Doctoral Thesis

Development of Reactor Multiphysics framework to analyze the effect of crossflow and dynamic gHTC for Depletion and REA transient

Awais Zahur

Department of Nuclear Engineering

Ulsan National Institute of Science and Technology

2023

Development of Reactor Multiphysics framework to analyze the effect of crossflow and dynamic gHTC for Depletion and REA transient

Awais Zahur

Department of Nuclear Engineering

Ulsan National Institute of Science and Technology

Development of Reactor Multiphysics framework to analyze the effect of crossflow and dynamic gHTC for Depletion and REA transient

A thesis/dissertation submitted to
Ulsan National Institute of Science and Technology
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Awais Zahur

06.16.2023

Approved by

Advisor

Deokjung Lee

Development of Reactor Multiphysics framework to analyze the effect of crossflow and dynamic gHTC for Depletion and REA transient

Awais Zahur

This certifies that the thesis/dissertation of Awais Zahur is approved.

06.16.2023

Signature

Advisor: Deokjung Lee

Signature

Douglas A. Fynan

Signature

Eisung Yoon

Signature

Joosuk Lee

Signature

Taewan Kim

Abstract

The aim of this research is to create a Multiphysics coupling framework called MPCORE (Multi-Physics CORE) to analyze the behavior of nuclear reactors. This framework couples fuel performance (FP) with neutron kinetics (NK) and thermal hydraulics (TH) modules for depletion and transient analysis. Coupling the FP code allows for accurate modeling of dynamic gap heat transfer for each pin. Converging all modules together provides a more meaningful insight into the variation of reactor parameters. Depletion studies with Multiphysics parameters are essential to understand safety parameters throughout a nuclear reactor's life. The study investigates the passive response of the reactor core to reactivity insertions caused by rod ejection accidents (REA). Most coupling frameworks only couple NK with TH, but this research also includes FP and uses two-way coupling between TH and FP modules to examine the impact on critical safety parameters.

The adaptive time-step feature of MPCORE reduces execution time, and the framework performs in-memory data transfer between modules. Verification and validation work for MPCORE coupled modules (RAST-K for NK, CTH1D/CTF for TH, and FRAPI for FP) has been performed for single assembly, 3x3 mini-core, and whole-core problems. The performance of the TH module is evaluated with and without crossflow for transient calculations in whole-core problems. The effect of dynamic and static gap heat transfer coefficient models on the FP module is quantified for assembly, mini-core, and whole-core transient problems. Difference between one-way and two-way coupling between FP and TH modules is quantified for whole-core depletion problems.

The study compares safety parameters such as departure from nucleate boiling ratio, linear power, fuel enthalpy, fuel centerline temperature, cladding outer surface temperature, coolant temperature, and cladding hydrogen concentration for different models. A best-estimate coupling framework has been developed and tested for uncertainty quantification (UQ) studies for assembly and mini-core problems. Random sampling and Latin hypercube sampling options are available for UQ studies in MPCORE. Standard deviation of different parameters in case of dynamic gap conductance has increased due to the difference of gap heat transfer in different cases.

KEYWORDS: CTF, FRAPTRAN, Multiphysics, external loose coupling.

Table of Contents

List of Figures

List of Tables

Technical Terms and Abbreviations

1D	One dimension
CTF	Coolant Boiling in Rod Arrays- Two Fluid
IFBA	Integra Fuel Burnable Absorber
FRAPI	Fuel Rod Analysis Programming Interface
UNM	Unified Nodal Method
ANM	Analytical Nodal Method
NEM	Nodal Expansion Method
FA	Fuel assembly
LWR	Light water reactor
MPCORE	Multi Physics CORE
STREAM	Steady-state and transient reactor analysis code with the method of characteristics
TH	Thermal hydraulic
FP	Fuel Performance

1 Introduction

Nuclear power provides an alternative to primary sources of electricity. Renewable energy has many benefits, but its low-capacity factor makes it highly dependent on external factors. Coal and oil-powered plants produce a significant share of electricity, but also pose a threat to the environment. Nuclear power, however, can serve as a reliable primary source of electricity with minimal dependence on external factors and no significant impact on the planet's survival.

Among the nuclear power plants in operation, pressurized water reactors (PWRs) are the most common. According to data from the International Atomic Energy Agency (IAEA) [1], PWRs account for 303 out of 437 operational nuclear reactors, producing 77% of the total nuclear energy output. Of the 56 nuclear reactors currently under construction, 48 are PWRs, which will produce 90% of the nuclear energy output. This popularity is due in large part to PWRs' promising safety features.

1.1 Pressurized water reactor

The PWR uses a two-loop system for power generation. The primary loop is responsible for transferring heat from nuclear fuel to the water coolant at high pressure. This high pressure prevents bulk boiling, and the system is designed to ensure that it never occurs. The primary water coolant, which is at a high pressure and temperature, transfers heat energy through thin U-tube steam generators. The water in the secondary loop of the PWR is at a lower pressure, and boiling occurs in this loop. The dried steam produced in the steam generators is then passed on to turbines for electricity generation. There is only one reactor pressure vessel and core in a PWR, but there can be multiple steam generators and secondary loops depending on the power of the PWR. Figure 1-1 shows a typical two-secondary-loop PWR.

The separation of the primary and secondary coolant adds an extra layer of safety to prevent the release of radioactivity into the environment. In the event of fuel failure, fission fragments and other contaminants can enter the primary coolant, but they may not interact with the secondary coolant. The primary coolant is continuously monitored for radioactivity levels, and if the levels exceed a certain threshold, a reactor shutdown signal is generated.

The coolant passing through the primary loop has two main objectives: first, to transfer heat from the core to the steam generators, and second, to act as a neutron moderator in the core. Water, which has two hydrogen atoms and is readily available, is the best coolant option for PWRs. As the temperature of the core and coolant increase, the water's neutron scattering properties decrease, which reduces moderation and, hence, the fission capability of the fuel material. The increase in fuel temperature also results in a decrease in neutron absorption in the fuel. These negative feedback mechanisms of water and fuel are promising features of PWRs that have made them an attractive option for nuclear engineers for many years.

The control rod drive mechanism located at the top of the core can be used for power maneuvering but is mostly used for a rapid shutdown of the nuclear reactor, also called a reactor scram. The control rod drive mechanism is held in place due to a magnetic latch system that fails in case of a power loss. In the event of power failure, the control rods drop down in the core due to gravity. Thus, this passive safety system is another promising feature of PWRs.
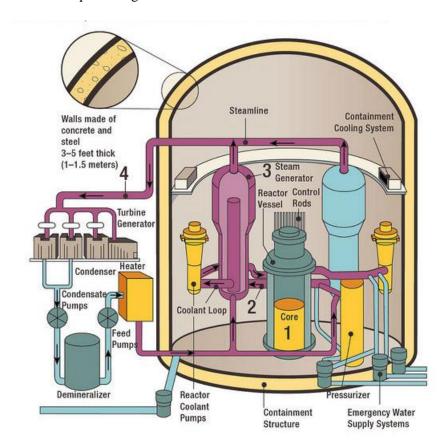
Figure 1-1: PWR components for primary and secondary loop (courtesy US NRC, 2023)

Many power plant systems are present in PWR that are related to heat exchange, pressure control, radioactivity control, reactivity, coolant temperature, coolant chemical control etc. All the systems are dependent on the nuclear core of the reactor.

1.2   Nuclear core

Heat is produced from the nuclear reactor's core, which contains an array of nuclear fuel assemblies ranging from 15 x 15 to 17 x 17, as depicted in Figure 1-2. Each fuel assembly consists of fuel rods or pins with a geometric arrangement of 15 x 15 to 17 x 17, where heat is generated by fission, producing more neutrons. The moderator thermalizes the neutrons, allowing fission to continue until a steady state is reached. The selection of nuclear fuel (such as uranium, uranium oxide, or uranium carbide) is based on melting temperature and thermal conductivity.

To prevent densification in case of temperature increase, fuel pellets are sintered to form a solid mass, and they are enclosed within a cladding material with high-temperature resistance and low neutron absorption. The fuel-clad gap is filled with helium to maintain high internal fuel pin pressure, as the coolant is under high pressure of approximately 15.5 MPa. Fuel pins are stacked together to create a fuel assembly with some guide tubes for the insertion of control rods later. These fuel assemblies are arranged in a two-dimensional array to form the core structure and are secured together by spacer grids, which slightly impede coolant flow. In addition to fuel and moderator, non-fuel materials can also absorb neutrons, leading to non-fuel absorption and decreasing reactivity.
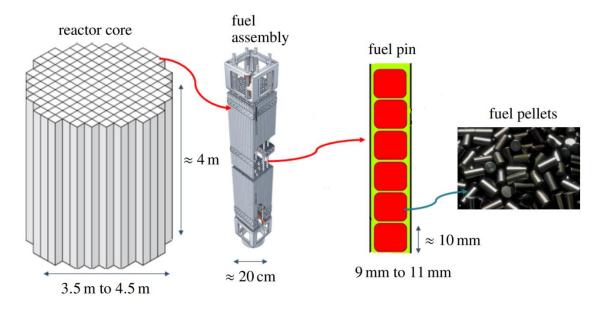
Figure 1-2: The discrete heterogeneous material regions in a PWR reactor core[2]

1.3 Single physics vs Multiphysics

The fuel pellet acts as the primary control barrier for reactivity, followed by the cladding surrounding the fuel rod. Heat is transferred from the fuel to the coolant through the fuel, gap, and cladding surfaces. The conductivity of the fuel and gap depends on the burnup and linear power. The oxide and crud layers on the cladding wall also hinder heat transfer from the cladding to the coolant, as shown in Figure 1-3. The heat transfer through solids and liquids is higher than that through gases, as seen in the figure, where the largest temperature gradient is observed from the fuel surface to the clad inner surface temperature. The reduction of gap conductance with burnup must also be modeled, as it can lead to pellet-clad mechanical interaction (PCMI).

In a nuclear reactor core, power produced depends on the critical state of a reactor. Criticality state often denoted with k_{eff} is given by,

$$k_{eff} = \frac{neutrons\ produced\ in\ generation\ i+1}{neutrons\ produced\ in\ generation\ i} \tag{1.1}$$

It is calculated by the four-factor formula (in case of k_∞) or six factor formula (in case of k_{eff}).

$$k_{eff} = k_\infty . P_f . P_t = \eta . \varepsilon . p . f . P_f . P_t . \qquad (1.2)$$

Where k_∞ is the infinity multiplication factor if we assume the reactor with infinite dimension. P_f and P_t are the fast and thermal neutron leakage probability. η is the reproduction factor, ε is the fast fission factor, p is the resonance escape probability, and f is the thermal utilization factor. Neutron thermalization is the key to maintaining the critical state in a PWR. Neutron cross-section is defined as the likelihood of neutron interaction with a target nucleus. If the interaction leads to fission than it is called fission cross-section, if it leads to scattering than it is called scattering cross-section and so on. Nuclear fuel has high fission cross-section for low energy neutrons and less fission cross-section for high energy neutrons. So, it is desired in a PWR to thermalize (decrease neutron energy) neutrons that are produced because of high energy fission reaction. For this purpose, water coolant is used that can thermalize neutrons and have less absorption cross-section.

The main computer programs that calculate a reactor's criticality are known as reactor physics, neutronics, or neutron kinetics (NK) codes. These programs focus on the neutron population in the reactor core. If the neutron production exceeds the neutron loss terms, the reactor is in a supercritical state, whereas for less neutron production than loss, the reactor is in a subcritical state. In steady-state operation, the neutron production is equal to the neutron loss term, and the reactor is in a critical state. In a new core, there is more fissile fuel present, which contributes to a positive reactivity in the core. To compensate for this, boric acid is used in the moderator as the primary control. Boron has a high neutron absorption cross-section, which decreases the neutron population. A control rod is also inserted from the top of the core as a secondary reactivity control mechanism. Some of the new fuel rods are coated with an Integral Fuel Burnable Absorber (IFBA), which adds negative reactivity at the beginning of the core but adds positive reactivity to the system after burnup. Additionally, hollow fuel pellets are added, which contribute positive reactivity to the system at the start of the cycle and negative reactivity at the end of the cycle. This is usually required at the axial top and bottom of the fuel rod. To calculate the effect of all these parameters, another quantity ρ is defined as reactivity and given by,

$$\rho = \frac{k_{eff} - 1}{k_{eff}} . \qquad (1.3)$$

Reactivity in a nuclear reactor must be kept equal to 0 for steady state operation. Change in reactivity is expressed by the following equation,

$$\frac{d\rho}{dt} = \frac{\partial \rho_{cs}}{\partial t} + \frac{\partial \rho}{\partial T_f} \frac{dT_f}{dt} + \frac{\partial \rho}{\partial T_m} \frac{dT_m}{dt} + \frac{\partial \rho}{\partial \alpha_m} \frac{d\alpha_m}{dt} . \qquad (1.4)$$

Where ρ_{CS} is the reactivity change by the reactivity control systems. T_f denotes the fuel temperature, T_m denotes the moderator temperature and α_m represents the void content. Increase in fuel and moderator temperature decreases the reactivity in a system. Increase in void content also decreases the scattering cross-section of moderator, hence, decrease in reactivity.
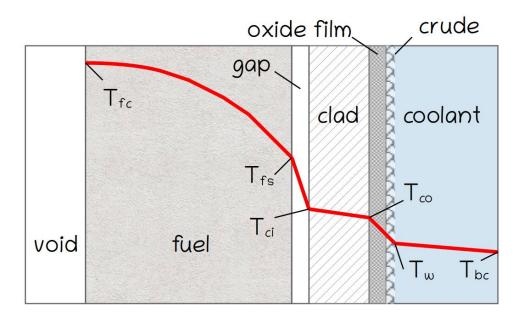
Figure 1-3: Heat transfer from fuel to coolant

In a nuclear reactor, one of the most significant phenomena is the thermal hydraulics of the coolant. The coolant serves as the ultimate heat sink in the core, with high pressure water entering from the bottom and increasing in temperature as it rises. This process is driven by the heat transferred from the cladding outer surface. Typically, in operational reactors, the surface temperature is higher than the saturated fluid temperature but below the critical heat flux, which leads to nucleate boiling on the surface. Thermal hydraulic codes are primarily concerned with the boiling regime in nuclear reactors, and in steady-state conditions, efforts are made to avoid a departure from nucleate boiling. In designed basis accidents, any departure from nucleate boiling is minimized.

The heat transfer from bottom to top is the dominant transfer process and is referred to as 1D modeling. However, when considering the pressure and temperature differences between different channels and modeling crossflow, the codes used are called sub-channel codes. Thermal hydraulic codes also model the heat transfer from fuel to cladding surface, using simplified heat conduction correlations.

Another important phenomenon in nuclear reactors is the fine heat transfer modeling from fuel to cladding surface. This process is modeled using fuel performance (FP) codes. As the fuel burns, it expands, and fission products accumulate, leading to changes in the gap between fuel and clad. The gap conductivity is a function of burnup and linear power, and fuel rods often have an IFBA coating that

reduces gap width and alters fuel heat conduction. The heat conduction correlations used for hollow fuel pellets are also different.

Increasing water and fuel temperatures lead to a decrease in the scattering cross-section and the resonance escape probability, respectively, resulting in a decrease in reactivity. Therefore, the NK codes are dependent on the thermal hydraulics and FP codes. Similarly, the FP codes depend on the NK codes for power input and thermal hydraulic codes for coolant information used as boundary conditions. Thermal hydraulic codes also depend on the FP codes for clad surface temperature and NK codes for moderator direct heating, taken as a fraction of the total linear power.

1.4 Steady state depletion

At the beginning of a nuclear reactor's operation, control rod withdrawal and appropriate boric acid concentration are used to achieve steady-state operation. The production of power in the reactor requires the fission of fissile material nuclides, and as the fuel burns, the inventory of fissile nuclides decreases, leading to a decrease in fuel reactivity. This decrease is compensated for by a decrease in boric acid concentration in the moderator.

To calculate the changes in critical boron concentration with burnup, reactor depletion must be modeled in NK codes. Depletion models not only calculate the decrease in fissile material but also the production of higher actinides and fission products using depletion modules. Xenon-135 and Samarium-149 are particularly important fission products as they have high neutron absorption cross-sections, which adds negative reactivity to the system. During normal power operation, Xenon and Samarium reach a saturation state, providing a fixed amount of negative reactivity. However, Xenon is crucial in controlling power excursion due to its short half-life. Therefore, nuclear reactor power increases and decreases cannot be solely controlled by power demand but must also take Xenon inventory into account.

Changes in the material composition of fuel pellets also affect fuel heat conduction properties. Depletion modules are therefore included in FP codes, as changes in fuel density, the presence of fission products, and irradiation defects all depend on fuel burnup. Depletion calculations model a nuclear reactor during the cycle life of the reactor and are often used to model the reactor at steady-state operation. The codes used to model power excursions that occur during the cycle length of a reactor are also of great importance.

1.5 Reactivity initiated accident

The licensing process for nuclear reactors requires an analysis of the reactor's response to design basis accidents. One such accident is the Rod Ejection Accident (REA), also known as the Reactivity Initiated Accident (RIA), which gained significance after the SL-1 accident. During this accident, four out of five control rods were inside the reactor core, and the ejection of the central single control rod made the reactor supercritical, causing an explosion that killed three people and released a small amount of

reactivity into the environment. Since then, regulatory requirements have ensured that the reactivity worth of the control mechanism is checked, and it is now regulated that the worth of a single control rod cannot be so high as to make the reactor critical.

The analysis of this accident requires NK, FP, and TH analysis, as the reactivity and power excursion lead to high heat transfer from fuel to cladding, which can cause Pellet Clad Mechanical Interaction (PCMI) and result in different flow regimes that do not occur during normal reactor operation. Reactivity control after control rod removal is usually by the negative feedback of fuel and coolant. The immediate increase in fuel temperature leads to doppler feedback, which decreases fuel reactivity. Doppler feedback occurs when the resonance spectrum of fuel broadens, leading to a low resonance escape probability. Furthermore, an increase in coolant temperature decreases the scattering cross-section of hydrogen atoms, which prevents neutrons from thermalizing.

Inadvertent control rod withdrawal during hot zero power and hot full power conditions can have different consequences. In normal reactor operation, none of the control rods are fully inserted in the core during HFP conditions, so the removal of control rods may not have catastrophic results. However, the coolant and fuel temperature are already very high, and even slight increases in their temperatures can be severe. In contrast, during HZP, all control rods are fully inserted in the reactor, and the fuel and coolant temperature are maintained at 291.5 $^{\circ}$C. Complete removal of a control rod during HZP can cause the reactor to become prompt critical, and the heat flux can surpass the Critical Heat Flux (CHF).

1.6   Thesis layout

In Chapter 1 of this thesis, the fundamentals of reactor depletion, REA transient, and the necessity of Multiphysics analysis for normal and transient modeling of a nuclear reactor are presented. Chapter 2 reviews various methods and techniques developed by researchers for performing Multiphysics coupling calculations. In Chapter 3, an overview of the modules that are coupled in the current research work is provided. Chapter 4 outlines the main features offered by the Multiphysics coupling framework, MPCORE, developed in this study. Chapter 5 analyzes the entire core depletion and transient studies conducted in this research. Chapter 6 analyzes the uncertainty studies performed for single assembly, using random sampling, and 3 x 3 mini core, using Latin hypercube sampling, problems. Chapter 7 examines the impact of crossflow in transient calculations and dynamic gap conductance in steady state and transient simulations. Finally, Chapter 8 concludes the research work, compares the results, and provides prospects for the developed multipurpose Multiphysics framework.

2   Literature Survey - Previous coupling frameworks

Previous coupling approaches involving NK, TH and FP codes are first discussed in this chapter. NK and TH coupling codes are also discussed later in this chapter.

2.1   PARCS-SUBCHANFLOW-TRANSURANUS

Multiphysics tool comprising of PARCS as neutronics code, SubChanFlow as thermal hydraulic code and TransUranus as fuel performance code was developed for the analysis of full core RIA transient at Karlsruhe Institute of Technology, Germany [3]. Two step neutronics models use lattice code calculations at assembly level to generate cross sections at different temperatures and boron concentration. Nodal code solves the diffusion equation in the second step by using the multi group cross sections generated by lattice code. In this research, Serpent2, a Monte Carlo code, is used to generate multigroup cross section and form functions for pin power reconstruction. PARCS is used in the second step to provide linear power of each pin at each axial level. PARCS is a 3D core simulator code that can solve steady state and time-dependent, multigroup diffusion equation as well as low order SP3 transport equations for orthogonal and hexagonal geometries. Two group nodal equations are solved using Analytic nodal method for nodes where k is not close to 1 and Nodal Expansion Method for k close to 1 (called critical nodes). Pin power reconstruction is performed in PARCS using analytic method. Pin power reconstruction in nodal code shows 1-5% difference with higher order solutions that can go up to 10-15% local differences for some cases.

