XIX International Conference on Water Resources
CMWR 2012
University of Illinois at Urbana-Champaign
June 17-22, 2012

COMPUTER GENERATED PARTICLE ARRANGEMENT FOR PORE SCALE MODELING

Rusen Sinir[*], Xiaofeng Liu[†], Yijiu Jiang[††]

[*] University of Texas at San Antonio
Department of Civil and Environmental Engineering
e-mail: rusensnr@gmail.com

[†] University of Texas at San Antonio
Department of Civil and Environmental Engineering
One UTSA Circle, San Antonio TX 78249 USA
e-mail: xiaofeng.liu@utsa.edu, web page: http://engineering.utsa.edu/~xiaofengliu/

[††] University of Texas at San Antonio
Department of Civil and Environmental Engineering
e-mail: jaq526@my.utsa.edu

Key words: Pore scale modeling, collision detection, rigid body dynamics, fluid dynamics

Summary. A new method based on collision detection and rigid body dynamics is proposed to generate physically correct particle arrangement for pore scale modeling. This method can automatically generate artificial porous media given the information of the 3D particle shapes. The surface shapes of moderately sized particles can be acquired through 3D scanner. This paper will demonstrate the process of digitization, automatic arrangement of the particles, and flow simulations.

1 INTRODUCTION

The creditability of pore scale modeling is affected by how well the real pore spaces are represented in the model. Past investigations have either assumed the particle arrays in space (most time spheres) or more recently used high resolution imaging techniques such as CT-scans. In this research, we propose to use the method in computer graphics and collision detection theories to artificially generate porous media. The porous media particles and pores should be physically correct which requires the particles to be stable in space and not penetrate into each other.

When the physically correct spatial arrangement of the particles is available, it is imported into a computational fluid dynamics (CFD) code to simulate the porous media flow. In the code, the particles are represented using immersed boundary method (IBM). This tool makes it possible to do pore-scale simulation based on physical and realistic pore spaces. It also makes it possible to estimate porous media properties such as hydraulic conductivity which provides a bridge between multiple scales. A demonstration is given to show the results from the computational model. Laboratory experiments are scheduled for future comparison.

2 MODEL DESCRIPTIONS

2.1 Generation of the porous media

The following steps are used to prepare the spatial arrangement of particles. First, the particles (gravel for this case) are digitized using our 3D laser scanner and the surface meshes are stored. Then we use the routines from computer graphics and collision detection theories to model the process of filling a cylindrical column with the digital gravels. The package we used is BulletPhysics[3]. We assumed that each gravel is a rigid body with certain properties such as density, friction, restitution, etc. When the column is full, the collision detection and rigid body dynamics algorithms continue until the system reaches equilibrium. The spatial configurations of these stone particles (center of mass and principle axis) are recorded which uniquely defines their correct position and orientation.

The beauty of this approach is that it is extremely easy to generate hundreds of virtual porous media columns in a very short period of time. It provides a viable way to do Monte Carlo type of simulations which is important due to the random nature of the pore space distribution.

2.2 Fluid dynamics model

The fluid dynamics model we used for the simulation of porous media flow is the open source computational fluid dynamics code OpenFOAM[1]. However, to do pore scale modeling, the existing fluid solvers in OpenFOAM are not suitable. The reason is that OpenFOAM requires an unstructured body-fitted mesh for the particles and the pores. This is impossible for our case due to the extreme complexity.

Instead, a new fluid solver was developed to implement an innovative immersed boundary method (IBM)[2]. Immersed boundary method was first proposed in Peskin (1977). This method does not need a body-fitted mesh to describe the solid boundary exactly. It only approximates the effect of the boundary by imposing the no-slip condition. The accuracy of the method depends on the mesh resolution and the specific method used to enforce the immersed boundary. Most of the immersed boundary methods are developed for structured meshes. However, an IBM based on unstructured meshes will be more attractive since other than the immersed boundaries there might be other features which can be better modeled with an unstructured mesh.

The essence of our new immersed boundary method is the interpolation technique for the ghost cells which are used to enforce the boundary condition with the internal cells. On structured meshes, the neighbor information of a cell point can be easily retrieved. However, on unstructured mesh, the neighboring relationships are less clearly defined. There are no definitions of east, west, north, south, top, and bottom. Cells are connected by common faces. We proposed a method which does not require the explicit notation of neighboring information. This method uses an influence sphere to search for the closest cells. The radius of the sphere cannot be too small or too big. If it is too small, there might be no cell in the influence sphere. On the other hand, if it is too big, the accuracy of the method will be low.

3 SIMULATION RESULTS

As the first step, our collision detection and rigid body dynamics tool was used to simulate the processing of placing particles into a cylindrical basket (Figure 1). These gravel particles were digitized using our 3D laser scanner. These particles are irregularly shaped as they are in nature. They have a mean diameter of 2 inches.

