Transactions of the Korean Nuclear Society Autumn Meeting
Changwon, Korea, October 20-21, 2022

MPI Parallelization of START and its Embedment in Pin-by-pin Calculation

Sicheng Wang [a,b], Yonghee Kim [b*], Yunzhao Li [a], Liangzhi Cao [a]
[a] *School of Energy and Power Engineering, Xi'an Jiaotong Univ., 28 West Xianning Rd., Xi'an, Shaanxi 710049, P. R. China*
[b] *Department of Nuclear and Quantum Engineering, Korea Advanced Institute of Science and Technology., 291 Daehak-ro, Yuseong-gu, Daejeon 34141, R. Korea*
[*] *Corresponding author: yongheekim@kaist.ac.kr*

1. Introduction

The pin-by-pin two-step method is an improved two-step method. The homogenization calculation is refined to the pin-wise scale compared to the conventional assembly-homogenization two-step method. A pin-wise few-group constants library could then be constructed. The resolution of the 3D whole-core neutronics calculation is refined to the pin-wise scale, and the pin-wise power distribution can be directly obtained without the pin-power reconstruction.

Accordingly, the pin-by-pin whole-core calculation requires pin-wise thermal-hydraulics (TH) feedback to obtain the pin-cell fuel temperatures and moderator temperatures for the calculation of pin-wise few-group constants. Consequently, the coupling of pin-by-pin neutronics code and the subchannel TH codes attracts increasing attention.

The purpose of this paper is to modify the subchannel code START developed by Korea Advanced Institute of Science and Technology (KAIST) [1] into a new MPI-based version, which can serve as the fundamental model for the embed coupling with the pin-by-pin neutronics code NECP-Bamboo2.0 developed by Xi'an Jiaotong University [2].

2. Methods

2.1 Brief introduction of START

START is a subchannel solver, known as Steady and Transient Analyzer for Reactor Thermal hydraulics, developed at KAIST as a fast and accurate TH-solver for coupled and multi-physics calculations. It is based on a two-phase homogeneous model with Equal Velocity and Equal Temperature approximations, considering the conservation equations of mass, energy, and momentum (axial and lateral). In addition, OpenMP-based parallelization has been applied to various portions of the original START code to achieve the goal of multi-physics analyses of full-scale PWR in a reasonable amount of time [3].

2.2 Brief introduction of NECP-Bamboo2.0

The PWR-core pin-by-pin fuel management calculation code system named NECP-Bamboo2.0 consists of the 2D lattice calculation code Bamboo-Lattice2.0 and the 3D steady-state whole-core pin-by-pin calculation code Bamboo-Core2.0. A built-in pin-wise single-channel model is employed in the original Bamboo-Core2.0 as the TH module to simulate the coolant flow in the pin-cell channel while ignoring the influence of cross-flow and turbulence.

Notably, a quarter assembly is referred to as a "block" in Bamboo-Core2.0, which plays the role of the minimum and most fundamental parallel unit. The block-wise parallel unit can balance the load balancing of each processor with the cost of communication for various cases. The expected domain decomposition results of an example mini core are shown in Figure 1, which contains 5 assemblies, and the total number of processors is 4. The block ID of the 1st processor (red frame) is also shown in Fig. 1.

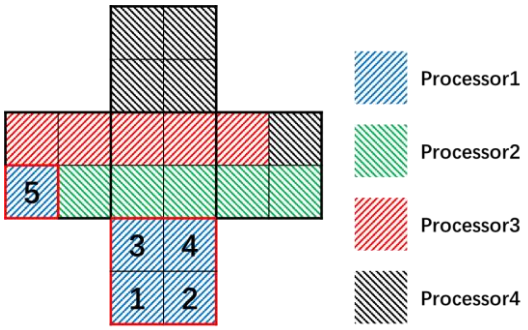

Fig. 1. Expected domain decomposition (5 assemblies, 4 processors)

2.3 Embed coupling approach

The embed coupling approach is chosen to implement the coupling of the START code and the NECP-Bamboo2.0 code to avoid the introduction of an additional supervisor code, which would have a negative impact on the calculation efficiency. Such an embed coupling approach shown in Fig. 2 has the following characteristics [4],