The linear power provided by PARCS is used by TU for fuel thermos mechanics calculation and calculates temperatures of fuel centerline, fuel doppler and clad outer temperature. TU can solve the steady state and transient problems that include fuel material phase changes as well. TU can solve one fuel rod at a time like general fuel performance codes. The active fuel height is divided into slices like axial meshes in nodal code. The rod is discretized in radial direction as well with each radial zone can have different composition. TU solves the heat transfer equations using the best estimations thus it replaces the thermal hydraulic code heat conduction solver. Different fuel rods are analyzed with different initial gap width are analyzed in this research that shows the gap width increase in start due to densification and then it starts decreasing later. The gap heat transfer coefficient varies from 5000 W/m^2K to around 130,000 W/m^2K.

Clad outer temperature and power is provided to SCF for performing crossflow enabled thermal hydraulic calculations. SCF solves the energy, mass and momentum conservation equations for the two-phase liquid and vapor mixture. momentum conservation is solved separately for axial and lateral direction; hence 4 conservation equations are solved. Thermal hydraulic safety parameters like minimum departure from nucleate boiling, critical heat flux or maximum pin power is evaluated for the most compromised fuel rods.

Two-way data transfer occur between all the constituent codes and the overall flow scheme for this coupling is shown in Figure 2-1. Fuel temperature is provided by TU code to neutronics, and coolant temperature and density are provided by TH code. Neutronics supplies linear power to TH and FP codes. Clad outer surface temperature is provided by TU to SCF code.
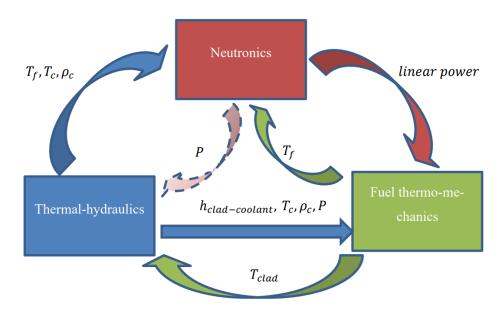
Figure 2-1: Feedback variables transferred in PARCS-SCF-TU coupling (courtesy ref [3]).

Multiphysics problem is treated as a function of three variables and three equations. They are solved using the Gauss Seidel algorithm. PARCS-SCF is made one code at first step and in the second step TU code is coupled with PARCS-SCF. PARCS-SCF solves for the node wise power and node wise coolant channel while TU solves each fuel rod independently. MPI parallel implementation is done so that each processor solves for single fuel assembly rods and all the processors send information back to the master node. Extension of nodal level PARCS-SCF to a pin level code is targeted as a future work. Adaptive time step control and a general coupling approach are set as the targets and has not been performed in this work.

2.2 APOLLO3-FLICA4-ALCYONE

APOLLO3 neutronics code, FLICA4 thermal hydraulic code and ALCYONE FP code are coupled at École Polytechnique, Université Paris-Saclay, France [4]. It was developed initially without fixed point convergence scheme and tested for REA at HZP BOC condition. The flow scheme is shown in Figure 2-2. APOLLO3 neutronics code is used that calculates multigroup cross-sections using APOLLO2 and calculate pin power using MINOS-MINARET approach. MINOS solves the transient equations with SP_n neutronics equations, 10 energy groups and zero flux boundary condition. MINARET solves Sn neutronic equation with static conditions, 30 energy groups and reflective boundary condition. From the scalar flux obtained from both the codes, most detailed space and energy representation of the pin

cell spectra is obtained. Power is calculated inside fuel and fluid by the moderator heat fraction constant. Cross-section is updated based on the temperature feedback from TH and FP codes.

Each fuel rod is independently modeled in ALCYONE thermomechanics code. It includes crack law, creep law, solid swelling and densification, gas swelling, thermal expansion, and fuel pellet physical data for the fuel pellet. Viscoplastic behavior, creep behavior thermal expansion and fuel pellet cladding physical data is also calculated at each timestep in ALCYONE FP code. 1.5D modelling of ALCYONE is used because of rotational symmetry. Doppler temperature feedback by FP code is important for cross section update in neutronics code. The researcher proposed a new formulation that takes into account the isotopic fraction of fissionable material.

$$T_{eff}^{Iso} = \frac{\sum_{i=1}^{R}\sum_{k=1}^{K}T_i A_i^k}{\sum_{i=1}^{R}\sum_{k=1}^{K}A_i^k} \tag{2.1}$$

Where R is the number of radial rings in fuel pin discretization and $A^k(r) = N^k(r)\sigma_a^k(r)\Phi(r)$. N^k is the quantity of each isotope k (mainly corresponding to U238 and Pu240). σ_a are the microscopic cross section and Φ is the prompt neutron flux. A dynamic gap heat conductance model is used, and clad surface temperature is given by thermohydraulic code and used as a boundary condition.

FLICA4 is used as a 3D thermohydraulic 2 phase code that solves 3 conservative equations and 1 fluid-gas equation. Fluid power provided by neutronics code and clad surface conduction variables provided by FP code are used to calculate moderator density and clad outer surface temperature.

Coupling was started step by step so initially two codes are coupled with the simplification of third. The last one with all the codes is named as a best effort code. It involves all the three codes with fuel temperature and moderator density input to the neutronics module, fuel power and clad surface temperature as input to FP code and fluid power and thermal pin coefficient as input to the thermohydraulic code. Fixed point convergence scheme is not applied in this work and all the modules are executed only once at any timestep. Neutronics is solved by taking 4 meshes in each assembly and 30 axial meshes. The same number of axial meshes are used in TH and FP formulations as well.
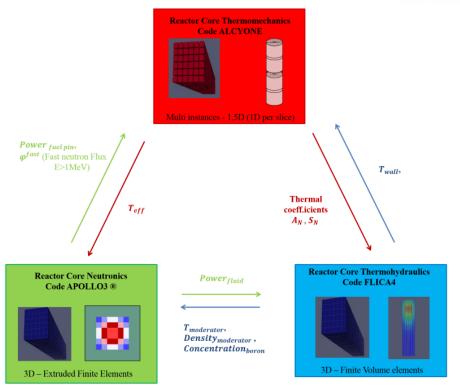
Figure 2-2: APOLLO3-ALCYONE-FLICA4 coupling scheme and exchanged variables (courtesy ref [4])

This best effort code was further changed into best estimate code by removing FP code ALCYONE completely. The best estimate code uses two group diffusion theory-based code APOLLO3 and 1D thermal hydraulic code FLICA4. Improved best estimate code with machine learning trained FP was used for uncertainty quantification studies [5]. The comparison between best effort, best estimate and improved best estimate is shown in Figure 2-3.
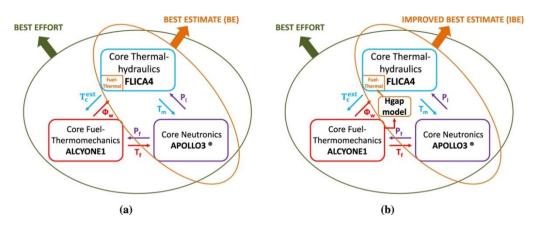
Figure 2-3: CEA coupling framework used for REA (a) and the proposed improvement in the BE scheme (b) (courtesy ref [5]).

2.3 MPACT-CTF-BISON

Virtual Environment for Reactor Applications (VERA) simulator capabilities has included BISON fuel performance code [6] with the already coupled MPACT/CTF code [7]. Inline approach involving one-way coupling of fuel performance code and tiamet fully coupled two-way approach is used for BISON code. In two-way coupling analysis the fuel temperature predicted by BISON is fed back to MPACT/CTF for refinement of the linear power and coolant temperature distribution. The two-way approach converges to a consistent solution at each depletion step by transferring data between the modules. 1.5D capability of BISON (neglect axial thermal conduction) is used in the coupling analysis. 3 radial rings are used in MPACT while 6 non-uniform radial nodes are used in BISON fuel region part. Radial average power is provided by MPACT to BISON. Axial planes in MPACT and BISON are the same, so no interpolation is performed for data exchange.

Although it is written by researchers that BISON fuel temperature is provided to MPACT/CTF coupled code, but the flow scheme shows (Figure 2-4) that fuel temperature is only updated in MPACT calculations. While no information from BISON is given to CTF, thus tiamet fully coupled approach also did one-way coupling between FP and TH codes. BISON uses Clad outer surface temperature provided by CTF which means that heat conduction in fuel and cladding are performed in CTF as well. This results in coolant temperature prediction partially independent from BISON fuel performance code. The word partially is used because BISON fuel temperature is provided to MPACT, so the change in power effect is translated in CTF calculations.
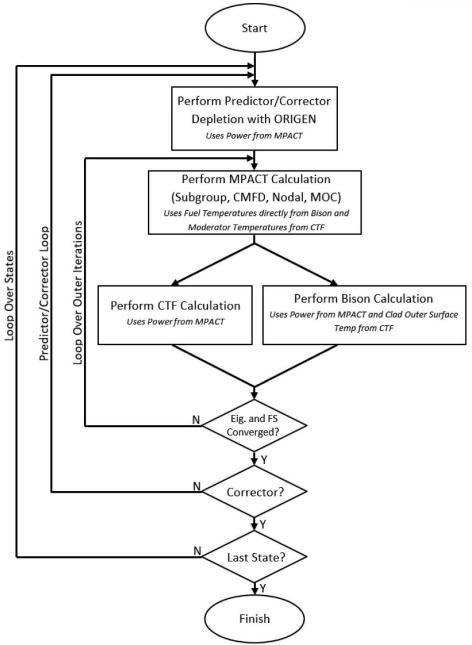
Figure 2-4: Flowchart of the Tiamet Fully coupled calculation (courtesy ref [6]).

2.4   Ants-SUBCHANFLOW-SuperFINIX

Nodal neutronics code ANTS, subchannel thermal hydraulic code SubChanFlow and 1.5D fuel performance code SuperFINIX is coupled at VTT Technical Research Center of Finland [8]. A central Multiphysics python-based driver code Cerberus was created to couple multiple physics code together. Data transfer between codes, time step control and convergence checking are provided in Cerberus. The first operating cycle of BEAVRS was simulated with the coupled methodology. Maximum fuel centerline temperatures and minimum DNBR are evaluated at three timepoints during the cycle. ANTS diffusion code results are compared with Serpent2. Cross sections for ANTS code were also generated with Serpent2 code. SCF code can solve steady state and transient problems and have simplified models

for cracking, swelling and gap conductance. FINIX solves one fuel rod at a time but for running whole core problem tens of thousands of fuel pin simulations are required. SuperFINIX is a wrapper code created for parallel implementation of multiple fuel rods in FINIX using MPI and OpenMP.

ANTS provides updated power field to SCF and SuperFINIX. SCF uses the power field and transfers coolant temperature and density fields to ANTS. Heat transfer coefficient and coolant temperature fields are also transported to SuperFINIX. A new solution for fuel rod temperature based on T_{eff} is supplied to ANTS. T_{eff} is calculated as the fraction of fuel center and fuel surface temperatures.

$$T_{eff} = 0.7*T_{surf} + 0.3*T_{center}. \tag{2.2}$$

The convergence is checked at each time step by comparing differences in critical boron concentration, fuel temperature field and coolant temperature field. Nothing is mentioned about the transfer of any data from SuperFINIX to SCF means that SCF is using its own simplified heat conduction solvers.

2.5 MCS-CTF-FRAPCON

In MCS-CTF-FRAPCON coupling done in Ulsan National Institute of Science and Technology [9], the burnup dependent fuel and gap conductance models are used instead of burnup independent models of CTF. Burnup dependent fuel conduction models of FRAPCON are considered superior to CTF models and CTF burnup dependent fuel conduction correlation were not used. Coolant temperature and pressure information is transferred from CTF to FRAPCON and MCS. Power is provided from MCS to CTF and FRAPCON. FRAPCON is updating the fuel temperature to MCS. No exchange of information took place from FPACON to CTF thus the coupling of TH and FP is one way coupling as shown in Figure 2-5.
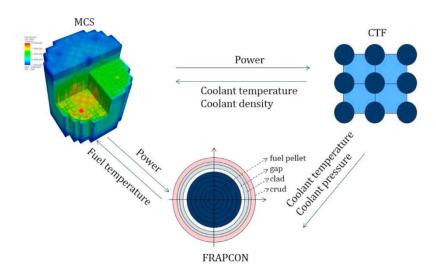
Figure 2-5: Coupling information exchange between CTF, FRAPCON and MCS (courtesy ref [9]).

The convergence might be achieved in a smaller number of iterations, but the CTF fuel conductivity correlations are still used for coolant temperature calculation. This study has reported only the steady state beginning of cycle results, so the burnup dependence effect couldn't be studied. Operator splitting method is used for the coupling interface. The coupling interface was developed as a module in MCS. Coupling interfaces of CTF and FRAPCON eases the information exchange between all the codes. The channel centered approach is used in CTF while MCS and FRAPCON uses the rod centered approach so all the parameters from CTF are transferred to MCS and FRAPCON by taking weighted average of all channels in contact with a rod. Internal convergence criteria of FRAPCON, CTF and MCS are in place. In addition to those, external convergence criteria between modules are applied. Relative error (ε) for neutron flux as a primary parameter δ was used for convergence given as:

$$ RMS\left(\left\|\frac{\delta^{n+1} - \delta^n}{\delta^n}\right\|\right) < \varepsilon. \tag{2.3} $$

Numerical results for the BOC HFP and HXP state for BEAVRS cycle 1 was presented to verify the accuracy and efficiency of MCS/CTF/FRAPCON.

2.6   NECPX-CTF-CALF

High fidelity transport theory based neutronics code NEXPX, subchannel thermal hydraulic code CTF and 1.5D fuel performance code CALF is coupled at School of Nuclear Science and Technology, Xi'an Jiaotong University [10]. NECP-X uses 69 energy group cross-section libraries based on ENDF/B-VII.0 for transport calculations. MOOSE platform is used to couple all the three physics modules. In memory data transfer is obtained by using the Master-Multiapp architecture of MooseObject. NECP-X and CTF used the same axial mesh division, so the values are directly transferred between them. The difference is the rod centered and channel centered approach, so average value of 4 adjacent channel values from CTF are transferred to neutronics code. Linear interpolation of parameters is done for NECP-CALF as it uses different axial meshes than neutronics and TH codes. Picard convergence criteria is used at each timestep. Fuel cycle depletion for a PWR assembly for an accident tolerant fuel U3Si2 is performed. The data transfer between TH and FP code is one-way and no data from FP code is transferred to CTF. The data transfer between modules is given in Figure 2-6.
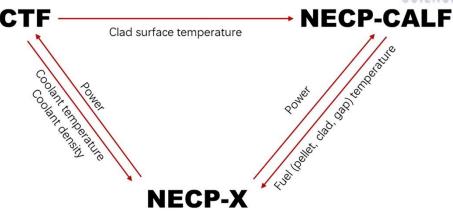
Figure 2-6: Data transfer in the coupling code system NECPX-CTF-CALF (courtesy ref [10]).

2.7   Serpent2-SubChanFlow-TransUranus

Serpent2, a continuous energy monte Carlo code, SubChanFlow, a subchannel thermal hydraulic code and TransUranus, a fuel performance code is coupled for the VVER-100- fuel assembly 360-day cycle depletion [11]. Initialization, termination, time-step control, depletion/transient calculations, and feedback variables exchange subroutine are written for all the codes. C++ classes derived from the same base class is written for all the codes. C++ supervisor program is written to couple all the codes. This utility allows in-memory transfer of data between modules and convergence is achieved between all the modules. In this mode of coupling, the interface of each class is the same and thus the coupling scheme is more simplified. The coupling derives from the traditional neutronics/thermal hydraulics coupling by just replacing the fuel solver in SCF with TU.

Two-way coupling is thus used between each of the constituent modules. Though SCF is developed using Fortran90 library, TU with Fortran95 library and Serpent2 with C library but the C++ interface has made it easier to couple all the codes without making changes to all the modules. Data from each code is obtained with its associated mesh and interpolation is performed with target mesh before providing data to target code. For the convergence of the coupled modules, multiplication factor convergence of 30pcm and 1% L_∞ norm for coolant density, fuel temperature and pressure are used. Higher fuel temperature corresponding to low multiplication factor is observed for the Serpent2-SCF-TU coupling compared with simple serpent2-SCF coupling. No major changes in coolant parameters are reported. Significant differences observed in gap conductance and fuel conductivity were observed. High burnup effect that could lead to larger differences in fuel models of SCF and TU were not studied in that research.

t_n

SUBCHANFLOW
(t_{n+1}, steady state)

$h_{clad-cool}$, T_{cool}, p

P

TRANSURANUS
($t_n \rightarrow t_{n+1}$, burnup)

T_{cool}, ρ_{cool} T_{fuel} P

Serpent2
($t_n \rightarrow t_{n+1}$, burnup, SIE)

Convergence at t_{n+1}?

t_{n+1}
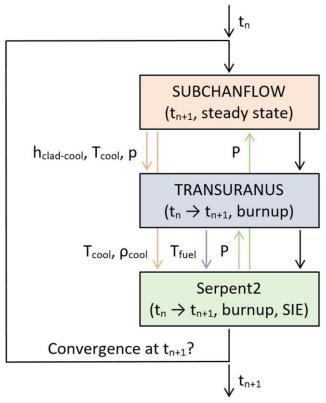
Figure 2-7: Depletion Scheme for Serpent2-SubChanFlow-TransUranus (courtesy ref [11]).

2.8 Neutronics-thermal hydraulic coupling frameworks

Although coupling of NK, TH and FP are done in this research and many of the previous research stated above but most of them were started with NK and TH coupling. In these two codes coupling, the fuel heat conduction part was solved with the simplified fuel conduction solvers of TH codes. A short description of each of the coupling platforms is given below.

2.8.1 RMC-CTF

VERA cycle 1 depletion problem has been solved with Monte Carlo code RMC coupled with subchannel thermal hydraulic code CTF [12]. Coupling convergence was investigated based on power distribution and Keff. The steady state power profile is compared with the benchmark results of MPACT/CTF and MC21/CTF and the whole core depletion analysis was performed. The Rod centered approach is used in RMC and channel centered approach is used in CTF. Average rod parameters based on neighboring channels data is provided from CTF to RMC. RMC generates script file to execute CTF with the power provided by RMC. CTF writes data in ".HDF5" files and RMC scripts read data for the neutronics calculation. File read and write is used for data transfer between the two codes. Convergence at each step is achieved based on the power distribution using Shannon entropy. Once the coupling converges, boron concentration search in RMC is performed. The maximum difference for assembly power distribution at HFP BOC steady state is -3% with 6 ppm error. The boron concentration for the whole cycle is larger (difference around 200 ppm) without coupling but after coupling the boron

concentration results matches with the measured boron concentration (maximum absolute difference of 20 ppm).

2.8.2   CTF-PARCS

Two-step neutronics calculation using Polaris lattice calculation and PARCS as nodal code is used as a neutronics options while subchannel TH code, CTF is used for CTF-PARCS coupling. Neutronics-thermal hydraulics coupling is used for uncertainty quantification studies. Cross-section perturbation is performed using Sampler while Dakota is used to sample the CTF input perturbations. The most correlated input found out by this research was the ^{238}U inelastic and elastic cross-sections above 1 MeV. The fuel performance is simplified in thermal solver as it is computationally intensive to couple FP code in N-TH coupling. In steady state and depletion mode, Picard convergence is used for the coupling. In transient calculation, each code is solved only once, and the feedback is provided to other code for next timestep calculation. All the physics codes used smaller timesteps and they still used the other physics solver results of previous timestep [13].

2.8.3   CTF-DYN3D

3D Diffusion equation for two energy group was solved using Nodal expansion method with DYN3D core simulator code was developed at Helmholz Zentrum Dresden Rossendorf, Germany. Homogenous results are followed with pin power reconstruction and coupled with subchannel based TH code, CTF. Coupling was performed using NURESIM, open-source software Salome based, platform [14].

2.8.4   PARCS-TRACE

WIMS lattice code was used for cross-section generation and PARCS nodal code is used afterwards for criticality calculations. PARCS was coupled with the system code of thermal hydraulic TRACE. This coupling as validated for Westinghouse AP1000 reactor [15].

2.8.5   MCNP-COBRA EN

Monte Carlo code, MCNP, was coupled with subchannel code COBRA-EN and results were validated with MCNP6 and CTF code coupling. Simulation results for VVER-1000 core were also compared with Final Safety Analysis Report results. Due to high computational requirement of Monte Carlo code, it is hard to find a Monte Carlo code coupled with TH and FP code for core depletion or transient calculations [16].

