Proceedings of the ASME 2010 3rd Joint US-European Fluids Engineering Summer Meeting and 8th International Conference on Nanochannels, Microchannels, and Minichannels
FEDSM-ICNMM2010
August 1-5, 2010, Montreal, Canada

FEDSM-ICNMM2010-30117

PARALLEL COMPUTATION FOR PARTICLE-GRID HYBRID METHOD

GUO Yun
College of Nuclear Science & Technology, Harbin Engineering University, Harbin 150001, China

Yuki ISHIWATARI
Nuclear Professional School, The University of Tokyo, Tokai-mura, Ibaraki 319-1188, Japan

Satoshi IKEJIRI
Nuclear Professional School, The University of Tokyo, Tokai-mura, Ibaraki 319-1188, Japan

Yoshiaki OKA
Faculty of Science and Engineering, Waseda University, Shinjuku-ku, Tokyo, 169-8555, Japan

ABSTRACT

In this paper, parallel computational technology is used in the numerical analysis code of particle-grid hybrid method. Particle-grid hybrid method is a rising solution method to analyze two-phase flow problem. It has shown its ability in several two-dimensional simulations. However, when this method was used to predict the droplet entrainment ratio of annulus flow the calculation time was insufferable. When the droplet occurs in the gas core, the density field varying near the droplet is strenuous. The pressure correction equations are very hard to be convergence. Hence, the MPI (Message Passing Interface) library is chosen as the parallel technology to decrease the calculation time. The grid and particle calculation parts are paralleled, separately. Jacobi point iteration method and ADI (alternating direction implicit) combined with divide and flow line programming techniques for grid calculation are discussed. All the particles are divided into several groups depending on the processor number. Then several cases are set for testing the parallel efficiency. Generally, with four processors the efficiency is about 60%. If the processor number is more than four the parallel efficiency will decrease rapidly. This method can accelerate the hybrid method; however, it still needs improving. Finally, some droplet entrainment cases calculated by parallel code are summarized.

INTRODUCTION

For some large scale problems it is necessary to use parallel computers [1]. However, in two-phase flow simulation field, even very small calculation domain with small scale grids, the parallel calculation is also necessary. Now, with the development of computation hardware the HPC (High Performance Cluster) with parallel code has became the key to solve such fluid problems. Richard T. Lahey Jr. presented the current state-of-the-art in our ability to perform the DNS (Direct Numerical Simulation) of various two-phase flows. Beyond all doubt, DNS need massively parallel processors [2]. In business codes, such as ANSYS CFD, it can supply more than 2000 processors to do parallel calculation with very high efficiency.

The particle-grid hybrid method is a new way to simulate two-phase flow. This method is developed by the University of Tokyo and it has been validated by several two-phase flow problems [3, 4]. Certainly, it needs more applications. When this method was tried to be optimized and used in large scale analysis the calculation time was bottleneck. In this paper, the particle-grid hybrid method will be parallelized to improve the calculation efficiency. LI proposed one parallel method for N-S equations based on MPI (message-passing interface) library [5]. It has been proved to be accurate and reliable in a 20 processors system with distributed memory. Zhou [6], Zhang [7], and Chi [8] studied the parallel algorithm of triangular equations by MPI, separately. Their results indicated that very high parallel efficiency can be obtained by MPI with Linux operation system. In present study, the original serial code was written by C Language, the MPI is chosen as the parallel library. This code is parallelized by two parts: the particle and the grid.

THE PARTICLE -GRID HYBRID METHOD

The hybrid method was described in detail in Liu and Guo's papers [3, 4]. Here it is only introduced briefly. In two-phase flow region, such as gas-liquid flow, the gas phase and liquid phase are represented by grid and particle respectively (fig. 1). The fixed black and green grids are set as the wall and the blue grids are the liquid. The main calculation for pressure and velocity fields is performed on the grid. The properties transfer from the particles to the grids by area weigh. After the pressure and velocity fields are updated by SIMPLE method the particles will obtain velocities decided by the coordinates and interpolation. Then the particles move to new positions. At last the new positions will be corrected by MPS (Moving Particle Semi-implicit) method [9, 10, and 11] for keeping the fluid density. The liquid surface is tracked by surface particles (if the particle density is smaller than a constant it is set as surface.). The CSF [12] model is used to compute the surface tension.

