Concurrent DSMC Method Combining Variable Time-Step and Adaptive Unstructured Mesh

Jong-Shinn Wu* and K.-C. Tseng

Department of Mechanical Engineering
National Chiao-Tung University
1001, Ta-Hsueh Road, Hsinchu, 30050, Taiwan

A parallel direct simulation Monte Carlo method is developed, which uses dynamic domain decomposition to solve the problem of load unbalancing among processors. In addition, variable time-step method and unstructured adaptive mesh are implemented, respectively, for reducing the computational time and increasing the accuracy of solution. Associated flow chart of the proposed DSMC method is illustrated in Fig. 1. Major difference between the current study and the previous related parallel DSMC studies [2,3] is the utilization of the state-of-the-art graph partitioning technique developed in the community of computer science. The current DSMC method is implemented on unstructured adaptive mesh using particle ray tracing technique by taking the advantages of the cell connectivity information [5]. Standard Message Passage Interface (MPI) is used to communicate data between processors. In addition, different strategies of applying Stop at Rise (SAR) [4] scheme are utilized to decide when to adjust the workload among processors. Briefly speaking, SAR method defines a degradation function, which represents the average idle time for each processor including the cost of repartition. Decision to repartition is made based on the detection of a minimum value of degradation function during the simulation. Corresponding parallel performance is analyzed using the results of a high-speed driven cavity flow on IBM-SP2 (memory-distributed, CPU 160 MHz, RAM 256 MB each) and IBM-SMP (memory-distributed, CPU 375 MHz, RAM 1 GB each) parallel machines up to 64 and 128 processors, respectively. In general, the implementation of dynamic domain decomposition reduces the runtime in the range of 30-60% as compared with that of static domain decomposition.

Variable Time-Step Method

Basic idea of variable time-step method in DSMC is to enforce the flux conservation (mass, momentum and energy) of moving simulated particle when crossing the interface between two neighboring cells. If we scale the local cell time-step to the local cell size (or local mean free path), then the best way to enforce flux conservation is to change the particle weight factor without destroying or cloning the particles during particle movement across the cell interface. The cloning of particle can generally induce unpredictable random-walk effects in a statistical simulation like DSMC. One of the advantages in implementing the variable time-step is to reduce both the simulated particle numbers and transient time to steady state, when the sampling normally starts in DSMC. This will result in appreciable time saving for the steady DSMC simulation as can be shown in Fig. 2, where the simulated particle numbers is illustrated as a function of simulation time steps for a 2-D DSMC computation using constant time-step and variable time-step, respectively.

Unstructured Adaptive Mesh

To obtain an accurate numerical solution in DSMC requires an optimum mesh adaptive to flow-field solution as that in CFD. We have implemented an *h-refinement* technique to locally refine the triangular (2-D) [6] and tetrahedral (3-D) mesh [7] following the DSMC solution. In

*E-mail: chongsin@cc.nctu.edu.tw

addition, combination with variable time-step method can be shown to be very effective in DSMC, especially for the external hypersonic flow with high varying density. One example of such solution-based adaptive mesh (2-D) is shown in Fig. 3.

