Parallel Computing Characteristics of Two-Phase Thermal-Hydraulics code, CUPID

Jae Ryong Lee[a*], Han Young Yoon[a]
[a]*Korea Atomic Energy Research Institute,1045 Daeduk-daero, Daejeon, Korea, 305-353*
[*]*Corresponding author: jrlee@kaeri.re.kr*

1. Introduction

The CUPID code [1], which aim is for multi-dimensional, multi-physics and multi-scale thermal hydraulics analysis, has been parallelized in order to fulfill the needs for complicated and long transient phenomena. Parallelized CUPID code has proved to be able to reproduce multi-dimensional thermal hydraulic analysis by validating with various conceptual problems and experimental data.

In this paper, the characteristics of the parallelized CUPID code were investigated. Both single- and two-phase simulation are taken into account. Since the scalability of a parallel simulation is known to be better for fine mesh system, two types of mesh system are considered. In addition, the dependency of the preconditioner for matrix solver was also compared. The scalability for the single-phase flow is better than that for two-phase flow due to the less numbers of iterations for solving pressure matrix.

2. Numerical Methodology

2.1 Governing equation

The CUPID code [1] adopts the two-fluid model for two-phase flows. In the two-fluid model, the mass, energy, and momentum equations for liquid and vapor phases are established separately, and then, they are linked by the interfacial mass, momentum, and energy transfer models. For a mathematical closure, the constitutive relations for the interfacial momentum transfer, the interfacial heat transfer and the wall heat partitioning are necessary.

2.2 Parallelization of the CUPID code

The CUPID code is parallelized based on the domain decomposition method with MPI (Message Passing Interface) [2]. MPI can be easily installed in any hardware and shows good performance for the parallelization by using internal MPI libraries. For domain decomposition, the CUPID code provides both manual and automatic method with METIS library [3]. Fig 1 shows the domain partitioning with METIS library.

For the effective memory management, the CSR (Compressed Sparse Row) format is adopted, which is one of the methods to represent the sparse asymmetric matrix. CSR format saves only non-zero value and its position (row and column).

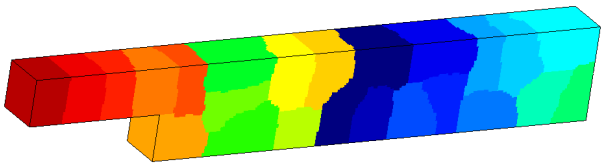

Fig 1. Domain partitioning with METIS library

2.3 Scalability analysis

The scalability of the parallel simulation is defined as the speedup ratio of computation time against the number of processors used as follows;

$$\Gamma = \frac{T_1}{T_n} \qquad (1)$$

where T: computation time, n: number of processors

Two types of grid system are taken into account in this study: 40,000 meshes for coarse system and 320,000 meshes for fine system. In the given geometry, both single- and two-phase calculations are conducted. In addition, since most of computation time is due to the pressure solver, the computation time and the scalability for some major subroutines is compared.

3. Results and Discussion

Fig 1 shows the scalability for the 40,000 mesh system. Both single- and two-phase results with different preconditioners are taken into account. In general, the performance of the parallel computing is estimated as ratio between computational time and communication time. If a communication time is longer than a computation time, the parallel computing is not effective. For the coarse mesh system, proportion of the computation is relatively greater than that of communication. Therefore, as increasing of the number of processors, the scalability is rapidly saturated. When four (4) processors are used as shown in Fig 2, a best scalability is obtained as almost linear performance. For forty (40) processors are used, the maximum speedup are observed. As increasing the number of processors over forty (40), the scalability is decreased because of over-partitioning of the computational domain. For a large number of computational grids per partitioned subdomain, the speedup ratio is expected to be improved in terms of scalability even though the computation itself takes more time. In addition, the Incomplete Lower- and Upper decomposition (ILU) preconditioner makes the number of iteration for

solving the pressure matrix rapidly decreased so that it is able to shorten the overall computation time. However, there is not much difference between diagonal and ILU preconditioner in terms of scalability.