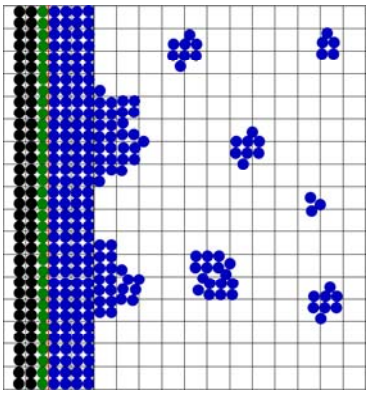

Fig. 1 the particle and grid

The serial code of hybrid code follows the calculation procedure shown in fig. 2. There are two parts can be parallelized: the mesh calculation by Finite Volume Method and the particle moving and position modifying. The former will cost about 80% CPU time and the later will cost about 13%. The left 7% is used for data transfer.

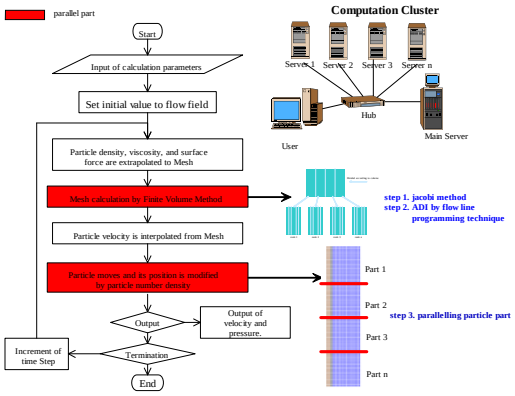

Fig. 2 Algorithm of the hybrid method

PARALLEL METHOD

1. Mesh Calculation

In mesh calculation the pressure correction equation is the solving point which can be written in the following:

$$\sum_{faces}\left[\frac{1}{\rho^{n+1}}\sum_{faces}\left(\overline{\overline{F}}_{press}\cdot\vec{n}\Delta S\right)^{n+1}\right]\cdot\vec{n}\Delta S=\frac{V}{\Delta t}\sum_{faces}\vec{v}^{*}\cdot\vec{n}\Delta S \qquad (1)$$

where t represents time, $\vec{v}$ and ρ are velocity vector and the density of the fluid, respectively. $\overline{\overline{F}}_{press}$ is the pressure flux, V is the volume occupied by the fluid, S is the surface area formed by the boundary of this volume, and $\vec{n}$ is the outward pointing unit vector, normal to S.

if we expanded eq. (1) in the grid of two dimension space shown in fig. 3 we can get:

$$\begin{aligned}&\frac{\Delta y}{\frac{1}{2}\left(\rho_{i,j}^{n+1}+\rho_{i+1,j}^{n+1}\right)}\left(dp_{i+1,j}^{n+1}\Delta y-dp_{i,j}^{n+1}\Delta y\right)+\frac{\Delta x}{\frac{1}{2}\left(\rho_{i,j}^{n+1}+\rho_{i,j+1}^{n+1}\right)}\left(dp_{i,j+1}^{n+1}\Delta x-dp_{i,j}^{n+1}\Delta x\right)\\&+\frac{\Delta y}{\frac{1}{2}\left(\rho_{i,j}^{n+1}+\rho_{i-1,j}^{n+1}\right)}\left(dp_{i-1,j}^{n+1}\Delta y-dp_{i,j}^{n+1}\Delta y\right)+\frac{\Delta x}{\frac{1}{2}\left(\rho_{i,j}^{n+1}+\rho_{i,j-1}^{n+1}\right)}\left(dp_{i,j-1}^{n+1}\Delta x-dp_{i,j}^{n+1}\Delta x\right)\\&=\frac{V}{\Delta t}\left(u_{i+1,j}^{*}\Delta y+v_{i,j+1}^{*}\Delta x-u_{i,j}^{*}\Delta y-v_{i,j}^{*}\Delta x\right)\end{aligned} \qquad (2)$$

where Δy and Δx are the boundary of control volume, $u_{i,j}^{*}$ and $v_{i,j}^{*}$ mean the temporary velocity. The subscripts i and j are the serial numbers of control volume.

