Conference Paper

Modeling of Gravitational Separation By the Method of Smoothed Particles Hydrodynamics (SPH)

V. P. Tsymbal[1], P. A. Sechenov[1], and A. A. Olennikov[2]

[1]The Siberian State Industrial University, 42, Kirova street, Novokuznetsk, Russia
[2]The Tyumen Industrial University, 6, Volodarsky street, Tyumen, Russia

Abstract

The article deals with the peculiarities of solving the problem of numerical simulation of gravity separation of dispersed particles. A simulation model is created by using the Monte Carlo method, in which the 'first principles' (elementary particles) are particles of the charge and reaction products. The object-oriented language ActionScript 3.0 was chosen as the programming language. At the same time, the most difficult (computational) task was to find neighbors (complexity N^2). In this article, the comparison analysis of the improved algorithm of neighbors search of complexity ($2 \cdot N \cdot k$) with standard neighbors search is given; the object of comparison is the quantity of the displayed particles moving in real time.

Keywords: modeling of flows, gravity separator, the Monte Carlo method, smoothed particles, complexity of the algorithm, neighbors search

Corresponding Author:
V. P. Tsymbal
tsymbal33@mail.ru

Received: 6 June 2018
Accepted: 15 June 2018
Published: 17 July 2018

Publishing services provided by Knowledge E

Selection and Peer-review under the responsibility of the TIM'2018 Conference Committee.

1. Introduction

In connection with the progress of computer technology, the role of mathematical modeling methods for studying complex interrelated physical processes continues to grow. Numerical modeling requires both powerful computer technology and numerical algorithms and codes that allow efficient use of computers and study interesting physical problems with high accuracy and a flexible approach for introducing new physical processes.

The aim of the article is to compare the speed of the neighbors search algorithms in the simulation model of the gravitational separator [1]. On the basis of existing and well-proven methods for solving the system of gas-dynamic equations, a large number of software packages have been developed for modeling currents in order to predict their characteristics and operating parameters of modern engineering devices: Fluent [2], FlowVision.

How to cite this article: V. P. Tsymbal, P. A. Sechenov, and A. A. Olennikov, (2018), "Modeling of Gravitational Separation By the Method of Smoothed Particles Hydrodynamics (SPH)" in *VII All- Russian Scientific and Practical Conference of Students, Graduate Students and Young Scientists on "Heat Engineering and Computer Science in Education, Science and Production"*, KnE Engineering, pages 199–204. DOI 10.18502/keg.v3i5.2670

2. Comparison of Numerical Methods

At present, the following methods are used from a wide range of numerical methods: Smooth Particle Hydrodynamics (SPH) and Euler methods on adaptive Mesh Refinement (AMR) networks [3]. In this article, the gridless Lagrange method is chosen, because the absence of a grid is especially important. In the SPH method, a spatially distributed set of particles is used to represent the gaseous medium. Each particle moves under the action of forces directed at it, the distribution of particles in the system changes in the course of its evolution. Hydrodynamic quantities, such as density or pressure, are defined as the result of the contribution of particles in a certain region.

One of the advantages of the SPH method over grid methods is the use of a constant amount of computing resources during the entire counting time. Unlike methods using adaptive grids, which change the number of calculated points in space in order to refine the results, the method of smoothed particles provides high resolution at a constant number of particles. Since particles are concentrated in areas of higher density, they guarantee high accuracy of calculations.

The task of creating an imitation model of the processes of physical interaction of disperse particles of charge, which are in the flow of gas, does not have a deterministic solution. One of the possible ways to solve it is simulation modeling using Monte Carlo methods and 'first principles'.

The Monte Carlo method appeared more than half a century ago [4], but interest in this method [5, 6] is increasing. The most complex and close to the imitation model problems [7], considered further are connected with the passage of beams of elementary particles through substances (obstructions).

3. Physical Formulation of the Problem

The core of the model considered further is the flow of a condensed particle over a vertical gas stream. First of all, let us consider the mechanism of the flow and interaction of dispersed particles of the charge and reaction products in the vertical flow of the carrier gas.