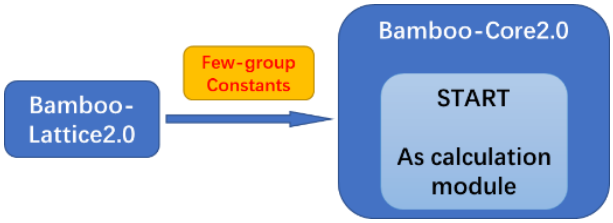

Fig. 2. Schematic implementation of the coupling approach

- Data transfer through memory between the neutronics calculation and the TH calculation.
- The calculation flow of the Bamboo-Core2.0 is

maintained, where the START is embedded as a calculation module.

- The initialization process of the coupling code is uniformly implemented by Bamboo-Core2.0, avoiding repetitive input definition between codes.

2.4 MPI-based parallelization of START

Even though the embed coupling can be realized when the pin-by-pin neutronics code and the subchannel code have different parallel strategies, it would be challenging in both computing time and computing resource requirements. If a multi-node or multi-processor platform is used for the multi-physics calculation, the disparity between MPI-based Bamboo-Core2.0's multi-processor computing and OpenMP-based START's single-processor multi-thread computing will inevitably lead to data exchange issues between different processors, necessitating a complex data interface and reducing the overall efficiency of parallel computing.

Consequently, a unified domain decomposition is employed in the coupling of NECP-Bamboo and START, meaning that a modified MPI-based START is developed in which the block is also introduced as the most fundamental parallel unit. Based on the unified domain decomposition, the data exchange between the neutronics module and TH module could be realized in a direct block(neutronics)-to-block(TH) way.

Noting that the effects of cross-flow and turbulence between neighboring channels are considered in the conservation equations of the subchannel theory, generally, the implementation of the MPI-based parallelization of subchannel code requires the concept of the "ghost" region. The ghost region's role is to provide the boundary information of the neighboring subdomains, which is necessary for the solution of the local subdomain. Fig. 3 depicts the ghost region of the blocks belonging to the 1st processor of the mini core shown in Fig. 1.

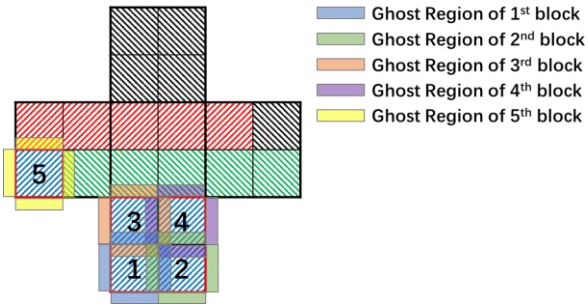

Fig. 3. Block-wise ghost region of blocks belonging to the 1st processor

Besides, due to the existence of domain decomposition, the START solution process also needs to be parallelized. The calculation flow of the MPI-based START is shown in Fig. 4. The diagram demonstrates that there are two different types of MPI communication requirements. The first type of communication, as shown by the blue line in Fig. 4, is the communication between different subdomains for the convergence judgment after the conservation equations are solved. This type of communication can be simply achieved by using the MPI's built-in functions, for example, the MPI-Allreduce function.