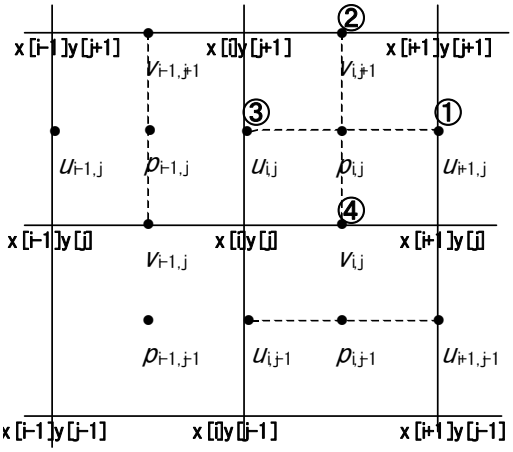

Fig. 3 p^{n+1} -control volume

The basic solution method of eq. (1) is Jacobi point iteration. It can be preceded by following equations.

$$dp_{i,j}^{n+1}=(B-C)/A \qquad (3)$$

Where:

$$A=\frac{\Delta y\Delta y}{\frac{1}{2}\left(\rho_{i,j}^{n+1}+\rho_{i+1,j}^{n+1}\right)}+\frac{\Delta x\Delta x}{\frac{1}{2}\left(\rho_{i,j}^{n+1}+\rho_{i,j+1}^{n+1}\right)}+\frac{\Delta y\Delta y}{\frac{1}{2}\left(\rho_{i,j}^{n+1}+\rho_{i-1,j}^{n+1}\right)}+\frac{\Delta x\Delta x}{\frac{1}{2}\left(\rho_{i,j}^{n+1}+\rho_{i,j-1}^{n+1}\right)}$$

$$B = \frac{\Delta y \Delta y}{\frac{1}{2}\left(\rho_{i,j}^{n+1} + \rho_{i+1,j}^{n+1}\right)} dp_{i+1,j}^{n+1} + \frac{\Delta x \Delta x}{\frac{1}{2}\left(\rho_{i,j}^{n+1} + \rho_{i,j+1}^{n+1}\right)} dp_{i,j+1}^{n+1}$$
$$+ \frac{\Delta y \Delta y}{\frac{1}{2}\left(\rho_{i,j}^{n+1} + \rho_{i-1,j}^{n+1}\right)} dp_{i-1,j}^{n+1} + \frac{\Delta x \Delta x}{\frac{1}{2}\left(\rho_{i,j}^{n+1} + \rho_{i,j-1}^{n+1}\right)} dp_{i,j-1}^{n+1}$$

$$C = \frac{V}{\Delta t}\left(u_{i+1,j}^{*} \Delta y + v_{i,j+1}^{*} \Delta x - u_{i,j}^{*} \Delta y - v_{i,j}^{*} \Delta x\right)$$

It is very simple and stable although its efficiency is low. And then the simulation domain can be divided into several regions, in each region the Jacobi point iteration is used and the data transfer only happens on the interface.

