Performance Assessment of GPU-Based nTRACER/ESCOT Coupled Simulations

Kyung Min Kim[a], Jaejin Lee[b], Namjae Choi[a], Han Gyu Joo[a*]
[a]*Department of Nuclear Engineering, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul, 08826, Korea*
[b]*Global Research for Safety, Schwertnergasse 1, Cologne, 50667, Germany*
[*]*Corresponding author: joohan@snu.ac.kr*

1. Introduction

Significant advances in CPU processing power and parallel computing technologies led to the era of high-fidelity simulation. It thus became more viable to simulate various phenomena in science and engineering problems with higher resolutions and fewer approximations. In the reactor physics field, it had enabled whole-core multi-physics simulations by coupling high-fidelity neutronics codes and sub-channel thermal-hydraulics (T/H) codes. DeCART [1] coupled with MATRA [2] and nTRACER [3] coupled with ESCOT [4] are the examples of the successful code systems that are capable of performing pin-resolved multi-physics simulations.

However, the improving trend of the CPU computing technology is now facing challenges due to the power consumption issue. As the result, more researchers are trying to exploit heterogeneous computing technologies driven by graphic processing units (GPU) to achieve even higher performance. And so far, it has succeeded in many computational physics areas. Especially in the reactor physics field, nTRACER had established a plan to offload entire core follow processes onto GPU. Until now, implementation of neutronics solvers for GPUs is complete, and it demonstrated significant improvements in performance over the conventional multi-core CPU-based parallelism [5]. Furthermore, the whole process of the pin-wise sub-channel T/H code ESCOT had been successfully offloaded onto GPUs, which had revealed that GPUs are highly effective for the T/H calculations as well [6].

As a progressive step in the nTRACER development roadmap, this paper introduces the nTRACER/ESCOT coupled multi-physics simulation scheme on GPUs and assesses its performance. Newly introduced features of ESCOT – fuel heat conduction on GPUs and Anderson acceleration method – are explained briefly with proper references where the details can be found. In addition, performance assessments with a realistic core problem are presented.

2. nTRACER/ESCOT Code System

In this section, brief explanations on calculation flows and coupling scheme of the two codes are given. Full details can be found in the reference papers.

2.1. nTRACER Calculation Scheme

The steady-state calculation in nTRACER proceeds by the alternation of five primary solution modules: planar method of characteristics (MOC), subgroup fixed source calculation, two-level coarse mesh finite difference (CMFD), 1D axial solver, and T/H feedback, as illustrated in **Figure 1**. The T/H feedback calculation, on which this work is focused, is externally driven by ESCOT, which should be accompanied by an efficient coupling scheme.

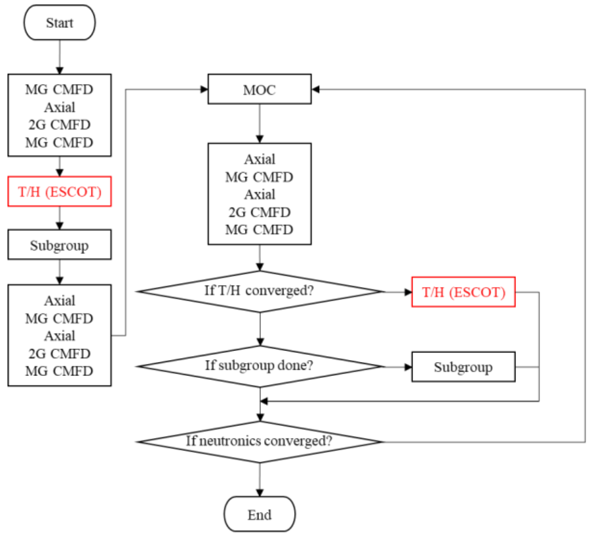

Figure 1. Calculation flowchart of nTRACER.

2.2. ESCOT Calculation Scheme

ESCOT is a pin-wise sub-channel T/H code based on four-equation drift-flux model. The calculation scheme of ESCOT is explained in **Figure 2**.

