Transactions of the Korean Nuclear Society Spring Meeting
Jeju, Korea, May 17-19, 2023

Development of multicycle capability in Multiphysics coupling framework MPCORE

Awais Zahur, Muhammad Rizwan Ali, Deokjung Lee*
Ulsan National Institute of Science and Technology, UNIST-gil 50, Eonyang-eup, Ulju-gun, Ulsan, 689-798, Korea
**Corresponding author: deokjung@unist.ac.kr*

1. Introduction

Multicycle depletion data from an operational nuclear power plant can be used to validate Multiphysics coupled codes [1, 2]. In a nuclear reactor, Neutron kinetics (NK), Thermal hydraulics (TH), and Fuel performance (FP) codes are the most important physics phenomena, which are inherently dependent on each other. To couple different physics codes, a Multiphysics coupling framework, MPCORE, has been developed at Ulsan National Institute of Science and Technology. MPCORE offers features such as in-memory data transfer, adaptive timestep, Picard convergence algorithm, and mesh overlay (different meshes in different modules). A Multicycle feature has now been added to MPCORE to forward the burnup related information in multicycle case. A python-based input processing utility is developed for MPCORE to independently model each module's input file. This utility allows detail modeling in FP code that may not be required for NK and TH codes. The FP code requires hollow fuel pin modeling and IFBA coating in selective fuel pins. Such detailed modeling necessitates using different fuel conduction correlations in FP. It is worth noting that the NK module in this research does not require detailed pin information when coupling with the TH and FP module.

2. Methodology

In MPCORE, the coupling of the NK, TH, and FP modules is performed externally. Certain modules require restart information from the previous cycle. The primary characteristics of these constituent modules are outlined below:

1. **NK:** In the MPCORE framework, the neutronics calculation follows a conventional two-step modeling approach [3]. The lattice code, STREAM, is used for generating cross-sections of fuel assembly and reflector. Several branch cases are created for the hot state with varying combinations of temperatures and boron concentration, covering burnup points from 0-80 MWd/kg. The nodal code, RAST-K, is utilized to solve the 3D diffusion equation for flux and power distribution in the whole or quarter core. RAST-K employs the Unified Nodal method with pin power reconstruction, and the Chebyshev Rational Approximation Method (CRAM) is employed for solving Bateman equations for depletion calculation. To model multicycle depletion problems, RAST-K has shuffling and rotation utility. Spacer grid homogenization is used in the nodal code solution, but for the present analysis, heterogeneity has been preserved in the nodal solution. Spacer grid part uses different cross-sections with spacer grid smeared in coolant, while non-spacer grid part uses cross-sections without zircaloy/Inconel smearing.
2. **TH:** Coolant boiling in rod arrays-Two Fluid (CTF) subchannel thermal hydraulic code was utilized in present research. Developed and maintained jointly by Pennsylvania State University (PSU) and North Carolina State University (NCSU), CTF can efficiently perform thermal hydraulic analyses with cross flow and is capable of fast computation through MPI-based parallelization. The coupling interface of CTF facilitates in-memory data exchange between CTF and other external codes. However, while neutronics and fuel performance modules exchange data on a rod/pin basis, CTF is a channel-centered code, and thus, rod-centered information is used when coupling with CTF.

   The conduction option is turned off in CTF, and the oxide surface temperature is determined by the FP module. In parallel processing, CTF requires the same number of processors as the number of assemblies, and each processor contains data for its own assembly rods [4]. With the quarter symmetry option, CTF can model a quarter core, reducing the number of modeled fuel assemblies and processors needed to run in parallel.
3. **FP:** Fuel Rod Analysis Programming Interface (FRAPI) is employed as the fuel performance code in the coupling framework [5]. FRAPI serves as an interface for coupling depletion and transient calculations using FRAPCON and FRAPTRAN, respectively. Restart files are produced at every burnup step for both depletion and transient calculations. During the nuclear fuel cycle, fuel expansion, cladding creep, hoop stress, and fuel cracks may occur due to power transients. These models are created in FRAPCON and used to analyze fuel deformity and pellet clad mechanical interaction. FRAPCON can also model IFBA coating, which is utilized to manage power peaking at the beginning of a cycle. Moreover, FRAPCON models a hollow fuel pin in selective axial meshes, different gap widths due to IFBA coating, and varying enrichment in the blanket region.