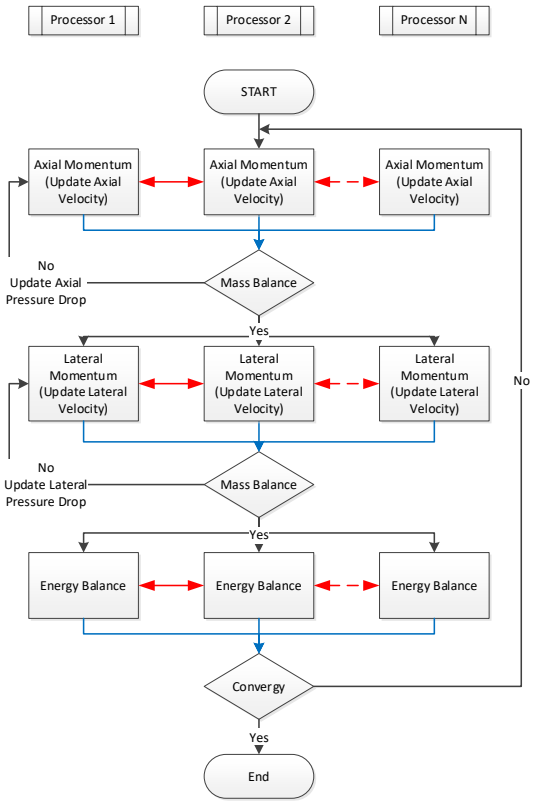

Fig. 4. Flowchart of MPI-based START

Another type is the communication for the ghost regions between and inside the parallel processors during the solution process, as shown by the red line in Fig. 4. A three-step MPI communication is implemented in START as shown in Fig .5 for the information exchange of this type. In this scheme, two communications between different processors are conducted first. The inside region information from the processor with the larger number is transmitted to the ghost region with the smaller number, followed by the transformation of inside region information belonging to the processor with the smaller number. After that, the information for the ghost regions belonging to the same processor is exchanged. According to this scheme, the ghost regions that extend beyond the radial active area would never send or receive any message, so their existence has no influence on the results.

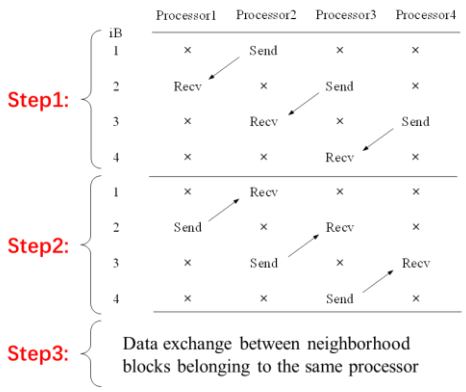

Fig. 5. Data exchange model in the MPI-based START

3. Results

3.1 3D AFA-3G single-assembly problem

The modified version of START, together with the coupling with Bamboo-Core2.0, is firstly verified based on a HFP 3D AFA-3G single-assembly problem. The boron concentration is fixed at 600ppm. There is no spacer grid and axial reflector for the verification. The turbulent mixing factor is set to 0, so only the cross-flow is considered.

This issue was solved using three distinct multi-physics models. The first is the original Bamboo-Core2.0 with a built-in pin-wise single channel, which can be viewed as a simplified subchannel model devoid of cross-flow and turbulence simulation. The second model is the direct coupling of the OpenMP-based original START and the MPI-based Bamboo-Core2.0. Due to the distinction between the parallel strategies, the calculation was implemented using a single processor. The final model is the newly developed coupling of MPI-based Bamboo-Core2.0 and MPI-based START. One to four processors were utilized for the calculation.

The calculation results are shown in Table I, where the pin-wise single-channel result is the one obtained from the NECP-Bamboo2.0 stand-alone calculation. Considering that the turbulence is not taken into account in this case, the difference between the NECP-Bamboo2.0 stand-alone calculation and the NECP-Bamboo2.0/START coupling calculation is no as observable, which is also physical. What's more, it can be illustrated that the MPI-based TH module is well parallelized. First of all, the calculation results of Bamboo-mpiSTART and Bamboo-ompSTART are identical. Besides, The TH calculation time is dependent on the number of processors. It is notable that the calculation time of 2 processors is similar to the one of 3 processors. Due to the existence of four blocks in this single-assembly problem, in the case of three processors, the number of blocks belonging to each processor is 2/1/1, with a maximum of 2. Meanwhile, in the case of two processors, the maximum number of blocks belonging to each processor is also 2. Consequently, the TH calculation times of the case with three processors are comparable to those of the case with two processors. The difference in total time between these two instances is due to the other modules of the whole-core calculation, such as initialization and finalization.