2.8.6   MCNP6-CTF

MCNP6 Monte Carlo code coupled with subchannel TH code, CTF, was performed and validated for full assembly problems. Relaxation acceleration technique uses the current iteration values and previous iteration values. It was verified that relaxation acceleration technique always show convergence with enough iterations [17].

2.8.7   *Serpent-SubChanFlow[18]*

VTT Technical Research Center of Finland developed a Monte Carlo code Serpent. It was coupled with KIT Germany subchannel TH code names SubChanFlow. The work was performed for the development of tool to couple MC and TH code primarily for transient problems but worked fine for steady-state and burnup calculations as well [18].

2.9   NK-FP, FP-TH, and other NK-TH coupling

There has been abundant number of coupling performed between other NK codes with TH code like development of ENIRCO framework that coupled OpenMC and Shift MC codes with Nek5000 CFD code[19]. Other Monte Carlo codes were also coupled with subchannel TH code [20]. Coupling of transport code nTRACER with MATRA subchannel thermal hydraulic code has been coupled and validated for OPR1000 and C5G7 standards benchmark problems. Average and maximum coolant outlet temperature predicted by open-channel (sub-channel) is less than that of closed-channel (1D) model [21]. High fidelity transport code NEXP-X is coupled with CTF [22]. CTF is called after every transport sweep of NECP-X. The interface was developed to allow in memory transfer of data between NECP-X and CTF. Crossflow TH model and gap conductance models were studied for VERA HFP assembly and core cases. The iteration scheme was shown to be better than the traditional Picard iteration schema as TH module is called after every transport sweep of NECP-X. The sensitivity analysis shows the importance of crossflow and dynamic gap conductance for the neutronics-thermal hydraulics coupling. Angular parallel implementation in transport code allows taking advantage of many characteristic rays in MOC. For transferring data between CTF and NECP-X, all the spatial domain data from one code is gathered at one place and distributed to the other code. This avoids the potential of maneuvering the codes too much and they can be used as black box for the other code. Same axial meshes are used for both the codes to avoid accuracy and computation burden for temperature merge and reconstruction. Internal 1D T/H model and constant gap model are proved to be more conservative models in steady state analysis of VERA whole core.

Apart from the NK and TH code coupling. There has been extensive research on the TH and FP codes as well. This include CUPID TH code coupling with FINIX FP code [23]. CTF and FLOCAL TH codes were coupled with DYN3D code for the verification of CTF [24].

NK codes coupling with FP codes also have so many examples. PARCS was coupled with FRAPCON code for uncertainty quantification studies [25]. iMC Monte Carlo code was coupled with 3D FP code, BISON in KAIST [26].

VERA design suite, and architecture was developed keeping in mind the generality of coupling tools. The coupling between any of the NK, TH and FP codes could take place [27].

3   Coupled modules

MPCORE framework developed has coupled modules that are validated by now. The modules are explained in this chapter while the coupling methodology of MPCORE is explained in the next chapter.

3.1   Neutronics

Conventional two-step approach based on assembly homogenization is performed for the neutronics calculation in MPCORE. STREAM/RAST-K two step code has been verified and validated against many benchmarks [28, 29]. The overall structure for the two-step method used is given in Figure 3-1. STREAM [30-33] is a lattice physics, Method of Characteristics based, transport code that can simulate whole core depletion/transient calculations or generate cross-sections to be used by nodal code. Input files defining the pin detailed geometry and material information are provided to STREAM code that generates STN files for each provided assembly. Different branch cases cross-sections are generated from STREAM based on fuel and moderator temperatures, boron concentration, control rod position etc.   STN files for all the core assemblies are combined by STORA program and make one cross-section file to be used by RAST-K. RAST-K [34] is a nodal code that takes input files and restart information file (in case of burnt fuel case) along with cross sections to solve steady state or transient diffusion equation.
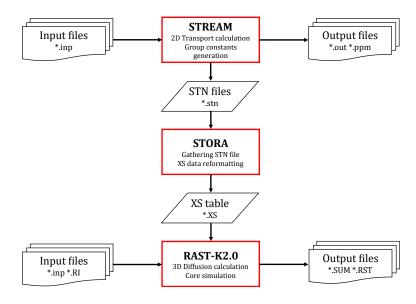
Figure 3-1: Flowchart of STREAM/RAST-K 2.0 code system (courtesy ref [28]).

RAST-K solves the two-group diffusion equation for the nodal power in 3D whole core geometry. Diffusion equations are solved using two-node nodal method with Coarse Mesh finite difference method. Unified Nodal method is used as a two-node method in RAST-K. Unified Nodal Method uses Analytic Nodal Method for all nodes except critical nodes. For the critical nodes, two-node NEM formulation is used for the local solution. CMFD solves global coupling by considering the current correction in neighboring nodes while nodal method corrects the local high order coupling of node. Pin

power reconstruction is performed in the next step to calculate linear power of each pin for every axial mesh. The two-group diffusion equation solved for the steady state is given as,

$$\nabla.J_1(r)+\Sigma_{r1}\phi_1(r) = \frac{1}{k}\left(\nu\Sigma_{f1}\phi_1(r)+\nu\Sigma_{f2}\phi_2(r)\right). \tag{3.1}$$

$$\nabla.J_2(r)+\Sigma_{a2}\phi_2(r) = \Sigma_{12}\phi_1(r). \tag{3.2}$$

The fast flux loss term includes Leakage J(r), absorption in the medium or scattering to the lower energy group. Absorption and scattering terms can be added to form removal term, because $\Sigma_{r1} = \Sigma_{a1} + \Sigma_{12}$, as written in first equation. Neutrons in lower energy are lost by leakage and absorption in medium. Up scattering to high energy groups is not considered, so for group 2 absorption and removal cross-section are the same thing. Self-scattering within a group is ignored as it does not cause any loss or gain in a group.

To solve the 2G equations given by Eq 3.1 and 3.2 using Analytic Nodal Method, Problem is converted to three 1D problems. Now the 1D problem is slightly different from the original 2G diffusion equation because it now includes Transverse Leakage term.

The transient two group diffusion equation is expressed with the time-dependent term because during transient part, the loss and gain term of neutrons may not necessarily be equal. Transient equation is given as,

$$\frac{1}{v_1}\frac{d\phi_1}{dt} = \frac{1-\beta}{k_{eff}}\left(\nu\Sigma_{f1}\bar{\phi_1}+\nu\Sigma_{f2}\bar{\phi_2}\right)+\sum_{k=1}^{6}\lambda_k C_k -\nabla.J_1 -\Sigma_{r1}\bar{\phi_1}. \tag{3.3}$$

$$\frac{1}{v_2}\frac{d\phi_2}{dt} = \Sigma_{12}\bar{\phi_2} -\nabla.J_2 -\Sigma_{a2}\bar{\phi_2}. \tag{3.4}$$

Where β is the delayed neutron fraction, ν is the neutron produced as a results of fission, λ is the mean precursor decay constant and C is the precursor concentration. The source term in Eq (3.3) has been written as the sum of prompt fission and delayed fission neutrons. Neutrons that are produced immediately after the fission reaction ($\sim 10^{-8}$s) are called prompt neutrons. Neutrons that are produced after some delay (0.1s-55s) are called delayed neutrons. Due to the wide difference in the emission time of delayed neutrons, they are often grouped based on their emission times. Six delayed energy groups are used in RAST-K. During steady state the delayed neutron precursors produced reached a saturation point so the neutrons contributing through delayed precursors stay same. In transient part, this is important because the delayed neutron precursors produced at current time state will express their effect later.

These equations are solved for each node using the current coming in and going out to calculate the leakage term. The average flux of node and eigenvalue is solved by Coarse Mesh Finite Difference (CMFD) method. The surface corrected current is calculated by,

$$J_g = \mp \tilde{D}_g \left(\phi_g^l - \phi_g^r \right) - \hat{D}_g \left(\phi_g^l + \phi_g^r \right).$$ (3.5)

Diffusion coefficient correction factor, $\hat{D}_g$ is calculated with the help of Coarse Mesh Finite Difference (CMFD) method. Nodal current is obtained and the homogenous flux for two adjacent nodes of the surface is known. Diffusion coefficient is calculated as,

$$\hat{D}_g = -\frac{J_g^{nodal} + \tilde{D}_g (\phi_g^r - \phi_g^l)}{\phi_g^r + \phi_g^l}.$$ (3.6)

Where,

$$\tilde{D}_g = \frac{2 D_g^r D_g^l}{D_g^r \Delta r + D_g^l \Delta l}.$$ (3.7)

These steps mentioned above will be repeated for each two-node problem. Homogenous cross-sections with assembly discontinuity factor are generated by the lattice physics code. To preserve the solution integrity at the corners with nodal code, discontinuity factor is used to calculate flux near the surface Discontinuity factor is the ratio of heterogeneous flux to homogenous flux at the surface. Form function for each pin is also generated by the lattice code that is used in pin power reconstruction part. The homogenous flux obtained by the nodal code is converted to pin wise values with the help of form function. Internal numerical convergence on flux and eigenvalue are checked at each iteration of RAST-K.

3.2 Thermal hydraulics

Thermal hydraulics modules deal with the transfer of energy from fuel rod surface to coolant. Coolant inlet temperature and outlet pressure is provided as an input to calculate the temperate axially along a fuel rod. Mesh average temperature is calculated by TH codes depending on fuel surface temperature and heat flux. Channel centered or rod centered approach, as shown in Figure 3-2, is used by TH codes where the former has been mostly used. Parameters values for rod can be taken from channel centered approach code by weight averaging the values across all the faces of the rod.

Coolant centered
subchannel

Rod centered
subchannel
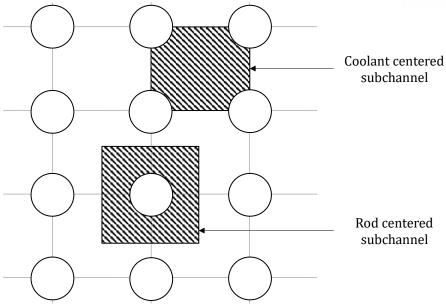
Figure 3-2: Fuel assembly sub-channels.

There are different flow regimes in a heating environment that depends on heat flux, saturation temperature and wall temperature at the position as shown in Figure 3-3.

Single Phase Convection	Nucleate Boiling		Transition Boiling	Film Boiling
Pure liquid	Isolated bubbles	Jets and columns	Disturbed slug flow	Annular flow

q''_{CHF}
Critical Heat Flux (CHF)

Onset of nucleate boiling (ONB)

q''_{Leid}
Leidenfrost Point

q''

$T_w - T_{sat}$

Figure 3-3: A typical boiling curve (courtesy ref [35])

Nucleate boiling is the area where heat transfer is higher from wall to the coolant. In a PWR, it is the operating region, as it offers the maximum heat transfer by remaining at the subcooled region. At the departure of nucleate boiling, the heat transfer drops significantly that leads to superheating and thus

the structure of fuel rod can get compromised. Difference between TH codes occur depending on the range of flow regimes, fluid type, and heat transfer between different subchannel fluids. TH codes that consider energy transfer in a lateral direction (crossflow) are called sub channel codes and the one that consider mass and energy transfer only in axial direction are called 1D codes. 1D and subchannel code are coupled as TH modules in coupled framework MPCORE.

3.2.1   1D TH code

CTH1D is the homogeneous single phase 1D heat transfer code coupled with MPCORE. It can model transient and steady-state simulation of coolant flow in a fuel rod channel. Pressure is assumed to be constant in the core, so only one-dimensional mass and energy conservation equations are solved in this code. Steam-water properties provided by FREESTEAM, an open-source implementation of international-standard IAPWS-IF97, are used. In addition, it can simulate coolant boiling effect based on correlations of CHF and heat transfer coefficients adopted from FRAPTRAN.

3.2.2   Subchannel TH code

Coolant-Boiling in Rod Arrays - Two Fluids (CTF) is the sub-channel code developed at Pennsylvania State University and North Carolina State University primarily for square geometries. It uses a two-phase three-field modeling approach [36]. Due to the pressure difference with the axial height and between subchannels, CTF solves the energy, mass, and momentum conservation equations. These equations ensure that mass, momentum, and energy are conserved for a specific computational unit. The two-fluid model is centered around this behavior, which is achieved by utilizing three conservation equations for each field, except for the liquid and droplet fields which share an energy equation. This approach involves modeling each phase with its own set of mass, momentum, and energy equations. The conservation equations are interdependent and connected by intera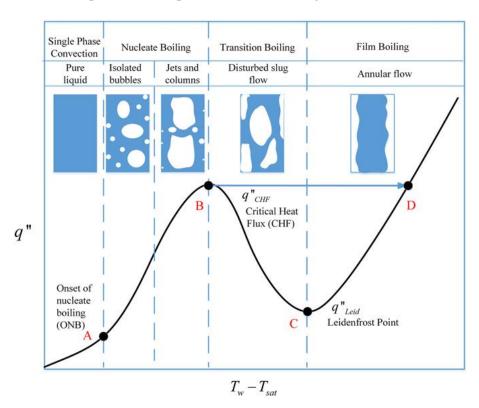ction terms that consider phenomena such as mass and heat transfer between phases, such as evaporation/condensation or entrainment/de-entrainment. The general mass conservation equation is presented as an example.

$$\frac{\partial}{\partial t}(\alpha_k \rho_k) + \nabla \cdot (\alpha_k \rho_k \vec{V}_k) = L_k + M_e^T, \tag{3.8}$$

The k subscript denotes the field under consideration; it can be l for the liquid film field, v for the vapor field, or e for the entrained droplet field. On the left-hand side (LHS) of Equation 2.1, the first term is the change of mass with time and the second term is the advection of the field mass into or out of the volume ($\vec{V}$ is the field velocity). On the right-hand side (RHS), the L_k term represents the mass transfer into or out of phase k — inter-phase mass transfer can occur by either evaporation/condensation or by entrainment/ de-entrainment.

The momentum conservation equation is given below,

$$\frac{\partial}{\partial t}(\alpha_k \rho_k \vec{V}_k) + \frac{\partial}{\partial x}(\alpha_k \rho_k u_k \vec{V}_k) + \frac{\partial}{\partial y}(\alpha_k \rho_k v_k \vec{V}_k) + \frac{\partial}{\partial z}(\alpha_k \rho_k w_k \vec{V}_k)$$

$$= \alpha_k \rho_k \vec{g} - \alpha_k \nabla P + \nabla \cdot \left[\alpha_k \left(\tau_k^{ij} + T_k^{ij} \right) \right] + \vec{M}_k^L + \vec{M}_k^d + \vec{M}_k^T, \tag{3.9}$$

On the LHS, the terms are change of volume momentum with time and advection of momentum. Note that the LHS terms multiply by the vector velocity $\vec{V}_k$ and so each term will have three components ($u_k \vec{i} + v_k \vec{j} + w_k \vec{k}$) and result in three separate momentum equations for each of the three directions (if using Cartesian coordinates). On the RHS, the terms are gravitational force, pressure force, viscous and turbulent shear stress, momentum source/sink due to phase change and entrainment, interfacial drag forces, and momentum transfer due to turbulent mixing. Pressure has no phase dependency because it is assumed to be equal in all phases and gravity is assumed to be the only body force. The $\vec{M}_k^L$ term is the momentum source/sink due to phase change and entrainment/de-entrainment.

The generalized energy equation is presented in Equation,

$$\frac{\partial}{\partial t}(\alpha_k \rho_k h_k) + \nabla \cdot (\alpha_k \rho_k h_k \vec{V}_k) = -\nabla \cdot \left[\alpha_k \left(\vec{Q}_k + \vec{q}_k^T \right) \right] +$$

$$\Gamma_k h_k^i + q_{wk}''' + \alpha_k \frac{\partial P}{\partial t}, \tag{3.10}$$

The LHS terms are change of phase energy with respect to time and advection of phase energy into or out of the cell. The RHS terms are k-phase conduction and turbulence heat flux, energy transfer due to phase change, volumetric wall heat transfer, and the pressure work term. Two different kinds of meshing is used for the solution of these conservation equations. Scalar mesh is used to find the scalar fluid properties while momentum mesh is used for quantities related to the velocity field of fluid.

The sub-channel approach is a simplification of the conservation equations that only considers two flow directions — axial flow and lateral flow. The lateral flow directions are not a set of fixed coordinates; instead, the term, "lateral flow" covers any orthogonal direction to the vertical axis. Because fixed coordinates are not defined for the lateral direction in the sub-channel approach, lateral flow has no direction once it leaves a gap. Lateral flow enters a sub-channel volume through "gaps" between the volume and other adjacent sub-channel volumes. The conservation equations include the terms of interfacial heat transfer, interfacial drag, and wall drag terms. Interfacial drag and wall drag are calculated for the momentum cells as they are required for momentum conservation equation while interfacial heat transfer, required for scalar mesh cells, is calculated by linear averaging of momentum cells. The flow regime is calculated based on the surface temperature related to CHF temperature and

the void coefficient of the coolant. The simplified flow diagram for selecting the flow regime is given in Figure 3-4.
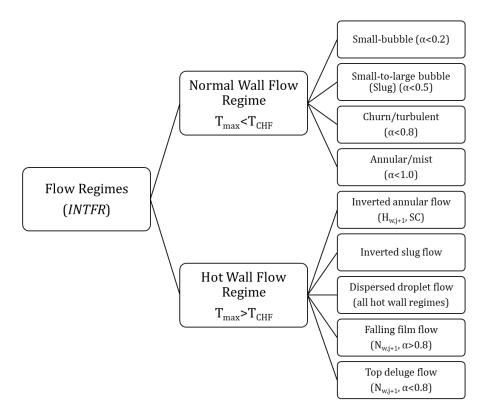
Figure 3-4: Flow Regime selection in CTF.

The flow regime maps as given in CTF manual is shown in Figure 3-5.
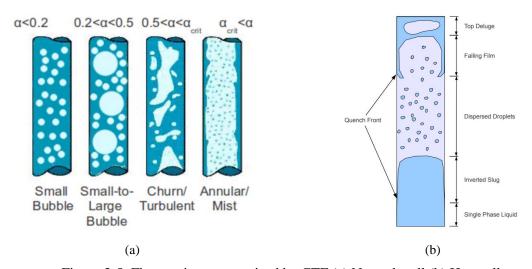
Figure 3-5: Flow regimes recognized by CTF (a) Normal wall (b) Hot wall.

In the conservation equation, other closure terms related to wall shear stress, inter-cell shear, wall heat transfer, turbulent mixing and void drift need to be calculated as well. Wall shear models include the friction loss of the surface, with liquid and vapor, and form loss of all the three phases. Apart from the flow regime, heat transfer regime is also calculated by CTF. In the heat transfer regime, the interface

between rod surface and fluid is important that governs the heat transfer coefficient value. Different models are used for each heat transfer regime. Heat transfer regimes recognized by CTF are given as:

1. Single-phase liquid convection (SPL)
2. Single-phase vapor convection (SPV)
3. Subcooled nucleate boiling (SCNB)
4. Saturated nucleate boiling (NB)
5. Transition boiling (TRAN)
6. Inverted annular film boiling (IAFB)
7. Dispersed droplet film boiling (IADF)
8. Dispersed droplet deposition heat transfer (DFFB)

Selection of the heat transfer regime depends on the heated surface temperature and void fraction. The simplified flow diagram for the selection of heat transfer regime is shown in Figure 3-6. The heat transfer to liquid and vapor is calculated and passed to the liquid and vapor conservation equations.
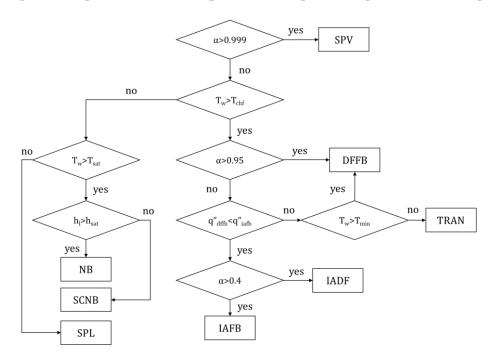
Figure 3-6: Heat transfer regime selection in CTF.