MPCORE coupling framework is used for exchanging data between different modules. Fig. 1 illustrates the main parameters that are exchanged between the modules.

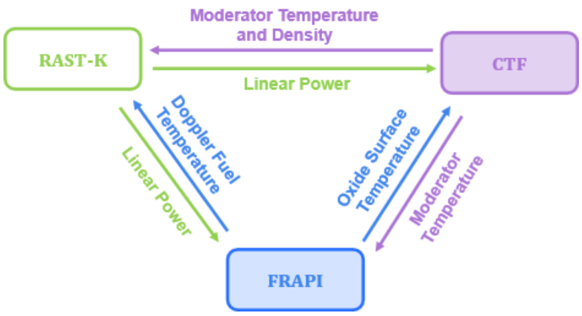

Fig. 1. Data exchange in MPCORE

3. Multicycle Strategy in MPCORE

At the start of the second or third cycle, burnt fuel rods have a reduced gap width and high gap conductance, which leads to efficient heat transfer in burnt fuel assemblies compared to fresh fuel assemblies. However, the fuel pellet thermal conductivity decreases with burnup. With the restart capability in FRAPI, FRAPCON restart files can be easily read for the next cycle, and the shuffling and rotation information is transferred to FRAPCON. FRAPI reads the appropriate fuel rod restart file based on the shuffling and rotation of assemblies. The transmission of burnup information has been confirmed for three cycles of the VERA core.

RAST-K and FRAPI modules can perform burnup depletion on their own, so no burnup information is transferred between these two modules. For the multicycle case, burnup-dependent gap conductance is used through the restart file reading capability in FRAPI.

4. Benchmark case

The cycle 1-3 core layouts of the Watts Bar reactor and a simplified power history can be found in [2], where the locations of Pyrex, IFBA, and WABA rods are taken from the benchmark document [1]. The gap conductance and fuel conductivity are both dependent on burnup and linear power. The current research also includes a coupled fuel performance code that calculates dynamic gap conductance based on burnup and power history.

5. Code-to-code comparison

Fig. 2 and Fig. 3 provide the burnup information at the End of Cycle (EOC) 1 and Beginning of Cycle (BOC) 2, respectively. The shuffling and rotation information has been verified based on the configuration of cycle 1 and cycle 2 [2]. Although the whole core can be modeled using the Nodal code to obtain accurate results, this study includes CTF and FRAPCON, resulting in longer computation times for the entire core.