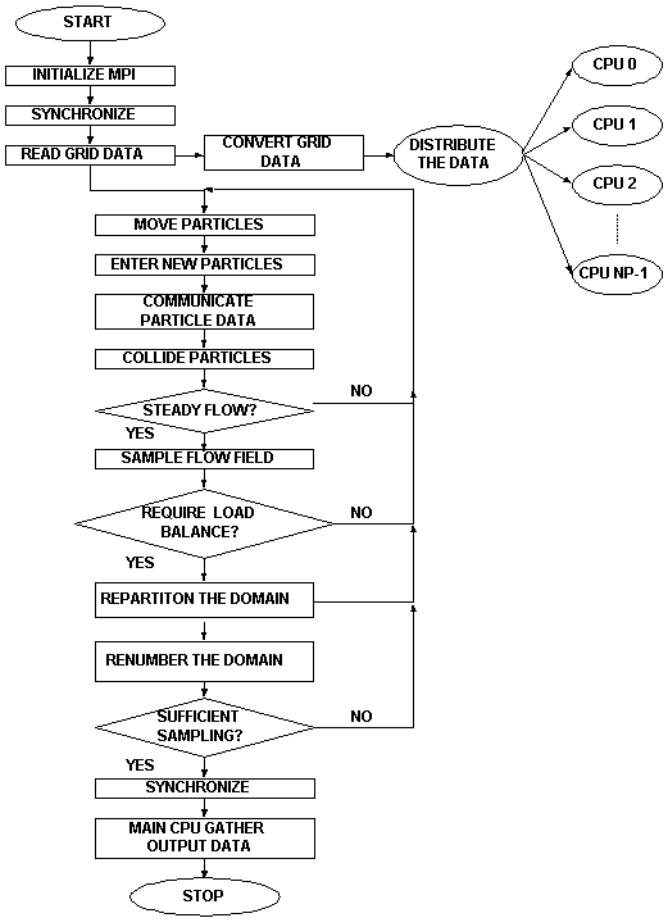

Fig. 1. Flow chart of proposed parallel DSMC method using dynamic domain decomposition.

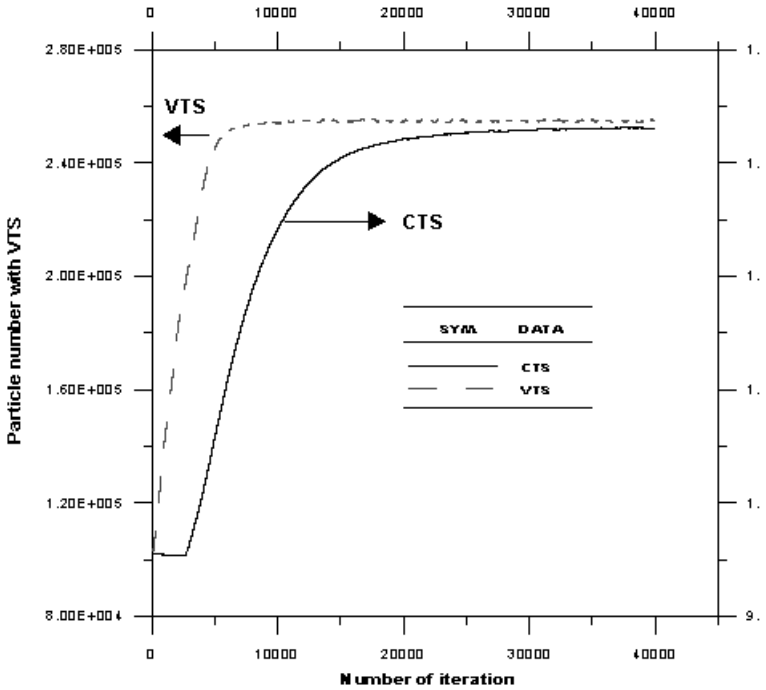

Fig. 2. Particle distribution of 2-D cylinder flow using constant time-step (CTS) and variable time-step (VTS), respectively.