The normal operating heat transfer regime is that of SCNB and NB where the heat transfer is by forced convection to liquid, vapor generation at wall and condensation of bubbles at the wall (condensation is true for SCNB only). The heat transfer coefficient calculation can be done by Chen correlation or Thom correlation (Chen correlation is used in this research). Heat transfer coefficient using Chen correlation is given as,

$$h_{chen} = h_{fc} + h_{nb}, \tag{3.11}$$

The forced convection part is calculated as,

$$h_{fc} = 0.023 F_{chen} \left(\frac{k_l}{D_h} \right) \text{Re}_l^{0.8} \text{Pr}^{0.4},$$ (3.12)

F_{chen} factor accounts for the difference in two-phase and single-phase Reynold number. The nucleate boiling component of HTC, modified from Forster-Zuber type of pool boiling equation, is given by,

$$h_{nb} = 0.00122 S_{chen} \left(\frac{k_f^{0.79} C p_f^{0.45} \rho_f^{0.49} g_c^{0.25}}{\sigma^{0.5} \mu_f^{0.29} H_{fg}^{0.24} \rho_g^{0.24}} \right) \left(T_w - T_f \right)^{0.24} \left(P(T_w) - P(T_{sat}) \right)^{0.75},$$ (3.13)

Where boiling suppression factor, S_{chen}, is a function of two-phase Reynold number. Thom correlation formulated a new correlation like Dittus-Boelter correlation for forced convection part, but in CTF, the un-modified Dittus-Boelter is used for forced convection part.

Heat conduction through fuel, gap and clad can be performed using CTF. CTF considers same fuel dimensions in a single assembly. Due to the use of independent FP code in this study, conduction option has been turned off in CTF and external surface temperature is provided by FP code. DNB calculation in CTF is an important feature that dictates the flow regime and heat transfer regime in the nuclear core. Different models are available to perform DNB and CHF calculation in CTF that includes Biasi model (standard), Bowring correlation, W-3 correlation and Groeneveld lookup tables. W-3 correlation (used in this research) developed by Tong is the most widely used correlation for evaluating DNB in PWRs. CHF is calculated by,

$$\frac{q_{w3}''}{10^6} = \left\{ (2.022 - 0.0004302 p) + (0.1722 - 0.0000984 p) \exp\left[(18.177 - 0.004129 p) x_e \right] \right\}$$
$$\left[(0.1484 - 1.596 x_e + 0.1729 x_e \mid x_e \mid) \frac{G}{10^6} + 1.037 \right]$$ (3.14)
$$(1.157 - 0.869 x_e) \left[0.2664 + 0.8357 \exp(-3.151 D_h) \right] \left[0.8258 + 0.000794 (h_f - h_{in}) \right],$$

Where,

q''_{w3} = critical heat flux, BTU/hr.ft^2

p = pressure, psi$_a$

x_e = local steam thermodynamic quality

D_h = equivalent hydraulic diameter, in

h_{in} = inlet enthalpy, BTU/lbm

G = mass flux, lbm/hr.ft^2

The W-3 correlation is applicable in the following range of operating conditions,

p = 800 to 2300 psi$_a$

$G/10^6$ = 1.0 to 5.0 lbm/hr.ft^2

D_h = 0.2 to 0.7 in

x_e = -0.15 to 0.15

L = 10 to 144 in

This correlation is valid for uniform axial heat flux profiles. In nuclear core, it is common to have non-uniform axial heat flux. For non-uniform axial heat profile, Tong F-factor is applied to correct the CHF.

$$q^{''}_{chf,nu} = \frac{q^{''}_{w3,u}}{F},$$
(3.15)

CTF has its own coupling interface that enables efficient data transfer between CTF and other coupled modules [36]. For commonly required variables, subroutines have been developed, and the values are recorded in common memory variables as defined in the "transfer_io" module developed by the CTF development team. Custom subroutines can be implemented in the CTF coupling interface if certain other variable values are required. A fraction of heat directly released to the coolant is modeled by CTF by supplying that heat to the liquid phase.

CTF can model full core with the quarter symmetry option. The number of modeled fuel assemblies decreases with the quarter symmetry option. The "Get_mirrored_rod" subroutine can be used to obtain the index of the modeled rod that replicates the unmodeled rod in assemblies that exist on the symmetry line. For parallel implementation, CTF uses the same number of processors as the number of assemblies[37, 38]. COBRA-TF includes a wide range of T/H models crucial to accurate LWR safety analysis including, but not limited to, flow regime dependent two-phase heat transfer, inter-phase heat transfer and drag, droplet breakup, and quench-front tracking.

3.3 Fuel performance

FRAPI fuel performance module is coupled with MPCORE. FRAPI is an interface that provides restart capability for FRAPCON and uses FRAPCON and FRAPTRAN for depletion and transient cases respectively [39].

3.3.1 FRAPCON

FRAPCON is a software program that uses fuel and cladding deformation subroutines to model fuel rod temperatures. Gas release, void volumes, and gas pressure are calculated for each steady state. The program applies internal convergence criteria to maintain a 1% relative error for temperature and

pressure. FRAPCON makes certain assumptions, such as predominantly radial heat transfer and negligible heat conduction in the axial direction due to a large length to diameter ratio. At each time step, constant boundary conditions for power and coolant temperature are maintained. FRAPCON is considered a 1.5D code because it solves a simple balance equation in the axial direction and heat conduction in the radial direction. Coolant temperature and linear power serve as the boundary conditions for calculating fuel/cladding temperatures and gap conductance. FRAPCON also computes oxide thickness with burnup, which depends on coolant temperature and burnup. The oxide layer causes a temperature drop between the cladding and coolant temperature, and as the oxide layer thickness increases, the temperature difference between the oxide layer and cladding outer layer increases. The minimum of two formulas given below determines the cladding surface temperature at any axial height z.

$$T_w(z) = T_b(z) + \Delta T_f(z) + \Delta T_{cr}(z) + \Delta T_{ox}(z) \tag{3.16}$$

$$T_w(z) = T_{sat} + \Delta T_{JL} + \Delta T_{ox}(z) \tag{3.17}$$

Where T_b is the bulk coolant temperature at height z, T_w is the rod surface temperature, ΔT_f is the forced convection film temperature drop, ΔT_{cr} is the crud temperature, ΔT_{ox} is the oxide surface temperature and T_{sat} is the coolant saturation temperature.

Temperature correlations for the fuel depends on the fuel burnup. Fuel burnup is calculated by FRAPCON by using the volumetric power provided by the neutronics code. Burnup calculation is performed by FRAPCON for each axial mesh based on the power provided by neutronics code.

$$dbu = \frac{q'''dt}{\rho_{fuel}} = \frac{\alpha}{\rho_{fuel}} \sum_k \sigma_{f,k} \bar{N}_k \phi dt \tag{3.18}$$

Where q''' is the volumetric heat generation rate, ρ_{fuel} is the fuel density, σ_f is the fission cross section and α is a conversion constant. Higher isotope production is performed based on Bateman equations.

Oxide thermal barrier model is also present in FRAPCON, so the oxide layer grows with time. Temperature drop across the zirconium oxide layer at elevation z is determined by:

$$\Delta T_{ox}(z) = \frac{q''(z)\delta_{ox}(z)}{k_{ox}}. \tag{3.19}$$

Where,

$\Delta T_{ox}(z)$ = oxide temperature drop at elevation z on the rod axis (K)

$\delta_{ox}(z)$ = oxide thickness at elevation z on the rod axis (m)

k_{ox} = oxide thermal conductivity (W/m−K)

Initially, the clad outer surface temperature and oxide surface temperature are same as there is no oxide layer. Due to the burnup and coolant flow, oxide layer starts growing so the temperature difference between clad outer and oxide layer widens. During nuclear fuel cycle, fuel expansion, cladding creep, hoop stress and fuel cracks might appear due to power transients. These models are developed in FRAPCON and used in industrial applications for fuel deformity and pellet clad mechanical interaction analysis. IFBA rods used to control power peaking at beginning of cycle can be modeled with FRAPCON. Helium production due to IFBA coatings of ZrB_2 increases the rod internal pressure. Empirical correlation used for helium production from IFBA liners is:

$$He_{prod} = -(A_1 IFBA + A_2)(B10)^2 + (B_1 IFBA + B_2)(B10) \qquad (3.20)$$

Where,

He_{prod} = helium production (atoms He/cm³-s)

IFBA = percent of fuel rods in a core containing IFBA liners

B10 = boron-10 enrichment (percent)

$A_1 = 6.23309 \times 10^{-9}$

$A_2 = 7.02006 \times 10^{-7}$

$B_1 = -1.35675 \times 10^{-7}$

$B_2 = 3.1506 \times 10^{-4}$

The amount of hydrogen that results from the interaction between metal and water and is absorbed by the cladding is called the pickup fraction. For zircaloy and PWR, a constant pickup fraction of 0.15 is used. In the FRAPCON hydrogen pickup models, information provided to the transient fuel rod analysis code includes permanent burnup effects like cladding creep down, fuel swelling, fuel densification, normalized radial power, burnup profiles, and fission gas inventory in the fuel matrix and the fuel rod void volume. An option available to developers that can significantly affect results is the ability to model gamma-ray heating, which deposits a portion of the energy directly into the coolant instead of all the energy going into the fuel. [40]. The code calculates all significant fuel rod variables with time including fuel and cladding temperatures, cladding hoop strain, cladding oxidation, hydriding, fuel irradiation swelling, fuel densification, fission gas release, and rod internal gas pressure. The default

mechanical model of FRACAS-I (finite difference model) and fission gas release model Massih is used in the analysis. The code is not validated above the melting temperatures of fuel and cladding, so the code stops once temperature reaches this value. The simplified model of FRAPCON is given in Figure 3-7.
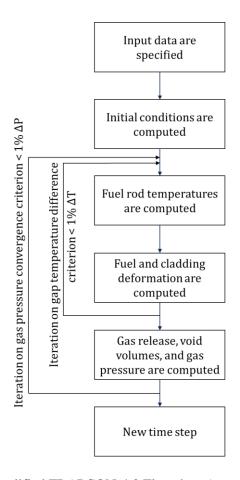
Figure 3-7: Simplified FRAPCON-4.0 Flowchart (courtesy ref [40])

3.3.2 FRAPTRAN

FRAPTRAN is a software that calculates the history of fuel rod temperature and deformation over time based on changes in fuel rod power and coolant conditions. Parameters related to burnup can be initialized from the FRAPCON single rod fuel performance code. FRAPTRAN uses a heat conduction model that uses a variable mesh spacing to handle power peaking at the pellet edge in high burnup fuel. FRAPTRAN analyzes the behavior of LWR fuel rods when there are rapid changes in power or coolant conditions, while FRAPCON calculates behavior over time when changes in power and coolant conditions are slow enough for the term "steady state" to apply.

The primary purpose of the cladding is to keep the fuel column and radioactive fission products contained. If the cladding doesn't rupture or melt during a reactor transient, the fission products remain inside the fuel rod. However, if the cladding is weakened by temperature increase, oxidized, or overstressed by mechanical interaction with the fuel, it can crack or rupture, releasing the radioactive

products into the coolant. The rupture or melting of one fuel rod's cladding can also alter the coolant flow and reduce cooling for nearby fuel rods.

FRAPTRAN assumes that the variables at any given axial node are independent of the variables at all other axial nodes, resulting in a one-dimensional, stacked solution. Although the code models critical heat flux (CHF) and post-CHF correlations, it's not intended to replace sub-channel codes like VIPRE or COBRA that provide more accurate modeling for departure from nucleate boiling (DNB) or post-DNB.

FRAPTRAN has several assumptions and limitations, which are listed in Table 1.1 of FRAPTRAN manual [41] and taken as input from FRAPCON as a restart file. The coolant conditions should be calculated by a thermal-hydraulic code and then input to FRAPTRAN, and the heat generation must be determined by a reactor physics analysis and input to FRAPTRAN. After calculating the temperature change across the oxide layer, the temperature change is added to the fuel rod surface temperature calculated from the coolant conditions. The flow scheme for the calculations in FRAPTRAN is shown in Figure 3-8.
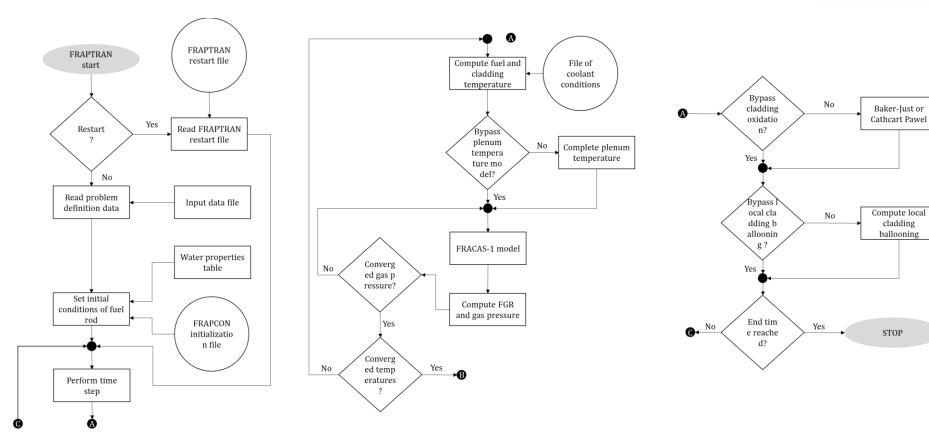
Figure 3-8: Flowchart of FRAPTRAN (courtesy ref [41])

The computer running time of FRAPI is directly proportional to the number of axial nodes but not as sensitive to the number of radial nodes. Doubling the number of axial nodes doubles the running time, while doubling the number of radial nodes increases running time by approximately 15 percent. For a full-length fuel rod, about 10 axial nodes and 15 radial nodes in the fuel are recommended, with unequal spacing permitted for both axial and radial nodes. The first radial node is located at the center of the fuel rod, and other radial nodes are placed at the fuel pellet surface and at the cladding's inside and outside surfaces. Additionally, an arbitrary number of radial nodes can be placed within the fuel and cladding.

Gap size deformation with burnup is shown in Figure 3-9 and the effect of each separate effect. These are modeled in fuel performance codes and formed the basis for the calculation of dynamic gap conductance in FP codes.
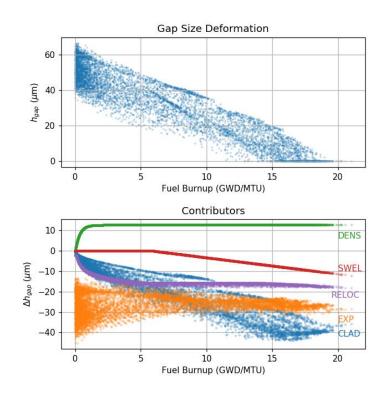
Figure 3-9: Effect of different physical phenomenon on gap size

3.4 Coupling parameters between modules

Different Multiphysics frameworks have been discussed in the last section. Among the earliest coupling approaches discussed in section 2.8, neutronics and thermal hydraulics codes are coupled together. This coupling approach is also used in the industrial approach as a first step followed by independent FP analysis in a second step [42, 43]. FP analysis is performed based on the power history and coolant temperature generated in the first step. As a first step, NK and TH coupling is performed in this study where pin power and exposure/burnup is provided by RAST-K to CTF and CTF supplies fuel doppler and moderator temperature to RAST-K as shown in Figure 3-10.
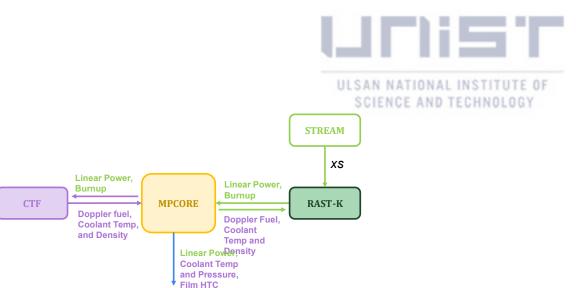
Figure 3-10: NK and TH code coupling

FP module is added with NK-TH code system like traditional approach, where FP code is not used in NK-TH Multiphysics calculation. After the successful coupling of TH module with NK module, the next step is to couple FP code as well, so that doppler temperature is given by FP code to NK code. In some of the previous coupling approaches discussed in Chapter 2 which use neutronics, TH, and FP, one-way coupling was performed between FP and TH modules. The clad surface temperature was used as a boundary condition for FP code in NECP-X, CTF, and NECP-CALF coupling [10]. Coolant temperature and pressure have been used as boundary condition for FP in MCS, CTF, and FRAPCON coupled analysis [9]. However, no data was transferred from NECP-CALF/FRAPCON to CTF, and it was a one-way coupling between FP and TH codes. In one-way coupling, coolant temperature and pressure are provided by TH code to FP code while no information is transferred from FP code to TH code. Thus, coolant temperature in CTF is calculated using its own fuel heat conduction solver. In two-way coupling, FP code provides clad outer surface temperature to TH code and no fuel heat conduction solver is used in TH code. CTF provides an option of applying no fuel heat conduction solver and uses the clad surface temperature provided by an external code. The cladding temperatures predicted by FP and the conduction option of CTF differ because the default thermal conductivity model in CTF overpredicts the fuel conductivity at higher temperatures [44]. FP codes are usually coupled with neutronics-TH as it gives burnup dependent fuel and gap conductance due to fuel swelling, densification, cracking, and cladding creep models implemented in FP code. Thereby, no actual advantage of FP coupling is achieved in one-way coupling. Two-way coupling is used, for FP and TH codes as well, in MPCORE to avoid using fuel heat conduction twice.

Core neutronics depends on the rod FP and coolant TH. FP, in turn, depends on both core neutronics and TH. Similarly, TH uses the boundary conditions calculated by core neutronics and FP codes. Presently, fuel heat conduction is performed only by FP code. This is because this code is specialized for performing swelling, densification, cracking, and dynamic gap heat conductance. Moreover, the conduction option is turned off in CTF, and the clad outer surface temperature is obtained from the FP

module. In this manner, TH calculates the coolant temperature, pressure, density, film heat transfer coefficient (fHTC), and DNBR using the linear power provided by neutronics code (for gamma heating) and clad outer surface temperature provided by FP code. Among the TH module's outputs, the neutronics module employs the coolant temperature to select the specific temperature-dependent cross-sections. In addition, fHTC, coolant temperature, and pressure from the TH module's output function as boundary conditions in the FP module for calculating the temperature profile in the fuel and cladding.

MPCORE structure facilitates the coupling of different physics modules independent of each other. The present coupled modules and the variable exchange between them are shown in Figure 3-11.
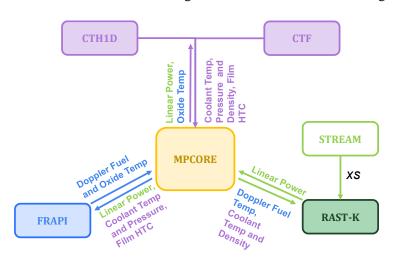

Figure 3-11: MPCORE Framework

In principle, to increase reliability, all the physics codes in a multi-physics framework should have a two-way coupling. Thereby, these interact to yield a more realistic outcome. This coupling methodology is generally recommended by the expert group meeting on multiphysics experimental data, benchmarks, and validation [1].

Output parameters that are used in the coupling or for the result analysis are shown in Table 1 in conjunction with the module that calculates these and their dimensions. Here, n_p is the total number of fuel pins, and n_a is the total number of axial meshes considered in the TH and FP modules. The flow scheme of different variable exchange and module calls in MPCORE is shown in Figure 3-12.