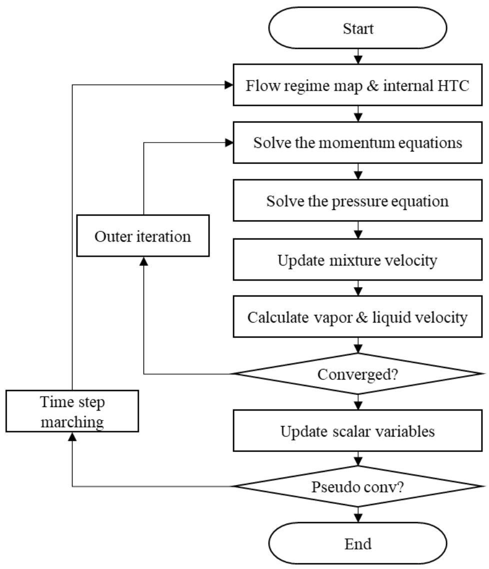

Figure 2. Calculation flowchart of ESCOT.

ESCOT sets several thermal and fluid properties as primary variables which are solved either in the form of linear systems or by SIMPLEC algorithm. On the other hand, there also exist secondary variables which are not solved but directly calculated from the equation of state.

2.3. Parallelization Topology for Coupling

Although it was recently demonstrated that assembly-wise radial decomposition is more effective in parallel efficiency for the massive parallelization based on CPUs [7], both GPU-accelerated nTRACER and ESCOT still employ plane-wise domain decomposition, resorting to the GPU's parallel computing power. In fact, ray-based fine-grain parallelism is much more beneficial on GPUs than the spatial decomposition. In addition, as the main purpose of the GPU acceleration module in nTRACER is to achieve high performance with limited amount of resources, plane-wise domain decomposition is a more suitable parallelization scheme. To keep the parallelism consistent, the GPU acceleration module of ESCOT also applies plane-wise decomposition, and as the result, nTRACER and ESCOT share the same MPI domain as shown in **Figure 3**.

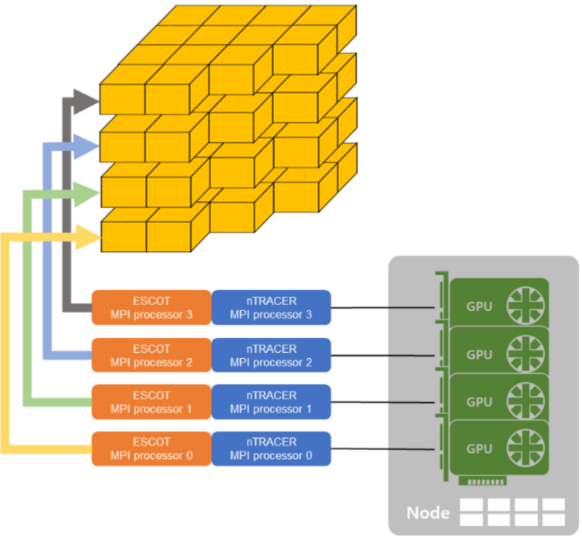

Figure 3. Computing topology for the coupled codes.

3. Extended Features of GPU Accelerated ESCOT

Several features were augmented in the ESCOT code to enhance accuracy and stability of coupled simulations. An inline fuel heat conduction solver was implemented for a realistic estimation of the fuel temperature profiles and the Anderson acceleration method was introduced to accelerate and stabilize the non-linear iterations. This section briefly introduces the additional features.

3.1. Fuel Heat Conduction Model

The fuel heat conduction model that ESCOT employs is based on the fuel conductivity correlations of the fuel performance code FRAPCON [8]. The correlations can take into account sub-pin level temperature, burnup, and gadolinium fraction distributions. It provides two types of correlations for UO_2 and MOX fuels, respectively:

$$k_{fuel}^{UO_2} = f(T, bu, w_{Gd}) \quad (1)$$

$$k_{fuel}^{MOX} = f(T, bu, w_{Gd}, \eta) \quad (2)$$

T : Temperature of fuel
bu : Burnup in GWD/MTU
w_{Gd} : Gadolinium weight fraction
η : Oxygen-to-metal ratio for MOX fuel

Initially, it was anticipated that the fuel conduction solution would not be expensive compared to the main T/H calculations even with CPUs, because it is executed only once when the primary calculation ends. However, due to the enhanced performance of the T/H procedures accelerated with GPUs, the cost of the fuel conduction became noticeable. Therefore, the conduction solver was ported onto GPUs as well to prevent the conduction solver from serving as an unnecessary bottleneck. The accuracy of the GPU module compared to the original CPU module were examined with an OPR1000 quarter core problem, whose results are illustrated in **Figure 4**, showing differences of only 10^{-3} °C at most.