Under dynamic equilibrium conditions, the D'Alembert principle for a moving particle leads to the equation:

$$F_G - F_C - F_A = m\frac{d\omega_s}{dt}, \tag{1}$$

where ω_s – speed of motion of a spherical particle. Depending on the ratio of these forces, the particle can rise, fall, or remain stationary.

Gravity is expressed as:

$$F_G = mg. \tag{2}$$

Force of pressure of a stream:

$$F_c = c \cdot S \cdot p \cdot \frac{\omega^2}{2}, \tag{3}$$

where S – the cross-sectional area of streamlined heat along the midsection (the area of the projection of the body on a plane perpendicular to the velocity vectors of the oncoming stream); p – mixture density; ω – mixture flow rate; C – coefficient of drag, which depends on the shape of the streamlined body and the Reynolds number.

Lifting or Archimedes force:

$$F_A = \frac{m}{\rho_s} \rho \cdot g. \tag{4}$$

From equations (1–4) we obtain:

$$\frac{d\omega_s}{dt} = g\left(1 - \frac{\rho}{\rho_s}\right) - \frac{A \cdot \rho \cdot \omega^2}{2 \cdot m} \cdot S. \tag{5}$$

At the same time, it is necessary to simultaneously display the process of interaction of a huge number of particles that collide: with each other, with walls and lattices of the gravitational separator, which imposes a certain imprint on the development of a technology for creating an imitation model of the processes discussed earlier. It also requires physical adequacy and visibility of animation, user-friendly (researcher) interface for statistical research and analysis.

4. Choice of Programming Language

Based on the analysis of existing approaches, it was decided to write a program on an object-oriented programming language, most suitable for these requirements. The following programming languages were considered: Delphi, MS Visual C# [8], and ActionScript 3.0 [9]. Due to the fact that the model being developed should be visualized and animated (motion, particle interaction in a gravity separator in real time), the object-oriented programming language ActionScript 3.0 was chosen which has the ability to display a large number of objects on the stage, the built-in functions of adding (removing) from an array of objects; functions of designing a 3D object on a plane, etc.

In the software implementation, the following main stages of computing:

1. Determine the new position of the particle, taking into account the forces acting on it, which depend on the characteristics of the flow (velocity and density), as well as checking the boundary conditions (collision with walls and gratings) and checking the particle emission from the separator.

2. Searching for neighbors for each particle, determining collisions and changes in velocity vectors for colliding particles.

3. Calculation of the flow (velocity and density) characteristics along the channel height, taking into account the current position of the particles.

The most time-consuming task is the problem of finding neighbors, which will be considered further.

5. Comparison of Neighbors Search Algorithms

As shown by the results of experiments on the simulation model, a complete neighbors search to determine the collisions and calculate the forces between the particles is not a trivial task of complexity O (N^2) [10]. As most particles are at a considerable distance from each other and only a small number of other particles are near it, it is not necessary to calculate the interaction forces with other particles. To cut off these particles, space is divided into cells, and the cell size should not be less than the possible movement of the particle in the direction in the time interval between the calculations of the neighbors search (in this case this is the distance the particle can fly in one-thirtieth of a second). Each of those cells contains objects that are in it. Thus, for each particle, the range of neighbors search is narrowed.

Algorithm for finding neighbors at each iteration:

1. Determine the position of the particles with allowance for the forces acting on one particle.

2. For each particle, find the number of the cell in which it is located.

3. For each particle, determine the intersection with the particles in the current cell and adjacent cells and, if they intersect, determine the new position and velocity vector of the particles.

The complexity of such an algorithm will be O (2·N·k), where N is the number of particles, and k is the number of neighboring particles in the cells.

The testing of this algorithm was carried out for a problem on the plane (two-dimensional space), where the particles in a real-time mode made a Brownian motion. Figure 1 shows the results of comparing two neighbors search algorithms.