Table 1: Output parameters and the responsible modules

Parameter	Module	Dimensions
Transient reactivity, $	RAST	1
Linear power, W/cm	RAST	n_p x n_a
Pellet centerline temperature, K	FRAPTRAN	n_p x n_a
Pellet doppler temperature, K	FRAPTRAN	n_p x n_a

Pellet surface temperature, K	FRAPTRAN	n_p x n_a
Clad outer surface temperature, K	FRAPTRAN	n_p x n_a
Fuel enthalpy, Cal/g	FRAPTRAN	n_p x n_a
Gap gas heat transfer coefficient, W/m^2K	FRAPTRAN	n_p x n_a
Film heat transfer coefficient, W/m^2K	CTF	n_p x n_a
Coolant density, kg/m^3	CTF	n_p x n_a
Coolant temperature, K	CTF	n_p x n_a
Departure from nucleate boiling ratio	CTF	n_p x n_a

Start

Initialize and setup RK/CTF/FRAPI

Solve for neutronics by RK

Decrease Δt by time step processing algorithm, calls time step rejection algorithm

Transfer q_l, $fHTC$, T_m and ρ_m to FRAPI

Solve for fuel performance by FRAPI

Transfer assembly averaged T_d, T_m and ρ_m to RK

Transfer q_l and T_{co} to CTF

Update t and Δt

Solve for thermal hydraulics by CTF

Call advance_fs or advance_ds

No

No

Ye(s)   Max. Iterations reached?   No   bc, q_l, T_{fc}, T_{fs}, T_d, T_{ci}, T_{co}, T_m, ρ_m, fHTC converged?   Yes   Last timestep

Yes

Convert **txt** output files to **h5** format

Finish

bc   :   boron concentration
q_l :   linear power
T_{fc}, T_{fs}, T_d, T_{ci}, T_{co}, T_m : (fuel centerline, fuel surface, fuel doppler, cladding inner, cladding outer, moderator) temperature
ρ_m :   moderator density
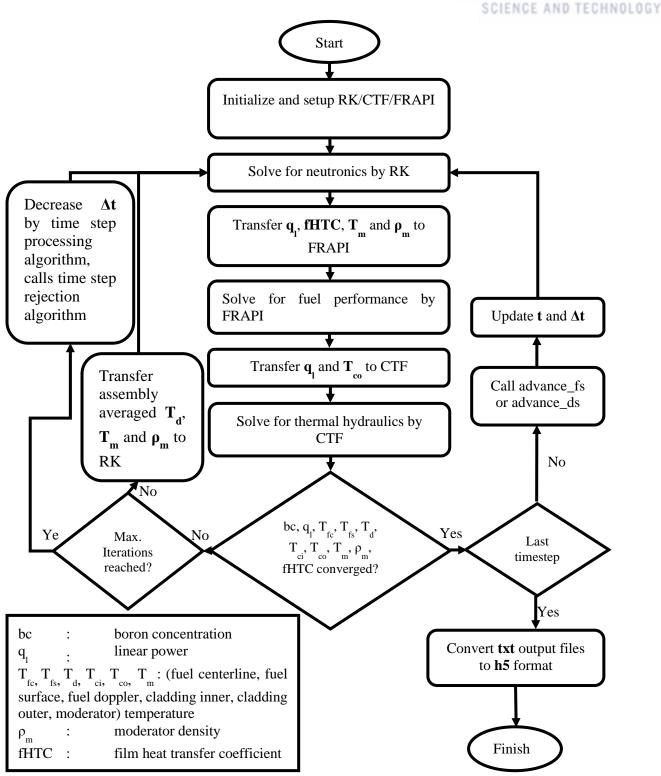fHTC   :   film heat transfer coefficient

Figure 3-12: Flowchart for coupled transient.

The meshing in neutronics code can differ from that in TH and FP codes. Mesh treatment given by MPCORE for different modules is given in section 4.5.

3.5   Parameters for comparison

In cycle depletion cases, boron concentration, fuel temperatures, cladding and oxide layer temperatures, coolant temperatures, DNBR, film HTC, gap HTC, maximum linear power, Cladding hydrogen concentration are compared with two different coupling approaches between fuel performance and thermal hydraulics modules. In transient cases as well, temperatures, DNBR and film HTC are compared. In addition to that, fuel enthalpy is also considered while boron concentration stays same during transient, so it is not compared. Transient cases used two way coupling approaches between all physics modules and the results are compared for dynamic gap conductance and coolant crossflow models [45-47].

Hydride precipitation can be calculated by FRAPCON, and it is one of the limiting factors for calculating the fuel rod life in nuclear reactor. Higher hydride concentration increases the cladding ductility, and the failure probability increases for cladding. Oxide thickness and hydrogen content are dependent on the burnup, and they get affected by the coolant temperature as well.

Independent fuel performance calculations with BISON were analyzed for local cladding failure probability [48]. Pellet-clad mechanical interaction and stress corrosion cracking results in fission products release to the primary coolant. This approach, considered to be traditional FP analysis method simulated the 3 cycles of Watts Bar Unit 1 using data generated from MPACT/CTF simulations. Maximum centerline fuel temperature, maximum clad hoop stress and minimum gap size were analyzed in this research. Axial maximum fuel centerline temperature and hoop stress, while axial minimum gap width for all three cycles, and all three states (BOC, MOC and EOC), using BISON independent of MPACT/CTF is shown. IFBA modeling capability has been enabled in BISON before this analysis.

4 Multiphysics coupling methodology

Developed coupling framework, MPCORE, was developed with the aim of coupling different UNIST CORE lab NK codes with TH and FP codes. CORE lab has developed Monte Carlo code MCS, Lattice physics-based transport code STREAM and nodal diffusion code RAST-K. RAST-K is explained in the previous chapter as only RAST-K neutronics code is coupled with MPCORE by now. In future work, STREAM, MCS and RAST-K pin-by-pin code can be coupled with MPCORE.

Different options available for Multiphysics coupling algorithm and structure of developed Multiphysics framework MPCORE is discussed in this chapter. In addition to that, different features developed in MPCORE for convergence, adaptive timestep, data exchange, mesh treatment, parallel implementation, shuffling/rotation, and quarter symmetry are also explained in this chapter.

4.1 Modeling/coupling choices

The basic coupling elements in a Multiphysics framework are shown in Figure 4-1. Among the many options, the ones that are used in MPCORE coupling framework are underlined. These options are:

1. Mesh Overlays: the meshes inside the modules are independent of the mesh control of MPCORE. The constituent modules can have different number of meshes according to their requirements and MPCORE will linearly interpolate the data points while mapping from one code domain to another.

2. Linking Interface: Codes are linked externally with each other. Modules don't have any information about where the data is provided to or provided from the code. MPCORE has established a connection with all the modules through their respective interfaces. Initialization, allocation, data transfer, deallocation is all performed on the requests from MPCORE. This requires little to no modification of constituent modules, modules are interchangeable, facilitates parallel and distributive processing of modules.

3. Fixed-point Iteration: In tight coupling approach, Newton iterations include the coupling effect between all the physics while in loose coupling each module is converged separately and then fixed-point iterations are performed until the system is converged. Tight coupling approach and loose coupling approach are compared by previous researchers [49]. In this coupling framework, as the modules are independent of each other, it was more meaningful to use loose coupling between them. It might have slow convergence as each module is converged for each fixed-point iteration. Performing partial convergence of modules and iterating between codes might reduce the computation time.

4. Time-Step Algorithms: time-step is controlled by MPCORE externally using adaptive timestep control while each coupled module has the capability to decrease the timesteps internally for convergence and supply the data to MPCORE at the requested timestep. Numerical integration of modules using Implicit Euler step-doubling method is implemented [50].

5. Modules inside MPCORE are executed in parallel wherever it is supported. For the codes that solve single fuel rod at a time, are made parallel by solving fuel rods in parallel using MPI. Thermal hydraulic code already has the capability to run in parallel using PetSc and MPI architecture.
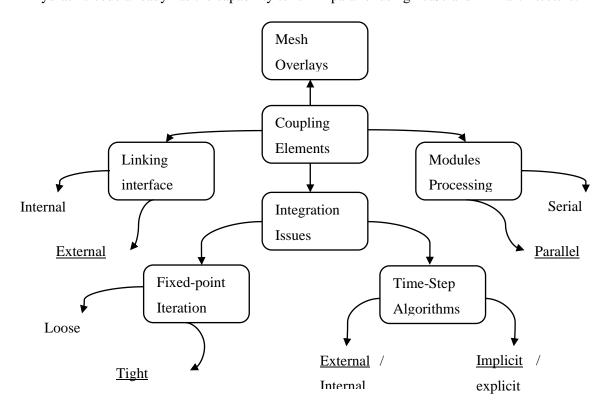

Figure 4-1: Coupling elements in a Multiphysics framework.

External tight coupling approach is used for coupling the modules, that require little or no change to the modules. The study was performed by performing two way coupling between all the physics modules [1, 11, 51]. However, some of the previous researchers used one way coupling between TH and FP modules [9, 10]. The difference between the two coupling approaches has been analyzed for whole core multicycle depletion case.

Multiphysics coupling framework is developed for sensitivity analysis and data generation for the deep learning module [51]. It has been emphasized that during external coupling of modules, all the exchanged variables should converge. In previous research, fuel temperature, flux or local power distribution was used for convergence but in this research axial power is estimated to be slowest in convergence. FUSE (Fuel performance, neUtronics and thermal hydraulicS Engine) was developed to ensure the two-way coupling between all the substituent modules. The variable exchange between neutronics-TH, neutronics-FP and TH-FP are explained in the research. Different neutronic, thermal hydraulics and fuel performance codes are used, and N-TH or N-TH-FP coupling is used based on the analysis requirement.

Burnup dependent models are implemented in CTF and benchmarked against the FRAPCON-3.4 predictions [44]. Validating CTF burnup dependent model results with FRAPCON itself verifies that FRAPCON burnup dependent conduction models are accurate. The results are within 5% of error but as the temperature and burnup increase, the difference between both the models increases. The primary focus of the comparison was fuel centerline temperature predictions by both the codes. Only solid rods are used for comparison as the hollow fuel rod modeling is not available in CTF. IFRAP=1 option is added in CTF to use the modified NFI fuel thermal conductivity model.

But they still use the same pin dimensions in single fuel assembly and cannot model the hollow fuel pins. Effect of burnup on fuel thermal conductivity [52].

4.2   Structure of MPCORE

Basic structure of MPCORE is developed in C++. Picard convergence algorithm, error control, adaptive timestep algorithm and subsequent calls to sub modules are through the driver program of C++. Overall, MPCORE consists of the modules A, B and C connected with the multi-physics modules of neutron kinetics NK, thermal-hydraulics TH, and fuel performance FP.
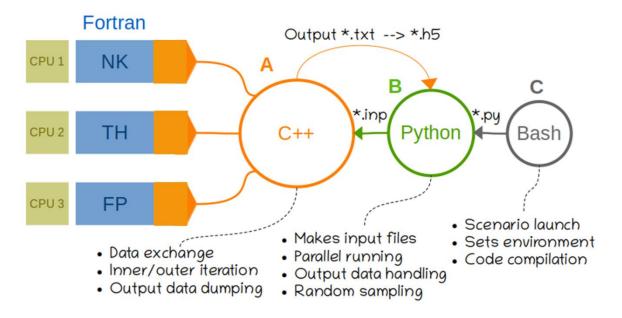

Figure 4-2: Structure of MPCORE

4.2.1   Module A

Module A written on C++ contains the multi-physics coupling algorithm based on the adaptive time step integration scheme and fixed-point inner iterations improved by a damping Picard convergence or the Anderson's acceleration approaches. On each cycle module A requests, the multi-physics modules to update their feedback vectors; then the feedback vectors are collected and transferred into the common memory space of module A. Associative Map container is created with each variable name associated with a sequential map array object. Time step size and feedback to different modules are

computed using the iterator to traverse all the values of parameters created. Transmitter class is created to send and receive values by each sub module. The new feedback vectors are re-sent to the multi-physics modules and iteration repeats. After the inner iteration convergence is achieved, module A yields the full list of multi-physics variables, including the iterated feedback vectors and the internal variables of NK, TH and FP. The output data is dumped into a temporary storage to be converted into HDF5 format. The general functions created for all the coupled modules are:

1. Make: It reads the input files for specific module. Allocation and assignment of all the variables specific to the modules are created in this subroutine.
2. Initialize: Initialization of steady state in each of the modules. Multiple calls to this subroutine occur till the steady-state convergence of all the modules.
3. Move: movement of modules at each time step h. modules are converged after each timestep h based on the feedback from other modules as well.
4. Yield: transfer of data from modules to MPCORE at each time step h. data transfer can occur for timestep h/2 as well if dynamic timestep is chosen at the input stage.
5. Save: current modules state is saved by MPCORE for reusing after the data update.
6. Load: loading of the previous module state if the current convergence state fails and timestep must be reduced for convergence.
7. Clean: deallocation of the dynamic variables created at the make stage. Memory is freed by the program to avoid cache overhead for next execution.

The execution sequence of different modules by MPCORE with the above defined functions is shown in Figure 4-3.

```cpp
// Initialization

a.make ();  // Initialization of the
b.make ();  // modules 'a' and 'b'

a.yield (x); // Put the initial guess
b.yield (x); // to common space 'x'

// Initial (steady) state calculation

a.save (); // Memorize the models state
b.save (); // (optional)

do:
    a.load (); // Load the state
    b.load (); // (optional)

    a.init (); // Calculate the initial
    b.init (); // states

    a.yield (x); // Update the shared
    b.yield (x); // variables in 'x'
```

```cpp
// Outer iteration loop (time steps)

time = 0; // set the current time to zero

do:
    a.save (); // remember the current
    b.save (); // states at time

    // Inner iteration loop (Picard)

    do:
        a.load (); // load the previous
        b.load (); // states of 'a' and 'b'

        a.move (h,time,x); // advance on
        b.move (h,time,x); // time step 'h'

        a.yield (x); // Update 'x'
        b.yield (x);

    time += h; // increment the time

// Finalization

a.clean (); // destroy all the models
b.clean ();
```

Figure 4-3: MPCORE C++ framework calling sequence.

4.2.2   Module B

Python module B is designed for random sampling and Latin Hypercube sampling of input data, processing and plotting of output data, and many query parallel run. Input file for neutronics modules is created with python module. Another input file for general options like selection of axial and radial dimensions, tolerance values and convergence algorithm are selected based on the input file created by python scripts. Data needed directly to change for each pin in FP code is managed in a separate input file created by python input scripting. FULLCORE class is created in python that creates specific input files for all the constituent modules. This class makes use of several other class/routines to check the geometry and execution conditions. Some of component classes/routines called by FULLCORE are:

1. Geometry: each fuel pin type, enrichment, pin to assembly mapping, core and assembly maps, id number of each fuel assembly and id of each fuel pin in assembly is given in geometry object.

2. Make: making of only RAST-K input file and checking with RAST-K input processing routine to satisfy that all the necessary information is provide to the MPCORE framework. Successful make will convert the geometry file to geom.h5.

3. Run: it will run the MPCORE C++ code by first verifying that geometry making was successful. it also assigns the file paths that should be used for input reading and output processing of .txt files. At the end of successful run from C++ code, "Replace" routine is called that converts .txt files of geometry, data, errors, and information to their respective .h5 format files for file size reduction and easy readability of python output processing routines.

4. Input Data: FULLCORE class has the objects of datatype IVP, RK2, FPI and TH for storing the data regarding initial value problem, RAST-K2, FRAPI and TH modules. Data that is required solely by the individual module must have its own class. IVP class has the values of relative and absolute tolerance, initial timestep size, iteration method and convergence driver. RK2 class has its own axial meshes, Control rod position, loading pattern, shuffling/rotation, depletion, mesh for creating RAST-K input file. FPI class has the data for all the pins regarding axial meshes, IFBA coating, hollow/non-hollow axial mesh, gap thickness etc. TH class has its own boundary conditions of temperature, pressure, mass flow etc.

5. Addons: special output processing routines are created to access data about the hottest, coolest fuel pin location. Data is available for each axial mesh of every fuel pin that can be used as such, or radial and axial integrated values are used for plotting.

6. Parameters: It lists all the Multiphysics parameters that are given in data or geometry output files. The parameter units, responsible module, and output processing functionalities are given in parameters file.

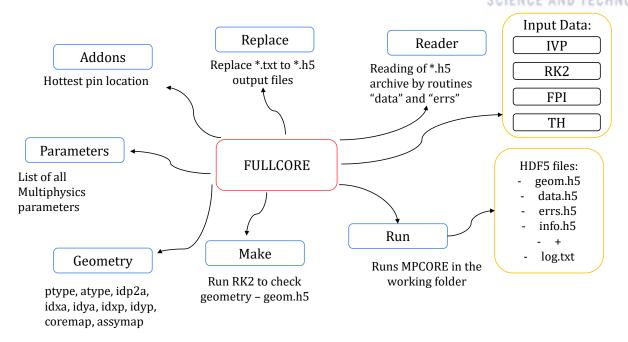The short description of python scripts is given in Figure 4-4.

Figure 4-4: Python Scripts in MPCORE

4.2.3 Module C

Module C is a shell script for multi-physics code compilation and execution. It arranges the framework environment to provide functionalities of code make, launch specific cases and running the output processing files written in python. Code compilation, setting of all the required compilers, path of the MPCORE files is set through the bash script. User friendly bash interface is provided to run different python scripts for specific modules integration, checking input, execution of specific scenarios, and output processing.

4.3 Convergence algorithm

MPCORE framework performs the external loose coupling of multi-physics modules. It allocates variable space internal to all the modules. Modules internal space, denoted by 'x_i' afterwards, are not affected by the MPCORE module and stays same if the modules are not converged for the timestep. Feedback vector space, denoted by 'u_i' afterwards, are kept in MPCORE and keep on updating its values till the convergence is achieved for a timestep. For data exchange between MPCORE and modules, each module maintains allocation 'A' and deallocation 'D' routines to create and destroy a module's internal memory space. Save 'S' and load 'L' routines to memorize and retrieve a module's state. Initial steady state 'I' and time step 'N' routines. Update routine 'U' is used for the feedback vector update.

The adaptive time step algorithm and fixed-point convergence algorithm can be implemented in the vector space u_i without an explicit operation of the corresponding state vector x_i. In MPCORE, damped Picard iteration method improved by Gauss–Seidel acceleration has been implemented. The improved Picard iteration algorithm used in MPCORE is given by,

$$P_X = (L_1 \cdots L_n X_1 U_1 \cdots X_n U_n).$$
(4.1)

Where 'X' denotes the initial steady state 'I' or time step 'N' routine. In traditional Picard iteration scheme, all modules are executed in parallel and the data exchange between them takes place afterwards. Timestep integration with fixed point convergence makes algorithm of MPCORE as given by,

$$Q_{MPCORE} = A_1 \ldots A_n U_1 \ldots U_n S_1 \ldots S_n P_I (S_1 \ldots S_n K_3) D_1 \ldots D_n.$$
(4.2)

Where K_3 is given as,

$$K_3 = P_N H_2 (L_1 N_1 N_1 U_1 \cdots L_n N_n N_n U_n) H_1.$$
(4.3)

H_2 is routine decreasing step h into two times while H_1 increases timestep by two times.

Therefore, the local memory space of the external modules is saved only once per time step iteration and the rejection of the step can be achieved by the loading of external modules from the saved memory space.

4.4   Adaptive time step method

MPCORE processes a time step according to (4.2) and the scheme shown in Figure 4-5. For a given reactor core state vector x and time step size h at the local time t, the Multiphysics framework advances the reactor core state from x to x′ using the step-doubling and Picard iteration methods [50]. The feedback parameters explained in section 3.4 are evaluated for the updated solution x′. The value of LTE e, for every time step is evaluated as follows,

$$e = \max_{k \in \{1,\ldots,m\}} \left\| \frac{\Delta_k}{A_{tol} + R_{tol} u_k} \right\|_2.$$
(4.4)

where m is the number of feedback parameters, Δ_k is the difference between two solutions obtained with the time steps h and h/2. A_{tol} and R_{tol} are the absolute and relative tolerances. For a given LTE update e′, the new time step is calculated according to the (4.5).

$$\text{ratio} = 0.7 * \sqrt{\frac{1}{e'}}.$$
$$h_{new} = \min(1.5, ratio) * h_{old}.$$
(4.5)

The comparison of the local error, e′ to 1 results in two logical branches: regular and rejected time steps.

For the regular branch, e′ ≤ 1, the vector state, local time and step size are updated in order to start the

next time step iteration. For the rejection branch, e′>1, the solution x′ is discarded and the iteration repeats with the smaller step size h′ computed for the local error e′. The rejection branch is used for exceptions thrown by the multi-physics code during the Picard iteration. The exception can be thrown by a substituent module that experiences the internal divergence or failure event. For the exception of any reason the rejected iteration is repeated using the time size reduced in two times. The rejection may be repeated several times, maximum iterations selected, until Picard iteration converges, and a small enough local error is achieved. The time step integration algorithm stops with an error if the number of rejecting cycles exceeds the limit or the minimum selected time step h′.
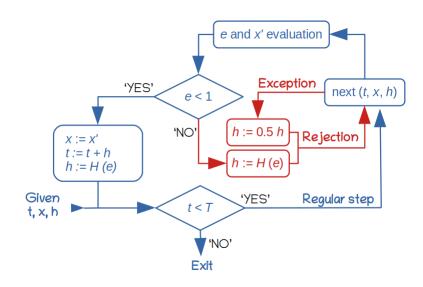
Figure 4-5: Time step processing flowchart.