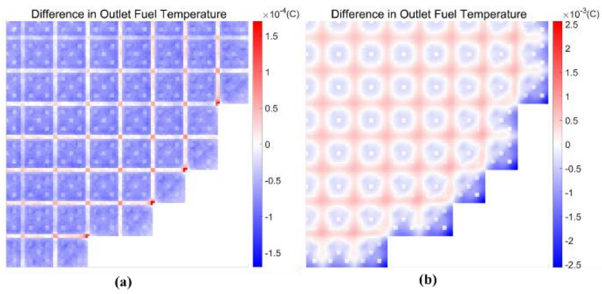

Figure 4. Difference in fuel center-line temperature between CPU and GPU modules for (a) constant thermal properties and (b) temperature dependent thermal properties.

3.2. Anderson Acceleration Method

Anderson acceleration method has been introduced as an alternative to a typical fixed-point iteration between neutronics and T/H calculations. Although the method was originally proposed to solve an integral equation [9], it had been revealed that it can accelerate as well as stabilize the convergence of the non-linear iterations in the multi-physics calculations [10]. **Table 1** shows the algorithm briefly, and details including derivations can be found in [11] and [12].

Table 1. Algorithm of Anderson acceleration.

Given $\mathbf{x}^0$ and $m \geq 1$,
Set $\mathbf{x}^1 = G(\mathbf{x}^0)$.
for k=1,2,....
Set $m_k = \min\{m,k\}$.

Compute $G(\mathbf{x}_k)$ and let $f_k = G(\mathbf{x}_k)-\mathbf{x}_k$.
Set $F_k = [f_{k-m_k},\ldots,f_k]$
Determine $\alpha^{(k)} = [\alpha_0^{(k)},\ldots,\alpha_{m_k}^{(k)}]^T$ that solves

$$\begin{cases} \min\limits_{\alpha=[\alpha_0,\ldots,\alpha_{m_k}]^T} \left\| F_k \alpha \right\|_2 \\ \text{s.t.} \sum\limits_{i=0}^{m_k} \alpha_i = 1 \end{cases}.$$

Set $\mathbf{x}_{k+1} = \sum\limits_{i=0}^{m} \alpha_i^{(k)} G\left(\mathbf{x}_{k-m_k+i}\right)$.

end

4. Results and Discussion

The performances and accuracies are examined for an APR1400 3D half-quarter core [13]. The problem specifications are in **Table 2**. The calculations were carried out in our own clusters, *Soochiro 3* and *Soochiro 4*. The former is for the original CPU codes, and the latter is for the new GPU codes, which is equipped with 4 commercial GPUs per computing node. The specifications of the clusters are listed in **Table 3** and **Table 4**.

Table 2. Specifications of APR1400 half-quarter core.

# of planes	16
# of pins / plane	20,032
# of flat source regions / plane	1,683,860
# of ray segments / plane	259,628,613
# of polar angles	4
Outlet pressure (MPa)	15.5141
Inlet temperature (°C)	291.3
Core power (MWth)	497.8759

Table 3. Specifications of the *Soochiro 3* cluster.

# of Nodes	27
CPU	2 × Intel Xeon E5-2640 v3 16 Cores, 2.8 GHz (Boost)
Memory	8 × 16GB DDR4 RAM
Interconnect	Infiniband (56 Gbps)
Compiler	Intel Fortran 14.0.3

Table 4. Specifications of the *Soochiro 4* cluster.

# of Nodes	9
CPU	2 × Intel Xeon E5-2630 v4 20 Cores, 2.4 GHz (Boost)
GPU	4 × NVIDIA GeForce GTX 1080 (3 nodes) 4 × NVIDIA GeForce RTX 2080 Ti (6 nodes)
Memory	8 × 16GB DDR4 RAM
Interconnect	Infiniband (56 Gbps)
Compiler	PGI Fortran 19.10

4.1 Accuracy Examination

The eigenvalues solved from the two different code systems matched exactly (0.98516). And absolute differences in axially integrated radial pin power and radially integrated axial power are shown in **Figure 5** and **Figure 6**. Since nTRACER adopts mixed precision techniques in most solution processes, the maximum difference in the radial pin power only occurs at the last significant digit (10^{-4}), which is negligible enough.