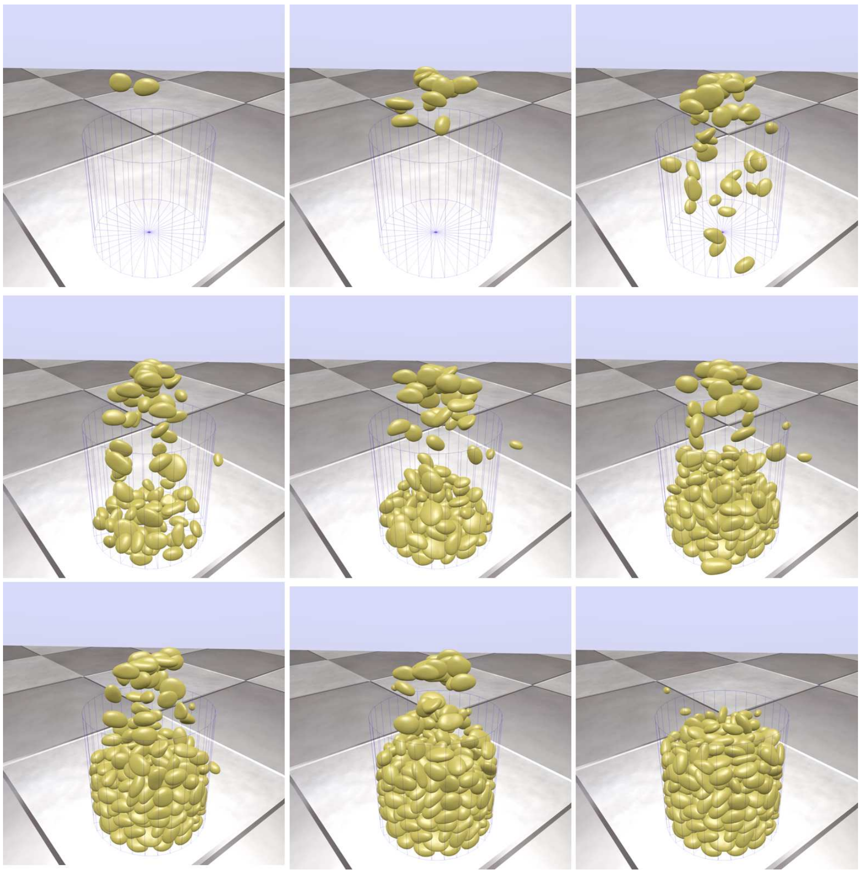

Figure 1. Generation of a virtual porous media column using computer tools from collision detection and rigid body dynamics. Irregularly shaped sand particles were randomly picked and dropped into a circular basket until it is full.

As the particles descending into the column, they will collide and interact with each other.

Our tool has the capability to model these interactions. After the column is filled up, the simulation will continue until every particle is in equilibrium. At this state, the configuration of the sand column is physical, which means sand particles will not penetrate into each other and they are stable in space. The final product of this step is the position and orientation of each particle. One could easily do further analysis on the properties of the porous media, such porosity, pore size distributions, etc. There properties are important factors for the porous media flow.

This second step is to import the generated porous media into our fluid simulation code. A flow velocity of 0.01 m/s was imposed on the top of the column and the flow was forced to move downward. In Figure 2, the streamlines are plotted to show the flow paths in the pore space.

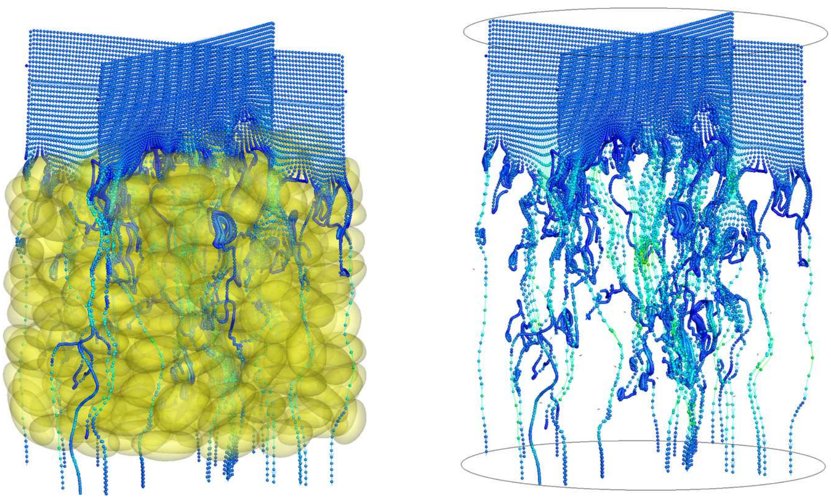

Figure 2. Streamlines originating from two cross-lines on top of the porous media column. Flow is from top to bottom. Details of the flow paths through the pores are clearly revealed.

The flow patterns can also be plotted over several cross sections. In Figure 3, the velocity magnitude on three horizontal cross sections at z=1/4H, 1/2H, and 3/4H are plotted, where H is the depth of the porous media column. It is observed that the model captured the high velocity at the narrow throat between particles. It is noted that the velocities inside the particles are not physical since they are the result of the immersed boundary method.

In Figure 4, the velocity and the pressure on the vertical cross section through the center are plotted. Again, the velocity vectors show the squeezing effect between particles. It is sometime also interesting to analyze the pressure distribution in the porous media. This pressure distribution can be used to compare with for example the classical Darcy's law.

