J. Plasma Physics (2006), *vol.* 72, *part* 6, *pp.* 845–849.
doi:10.1017/S0022377806005009 Printed in the United Kingdom

Parallel implementation of a three-dimensional electrostatic PIC–MCC method using an unstructured tetrahedral mesh

J.-S. WU and K.-H. HSU

Department of Mechanical Engineering, National Chiao-Tung University, Hsinchu 30050, Taiwan

(Received 12 August 2005 and accepted 4 November 2005)

Abstract. A parallel three-dimensional electrostatic particle-in-cell Monte Carlo Collision (PIC–MCC) code using an unstructured tetrahedral mesh is developed and validated in this paper. Poisson's equation is discretized by a nodal finite element method using a linear interpolation function. Charged-particle motion is traced cell by cell using a leap-frog method and the Boris scheme. Code is validated using quasi-one-dimensional radiofrequency gas discharge by observing the bi-Maxwellian electron energy probability distribution function at low pressure. Parallel performance of the PIC code is tested using a three-dimensional radiofrequency gas discharge on a PC-cluster system. Results show that parallel efficiency of 83% can be achieved at 32 processors with dynamic domain decomposition.

1. Introduction

In the past, there have been tremendous efforts in developing particle-in-cell Monte Carlo Collision (PIC–MCC) methods for plasma simulation using structured mesh (Birdshall 1991), while there have been very few studies concentrating on developing PIC–MCC methods using an unstructured mesh. Two recent rare examples using an unstructured tetrahedral mesh include Celik et al. (2003) and Spirkin and Gatsonis (2004). Not only does the unstructured mesh offer the flexibility of handling complicated geometry, but it also provides the possibility of applying dynamic domain decomposition (DDD) for parallel processing. In the current study, we intend to develop a parallel PIC–MCC method using a three-dimensional unstructured tetrahedral mesh with DDD for low-temperature plasma simulation.

2. Parallel three-dimensional PIC–FEM method

In the proposed PIC finite-element method (FEM), Poisson's equation for the electrostatic potential is discretized using Galerkin FEM with a linear shape function, while particles are pushed cell by cell on an unstructured tetrahedral mesh using a leap-frog method and the Boris scheme. The particle tracing algorithm is similar to that developed previously in the direct simulation Monte Carlo (DSMC) method (Wu and Lian 2003) by using cell-neighboring information derived from the element connectivity. Application of the current particle-pushing technique to

other types of cell is straightforward. Only non-zero entries of the coefficient matrix of the discretized Poisson's equation are stored using the compressed sparse row (CSR) scheme. The resulting sparse matrix equation is then solved by a parallel preconditioned (Jacobi) conjugate gradient method.

The PIC–MCC module with DDD is essentially the same as the serial version except for two places which require data communication between processors. One is as particles move across the inter-processor boundary, while the other is as the Poisson's equation is solved in parallel. Charge assignment from particle to nodes is obtained using the volume coordinates originated from FEM, while the field force on particles is obtained directly from the finite-element solution of the Poisson's equation. In addition, a sub-cycling technique, in which the ion time step is much larger (10 times) than the electron time step, is used to reduce the computational time throughout the study unless otherwise specified. Collision models include ionization, excitation and elastic collisions between electrons and neutrals, and charge exchange and elastic collisions between ions and atoms, which is similar to those used previously (Vahedi and Surendra 1995). With Nanbu's method (Nanbu 1994), only one random number is used to simultaneously determine whether collision occurs and which collisional event occurs at the end of the collisionless motion.

Further, DDD via a multi-level graph-partitioning technique (Karypis and Kumar 1995) is utilized to adaptively balance the workload among processors during the runtime. In the current implementation, we only repartition the domain at a constant number of time steps, rather than using the SAR (start-at-rise) scheme used previously (Wu and Tseng 2005). The choice of this implementation lies in its simplicity, while fairly satisfactory results can be obtained (shown later). In addition, the weighting of each graph vertex (cell center) for graph partitioning is set in proportion to the number of particles in each cell. Finally, this parallel PIC code is implemented on a memory-distributed machine (e.g. a PC cluster) using MPI for communication.

3. Results and discussions

3.1. Validation: quasi-one-dimensional radiofrequency gas discharge

A quasi-one-dimensional argon gas discharge with application of radiofrequency (RF) voltage source at the cathode and the grounded anode is considered to validate the parallel PIC–MCC code. Simulation conditions include: $Y = 20$ mm (gap), $f = 13.56$ MHz (RF), $V_{p-p} = 1000$ V (voltage amplitude), $\gamma = 0$ (secondary electron emission), $P = 20$ and 50 mTorr (background pressure). The electron time step of simulation is 3.7×10^{-11} s with approximately 200 cells across the gap using around 50 particles per cell. The ion time step is 10 times larger than that of electron time step. Results presented in the following are sampled after 300 RF cycles with an initial uniform number density of electrons and ions.