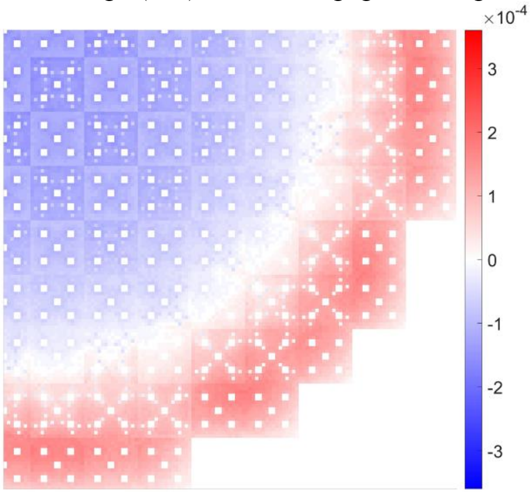

Figure 5. Absolute difference of integrated pin power.

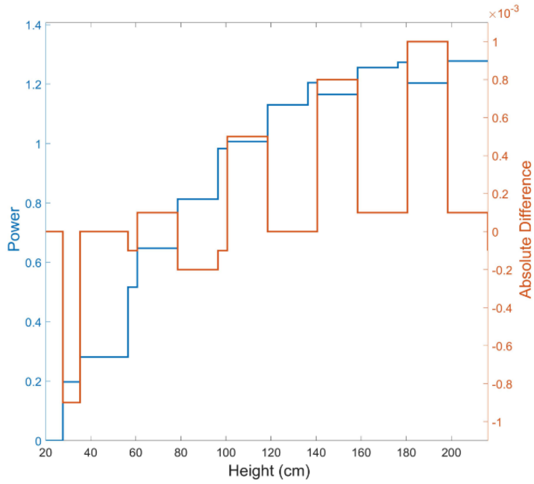

Figure 6. Absolute difference of integrated axial power.

4.2 Performance Examination

The GPU-accelerated coupled codes took about 4 minutes and 30 seconds to solve the problem. 5 MOC outer iterations were required to converge, and the subgroup resonance treatment was carried out 2 times, when the first T/H feedback was done and when the T/H feedback converged. Detail computing times of each procedure are listed in **Figure 7** and **Figure 8**.

As shown in the both figures, the T/H calculation only occupies 16 % of the whole computing time. However, in the T/H computing time, it shows most of it is taken by solving the pressure equation in ESCOT, which is due to its poor parallel performance.

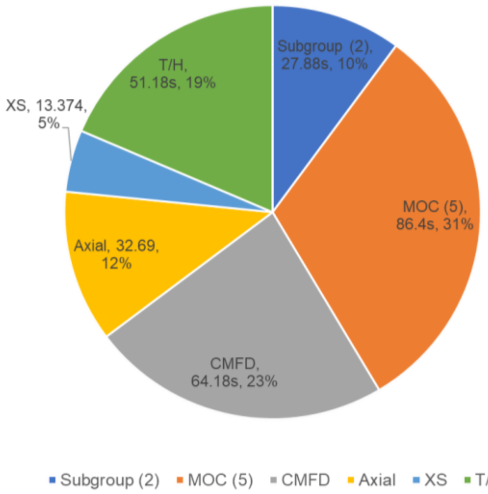

Figure 7. Computing time of each process in the coupled calculation.

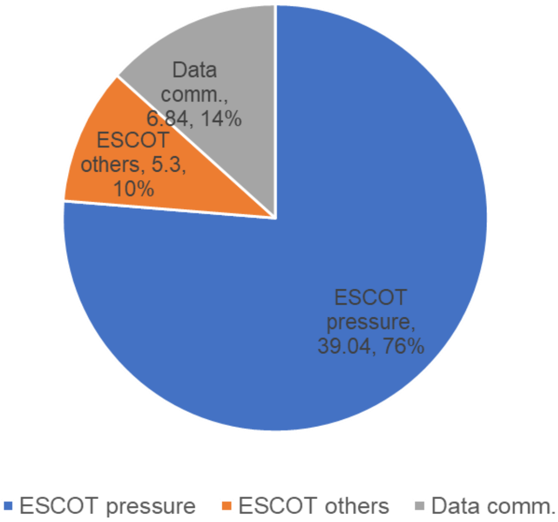

Figure 8. Computing time in the T/H part.