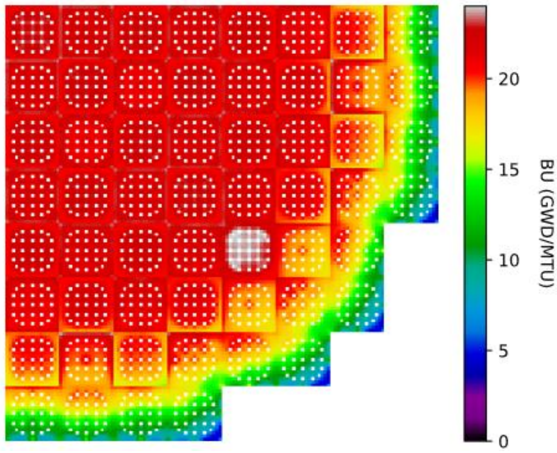

Fig. 2. FRAPCON Burnup at EOC 1

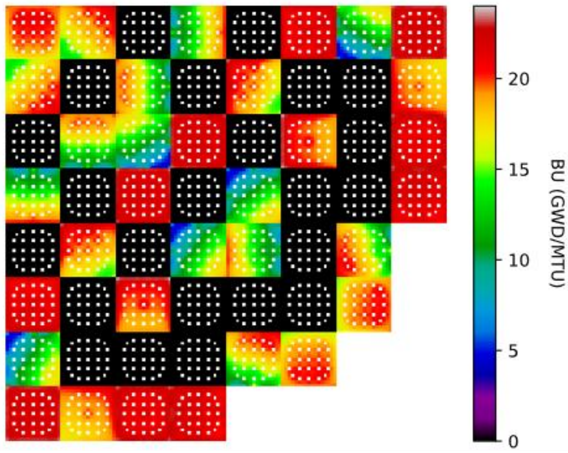

Fig. 3. FRAPCON Burnup at BOC 2

The burnup information at EOC 2 and BOC 3 is presented in Fig. 4 and Fig. 5 respectively. The shuffling and rotation information has been verified using cycle 2 and cycle 3 configurations [2]. In cycle 3, one assembly is directly loaded from cycle 1. In RAST-K, a single restart file is needed for the entire cycle, while FRAPI requires multiple restart files equal to the number of burnt fuel rods.

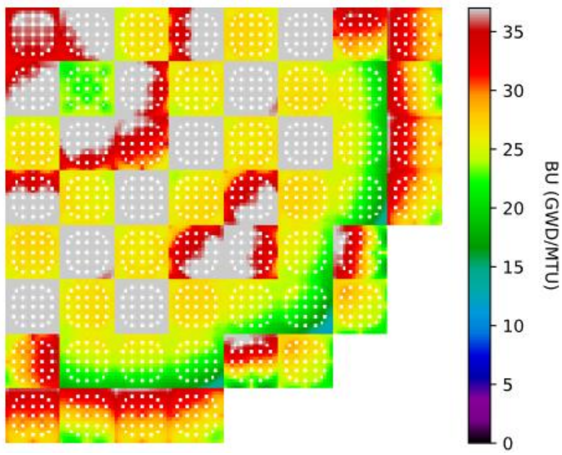

Fig. 4. FRAPCON Burnup at EOC 2

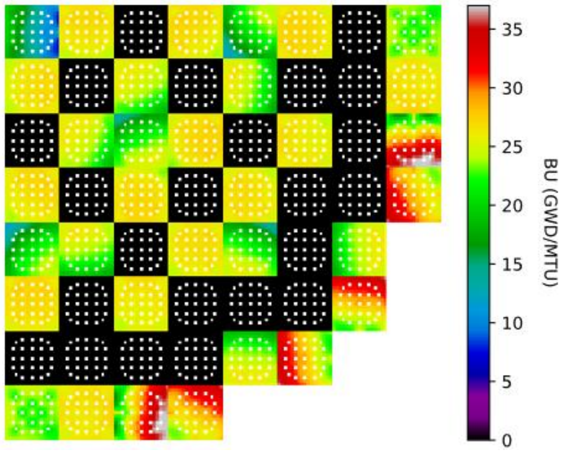

Fig. 5. FRAPCON Burnup at BOC 3

The integration of shuffling and rotation capability for RAST-K and FRAPCON into MPCORE has been successful. Along with benchmark information, gap conductance as a function of power and burnup needs to be examined. Fig. 6 presents the axial integrated burnup and linear power at 3.7 EFPD for cycle 2, as well as the axial integrated gap conductivity values.

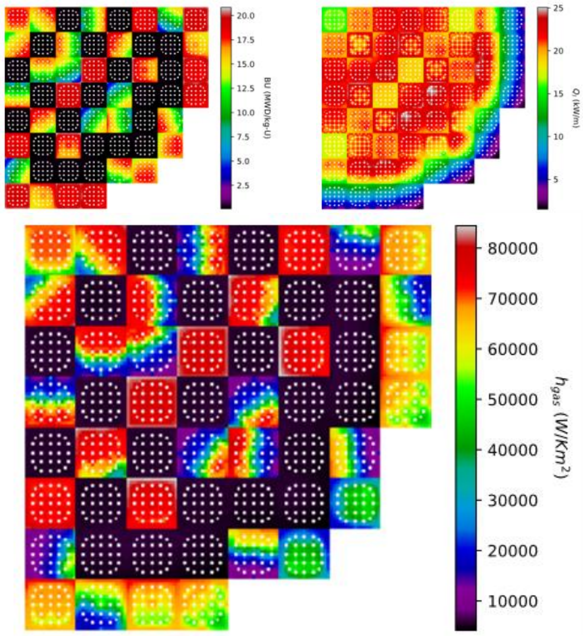

Fig. 6. Axial integrated gap conductivity given by FRAPCON with burnup and power at 3.7 EFPD cycle 2.