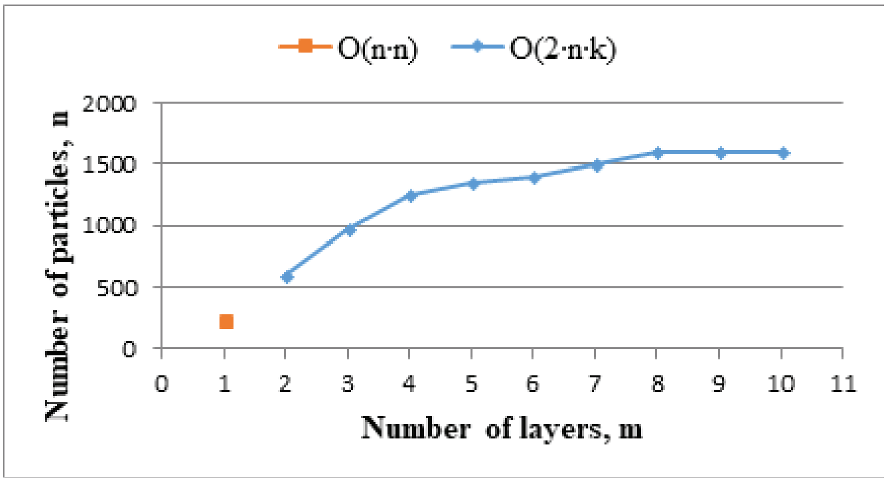

Figure 1: Dependence of the number of displayed particles on the choice of the algorithm and the number of layers in it.

In the problem of gravitational dust separation of manganese production [11] the number displayed particles increased from 150 to 600. This arises from the fact that for each particle a more complex problem (including a three-dimensional problem) is solved, and it is necessary to take into account the velocity distribution and density of the column reactor.

This approach to simulation modeling in combination with the proposed method allows solving a wide range of problems related to the study of the motion of heterogeneous streams in which particles with a large range of grain size distribution are weighed, that is, problems in which homogeneity or pseudo-homogeneity is not possible.

6. Summary

In the numerical implementation of the simulation model the method of smoothed particles was chosen, which makes it possible to solve this problem more effectively, since the uneven distribution of the height of the column reactor does not affect the accuracy of the results in comparison with the method on adaptive grids. In the implementation of the gravitational separation algorithm, the most resource-intensive task was to find the neighbors search, for which a more efficient method was proposed, which made it possible to increase the number of displayed and calculated particles by 4 times.

References

[1] Tsymbal, V. P., Pavlov, V. V., Sechenov, P. A., et al. (2016). Simulation modeling of interaction of dispersed particles in the jet-emulsion unit and gravitational separation. *Chernye Metally*, vol. 6, pp. 54–60.

[2] Valger, S. A., Fedorov, A. V., and Fedorova, N. N. (2013). Simulation of incompressible turbulent flows in the vicinity of poorly flowing bodies using a PC ANSYS Fluent. *J. Computing Technologies*, vol. 18, no. 5, pp. 27–40.

[3] Lazareva, G. G. (2010). Modern numerical methods of gravitational gas dynamics. *J. Vestnik Novosibirsk State University. Series: Mathematics*, vol. 10, no. 1, pp. 40–64.

[4] Metropolis, N. and Ulam, S. (1949). The Monte Carlo method. *Journal of American Statistical Association*, vol. 44, no. 247, pp. 335–341.

[5] Kroese, D. P., Brereton, T., Taimre, T., et al. (2014). Why the Monte Carlo method is so important today. *WIREs Computational Statistics*, vol. 6, pp. 386–392.

[6] Dunn, W. L. and Shultis, J. K. (2011). *Exploring Monte Carlo Methods*. Elsevier.

[7] Kroese, D. P., Taimre, T., and Botev, Z. I. (2011). *Handbook of Monte Carlo Methods*. Wiley.

[8] Lavrov, V. V., Spirin, N. A., Burikin, A. A., et al. (2012). Technology and software development tools used in the creation of the computer decision support system to solve technological problems in the blast-furnace process. J. *Athematic and Software Systems in the Industrial and Social Spheres*, vol. 2, pp. 147–152.

[9] http://help.adobe.com/en_US/FlashPlatform/reference/actionscript/3/index.html

[10] Gusev, D. I. (2012). Nearest neighbor search algorithm, *J. Software Products and Systems*, vol. 3, pp. 231–234.

[11] Sechenov, P. A., Tsymbal, V. P., and Olennikov A. A. (2016). Simulation model for separation of dust constituents of manganese production. *J. Cybernetics and Programming*, vol. 2, pp. 34–41.