5. Conclusion

As a follow-up study of introducing GPU acceleration into nTRACER and ESCOT respectively, the two codes were coupled to realize GPU-based whole-core multi-physics calculations. The accuracy and the performance of the coupled system had been verified with a realistic core problem. It was proved that the mixed precision technique introduced in the neutronics solvers to exploit the single precision computing power of GPUs can still retain sufficient accuracy under feedback calculations. Furthermore, it was demonstrated that whole-core multi-physics simulations can be carried out in a few minutes.

Nevertheless, there are several remaining tasks. First of all, various coupling schemes should be investigated and the optimal one should be determined. For example, the Jacobi fixed-point iteration, in which the neutronics and the T/H calculations proceed in tandem, is generally known to be less effective than the Gauss-Seidel scheme. However, as the neutronics calculations have become way much faster than before, different results may be obtained. Additionally, more optimizations of remaining bottlenecks, such as the solution to the pressure matrix in ESCOT, should be made for further performance enhancements.

ACKNOWLEDGEMENTS

This research is supported by National Research Foundation of Korea (NRF) Grant No. 2016M3C4A7952631 (Realization of Massive Parallel High Fidelity Virtual Reactor).

REFERENCES

[1] J. Y. Cho, S. Yuk, “Massive Parallel Computation for an Efficient Whole Core Transport Calculation,” Transactions of the Korean Nuclear Society Spring Meeting, Jeju, Korea, May 17 – 18, 2018.

[2] H. Kwon et al., “Validation of a Subchannel Analysis Code MATRA Version 1.1,” KAERI/TR-5581/2014, Korea Atomic Energy Research Institute, 2014.

[3] Y. S. Jung, C. B. Shim, C. H. Lim and H. G. Joo, "Practical Numerical Reactor Employing Direct Whole Core Neutron Transport and Subchannel thermal/hydraulic Solvers," Annals of Nuclear Energy, vol. 62, pp. 357-374, 2013.

[4] J. Lee, A. Facchini, H. G. Joo, “Development of a Drift-Flux Model Based Core Thermal-Hydraulics Code for Efficient High-Fidelity Multiphysics Calculation,” Nuclear Engineering and Technology, vol. 51, issue 6, pp. 1487-1503, 2019.

[5] N. Choi, J. Kang, H. G. Joo, “Preliminary Performance Assessment of GPU Acceleration Module in nTRACER,” Transactions of the Korean Nuclear Society Autumn Meeting, Yeosu, Korea, October 25 – 26, 2018.

[6] K. M. Kim, N. Choi, J. Lee and H. G. Joo, “Preliminary Performance Assessment of the GPU Acceleration Module in a Pinwise Core Thermal Hydraulics Code ESCOT,” Proceedings of Reactor Physics Asia, Osaka, Japan, December 2 – 3, 2019.

[7] J. Y. Cho, et al., “Performance of a Whole Core Transport Code, nTER,” in NURER2018, Jeju, Korea, 2018.

[8] K. J. Geelhood, W. G. Luscher, P. A., Raynaud and I. E. Porter, “FRAPCON-4.0: A Computier Code of Steady-State, Thermal-Mechanical Behavior of Oxide Fuel Rods for High Burnup,” PNNL-19418, Vol. 1 Rev. 2, Richland, Whashington, USA , 2015.

[9] H. R. Fang and Y. Saad, “Two classes of multisecant methods for nonlinear acceleration,” Numer. Linear Algebra Appl. vol. 16, pp. 197-221, 2009.

[10] J. Lee, J. Y. Cho and H. G. Joo, “Initial Assessment of Anderson Acceleration on Pinwise Coupled Neutronics/Thermal-Hydraulics Code nTER/ESCOT,” Proceedings of the Reactor Physics Asia, Osaka, Japan, December 2-3, 2019.

[11] An H., et al., “Anderson Acceleration and Application to the Three-Temperature Energy Equations,” J. Comput. Phys., 347, 1-19, 2017.

[12] Walker H. F., “Anderson Acceleration: Algorithms and Implementations,” Research Report, MS-6-15-50, Worcester Polytechnic Institute Mathematical Sciences Department, 2011.

[13] H. Hong, H. G. Joo, “Analysis of the APR1400 PWR Initial Core with the nTRACER Direct Whole Core Calculation Code and the McCARD Monte Carlo Code,” Transactions of the Korean Nuclear Society Spring Meeting, Jeju, Korea, May 18-19, 2017