The assemblies depicted in black represent fresh fuel assemblies, and their gap conductance is approximately 5000 W/Km^2. On the other hand, for burnt assemblies, the gap conductance has increased to up to 80000 W/Km^2. Fig. 7 shows the variation of gap conductivity with burnup and power for cycle 3 at 5.1 EFPD. For twice-burnt fuel, the gap conductivity for burnt fuel rods has increased up to 100,000 W/Km^2.

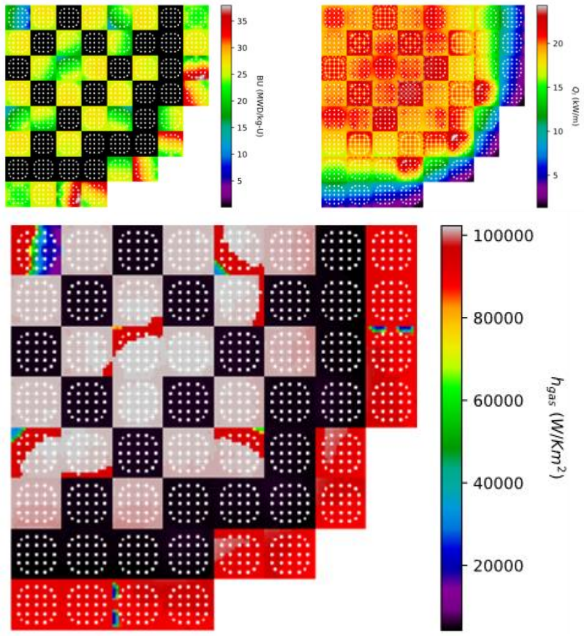

Fig. 7. Axial integrated gap conductivity given by FRAPCON with burnup and power at 5.1 EFPD cycle 3.

Fuel rods featuring axial meshes coated with IFBA and hollow blanket meshes were modeled in FRAPI using Python input processing. In cycles 2 and 3, IFBA coating was applied to selective fuel rods to balance high reactivity at BOC. The IFBA coating was aligned at the center, and to verify the model, the center mesh of fuel rods in FRAPI is presented in Fig. 8 for cycle 2 and Fig. 10 for cycle 3. The gap width at the center mesh in FRAPI is also displayed in Fig. 9 for cycle 2 and Fig. 11 for cycle 3. Burnt fuel rods have reduced gap width, as determined from restart files, while fresh pins have reduced gap width due to the IFBA coating.

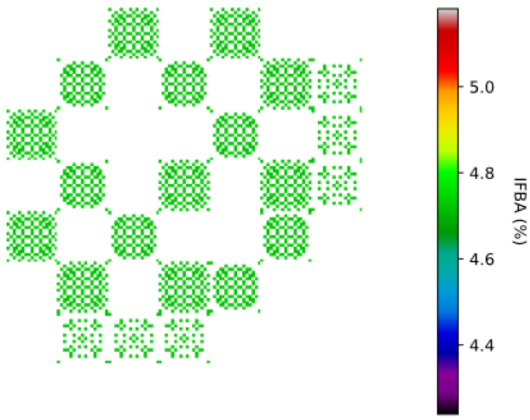

Fig. 8. IFBA coated pins in cycle 2.

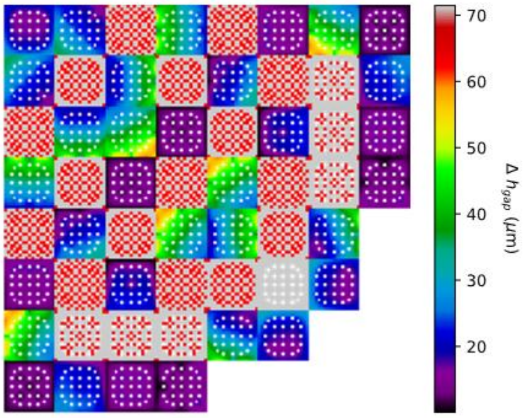

Fig. 9. Gap width at BOC cycle 2.