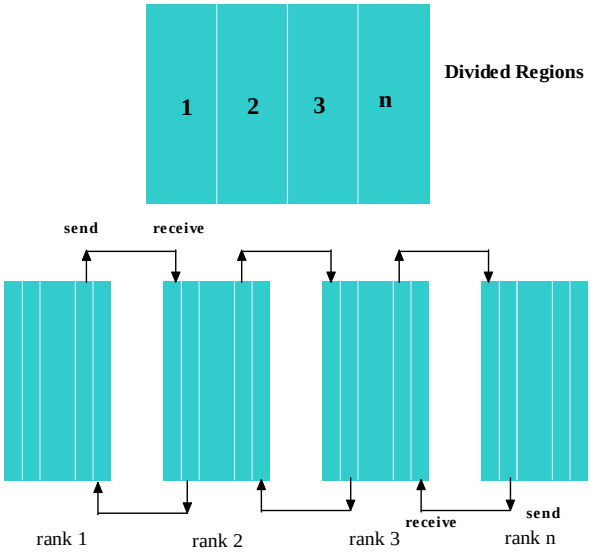

Fig. 4 Divided regions and data transfer

The traditional solution method is ADI (alternating direction implicit). For every direction the TDMA (tridiagonal matrix method) is used. The simulation domain can be divided into several regions just like that in Jacobi method. In each region the ADI and TDMA are used. The values of boundaries exchange after some time steps. If the transient varies quickly the values of boundaries exchange in every time step. Noticeable, the outside subregion with original boundary will start the calculation, and then is the inner parts. All the subregions are not synchronizing.

2. Particle Calculation

As mentioned before, the particle position needs correction according to the fluid density n^0. The modification of the particle number density n' is related to the modification of the velocity $\vec{v}'$:

$$\frac{1}{\Delta t}\frac{n'}{n^0} = -\nabla \cdot \vec{v}' \tag{4}$$

The velocity correction value $\vec{v}'$ is derived from a correction pressure gradient term as:

$$\vec{v}' = -\frac{\Delta t}{\rho}\nabla p' \tag{5}$$

where p' is the correction pressure, ρ is the density of fluid represented by particles.

All the moving particles are divided into several data groups. The data groups include not only the information of the particles itself but also the information of their neighbor particles. The new particles' positions are obtained in separate processor and then all the information will be collected by the main server. Hence all the particles can decide their new neighbors. This procedure makes the parallel efficiency lower.

PROBLEM SET-UP

The calculation domain is very similar to that shown in fig. 1. The area is 5mm×24mm and the grid number is 64×300. 5400 particles are used to represent the fluid. At initial the fluid film thickness is 0.24mm represented by 6 particles. Periodical condition is set for inlet and outlet. The properties of two-phase fluid are listed in table 1.

Table 1 Properties of two phases

Item	Gas	Liquid
Density(kg/m^3)	1.0	739.7
Viscosity(kg/m/s)	1.81e-5	150e-5
Surface tension(N/m)	20e-3	20e-3

However, for testing the parallel efficiency, different sets of grid and particle are also done. The detail will be shown in next section.

The parallel computational system consists with 8 processors and every 4 processor shared 32G Memory.

RESULTS

1. Validation of parallel code

Fig. 5 shows that the results of parallel code have a good agreement with those of serial code. The curves of gas and fluid velocities are almost overlap. Hence, the parallel code is right. It can be used in the next study.

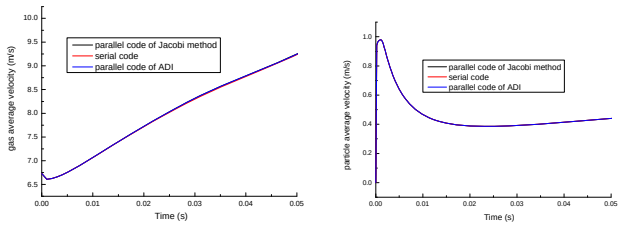

Fig. 5 Average velocities of two phases calculated by different codes

2. Parallel efficiency of Jacobi method of grid calculation

As mentioned before, Jacobi method is very simple and stable, and it is very easy to be paralleled. Table 2 includes the efficiencies based on this method by testing three grid structures. The efficiency of one CPU is set as *1*. In smaller scale grid the efficiency is very low, such as 64*300 grids, if four CPUs is used, the efficiency is only 0.16, which is even slower than one CPU. The reason is data transfer costs much time. However, in larger scale grid the efficiency becomes higher. In 256*1200 grids, with four CPUs the efficiency is 0.59.