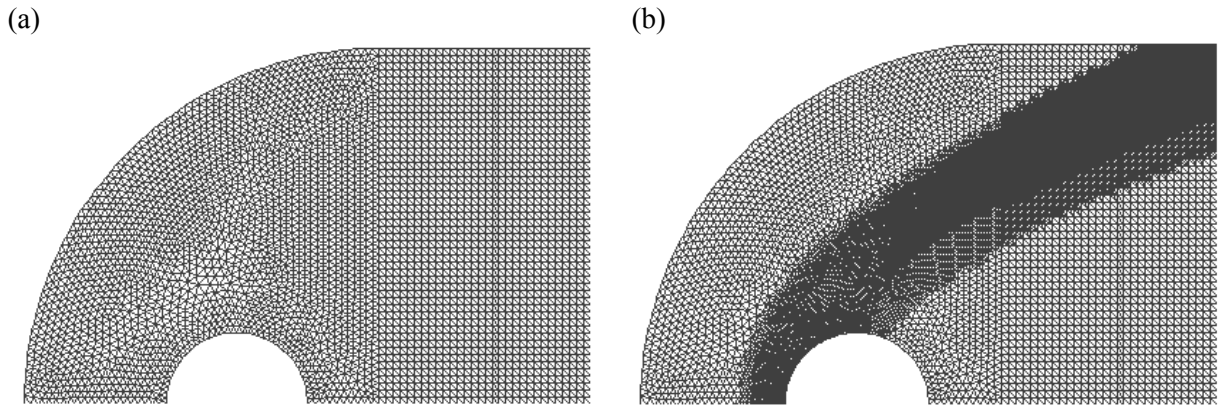

Fig. 3. Initial (a) and final (b) adaptive mesh
for 2-dimenstional hypersonic cylinder flow problem.

Parallel Implementation Using Dynamic Domain Decomposition

A square high-speed driven cavity flow with bottom plate moving to the right ($V=8*C_{mp}$) is considered as the test problem. Related flow conditions include argon gas, 300K of wall temperature and diffusive wall. No Time Counter (NTC) method [8] and Variable Hard Sphere (VHS) molecular model [8] is used for collision kinetics and reproduction of real fluid properties, respectively. Different problem sizes, including Small (22,500), medium (900,000) and large (3,600,000) problem size, are considered for simulation, where the number in the parenthesis represents the simulated particle numbers. Nearly uniform triangular mesh is used throughout the study. Simulations are run for 50,000 time steps with time step about 1/2 of the mean collision time step. DSMC code is implemented on IBM-SP2 with processor numbers in the range of 1~64.

Different strategies of activating SAR at intervals of $2\Delta t$, $10\Delta t$ and $20\Delta t$ are implemented. Typical evolution of domain decomposition using graph partitioning technique is shown in Fig. 4 for the large problem size using 16 processors applying SAR at intervals of $2\Delta t$. It is clear that regime of each domain is changing as simulation evolves by considering the load balancing among processors.