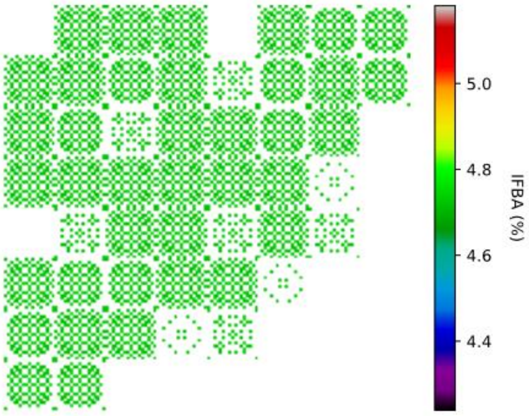

Fig. 10. IFBA coated pins in cycle 3.

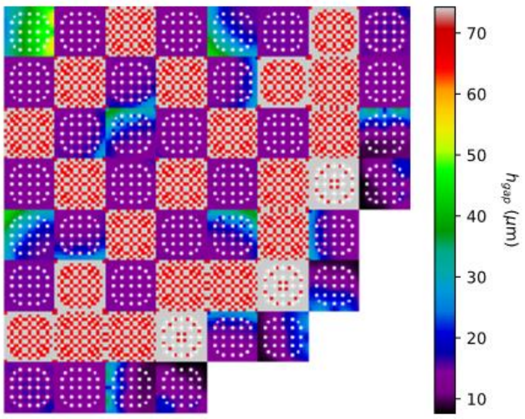

Fig. 11. Gap width at BOC cycle 3.

5. Conclusion

In MPCORE, multicycle capability has been developed and verified for three cycles of Watts bar reactor. The Python input processing utility in MPCORE enables detailed modeling in different modules. Since RAST-K is a nodal code, fine fuel rod modeling is not necessary at this level, and instead, fine fuel rod details are performed at the cross-section generation part in STREAM. FRAPI, on the other hand, utilizes individual fuel rod details to accurately model fuel behavior with burnup. The reduction in gap width, accumulation of internal gas pressure, fuel cracking, and fuel thermal conductivity all depend on the IFBA coating, hollow fuel pins, and fuel enrichment. FRAPCON has the capability to model different axial meshes with varying characteristics. Information is passed to each module through its own input file, which is processed by the Python input processing utility of MPCORE.

ACKNOWLEDGEMENTS

This research was supported by the project(L20S089000) by Korea Hydro & Nuclear Power Co. Ltd..

REFERENCES

[1] A. T. Godfrey, "VERA core physics benchmark progression problem specifications.," Oak Ridge National Laboratory, Oak Ridge, CASL-U-2012-0131-004., 2014.

[2] S. Stimpson *et al.*, "Pellet-clad mechanical interaction screening using VERA applied to Watts Bar Unit 1, Cycles 1–3," *Nuclear Engineering and Design*, vol. 327, pp. 172-186, 2018.

[3] J. Park *et al.*, "RAST-K v2—Three-Dimensional Nodal Diffusion Code for Pressurized Water Reactor Core Analysis," *Energies*, vol. 13, no. 23, 2020.

[4] R. K. Salko, R. C. Schmidt, and M. N. Avramova, "Optimization and parallelization of the thermal–hydraulic subchannel code CTF for high-fidelity multi-physics applications," *Annals of Nuclear Energy*, vol. 84, pp. 122-130, 2015.

[5] A. Cherezov, H. Kim, J. Park, and D. Lee, "Fuel Rod Analysis Programming Interface for Loosely Coupled Multiphysics System," presented at the In Proceedings of the American Nuclear Society Annual Meeting, Chicago, IL, USA, 16-19 November, 2020.