Table 2 Parallel efficiencies of different grid structures

Grid structure	CPUs	Efficiency
64*300	1	*1*
64*300	2	0.33
64*300	4	0.16
168*800	1	*1*
168*800	2	0.62
168*800	4	0.49
256*1200	1	*1*
256*1200	2	0.83
256*1200	4	0.59

3. Parallel efficiency of ADI method of grid calculation

The results of parallel efficiency of ADI method are listed in table3 for three grid structures. The calculation condition is the same as that of Jacobi method. Compared with Jacobi method the parallel efficiency of ADI method is higher. In 256*1200 grids, with four CPUs the efficiency is 0.77.

Table 3 Parallel efficiencies of different grid structures

Grid structure	CPUs	Efficiency
64*300	1	*1*
64*300	2	1
64*300	4	0.62
168*800	1	*1*
168*800	2	0.97
168*800	4	0.83
256*1200	1	*1*
256*1200	2	0.99
256*1200	4	0.77

4. Parallel efficiency of particle calculation

Although particle calculation is faster than that of mesh its parallel efficiency is very low in one processor. According to the results shown in fig. 6, if the CPUs number is greater than 4, the speedup ratio is almost invariable.

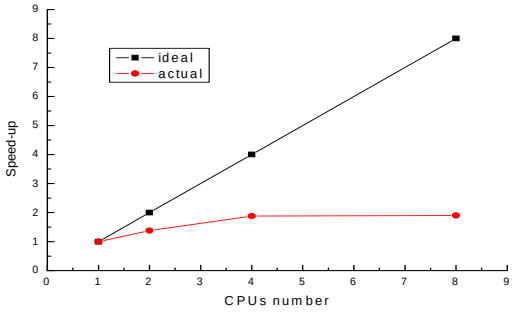

Fig. 6 Speed up of particle calculation

5. Some initial results of droplet entrainment ratio obtained by parallel code

Based on the parallel code some droplet entrainment problems are simulated. The initial fluid film thickness is 0.24mm, the surface tension coefficient is 0.03N/m, the densities are 998.2 and 1.1kg/m^3 (water and air), and the viscosities are 0.001 and 18.1×10^{-6}kg/m/s.

With four CPUs the speedup ration is about 2. The case cost about 96 hours for getting 0.13s simulation time shown in table 4. Fig. 7 shows the curves of gas and liquid velocities. Both of them are varied with time. Ishii and Wallis correlations are chosen as the reference. The comparing results are listed in table 4. Large differences between the numerical results and experimental correlations exist. The numerical results are still unstable. Some new droplets appear in the gas core and some droplets deposit on the fluid film. The simulation time is not enough.

Table 4 Results of droplet entrainment ratio

Item	**0.11(9.6, 0.2)*	0.12(8.5, 0.18)	0.13(7.5, 0.13)
Numerical result	**0.130*	0.177	0.279
Ishii correlation	0.024	0.017	0.012
Wallis correlation	0.103	0.082	0.079