Table I: NECP-Bamboo2.0 calculations

	Eigenvalue	Max. T_F	Time (Total/TH)
Pin-wise single channel (stand-alone)	1.09923	832.6K	211s/0.1s
ompSTART 1 processor	1.09924	832.4K	230s/17s
mpiSTART 1 processor	1.09924	832.4K	231s/18s
mpiSTART 2 processors			113s/9s
mpiSTART 3 processors			101s/9s
mpiSTART 4 processors			53s/5s

3.2 3D VERA#6 benchmark problem

VERA Core Physics Benchmark Problem 6 is also a 3D PWR fuel assembly at HFP conditions. This problem considers two different types of spacer grids and the existence of the axial reflector. The effects of TH feedback are verified with the calculations of eigenvalue, and coolant temperatures. Bamboo-Lattice2.0 is employed to provide pin-cell homogenized few-group constants for fuel assemblies and reflectors. The turbulent mixing factor is set to 0.02, which is the recommended value of START.

Table II: Problem Description

	Eigenvalue	Average coolant outlet temperatures
MPACT/CTF	1.16361	600.04K
MC21/COBRA-IE	1.16424	600.12K
NECP-Bamboo2.0 (Pin-wise single channel)	1.16379	600.34K
NECP-Bamboo2.0 (mpiSTART)	1.16385	600.31K

As in Table II, the calculation results of the eigenvalues and the average coolant outlet temperatures of the stand-alone pin-wise single-channel NECP-Bamboo2.0 and the coupled NECP-Bamboo2.0-mpiSTART are at the same level as the results of high-fidelity codes [5]. Fig. 6 shows the coolant outlet temperature distribution of different codes. Compared to the reference code (MPACT/CTF), the deviation of coolant outlet temperature distribution is shown in Fig. 7.

600.85	600.95	600.85	600.75	600.65	600.45	600.15	599.85	599.75
600.95	600.95	600.85	600.75	600.65	600.45	600.15	599.95	599.75
600.85	600.85	600.85	600.75	600.65	600.45	600.15	599.85	599.65
600.75	600.75	600.75	600.65	600.45	600.35	599.95	599.75	599.55
600.65	600.65	600.65	600.45	600.35	600.15	599.75	599.55	599.35
600.45	600.45	600.45	600.35	600.15	599.85	599.55	599.35	599.15
600.15	600.15	600.15	599.95	599.75	599.55	599.35	599.15	598.95
599.85	599.95	599.85	599.75	599.55	599.45	599.15	598.95	598.75
599.75	599.75	599.65	599.55	599.35	599.15	598.95	598.75	598.55

(a) MPACT/CTF

598.44	600.80	600.81	598.45	600.75	600.66	598.11	599.98	600.02
600.80	601.88	601.90	600.82	601.85	601.75	600.46	601.10	599.99
600.81	601.90	601.92	600.86	601.93	601.84	600.50	601.11	599.97
598.45	600.82	600.86	598.61	601.09	601.07	598.33	599.97	599.93
600.75	601.85	601.93	601.09	601.33	600.09	599.51	600.91	599.68
600.67	601.76	601.84	601.07	600.09	598.52	600.12	601.55	599.33
598.11	600.46	600.50	598.33	599.51	600.12	600.48	601.01	599.04
599.98	601.11	601.11	599.97	600.91	601.55	601.01	600.68	598.91
600.03	599.99	599.97	599.93	599.68	599.33	599.04	598.91	597.33

(b) NECP-Bamboo2.0 (Pin-wise single channel)

601.01	601.12	601.12	600.98	601.00	600.85	600.40	600.09	599.78
601.12	601.20	601.20	601.09	601.08	600.92	600.51	600.16	599.80
601.12	601.20	601.20	601.09	601.08	600.90	600.48	600.13	599.76
600.98	601.09	601.09	600.93	600.92	600.71	600.26	599.96	599.64
601.00	601.08	601.08	600.92	600.78	600.49	600.13	599.89	599.52
600.85	600.92	600.90	600.71	600.49	600.20	599.99	599.78	599.34
600.40	600.51	600.48	600.26	600.13	599.99	599.81	599.53	599.05
600.09	600.16	600.13	599.96	599.89	599.78	599.53	599.15	598.64
599.78	599.80	599.76	599.64	599.52	599.34	599.05	598.64	598.17

(c) NECP-Bamboo2.0 (mpiSTART)