The averaged number density distribution and averaged energy distribution of both electrons and ions at $P = 50$ mTorr are shown, respectively, in Figs 1(a) and (b). Results (Fig. 1(a)) show that the averaged number densities of both electrons and ions are symmetric and reach 1.6×10^{16} m^{-3} in the center. It is essentially neutral in the center part of the discharge ($Y = 6$–14 mm), while it is positively charged in the sheath region near both electrodes. In addition, the averaged electron energy is about 3 eV in the bulk region, while it decreases down to zero at the electrode in

https://doi.org/10.1017/S0022377806005009 Published online by Cambridge University Press

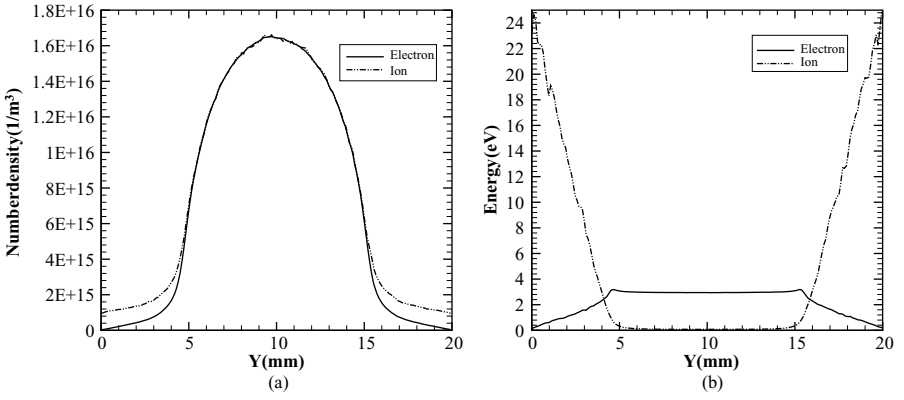

Figure 1. (a) Number density distributions (b) energy distributions of electron and ion in a RF gas discharge (argon, $f = 13.56\,\mathrm{MHz}$, $Y = 20\,\mathrm{mm}$).

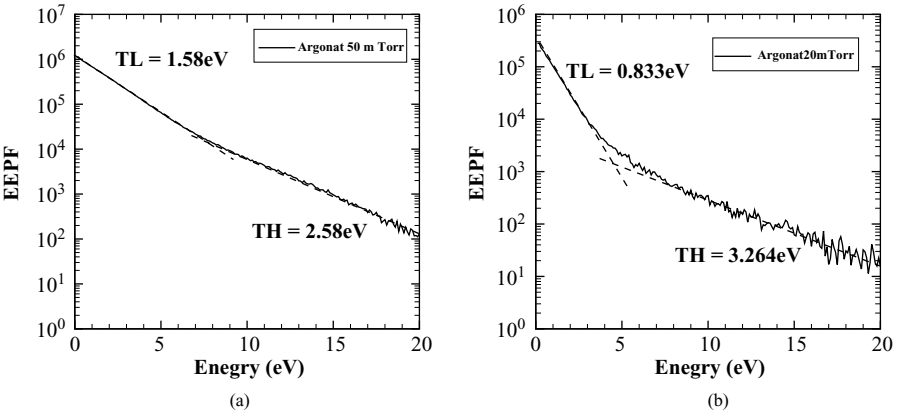

Figure 2. EEPF of a RF gas discharge: (a) $P = 50\,\mathrm{mTorr}$; (b) $P = 20\,\mathrm{mTorr}$.

the sheath region. The averaged ion energy is nearly zero in the bulk region, while it increases up to 25 eV at the electrodes due to the sheath potential.

Figures 2(a) and (b) respectively, illustrate the electron energy probability distribution function (EEPF) at pressures of 50 and 20 mTorr. At $P = 50\,\mathrm{mTorr}$, EEPF shows a weakly bi-Maxwellian distribution ($T_\mathrm{L} = 1.58\,\mathrm{eV}$, $T_\mathrm{H} = 2.58\,\mathrm{eV}$), while at $P = 20\,\mathrm{mTorr}$, it shows a strong bi-Maxwellian distribution ($T_\mathrm{L} = 0.833\,\mathrm{eV}$, $T_H = 3.264\,\mathrm{eV}$), which is comparable with previous experimental studies under similar conditions (Turner et al. 1993). At low pressures, the bi-Maxwellian EEPF reveals the stochastic electron heating mechanism, leading to the formation of cold bulk and oscillating hot tail electrons.