**0.11(9.6,0.2):time(gas velocity, fluid velocity)*

**0.130:* $E = j_{fe} / j_f$ *, entrainment ratio*

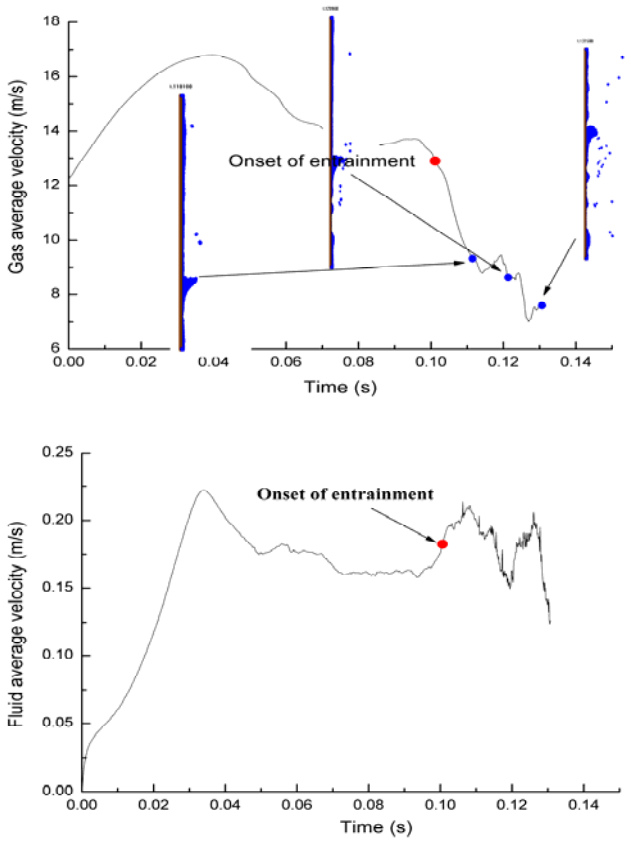

Fig. 7 Curves of gas and liquid velocities

CONCLUSIONS

In this paper a parallel computation method for hybrid method is introduced. And the parallel code is used to do the simulation of droplet entrainment ratio. Not to mince matters the method is not good enough to solve the calculation time problem of hybrid method. In the future, we try to increase the parallel efficiency by two ways. One is improving the parallel calculation method; the other is adopting new hardware equipment.

ACKNOWLEDGMENTS

The authors thank the Fundamental fund of Harbin Engineering University (HEUFT08013) and National Natural Science Funds (10805011) for financial support of this work.

REFERENCES

[1] G. Tryggvason, B. Bunner, A. Esmaeeli, et al., “A Front-Tracking Method for the Computations of Multiphase Flow”. Journal of Computational Physics, 169 (2001), 708–759.
[2] Richard T. Lahey Jr., “On the direct numerical simulation of two-phase flows”, Nuclear Engineering and Design, 239 (2009), 867-879.
[3] Jie Liu, Seiichi Koshizuka, Yoshiaki Oka, “A hybrid particle-mesh method for viscous, incompressible, multiphase flows”, Journal of Computational Physics, Vol. 202, pp.65-93.
[4] Guo Yun, et al., “Numerical analysis of the onset of droplet entrainment in annular two-phase flow by hybrid method”, Annals of Nuclear Energy, 37(2010), 230-240.
[5] LI Ning, Luo Jisheng, “Research on parallel computing algorithm of incompressible N-S equation based on MPI”. Chinese Computer Engineering and Applications, 2007, 43(9), 8-10.
[6] Zhou Junhu, et al., “Parallel solution of triangular system in direct numerical simulation”. Chinese Journal of Zhejiang University (Engineering Science), 2005, 39 (9), 1439-1444.
[7] Zhang Jing, Wang Fujun, Yu Yingduo, et al., “A Parallel Algorithm of Triangular Linear Systems Based on MPI”. Chinese Research on mechanization of farming, 2005, 4, 8-83.
[8] Chi Lihua, Liu Jie, Li Xiaomei, “An Effective Parallel Algorithm of Triangular Linear Equations”, Chinese Journal of Computers, 1999, 22(2), 218-221.
[9] S. Koshizuka, K. Okamoto and K. Furuta, “Development of computational techniques for nuclear engineering,” Prog. in Nucl. Energy, Vol. 32(1/2), 209.
[10] S. Koshizuka, A. Nobe and Y. Oka, “Numerical analysis of breaking waves using the moving particle semi-implicit method,” Int. J. Numer. Methods Fluids, Vol. 26, 751.
[11] K. Nomura, S. Koshizuka, Y. Oka and H. Obata, “Numerical analysis of droplet breakup behavior using particle method”, J. Nucl. Sci. Technol., Vol. 38, 1057.
[12] J. U. Brackbill, D. B. Kothe and C. Zemach, “A continuum method for modeling surface tension,” J. Comput. Phys., 100, 333