packing of Hard Sphere,'' *Powder Technol.*, **72**, 149 (1992).

Nolan, G. T., and P. E. Kavanagh, ''Computer Simulation of Random Packings of Spheres with Log-Normal Distributions,'' *Powder Technol.*, **76**, 309 (1993).

Nolan, G. T., and P. E. Kavanagh, ''The Size Distribution of Interstices in Random Packings of Spheres,'' *Powder Technol.*, **78**, 231 (1994).

Ouchiyama, N., and T. Tanaka, ''Estimation of the Average Number of Contacts Between Randomly Mixed Solid Particles,'' *Ind. Eng. Chem. Fundam.*, **19**, 338 (1980).

Ouchiyama, N., and T. Tanaka, ''Porosity of a Mass of Solid Particles Having a Range of Sizes,'' *Ind. Eng. Chem. Fundam.*, **20**, 66 (1981).

Ouchiyama, N., and T. Tanaka, ''Porosity Estimation for Random Packings of Spherical Particles,'' *Ind. Eng. Chem. Fundam.*, **23**, 490 (1984).

Ouchiyama, N., and T. Tanaka, ''Porosity Estimation from Particle Size Distribution,'' *Ind. Eng. Chem. Fundam.*, **25**, 125 (1986).

Ouchiyama, N., and T. Tanaka, ''Porosity Estimations of Fixed Assemblages of Solid Particles with Different Packing Characteristics,'' *J. Chem. Eng. Jpn.*, **21**, 157 (1988).

Ouchiyama, N., and T. Tanaka, ''Predicting the Densest Packings of Ternary and Quaternary Mixtures of Solid Particles,'' *Ind. Eng. Chem. Res.*, **28**, 1530 (1989).

Plastock, R. A., and G. Kalley, *Schaum's Outline of Theory and Problems of Computer Graphics*, McGraw-Hill, New York (1986).

Ponamgi, M., D. Manocha, and M. Lin, ''Incremental Algorithms for Collision Detection Between Solid Models,'' *IEEE Trans. Visualization Comput. Graphics* **3**, 51 (1997).

Powell, M. J., ''Computer-Simulated Random Packing of Spheres,'' *Powder Technol.*, **25**, 45 (1980).

Roblee, L. H. S., R. M. Baird, and J. W. Tierne, ''Radial Porosity Variations in Packed Beds,'' *AIChE J.*, **4**, 460 (1958).

Scott, G. D., ''Packing of Equal Spheres,'' *Nature*, **188**, 980 (1960).

Shah, Y. T., ''Recent Advances in Trickle Bed Reactors,'' *Concepts and Design of Chemical Reactors*, S. Whitaker and A. E. Cassano, eds., Gordon & Breach, New York (1986).

Suzuki, M., and T. Oshima, ''Verification of a Model for Estimating the Void Fraction in a Three-Component Randomly Packed Bed,'' *Powder Technol.*, **43**, 147 (1985).

Suzuki, M., A. Yagi, T. Watanabe, and T. Oshima, ''Estimation of the Void Fraction in a Bed Randomly Packed with Particles of Three Sizes,'' *Int. J. Chem. Eng.*, **26**, 491 (1986).

Thadani, M. C., and F. N. Peebles, ''Variation of Local Void Fraction in Randomly Packed Beds of Equal Spheres,'' *Ind. Eng. Chem. Process Des. Dev.*, **5**, 265 (1966).

Tory, E. M., B. H. Church, M. K. Tam, and M. Ratner, ''Simulated Random Packing of Equal Spheres,'' *Can. J. Chem. Eng.*, **51**, 484 (1973).

Visscher, W. M., and M. Bolsterli, ''Random Packing of Equal and Unequal Spheres in Two and Three Dimensions,'' *Nature*, **239**, 504 (1972).

Wakao, N., and S. Kaguei, *Heat and Mass Transfer in Packed Beds*, Gordon & Breach, New York (1982).

Yu, A. B., and N. Standish, ''Porosity Calculations of Multi-Component Mixtures of Spherical Particles,'' *Powder Technol.*, **52**, 233 (1987).

Yu, A. B., and N. Standish, ''An Analytical-Parametric Theory of the Random Packing of Particles,'' *Powder Technol.*, **55**, 171 (1988).

Yu, A. B., and N. Standish, ''Estimation of the Porosity of Particle Mixtures by a Linear-Mixture Packing Model,'' *Ind. Eng. Chem. Res.*, **30**, 1372 (1991).

Manuscript received Mar. 5, 1999, and revision received June 30, 1999.