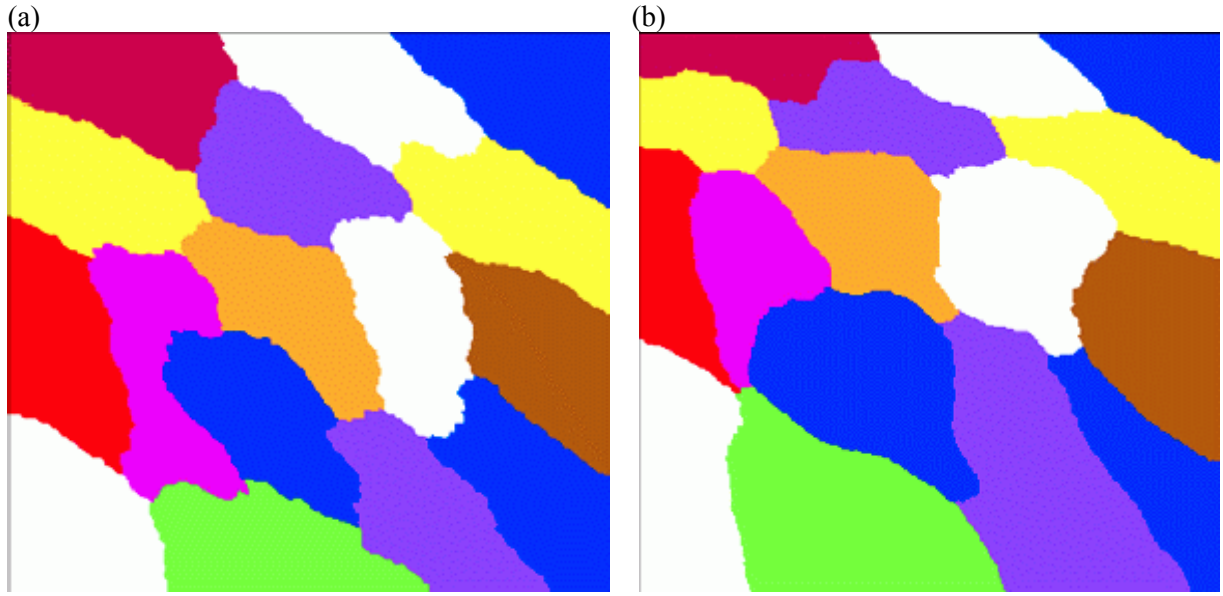

Fig. 4. Initial (a) and final (b) domain decomposition for medium driven-cavity flow problem (16 CPUs).

Typical results for parallel speedup are presented in Fig. 5 for a medium problem size. Results generally show that parallel performance with dynamic domain decomposition is 50~60% better than that with static domain decomposition for the range of processor numbers tested. Super-linear speedup is observed for processor numbers up to 48. Similar trends are found for small and large problems. The optimal strategy of applying SAR, however, strongly depends on the problem size. Details will be presented in the meeting upon reception. In addition, scalability analysis based on the simulation results is currently in progress and will be reported in the meeting. Finally, the application of this powerful computational technique will be demonstrated in the meeting by computing the interaction between two supersonic jets in the near-continuum regime.

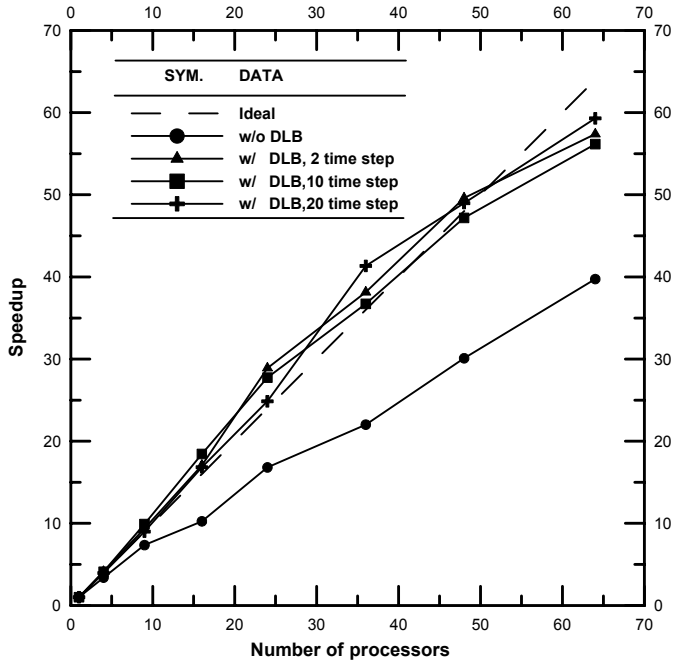

Fig. 5. Parallel speedup for medium problem with and without dynamic domain decomposition on IBM-SP2.

References

1. Walshaw, C., Cross, M., and Everette, M., J. Par. Dist. Comput. 47, 102-108, 1997.
2. Robinson, C.D., "Particle Simulation on Parallel Computers with Dynamic Load Balancing," PhD Thesis, Imperial College of Science, Technology and Medicine, London, UK, 1996.
3. LeBeau, G.J., Comput. Methods Appl. Mech. Engrg. 174, 319-337, 1999.
4. Nicol, D.M., and Saltz, J.H., IEEE Transactions on Computers 39, 1073-1087, 1988.
5. Wu, J.-S., and Tseng, K.-C., Computer & Fluids 30, 711-725, 2001.
6. Wu, J.-S., Tseng, K.-C. and Kuo, C.-H., International J. Numerical Methods in Fluids 38 (4), 351-375, 2002.
7. Wu, J.-S., Tseng, K.-C. and Wu, F.-Y., International J. Numerical Methods in Fluids (submitted).
8. Bird, G.A., "Molecular Gas Dynamics and the Direct Simulation of Gas Flows," Clarendon Press, Oxford, 1994.