Fig. 6. Coolant outlet temperature distribution

-2.41	-0.15	-0.04	-2.30	0.10	0.21	-2.04	0.13	0.27
-0.15	0.93	1.05	0.07	1.20	1.30	0.31	1.15	0.24
-0.04	1.05	1.07	0.11	1.28	1.38	0.35	1.26	0.32
-2.30	0.07	0.11	-2.04	0.64	0.72	-1.62	0.21	0.38
0.10	1.20	1.28	0.64	0.98	-0.06	-0.24	1.36	0.33
0.21	1.30	1.39	0.72	-0.06	-1.33	0.57	2.20	0.18
-2.04	0.31	0.35	-1.62	-0.24	0.57	1.13	1.86	0.09
0.13	1.15	1.26	0.22	1.36	2.10	1.86	1.73	0.16
0.27	0.24	0.32	0.38	0.33	0.18	0.09	0.16	-1.22

(a) NECP-Bamboo2.0 (Pin-wise single channel)

0.16	0.17	0.27	0.22	0.35	0.40	0.25	0.24	0.03
0.17	0.25	0.35	0.34	0.43	0.47	0.36	0.21	0.05
0.27	0.35	0.35	0.34	0.43	0.45	0.33	0.28	0.11
0.22	0.34	0.34	0.28	0.47	0.36	0.31	0.21	0.09
0.35	0.43	0.43	0.47	0.43	0.34	0.38	0.34	0.17
0.40	0.47	0.45	0.36	0.34	0.35	0.44	0.43	0.19
0.25	0.36	0.33	0.31	0.38	0.44	0.46	0.38	0.09
0.24	0.21	0.28	0.21	0.34	0.33	0.38	0.19	-0.11
0.03	0.05	0.11	0.09	0.17	0.19	0.09	-0.11	-0.38

(b) NECP-Bamboo2.0 (mpiSTART)

Fig. 7. Deviation of coolant outlet temperature distribution

It could be demonstrated in Fig. 6 and Fig. 7 that the employment of the subchannel model in NECP-Bamboo2.0/mpiSTART can accurately simulate the mass/momentum/energy exchange between different subchannels. Compared with the pin-wise single channel model, the improvement of the subchannel model can be demonstrated, where the mixing between different channels can be clearly found and the maximum deviation can be reduced from 2.41℃ to 0.47℃.

4. Conclusions

In this paper, the subchannel thermal-hydraulic code START is parallelized based on MPI and embedded into the pin-by-pin neutronics code NECP-Bamboo2.0 to carry out PWR whole-core pin-by-pin coupled neutronics and thermal-hydraulics calculations. The numerical results demonstrate that the coupled code possesses good parallelism and computational precision. Regarding the coupling code development. The parallel computation time based on the different number of processors is as expected.

In terms of the overall accuracy of the coupling code, the subchannel module is able to simulate the mixing effect between channels more accurately than the original pin-wise single-channel model in the NECP-Bamboo2.0 code. Both the eigenvalue and the moderator temperature distribution are in good agreement with the reference solution.

REFERENCES

[1] Khurrum Saleem Chaudri, Jaeha Kim and Yonghee Kim, Development and validation of a fast sub-channel code for LWR multi-physics analyses, Nuclear Engineering and Technology, Vol. 51, pp. 1218-1230, 2019.
[2] LI YZ, YANG W, WANG SC, et al., A three-dimensional PWR-core Pin-by-pin analysis code NECP-Bamboo2.0. Annals of Nuclear Energy, Vol. 144, No. 107507, 2020.
[3] Khurrum Saleem Chaudri, Sung Joong Kim and Yonghee Kim, CHF Prediction Evaluation for START Sub-channel Code, Transactions of the Korean Nuclear Society Spring Meeting, Jeju, Korea, May 23-24, 2019.
[4] Diego Ferraro, Manuel García, Ville Valtavirta, et al., Serpent/SUBCHANFLOW pin-by-pin coupled transient calculations for a PWR mini core, Annals of Nuclear Energy, Vol. 137, No. 107090, 2020.
[5] Brian N. Aviles, Daniel J. Kelly, David L. Aumiller, et al., Coupled MC21 and COBRA-IE solution to VERA core physics benchmark problem#6, PHYSOR 2016, Sun Valley, USA, May 1-5, 2016.