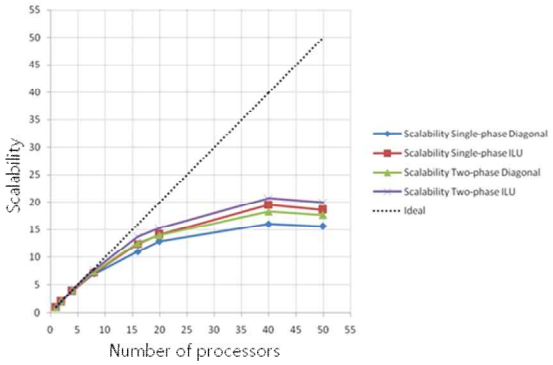

Fig 2. Scalability for 40,000 mesh system

Fig 3 shows the scalability for 320,000 mesh system. Comparing with Fig 2, the scalability is far improved. In addition, the effect of flow regime and preconditioner is observed. For the single-phase flow, the scalability is better than that for two-phase flow. The single-phase calculation with the diagonal preconditioner shows almost thirty-seven (37) times faster than a serial calculation for fifty (50) processors used. When four (4) ~ eight (8) processors are used, a speedup shows super-scalable which means that the speedup ratio exceeds the number of processors [4].

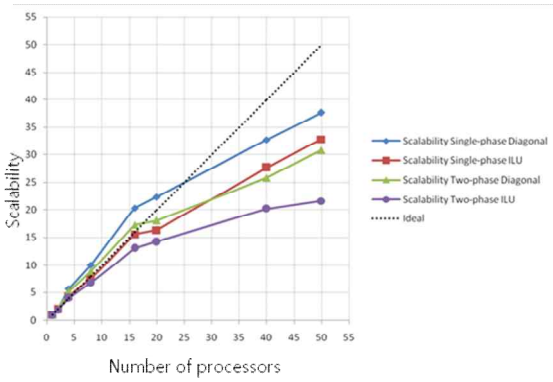

Fig 3. Scalability for 320,000 mesh system

Since the CUPID code takes most of time to solve the pressure matrix, it needs to evaluate the performance of parallel calculation of the pressure solver. The iterative solver in the CUPID code needs a couple of MPI calls. Since the number of iterations to solve the pressure matrix for two-phase flow simulation increases, correspondingly the number of communication calls is dramatically increased so that the scalability becomes saturated as increasing of the number of processors. Fig 4 shows the speedup of the pressure solver for different mesh systems, flow regimes and preconditioners. The scalability strongly depends on the number of mesh applied rather than the flow regime or which preconditioner is applied. Eighteen-time (18) speedup is obtained with fifty (50) processors for coarse mesh, whereas almost fourty-five-time (45) speedup for fine mesh. That is, even though the overall calculation time for fine mesh takes more than that for coarse mesh, the proportion of the communication time is decreased and the scalability is improved. For a general two-phase flow simulation, the preconditioner plays an important role to shorten the computation time. Even though the scalability for ILU preconditioner does not better than that for diagonal preconditioner, the ILU preconditioner makes the real computation time shorten in half.

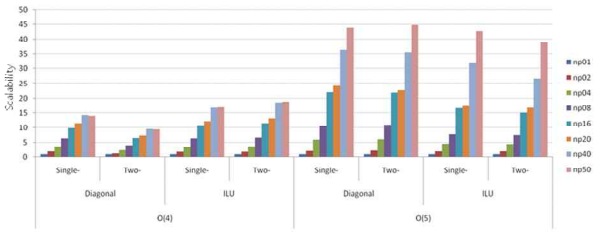

Fig 4. Scalability of pressure solver

4. Conclusions

The CUPID code was investigated the parallel performance in terms of scalability. The CUPID code was parallelized with domain decomposition method. The MPI library was adopted to communicate the information at the interface cells. As increasing the number of mesh, the scalability is improved. For a given mesh, single-phase flow simulation with diagonal preconditioner shows the best speedup. However, for the two-phase flow simulation, the ILU preconditioner is recommended since it reduces the overall simulation time.

Acknowledgement

This work was supported by Nuclear Research & Development Program of the NRF (National Research Foundation of Korea) grant funded by the MEST (Ministry of Education, Science and Technology) of the Korean government(Grant code: 2012M2A8A4025647).

REFERENCES

[1] H.Y. Yoon, et al., “Multi-Scale Thermal-Hydraulic Analysis of PWRs using the CUPID Code”, Nuclear Energy and Technology, Vol.44, pp.831-846, 2012.
[2] http://www.mpi-forum.org
[3] Karypis, G. et al., "A software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices V.4.0," U. of Minnesota, 1998
[4] Y.S. Nam, H.G. Choi and J.Y.Yoo, “AILU preconditioning for the finite element formulation of the incompressible Navier–Stokes equations”, Computer methods in applied mechanics and engineering, Vol.191, pp.4323-4339, 2002.