3.2. Parallel performance test: three-dimensional RF gas discharge

To test the parallel performance of the current parallel PIC–MCC code using DDD, we have used a three-dimensional RF gas discharge chamber at different simulation conditions. The argon gas discharge takes place between two parallel circular electrodes enclosed in a cylindrical chamber. A chamber wall with very high dielectric

https://doi.org/10.1017/S0022377806005009 Published online by Cambridge University Press

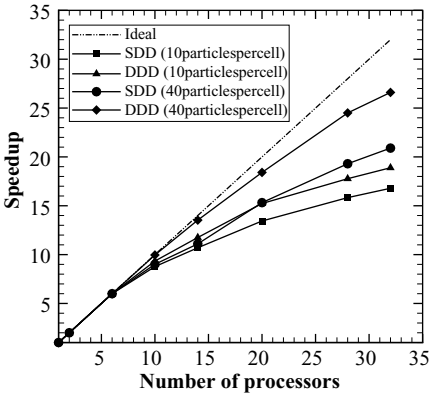

Figure 3. Parallel speedup as a function of the number of processors (HP-Itanium 64-bit).

constant is assumed, where we use Neumann boundary conditions for Poisson's equation solver. Considering geometrical symmetry, we only use $\frac{1}{16}$ of the full chamber for simulation. Simulation conditions are similar to those presented in one-dimensional RF gas discharge in the previous section except that 164 865 tetrahedral cells are used. The speedup test is conducted on HP-IA (64-bit) cluster parallel machines with up to 32 processors.

Figure 3 illustrates the resulting speedup as a function of the number of processors. Results show that speedup increases with increasing number of simulation particles. Also the use of DDD every $1500\Delta t$ improves the parallel performance appreciably as compared with the case using static domain decomposition (SDD), especially when there are more particles in the simulation. Note that DDD activated every $1500\Delta t$ results in the best parallel performance in the current simulation. Parallel efficiency as high as 83% can be obtained at 32 processors if 40 particles per cell is used. From the time breakdown analysis of the major components of the parallel code, we have found that the Poisson's equation solver is not efficient enough at this stage, which takes 31% and 16% of the total time, respectively, for the cases of 10 and 40 particles per cell. Further improvement of this solver using either a parallel preconditioner or direct matrix solver is currently in progress and will be reported in the near future.

Acknowledgements

The authors would like to acknowledge the financial support through the grants NSC-93-2120-M-009-004 and NSC-93-2212-E-009-015 of Taiwan. The parallel computing resources provided for this study by NCHC of Taiwan is also highly appreciated.

References

Birdsall, C. K. 1991 Particle-in-cell charged-particle simulations, plus Monte Carlo collisions with neutral atoms, PIC–MCC. *IEEE Trans. Plasma Sci.* **19**, 65–85.

Celik, M., Santi M., Cheng, S., Martinez-Sanchez, M. and Peraire, J. 2003 Hybrid-PIC simulation of a hall thruster plume on an unstructured grid with DSMC collisions. *Proceedings of the 28th International Electric Propulsion Conference*, Toulouse, France, IEPC-03-134.

Karypis, G. and Kumar, V. 1995 *Metis: Unstructured Graph Partitioning and Sparse Matrix Ordering, Version 2.0 User Manual*, Minneapolis, MN: Computer Science Department, University of Minnesota.

Nanbu, K. 1994 Simple method to determine collisional event in Monte Carlo simulation of electron–molecule collision. *Japan. J. Appl. Phys.* **33**, 4752–4753.

Spirkin, A. and Gatsonis, N. 2004 Unstructured 3D PIC simulation of the flow in a retarding potential analyzer. *Comput. Phys. Comm.* **164**, 383–389.

Turner, M. M., Doyle, R. A. and Hopkins, M. B. 1993 Measured and simulated electron energy distribution functions in a low-pressure radio frequency discharge in argon. *Appl. Phys. Lett.* **62**, 3247–3249.

Vahedi, V. and Surendra, M. 1995 Monte-Carlo collision model for particle-in-cell method: application to argon and oxygen discharge. *Comput. Phys. Comm.* **87**, 179–198.

Wu, J.-S. and Lian, Y.-Y 2003 Parallel three-dimensional direct simulation Monte Carlo method and its applications. *Comput. Fluids.* **32**, 1133–1160.

Wu, J.-S. and Tseng, K.-C. 2005 Parallel DSMC method using dynamic domain decomposition. *Int. J. Num. Meth. Eng.* **63**, 37–76.