The 18th International Conference on Finite Elements in Flow Problems (FEF2015)
16-18 March 2015, Taipei, Taiwan

Solving Euler Equation on Multiple Graphics Processing Units with Immersed Boundary Method

Fang-An Kuo[1,2] and Jong-Shinn Wu[1*]

[1] Department of Mechanical Engineering,
National Chiao Tung University, Hsinchu, Taiwan
[2] National Center for High-Performance Computing,
National Applied Research Laboratories, Hsinchu, Taiwan
* E-mail: chongsin@faculty.nctu.edu.tw

Key Words: *GPGPU, Finite volume method, Parallel computing*

Presented is an explicit cell-centered finite-volume solver for the Euler equation with a ghost-cell immersed boundary method on the multiple graphics processing units with a hybrid MPI-CUDA paradigm. The solver has been thoroughly validated with previously published simulations. The results show that the speedup can exceed 100 times on a single GPU (Nvidia Telsa M2070) as compared to a single thread of Intel Xeon X5472. In addition, more than 66% of parallel efficiency can be obtained when 96 GPUs are used for a problem size of 48 million cells. More details of the implementation and test results will be presented in the meeting.

The current finite volume method which is named Split-HLL scheme (SHLL) is a GPU-suitable designed scheme for solving Euler equations by using the accelerated computing units, like graphical processing units (GPU), and it based on Cartesian grid, which is good to make accessing data faster from memory. The scheme has been published by the journal “Computer and fluids” on 2011 [1]. SHLL scheme is a highly local scheme and the approximated scheme to the split-flux form of HLL scheme which is a traditional TVD scheme to solving the hyperbolic equation. The SLL scheme forms:

$$F_{net} = F^{+} + F^{-} = \frac{F_L - S_L U_L}{\left(1 - \frac{S_L}{S_R^{\text{A}}}\right)} + \frac{-F_R + S_R U_R}{\left(\frac{S_R}{S_L^{\text{A}}} - 1\right)} \tag{1}$$

The flux, named F_{net}, across the surface between the two neighbouring cells are been split to 2 fluxes which are F^+ and F^-, which is a different approximated expression as compared to the split-flux from of HLL scheme published by Toro [2]. We replace the wave speed terms by $S_L^A = u_R - a$ and $S_R^A = u_L + a$ in the split-flux HLL scheme due to accelerate the numerical computation performed on GPU.

Optimization strategy on parallel computing is an important issue of the scientific computing. Apart from the numerical algorithm, there are two main issue to effort the parallel computing performance. The first issue is the loading balance method between those processors (or coprocessors). In our case, we balance the number of fluid cells uniformly on each MPI which access the problem subdomain that separate the problem domain to several subdomain. The second issue is the communicate method. In our benchmarking environment, the faculty assembles Infiniband device that is enable to support GPU-Direct RDMA technology (GDR). The result of the optimization strategy can improve the efficiency up to 89 % in our benchmarking case.

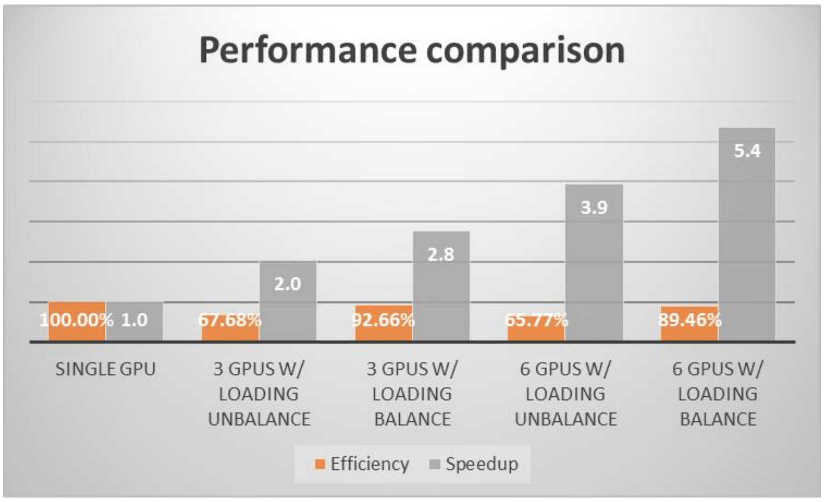

FIGURE 1 Comparison of those cases under the conditions including the different GPU numbers and loading balance method

REFERENCES

[1] F.-A. Kuo, M.R. Smith, C.-W. Hsieh, C.-Y. Chou and J.-S. Wu, “GPU acceleration for general conservation equations and its application to several engineering problems”, Computers and Fluids, 2011, volume 45: pp. 147-154.

[2] E.-F. Toro, M. Spruce and W. Speares, “Restoration of the contact surface in the HLL-Riemann solver”, Shock Waves, 1994, volume 4: pp. 25–34

[3] Ferziger JH *Numerical Methods for Engineering Application*. 2nd Edition, Wiley, 1998.