4.5   Mesh treatment

An interpolating procedure was developed to realize consistent data exchange between neutronics and the other modules. Algorithm 1 shows the procedure for interpolating 'x' mesh values to 'y' mesh values.

```
Algorithm 1: Linear Interpolation procedure
INPUT x_meshsize: ARRAY [1: total meshes in x]
INPUT y_meshsize: ARRAY [1: total meshes in y]
INPUT x_val: ARRAY [1: total meshes in x]
OUTPUT y_val: ARRAY [1: total meshes in y]

y_prev_len ← 0

x_prev_len ← 0

interp_len ← 0
```

```
i ← 1

j ← 1

x_total_length ← sum(x_meshsize)

WHILE abs(interp_len-x_total_length) > 1.D-10

y_curr_len ← y_prev_len + y_meshsize[j]

x_curr_len ← x_prev_len + x_meshsize[i]

IF x_curr_len > y_curr_len THEN

y_val[j] ← y_val[j] + x_val[i] * (y_curr_len – interp_len) / y_meshsize[j]

j ← j+1

y_prev_len ← y_curr_len

interp_len ← y_curr_len

ELSEIF y_curr_len > x_curr_len THEN

y_val[j] ← y_val[j] + x_val[i] * (x_curr_len – interp_len) / y_meshsize[j]

i ← i+1

x_prev_len ← x_curr_len

interp_len ← x_curr_len

ELSE

y_val[j] ← y_val[j] + x_val[i] * (x_curr_len – interp_len) / y_meshsize[j]

i ← i+1

j ← j+1

x_prev_len ← x_curr_len

y_prev_len ← y_curr_len

interp_len ← x_curr_len

ENDIF
ENDWHILE
```

This utility can also be used for other MPCORE modules according to the meshing requirements. For example, as shown in Figure 4-6, the location of spacer grids requires different cross-section depending on spacer grid homogenization while the FP is not affected by spacer grid location. The presence of

IFBA (coating length 304.8 cm or 335.28 cm, centered aligned) in cycle 2 and cycle 3 for VERA requires special treatment in FP code but in neutronics code, there is no such requirement.
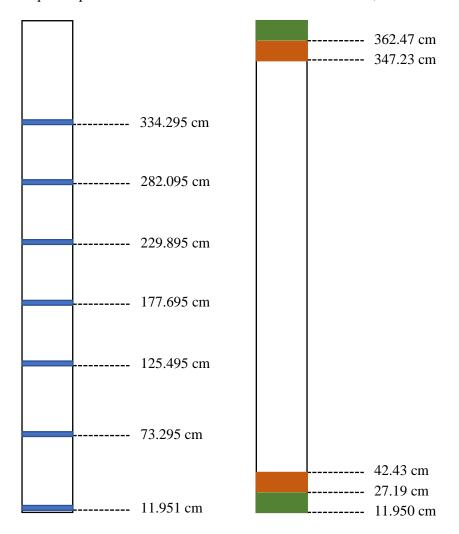

Figure 4-6: (a) Spacer Grid axial location (b) IFBA coating in cycle 2

Interpolation is also required to provide pin values of FP and TH as node average values to NK module. FP and TH axial mesh values are converted to NK axial mesh interpolated values and then node wise average is obtained by,

$$\delta_{node} = \sum_{ipin=1}^{nodepins} \delta_{ipin} * \frac{w_{ipin}}{w_{node}} \tag{4.6}$$

Where 'δ' is the parameter value and 'w' is the weight. w_{pin} denotes the pin weight as 0.25, 0.5 or 1 depending on the location while w_{node} represent the sum of weight of all pins in respective node.

4.6 MPI Parallel implementation

MPCORE runs different modules that may take less execution time like RAST-K or takes higher computation time like CTF. In the case of modular approach as in MPCORE, it is easy to divide the problem and run in parallel the module that takes most of the execution time.

Most of the execution time is taken by subchannel thermal hydraulic module, CTF. MPI parallel implementation is already present in CTF code but it requires same number of processor as the number of assemblies for parallel processing [38]. The data exchange between MPCORE and CTF is also done in parallel. With 'rod owner' functionality in CTF, each processor sets data for its own rods. Core assembly information is already present in MPCORE so specific information is fed to and taken from CTF by respective processors. 'MPI_allreduce is' used at the end of getting information from all processors. Information on MPCORE level is shared by all the processors because all the modules are not using parallel processing and it is difficult to strictly assign the rod to same processor in all the modules.

After CTF, FRAPI takes most of the exaction time and the memory requirement of FRAPI is above all the other modules including CTF. FRAPI models each fuel rod independently, so it is easy to divide the fuel rods among the available processors. Due to requirement of CTF, minimum processors used are equal to number of assemblies, so each processor is solving fuel rods of one fuel assembly at maximum. Sharing data between MPCORE and FRAPI is straight forward. In FRAPI, only the owner processor can assign or retrieve data for a fuel rod while on MPCORE level, 'MPI_allreduce' is called afterwards.

RAST-K is a nodal code, and it takes least time even if CTF is using parallel processing in case of depletion cases. So RAST-K is still running in serial.

MPCORE decision for adaptive timestep depends on the error norm of different parameters for step size 'h' and step size 'h/2'. The dimension of each parameter is equal to the number of rods times the number of axials meshes. So, if the fuel rods are 14784 and the axial meshes are 24. The dimension of each parameter is around ~350,000. Checking for the error of 10 parameters with dimension of 350,000 after each iteration is performed by MPCORE. As already stated, data on MPCORE level is shared by all the processors so for error control the work is divided among all the processors. The algorithm for error control is given in section 4.3. The error control formula summation is divided equally among all processors and the sum is finally calculated and distributed to all processors with MPI_allreduce function.

4.7 Quarter symmetry modeling

Quarter symmetry option is available in RAST-K and CTF modules. In the case of FRAPI, individual fuel rods are modeled so no symmetry option is required. RAST-K gives linear power for all the pins in the symmetry line assemblies. CTF can model full core with quarter symmetry option. With quarter

symmetry option number of modeled fuel assemblies decreases and need less processors to run in parallel. "Get_mirrored_rod" subroutine can be used to get the index of modeled rod that imitates the unmodeled rod in assemblies that exist on symmetry line. In FRAPI, all the fuel rods on symmetry line assemblies are modeled. In case of MPI parallel processing, the number of processors must at least be equal to number of assemblies so modeling all the fuel rods on symmetry line assemblies doesn't cause any overhead as all the processors are modeling equal number of fuel rods.

4.8 Shuffling/rotation for multicycle depletion

Multicycle depletion analysis requires restart information for fission fragments and burnt fuel in each fuel pin. Depletion in fuel affects the NK as the burnt fuel assembly has less fissile material available. Restart file reading with shuffling and rotation of burnt assembly is present in RAST-K of MPCORE. In addition to neutronics, the fuel burnup affects the thermal conductivity of fuel. Pellet clad gap thickness and internal pressure also depends on the fuel burnup. FRAPI coupled with MPCORE has the capability to read the restart file and differentiate between the fresh and burnt assembly. The shuffling and rotation information to constituent modules is transferred through MPCORE interface. The shuffling and rotation capability in fuel performance is confirmed by comparing the end of cycle burnup information with next cycle beginning burnup information.

Burnup information at the End Of Cycle (EOC) 1 and Beginning Of Cycle (BOC) 2 is given in Figure 4-7. Cycle 2 looks like a quarter symmetric core but in the assemblies B-8 and H-14 in BOC 2, less burnt part is modeled for B-8 and more burnt part is modeled in H-14 assembly. In nodal code, reflective boundary condition is used so there will be slight difference in power profile of B-8 and H-14 assembly. Similar burnup difference can be observed in G-8 and H-9 assemblies. Whole core can be modeled in Nodal code for exact results, but in current research CTF and FRAPCON are included so the computation time increases for whole core.

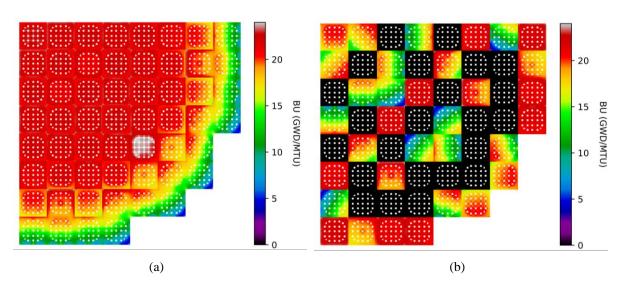(a)   (b)

Figure 4-7: FRAPCON Burnup at (a) EOC 1 (b) BOC 2

EOC 2 and BOC 3 burnup information is given in Figure 4-8. G-8 and H-9 assemblies may not give the same power due to different burnup part modeled in quarter symmetry. Similar differences can be observed in D-8 and H-12.

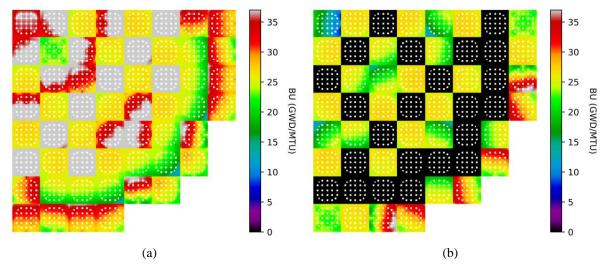(a)   (b)

Figure 4-8: FRAPCON Burnup at (a) EOC 2 (b) BOC 3

Shuffling and rotation capability for RAST-K and FRAPCON has successfully been integrated into MPCORE. Another point to be verified with the benchmark information is the value of gap conductance as a function of power and burnup. Benchmark document has given the values for gap conductance as a function of linear power and burnup given in Table 2.
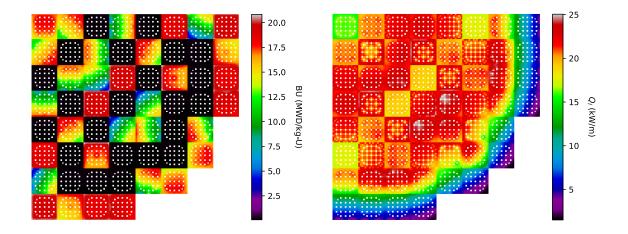
Table 2: Hgap as a function of burnup for a given linear power [courtesy ref [benchmark doc]]

Power [kW/m]	Burnup [MWD/kg-U]	Hgap [W/(m²-K)]
3.28	0.00	4855±105
6.56	0.00	5230±220
9.84	0.00	5645±365
13.12	0.01	6125±545
16.4	0.01	6675±775
19.69	0.01	7340±1090
23.39	1.92	9320±2280
23.39	3.85	19935±11265
23.39	5.78	50400±40100
23.39	7.72	51700±39300
23.39	9.65	52900±38200
23.39	11.58	62050±43950
23.39	13.51	65350±41650
23.39	15.45	70550±36450

23.39	17.38	83900±22100
23.39	19.31	96050±9950
23.39	21.24	95600±9400
23.39	22.40	95050±8950
19.69	23.05	92700±7300
19.69	24.68	93000±8000
19.69	26.31	92850±8150
19.69	27.93	92650±8350
19.69	29.56	92450±8550
19.69	31.19	97050±3950
19.69	32.81	96650±3350
19.69	34.44	96400±3000

These values are used by NK/TH coupled codes. MPCORE has the coupled FP code so gap conductivity given by MPCORE is verified against benchmark values. The point missed by NK/TH coupled codes is that not only the gap conductance, but fuel conductance is also a function of burnup and linear power. Axial integrated burnup and linear power at 3.7 Effective Full Power Days (EFPD) for cycle 2 is given in Figure 4-9 with the axial integrated gap conductivity values.
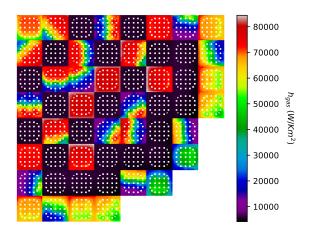
Figure 4-9: Axial integrated gap conductivity given by FRAPCON with burnup and power at 3.7 EFPD cycle 2.

Similarly, gap conductivity with burnup and power is given in Figure 4-10 for cycle 3 at 5.1 EFPD.
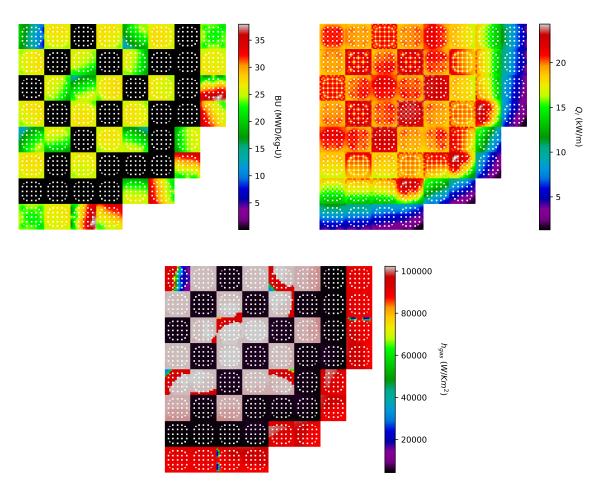
Figure 4-10: Axial integrated gap conductivity given by FRAPCON with burnup and power at 5.1 EFPD cycle 3.

Fuel rods with IFBA coated axial meshes and hollow blanket meshes were modeled by the python input processing. For cycle 2 and 3, IFBA coating is performed in selective fuel rods to balance the high

reactivity in BOC. IFBA coating is center aligned, so for verification, the center mesh of fuel rods in FRAPI is shown in Figure 4-11 for cycle 2 and Figure 4-12 for cycle 3. Burnt fuel rods have reduced gap width as read from restart files, while fresh pins have reduced gap width based on IFBA coating.

| (a) | (b) |
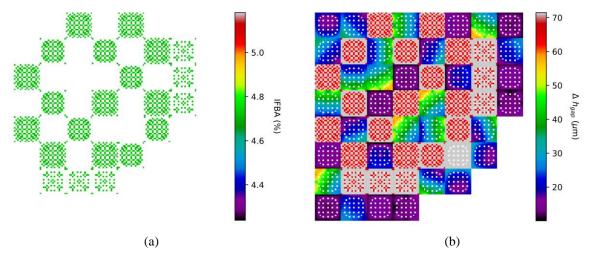
Figure 4-11: (a) IFBA coated pins in cycle 2. (b) Gap width at BOC cycle 2.

| (a) | (b) |
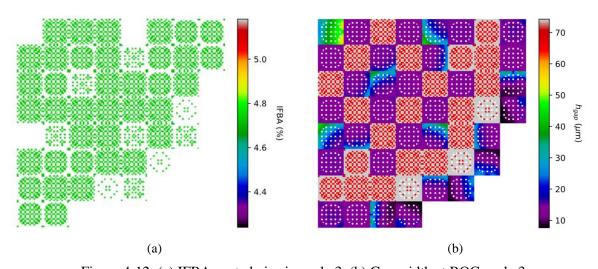
Figure 4-12: (a) IFBA coated pins in cycle 3. (b) Gap width at BOC cycle 3.

In MPCORE, multicycle capability has been developed and verified for 3 cycles of Watts bar reactor. Python input processing utility of MPCORE allows fine modeling in different modules. FRAPI uses individual fuel rod details for accurate modeling of fuel behavior with burnup. Gap width reduction, internal gas pressure accumulation, fuel cracking and fuel thermal conductivity depends on the IFBA coating, hollow fuel pins and enrichment of fuel. Different axial meshes can have different width, enrichment, and hollow/non-hollow characteristics and FRAPCON has the capability to model these fine details.

5   Case studies/Results and comparison

5.1   Watts bar unit 1 description

The Watts Bar reactor given in the VERA benchmark has been modeled for the present work [53]. The Watts Bar reactor consists of 193 fuel assemblies of the Westinghouse design. Each assembly consists of a 17×17 lattice of fuel rods and guide tubes. Among these, 264 are fuel rods, and 25 are guide tubes. Reactor houses eight control rod banks and separated into two categories: regulatory control banks for operation (A, B, C, and D) and safety banks for reactor shutdown (SA, SB, SC, and SD). The rod cluster control assembly (RCCA) for the VERA benchmark is shown in Figure 5-1. The geometrical and material details are specified in the VERA benchmark document.
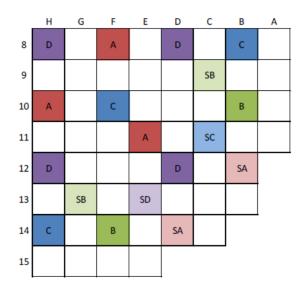
Figure 5-1: RCCAs Bank Positions (Quarter Symmetry)

5.2   Steady state multicycle analysis

The benchmark document has the data for the depletion of VERA core 1. Three cycle fuel performance study has been performed by VERA team where they used MPACT/CTF simulation results to model fuel rods in BISON [48]. The power history given in the document is used to model three cycles of vera with only NK-TH coupling and NK-TH-FP coupling. Among the FP coupling options, one-way FP-TH and two-way FP-TH coupling scheme were used.

5.2.1   Cycle 1

Cycle 1 core loading pattern is shown in Figure 5-2. Three different kinds of fuel enrichment is used in this cycle. Some of the enriched fuel assemblies have Pyrex rods as well to control the initial positive reactivity. The outside core barrel and water reflector is modeled in RAST-K by using reflector cross sections generated from STREAM. Fuel structure is same axially, but different cross sections are used in spacer grid region to model axial heterogeneity as much as possible. Fuel rod modeling in FRAPCON

for this cycle uses the same parameters in axial direction for every fuel pin. 24 axial meshes are taken in FRAPCON and CTF, while 54 axial meshes are taken in RAST-K.

Figure 5-2: Cycle 1 loading configuration

Results for NK-TH coupling, one-way FP-TH, and two-way FP-TH coupling in MPCORE framework are compared. Thermal conductivity of fuel decreases with the increase in burnup and temperature while gap conductivity increases with the increase in burnup due to gap closure. Dynamic gap conductance model of CTF does not include the burnup effect, hence the gap conductance stays in the range of 5000-6000 W/Km^2. It has been shown that fuel temperature predicted by CTF using its own gap conductance is higher than using gap conductance given by FRAPCON [54]. The fuel doppler temperature given by CTF to RAST-K is higher throughout the cycle so the CBC value in case of NK-TH coupling is lower. The coolant temperature is higher in one-way coupling due to high fuel thermal conductivity in CTF. FRAPCON consider burnup dependent conduction correlations. Another difference between CTF and FRAPCON fuel rod modeling is that CTF considers the same fuel and gap parameters for each fuel rod in an assembly while FRAPCON can have different gap width in each axial mesh of every fuel rod. Hollow fuel rod can also be modeled in FRAPCON.

The difference between CBC and fuel temperature is significant in the case of NK-TH coupling and NK-TH-FP coupling. However, one-way, and two-way coupling have small difference in NK parameters. The maximum difference between the two coupling approaches, in boron concentration, is 4 ppm. So, to verify the modeling, only two-way coupling results are compared with MCS/TH1D results as shown in Figure 5-3. The maximum absolute difference between MPCORE and MCS [55] is around 25 ppm. The difference between power profile using both approaches is also negligible as shown in Figure 5-4 and close to MCS/TH1D reported results. After verification of MPCORE modeling, several parameters are compared for one-way and two-way coupling.
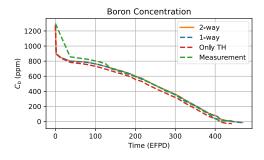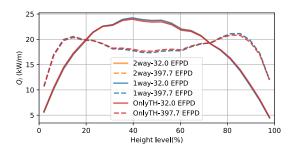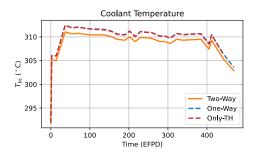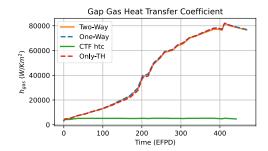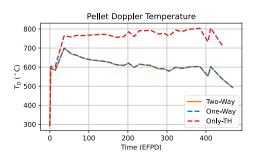
Figure 5-3: Boron concentration cycle 1.

Figure 5-4: Linear power profile at 32 EFPD and 397.7 EFPD

Core maximum parameter values obtained by 'Only TH', two-way and one-way coupling are shown in Figure 5-5. 'Only TH' and one-way coupling results show higher coolant temperatures throughout the cycle as the fuel conductivity correlations in thermal hydraulic code doesn't have the burnup effect. This coolant temperature and linear power drives the FRAPCON conduction. Higher coolant temperature leads to higher oxide surface temperature. Cladding hydrogen concentration is another parameter of interest for fuel analysis and accident scenarios. Multiphysics coupling analysis has been performed to analyze the hydrogen uptake by zirconium cladding [45]. BISON was used based on the results from DECART and CTF coupling. BISON was not used in the coupling, and it uses the coolant temperature predicted solely by CTF as boundary condition. Maximum cladding hydrogen concentration in current research also shows that one-way coupling over predicts the cladding hydrogen concentration. The difference in parameters increases with burnup which shows that for multicycle depletion, two-way coupling should be used.
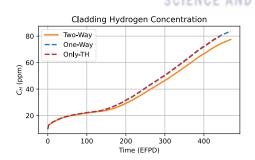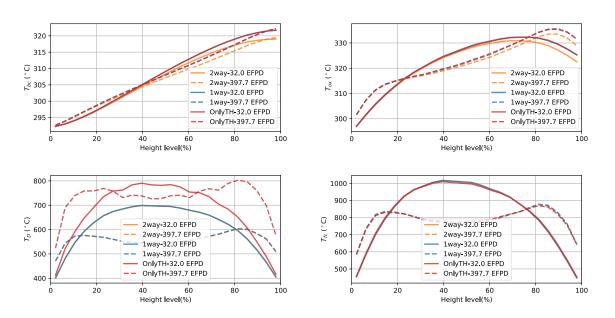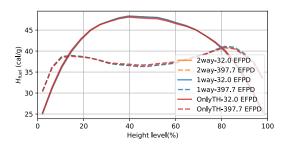
Figure 5-5: Maximum assembly value comparison between Only TH, one-way and two-way coupling.