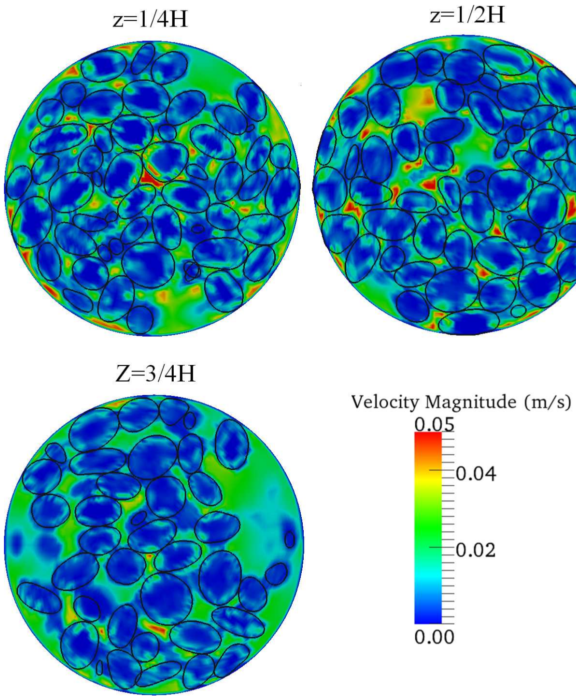

Figure 3. Velocity magnitude contour at three horizontal cross sections at z=1/4H, 1/2H, and 3/4H.

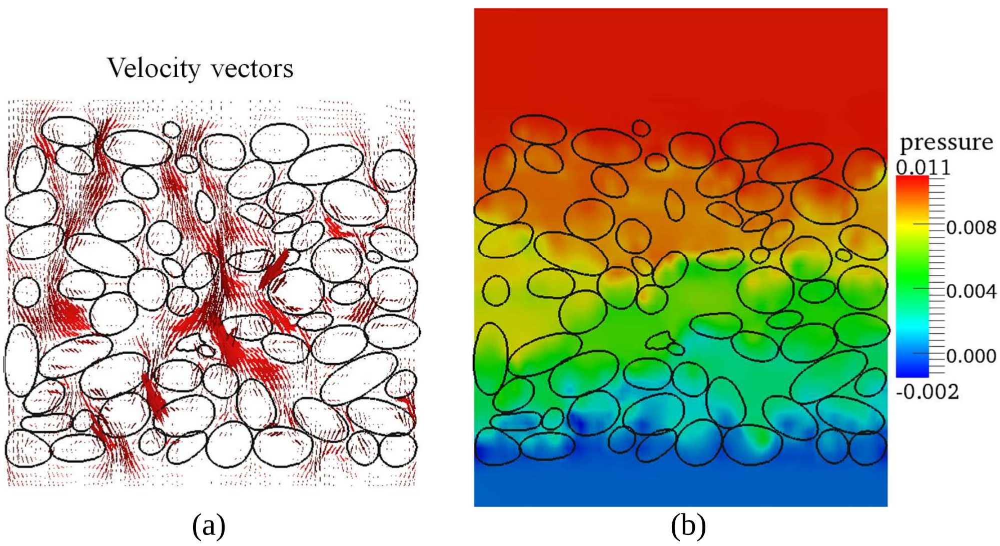

Figure 4. Plots of simulation result on a vertical plane through the middle of the porous media column: (a) Velocity vectors, (b) Pressure distributions.

4 DISCUSSION AND CONCLUSIONS

This paper introduced a new methodology to artificially generate porous media. The physically correct configuration of the particles from this method can be imported into a fluid dynamics solver to do pore-scale modeling. This makes it very attractive since more physics can be added to this model. For example, one can track the transport of nano particles or contaminants at the pore scale. The dynamics of these transports can be directly added to the model. This methodology is also very versatile since theoretically it can generate any pore distribution as long as one can provide a reprehensive distribution of particle shapes.

However, there are still some drawbacks for this method. First, the accuracy of this method depends on how accurate the particles surfaces are digitized. The surfaces cannot be too refined since the performance of the collision detection algorithms deteriorates very quickly when the number of faces increases. Secondly, the accuracy of the flow field depends on the resolution of fluid simulations. These accuracy issues, however, can be improved with the increase of computational power and capacity.

5 ACKNOWLEDGEMENTS

We acknowledge the financial support through a Tenure-Track Research Award Competition (TRAC) award by the University of Texas at San Antonio (UTSA).

REFERENCES

[1] OpenCFD. OpenFOAM User's guide, version 2.0.1, http://www.openfoam.com/. (2011).

[2] X. Liu. Toward More Realistic 3D Flow Simulations around a Pile with Porous Scour Protection, In the proceedings of the 6th International Conference on Scour and Erosion, Paris, France, (2012).

[3] E. Coumans. Bullet 2.76 Physics SDK Manual, http://bulletphysics.com.(2011)

[4] C.S. Peskin. Numerical analysis of blood flow in the heart, Journal of Computational Physics. 25: 220–252. (1977)