The axial distribution of core wise parameters with both the models is shown in Figure 5-6. Solid lines show the results at 32.0 EFPD while dotted lines show the results at 397.7 EFPD. Doppler temperature difference between 'Only TH' coupled and FP coupled is highest. In 'Only TH' coupled method, doppler fuel temperature is given by CTF to RK, while in other coupling approaches, FP code is supplying the fuel temperature. The inclusion of FP code is evident from the fact that appropriate fuel doppler temperature is required for exact feedback modeling. The difference between coolant temperature and oxide layer temperature keeps on increasing with height between two-way coupled and only TH/1-way coupled result. High coolant temperatures throughout the cycle length causes more hydrogen absorption in cladding layer. Thus, cladding hydrogen concentration was nearly equal in both coupling approaches at the start of cycle, but the difference widens at the end of cycle 1. More difference, for cladding hydrogen concentration, is observed at upper region of core where the temperature in one-way approach was higher. The other parameters that are provided by FP code in all coupling approaches or TH code in all coupling approaches have minimal difference.
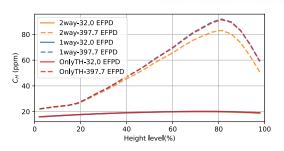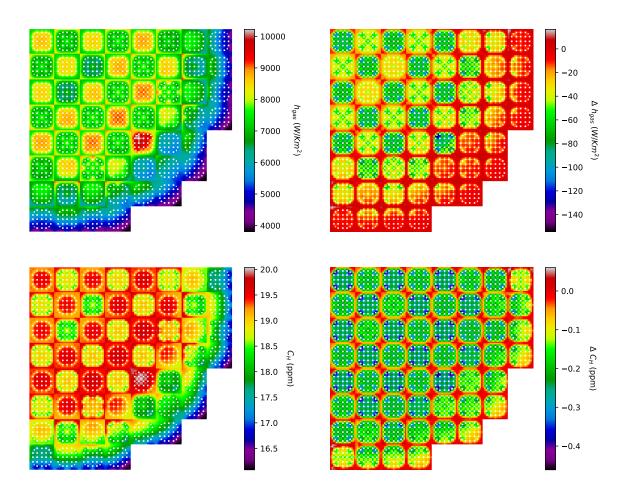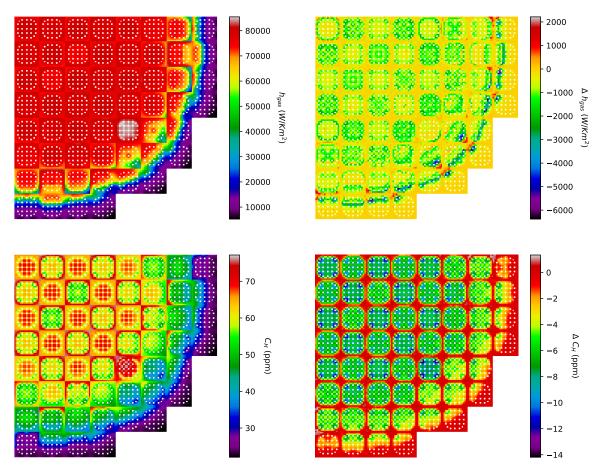
Figure 5-6: Axial distribution of parameters for two coupling approaches.

The parameter values except doppler temperature is same for 'Only TH' and 'one-way' coupled system. So Axial integrated radial pin parameter values are compared for one-way and two-way coupling. Two timesteps of 32.0 EFPD and 397.7 EFPD are selected for this purpose as well. For 32.0 EFPD:

For 397.7 EFPD:

5.2.2   Cycle 2

Cycle 2 core loading pattern is shown in Figure 5-7. Burnt fuel assemblies from cycle 1 and 1 fresh fuel assembly type is added in cycle 2. Pyrex rods in cycle 1 assemblies are removed and IFBA, Wet Annular Burnable Absorber, Tritium Producing Burnable Absorber Rods are added in batch 4 assemblies. The height of IFBA coating is less than complete fuel rod length and is centrally aligned. IFBA coating in central fuel pin reduces the gap width from 84 μm to 74 μm in the start. IFBA percentage is added in FRAPCON modeling as well. The top and bottom of fuel pins also have blanket pellets that have less enrichment. Fuel rod modeling in FRAPCON for this cycle needs to have different parameters axially for selected fuel pins. This modeling is verified and shown in section 4.7. 24 axial meshes are taken in FRAPCON and CTF, while 54 axial meshes are taken in RAST-K.
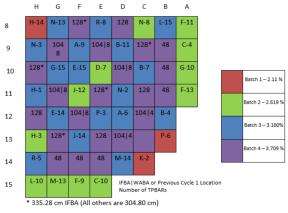
Figure 5-7: Cycle 2 core loading configuration.

Results for 'Only-TH', 'One-Way' and 'Two-Way' coupling in MPCORE framework are compared. In one-way coupling, CTF uses its own conduction solvers for coolant profile calculation. No restart file reading is done in CTF. So, it is not possible for CTF to differentiate between burnt and fresh fuel. On the other hand, FRAPI has developed the capability of reading restart files for FRAPCON as well. This utility is also verified in section 4.7 where gap conductance and burnup are read and the change in location is identical as the change in core configuration. The difference in boron concentration is prominent for 'Only-TH' coupled model while it is around same in one-way and two-way coupled models as shown in Figure 5-8. In one-way and two-way coupled models, neutronics code uses the fuel temperature provided by FRAPCON and coolant temperature provided by CTF. In 'Only-TH' model, both the temperatures are provided by CTF.
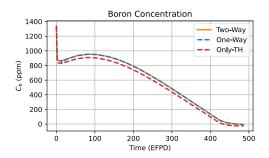

Figure 5-8: Boron letdown curve cycle 2.

The axially integrated maximum assembly value for some parameters during the cycle length are shown below:
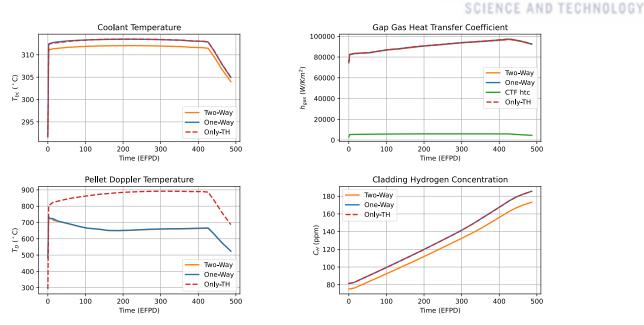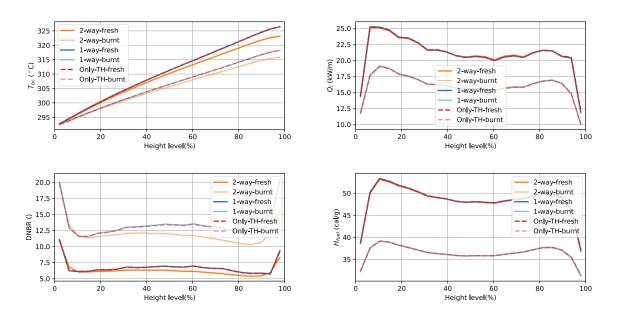
Figure 5-9: Maximum assembly value comparison between coupling approaches cycle 2.

The axial distribution of core wise parameters with both the models at 3.7 EFPD is shown in Figure 5-10. The average for fresh and burnt fuel is taken separately to see the effect on once burnt and fresh fuel. DNBR value for burnt fuel has higher difference because in FRAPCON provided surface temperature, the DNBR approaches low value. It is easy to differentiate between NK/TH coupling to NK/FP/TH coupling code. In this analysis FP code is used in one-way and two-way coupling approaches. FP code is also used in NK/TH coupled system, but it acts as a standalone code.
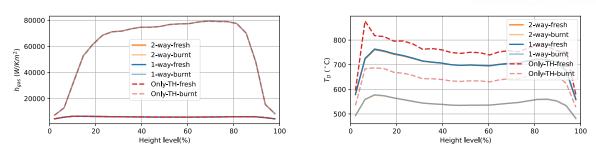
Figure 5-10: Axial parameters value at 3.7 EFPD cycle 2.

The axial distribution of core wise parameters with both the models is compared at 425.9 EFPD, shown in Figure 5-11. After burning the fuel for 1 cycle, batch 4 fuel gap conductance and gap width is the same as old fuel. DNBR value for now twice burnt fuel has still higher difference but the safety margin with both the models is higher as the share of power from older fuel is less.
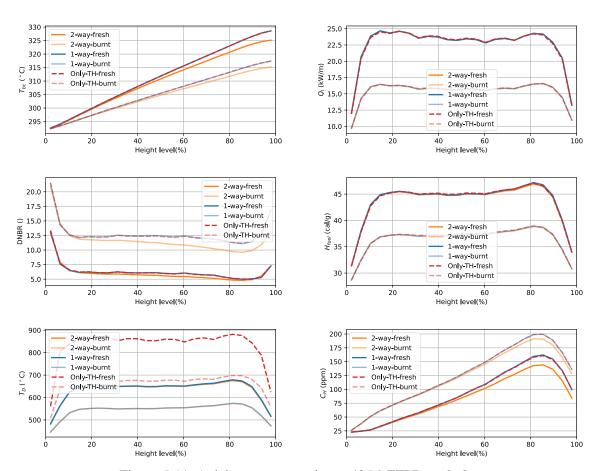

Figure 5-11: Axial parameters value at 425.9 EFPD cycle 2.

The radial difference between different parameters for one-way and two-way coupling at BOC and EOC steps during cycle 2 are shown in Figure 5-12.
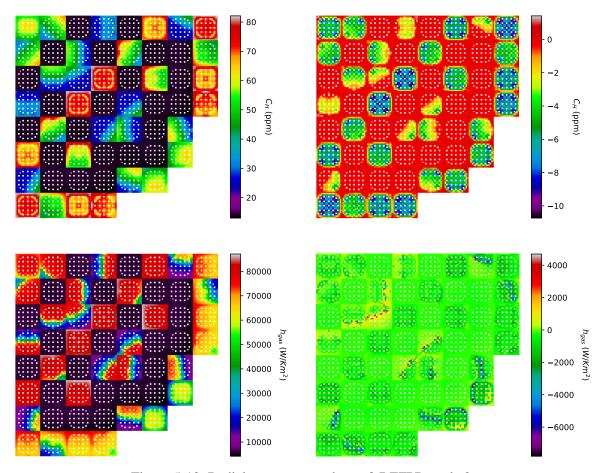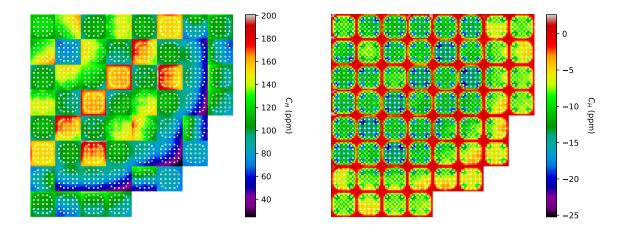
Figure 5-12: Radial parameters value at 3.7 EFPD cycle 2.

The radial difference between one-way and two-way coupling at 425.9 EFPD is also shown in Figure 5-13.
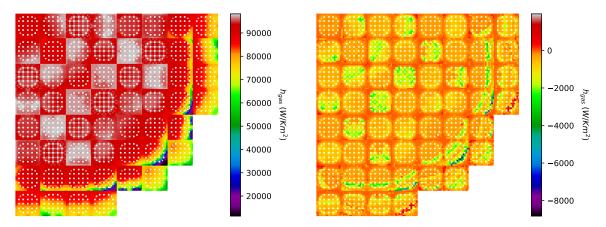
Figure 5-13: Radial parameters value at 425.9 EFPD cycle 2.

5.2.3 Cycle 3

Cycle 3 core loading pattern is shown in Figure 5-14. Burnt fuel assemblies from cycle 1 and 2 are used in cycle 3. Two different kinds of fresh fuel assemblies are also used in cycle 3. The enrichment in cycle 3 is even higher than cycle 2 enriched assemblies. The other unique feature in cycle 3 is the use of hollow fuel pins in the blanket fuel region. The blanket fuel region adds positive reactivity in the start of core while at the end the central void would remove that positive reactivity. This is especially essential because in BOC, the flux profile peak is at the center, so the top and bottom fuel part tries to make it linear. At EOC, flux is shifted to top and bottom part with dip in center. The top and bottom hollow pins once again try to make profile flat. Hollow fuel inner diameter is provided to FRAPCON. FRAPCON has the capability to model fuel heat conduction for hollow fuel pins. The height of IFBA coating again is less than complete fuel rod length and is centrally aligned. The top and bottom hollow fuel pin part also has less enrichment. Fuel rod modeling in FRAPCON for this cycle also needs to have different parameters axially for selected fuel pins. This modeling is verified and shown in section 4.7. 24 axial meshes are taken in FRAPCON and CTF, while 54 axial meshes are taken in RAST-K.
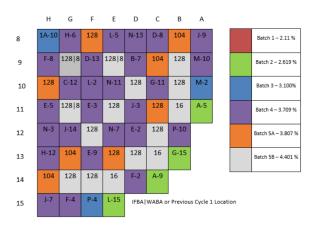

Figure 5-14: Cycle 3 core loading configuration.

Results for 'Only-TH', 'One-Way' and 'Two-Way' coupling in MPCORE framework are compared. Shuffling and rotation capability of MPCORE for cycle 3 (using burnt fuel from two different cycles) has been verified in section 4.7 where gap conductance and burnup are read and the change in location is identical as the change in core configuration. The result for CBC with 'Only TH', 'One-way' and 'Two-Way' modeling is shown in Figure 5-15.
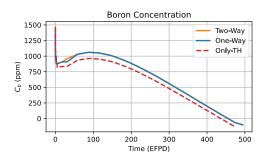

Figure 5-15: Boron letdown curve cycle 3.

The core maximum parameters during the cycle length are shown in Figure 5-16. Average gap thickness at the end of cycle3 is higher than that of cycle 2, thus gap heat transfer coefficient is less in EOC 3 than EOC 2. The difference can be due to the modeling of hollow fuel pins in blanket part of fresh fuel pins. The rod internal pressure let the fuel compress from the top and bottom blanket as well thus average core gap width is higher. Maximum coolant temperature throughout the cycle is again higher for one-way coupling due to burnup deficient heat conduction modeling in CTF.
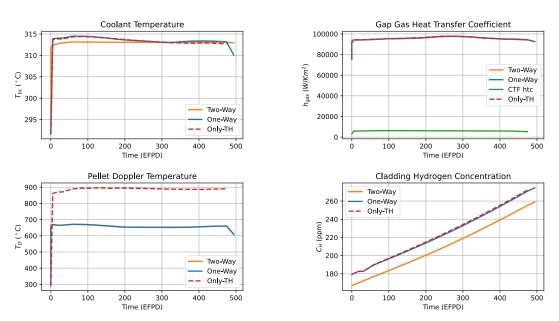

Figure 5-16: Assembly maximum value comparison between coupling approaches cycle 3.

The axial distribution of core wise parameters with both the models at 5.1 EFPD is shown in Figure 5-17. The average for fresh and burnt fuel is taken separately to see the effect on once burnt and fresh

fuel. DNBR value for burnt fuel has higher difference because in FRAPCON provided surface temperature, the DNBR approaches low value.
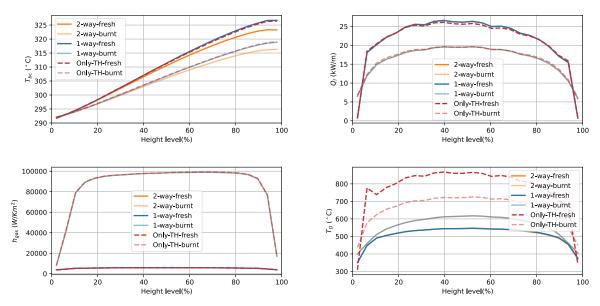
Figure 5-17: Axial parameters value at 5.1 EFPD cycle 3.

The axial distribution of core wise parameters with both the models is compared at 408.5 EFPD, shown in Figure 5-18. After burning the fuel for 1 cycle, batch 5 fuel gap conductance and gap width has changed much but the conductance is still less than older fuel. DNBR value for now twice burnt fuel has still higher difference but the safety margin with both the models is higher as the share of power from older fuel is less.
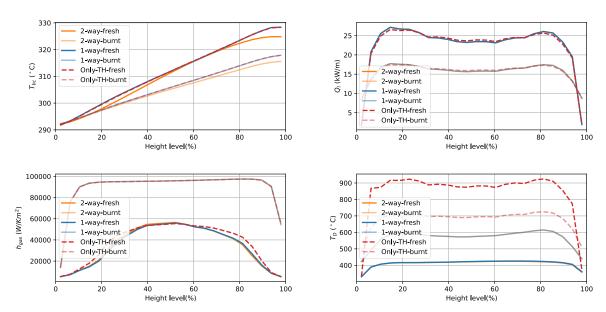
Figure 5-18: Axial parameters value at 474.4 EFPD cycle 3.

The radial difference for one-way and two-way coupling approaches for different parameters at 5.1 EFPD during cycle 3 is shown in Figure 5-19.
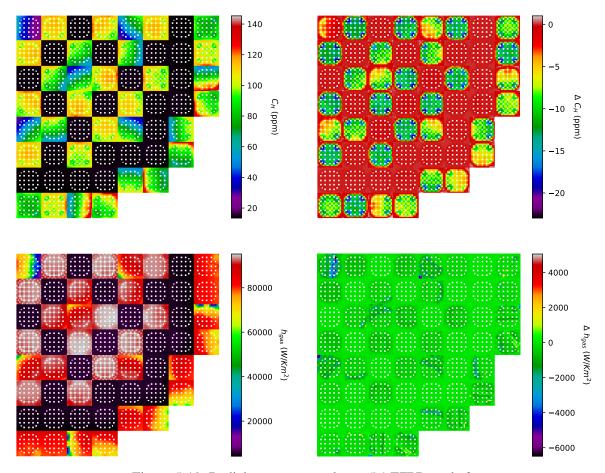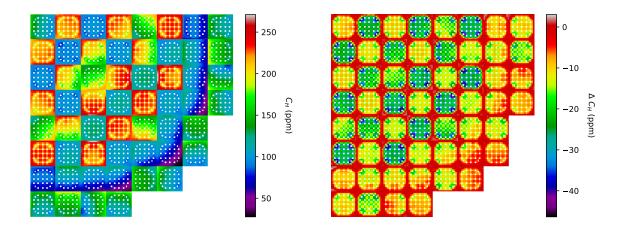
Figure 5-19: Radial parameters value at 5.1 EFPD cycle 3.

The radial difference between one-way and two-way coupling at 474.4 EFPD is shown in Figure 5-20. The difference in the gap conductance by two models is even higher at the EOC 3 as compared to EOC 2. This is due to the presence of hollow fuel pellets at the axial edges in addition to the IFBA coating on the fuel pins. IFBA coating and hollow fuel pins greatly affect the heat transfer process and FRAPCON has the capability to model these while calculating heat conductance through fuel.
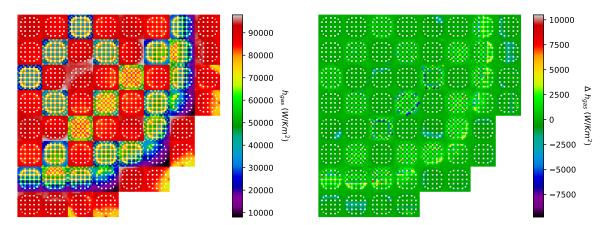
Figure 5-20: Radial parameters value at 474.4 EFPD cycle 3.

5.3 Transient analysis

Control-rod insertion limits are imposed during the normal operation of a plant such that the shutdown margin is minimum even in the most reactive case. The control bank withdrawal safety requirement was imposed after the SL-1 accident. Moreover, to prevent fuel failure, the fuel pellet average enthalpy should be less than 170 Cal/gm for fresh fuel and less than approximately 130 Cal/gm for burnt fuel [56]. Furthermore, fuel melting also needs to be less than 10% at the innermost fuel pellets. The purpose of these requirements is to prevent criticality by the withdrawal of a control device. This work assumes a hypothetical position of control bank D such that the rod ejection can make the reactor prompt critical.

A rod ejection case is considered at the first BOC for hot zero power (HZP) and hot full power (HFP) conditions. For both the cases, TH code is employed with and without crossflow option. Similarly, FP code is used with static and dynamic gap heat transfer coefficient to observe the effect on the overall safety parameters. Quarter core modeling is performed for all the cases. Thereby, the number of modeled assemblies is 56 rather than 193.

The results obtained using the following four models are analyzed in this section:

1. Dynamic gap HTC with crossflow (named as *dynamic-SC*)
2. Dynamic gap HTC with no crossflow (named as *dynamic-1D*)
3. Static gap HTC with crossflow (named as *static-SC*)
4. Static gap HTC with no crossflow (named as *static-1D*)

These models are executed for both accident scenarios. The safety parameters (DNBR, q_l, h_f, T_{fc}, T_{co}, and T_{bc}) at peak are considered for both the case scenarios.

5.3.1 Convergence Criteria:

RAST-K uses four nodes for each assembly with 50 axial meshes for fuel region and 1 axial mesh each for the top and bottom reflectors. The convergence criterion inside RAST-K is set as 1×10^{-5} for power

and flux between successive iterations. In the present work, 10 axial meshes are considered for the TH module (CTF), with a W-3 correlation for the critical heat flux and Chen correlation for fHTC calculation. CTF uses its internal convergence criteria. This criterion is set for the energy stored in the fluid and solids, global energy balance, amount of mass stored in the system, and global mass balance. These criteria are 0.1%, 0.1%, 0.001%, 0.1%, and 0.001%, respectively, by default. In FP code, 10 axial meshes, and 8 radial meshes are used. Among the radial meshes, 5 are in the pellet region, and 3 are in the cladding region. The fixed point method selected for convergence is the Picard method with an absolute tolerance of 1×10^{-5} and a relative tolerance of 1×10^{-3} in both HZP REA and HFP REA cases.

5.3.2 HZP REA BOC:

Control bank D is ejected partially (from 20 steps to 100 steps, out of the total 230 steps) at the BOC with HZP condition. The entire control bank withdrawal is simulated to utilize the quarter core model, which requires lesser execution time. The other regulatory control banks for operation (A, B, and C) are inside the core, whereas the safety banks for shutdown (SA, SB, SC, and SD) are out of the core. The normal trip mechanism is not considered in this case. A 10 s simulation is performed to analyze the passive safety features, namely, the fuel doppler coefficient of reactivity and moderator temperature coefficient of reactivity. The initial temperature of fuel, moderator, and coolant are in equilibrium at 291.85 °C, whereas the critical boron concentration at the start of transient is 755 ppm with an initial power equal to 10^{-4} % of the full power (3411 MW$_{th}$).

The results for power and the transient reactivity are shown in Figure 5-21 and Figure 5-22. In the first figure, 'Only TH' coupled model result is compared with 2-way FP coupled model. The reactivity and power reach the peak early in 'Only-TH' coupled module. The safety parameters for 'Only-TH' and '2-way' are also compared in Figure 5-23. The difference in 'Only TH' coupled module is evident due to different power. Results for 4 coupling approaches involving FP code obtained by the four models are plotted in Figure 5-24. The difference between the values is evident in Table 3, where the peak value of each parameter and their respective time are presented. The nomenclature for Table 3 is provided in Figure 3-12. Prompt criticality (reactivity = 1$) is achieved at 66.3 ms for all the cases. It results in a rapid increase in power and subsequently, temperatures. Figure 5-24 indicates that because of the prompt critical condition, first, the fuel centerline temperature increases. This is followed by the clad temperature and subsequently, the coolant temperature. Hence, fuel doppler feedback is the first reaction after the rapid power increase. The moderator feedback effect plays its role after approximately 0.2 s. The dynamic gap effect in FP resulted in a slower heat transfer from fuel to cladding in the initial stage (3280 W/(m^{2}K)). However, it attains a higher value (up to 8600 W/(m^{2}K)) as the power increases. Therefore, the peak fuel temperature in this case is low owing to the rapid heat transfer from the fuel. The static gap HTC is considered as a constant in our analysis (5640 W/(m^{2}K)).

A 2D comparison for the peak axial values at the time mentioned in Table 3 is shown in Figure 5-25. The difference in Figure 5-25 is computed using Equation (4.1):

$$Difference = AP_{dynamic-SC} - AP_{dynamic-1D} \qquad (1.1)$$

where $AP_{dynamic-SC}$ and $AP_{dynamic-1D}$ are the axial peak values obtained for each pin with the dynamic-SC model and dynamic-1D model, respectively. A positive difference implies that the values predicted by the dynamic-SC model are higher, and vice versa. From Figure 7, the minimum DNBR is obtained in the D-12 assembly (refer to Figure 4), the assembly with control bank D. The linear power and bulk coolant temperature values are also maximum in the D-12 assembly. The effect of crossflow is clearly visible because the DNBR for fuel pins is smaller (less safety margin) in case of no-crossflow. Owing to the coolant mixing between different sub-channels, the DNBR remains marginally higher for most reactive pins as well in the dynamic-SC case. The peak reactivity added in the system is approximately 2$ at 110 ms. The peak power obtained is approximately 420 GW (~124 times of full power) at 145 ms.

All the results in the case of HZP REA BOC show that the selection of dynamic or static gap heat transfer has a significant effect on the outcome. The safety margin for fuel centerline and DNBR increase in the case of crossflow (a high value of DNBR and lower value of temperature is desirable). However, the fuel enthalpy safety margin decreases in the case of crossflow. This shows that 1D code (no-crossflow) may not yield conservative results for all the parameters simultaneously. Thus, in the search for conservative results for a safety parameter, the other safety parameters are compromised.
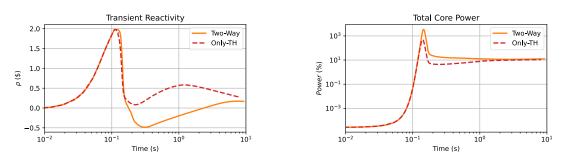

Figure 5-21: 'Only-TH' and FP coupled reactivity and power in HZP REA BOC
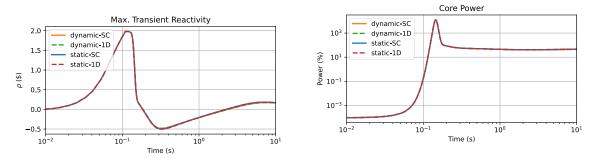

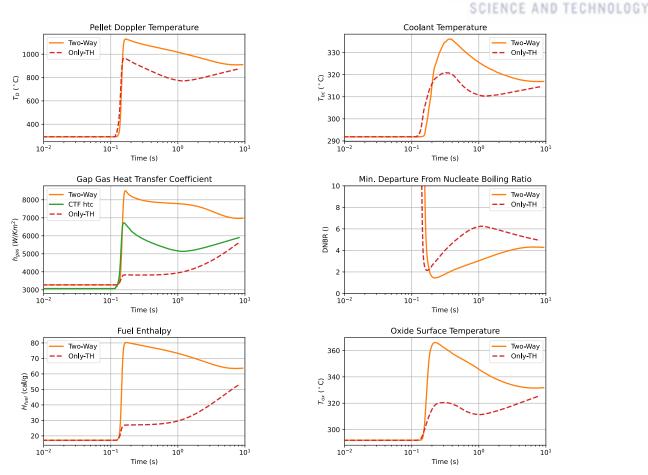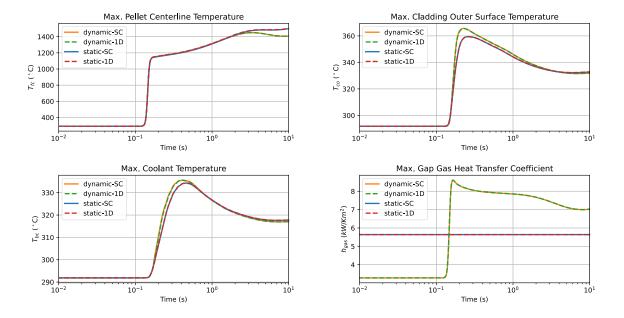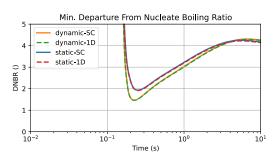Figure 5-22: Reactivity and core power variation in HZP REA BOC

Figure 5-23: Pin peak values with time for 'Only TH' and 2way FP coupled HZP REA BOC
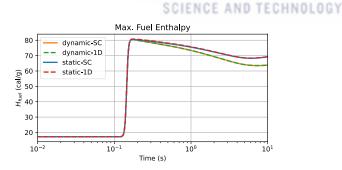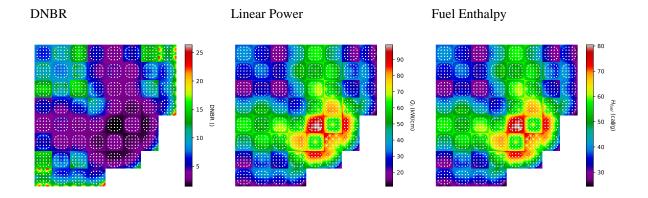
Figure 5-24: Pin-wise peak values of parameters for HZP REA with time

The minimum time-step for the 10 s transient is 1 ms (at the start of the transient), and the maximum time step is 0.9 s (in the final five time-steps).

Table 3: Peak values of the parameters and occurrence time for HZP REA

Parameter	dynamic-SC	dynamic-1D	static-SC	static-1D
Power (%)	12432 (145 ms)	12432 (145 ms)	12324 (145 ms)	12440 (145 ms)
T_{fc} (°C)	1450 (3.302 s)	1450 (3.311 s)	1500 (10 s)	1503 (10 s)
T_{fs} (°C)	1063.6 (159 ms)	1063.6 (159 ms)	1075.5 (161 ms)	1079.4 (161 ms)
T_{ci} (°C)	474.3 (205 ms)	474.1 (204 ms)	441.8 (229 ms)	442.1 (229 ms)
T_{co} (°C)	365.7 (227 ms)	365.5 (228 ms)	359.4 (259 ms)	359.4 (259 ms)
T_m (°C)	335.63 (418 ms)	335.59 (413 ms)	334.34 (464 ms)	334.36 (459 ms)
Min. DNBR	1.46 (222 ms)	1.46 (226 ms)	1.93 (248 ms)	1.92 (249 ms)
ρ_m (kg/m^3)	627.57 (418 ms)	627.80 (413 ms)	632.81 (464 ms)	632.78 (459 ms)
fHTC (kW/km^2)	60.56 (266 ms)	60.45 (267 ms)	51.80 (291 ms)	52.19 (291 ms)
Gap width (µm)	33.72 (1.38 s)	33.70 (1.36 s)	32.57 (1.86 s)	32.41 (1.85 s)
gHTC (kW/km^2)	8.60 (165 ms)	8.61 (165 ms)	-	-
h_f (cal/g)	80.45 (170 ms)	80.45 (170 ms)	80.49 (174 ms)	80.80 (174 ms)

DNBR   Linear Power   Fuel Enthalpy

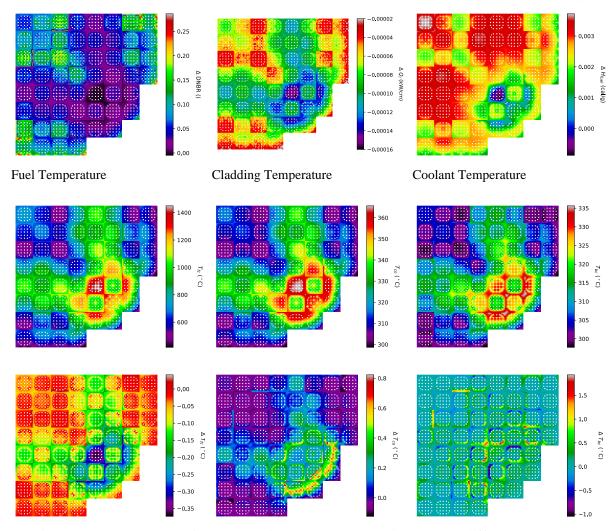Fuel Temperature   Cladding Temperature   Coolant Temperature

Figure 5-25: Peak values of parameters for HZP REA dynamic-SC (top), difference between dynamic-SC and dynamic-1D (bottom)

5.3.3 HFP REA BOC:

Like the previous case, the symmetric REA is simulated in this case as well. The partial ejection of control bank D provides a prompt critical transient. Bank D is considered as 20 steps withdrawn (out of the total 230 steps), and it shifts 80 steps out in an interval of 0.1 s (from 10 ms to 110 ms). The other regulatory control banks for operation (A, B, and C) and safety banks for shutdown (SA, SB, SC, and SD) are out of the core. This hypothetical accident is simulated in MPACT with internally coupled TH for the HFP condition [57]. For this case, the critical boron concentration at the start of transient is 967 ppm for the dynamic gap model and 966 ppm for the static gap model. The initial power is 100% full power 3411 MW.

For the HFP case, the peak reactivity added in the system is 1.16$ at 107 ms. Prompt criticality (reactivity = 1$) is achieved at 86.2 ms, 86.6 ms, 87.0 ms, and 87.6 ms for the dynamic-1D, dynamic-SC, static-SC, and static-1D cases, respectively. Prompt criticality results in a rapid increase in the power and subsequently, temperatures. Figure 5-26 shows that the reactivity and total core power at the

end of transient are higher for the dynamic gap HTC cases. The effect of crossflow is not visible in Figure 5-27. However, it clearly discriminates the dynamic/static gap effect. The minimum DNBR remains low (less safety margin) for the dynamic gap cases. This is because dynamic cases result in higher heat transfer to cladding and thereby, to the coolant. The dynamic gap effect in the FP results in fast heat transfer from fuel to cladding. Thereby, the peak fuel temperature in this case is low compared with the static gap cases. Furthermore, fuel enthalpy is also lower in the dynamic gap cases owing to the high heat transfer, low fuel centerline, and low clad surface temperature. In the case of the HFP initial condition, the dynamic gap width predicted by FRAPTRAN initially is higher than the static fixed gap width (5640 W/(m^{2}K)) considered in this work. Thereby, after the REA, the value of the dynamic gap HTC remains higher than the static value. The peak values of the safety parameters with their respective times are shown in Table 3.

The effect of crossflow is apparent in the 2D quarter core comparison shown in Figure 5-28. The difference in Figure 5-28 is calculated using Equation (4.1). Because it is a partial REA, the upper part of the control-bank-D assemblies has less power. In the sub-channel case, the crossflow effect enables heat transfer from the neighboring hottest channel. Therefore, the maximum coolant temperature is less (as shown in Figure 5-28). Consequently, the coolant heat is transferred from other assemblies to the bank D assemblies. This, in turn, results in a higher cladding outer temperature and fuel enthalpy in the SC cases, particularly in the bank-D assemblies. Although 1D TH is more conservative approach, but the higher cladding outer temperature for SC case show that detail analysis is required for the justification of 'conservative' approaches especially in the case of partial ejection of control rod. Partial rod removal limits the coolant in control bank D assembly to take heat from the top part of D control bank assembly.
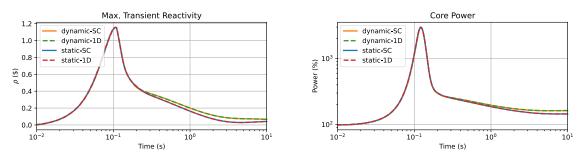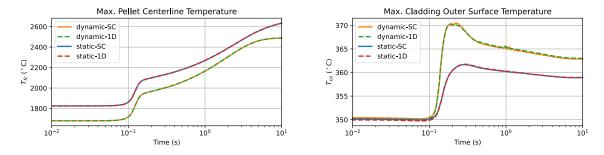

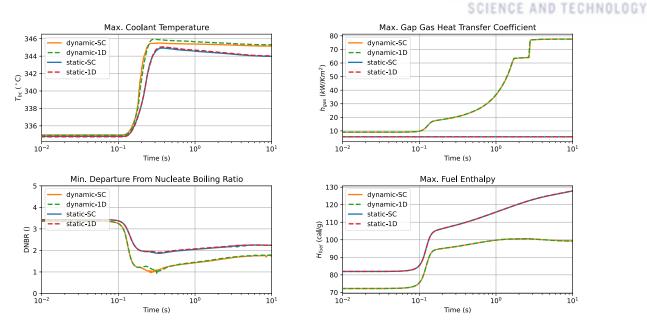Figure 5-26: Reactivity and core power variation in HFP REA BOC

Figure 5-27: Pin-wise peak values of parameters for the HFP REA case with time

The minimum time-step for the 10 s transient is 0.6 ms (during the rapid power increase), and the maximum time-step is 0.52 s (in the final four time-steps).

Table 4: Peak values of parameters and occurrence time for HFP REA

Parameter	dynamic-SC	dynamic-1D	static-SC	static-1D
Power (%)	2923 (121 ms)	2928 (121 ms)	2995 (122 ms)	2992 (122 ms)
T_{fc} (°C)	2489.5 (10 s)	2488.7 (10 s)	2634.1 (10 s)	2634.8 (10 s)
T_{fs} (°C)	757.4 (143 ms)	757.7 (143 ms)	894.6 (153 ms)	894.5 (153 ms)
T_{ci} (°C)	487.0 (167 ms)	487.2 (168 ms)	436.1 (202 ms)	436.1 (205 ms)
T_{co} (°C)	370.5 (222 ms)	370.3 (193 ms)	361.7 (283 ms)	361.8 (288 ms)
T_m (°C)	345.51 (339 ms)	345.97 (299 ms)	344.93 (372 ms)	345.06 (376 ms)
Min. DNBR	0.96 (267 ms)	0.94 (316 ms)	1.86 (336 ms)	1.91 (342 ms)
ρ_m (kg/m^3)	378.95 (328 ms)	390.72 (349 ms)	479.68 (392 ms)	487.52 (396 ms)
fHTC (kW/km^2)	85.69 (232 ms)	86.76 (253 ms)	73.32 (342 ms)	73.98 (342 ms)
Gap width (µm)	0.0 (2.70 s)	0.0 (2.79 s)	0.0 (0.57 s)	0.0 (0.57 s)
gHTC (KW/km^2)	77.66 (6.69 s)	77.61 (7.07 s)	-	-
h_f (cal/g)	100.64 (2.70 s)	100.64 (2.67 s)	127.83 (10 s)	127.93 (10 s)

DNBR   Linear Power   Fuel Enthalpy
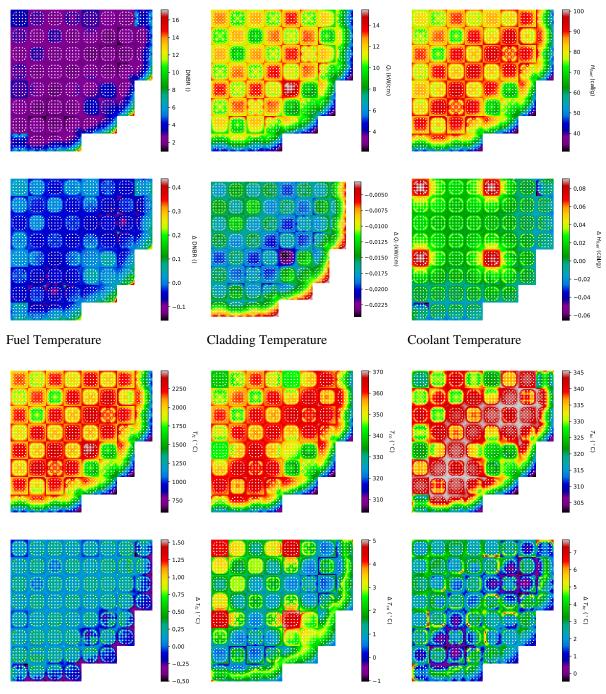
Figure 5-28: Peak values of parameters for HFP REA dynamic-SC (top), difference between dynamic-SC and dynamic-1D (bottom)

The power peak in the case of RAST-K for HFP REA is early and higher than the one published with MPACT/TH [57]. MPACT performs one-step integral transport calculations using MOC. Meanwhile, RAST-K (used as the neutronics module in this work) involves a two-step calculation with nodal homogenization and pin power reconstruction using the form functions. The macroscopic cross-sections are generated at the assembly level (2D) assuming no effect of neighboring assemblies. One of the most important differences is the treatment of the doppler feedback in both the codes. The fuel and moderator

temperature are provided locally in MPACT, whereas RAST-K uses the node averaged fuel and moderator temperature. A similar difference between transport code and diffusion code has been analyzed in a few of the previous studies using DeCART (transport code) and PARCS (nodal diffusion code) [58, 59].

5.3.4   HZP REA MOC:

Positive reactivity is added again at MOC stage, 163.3 EFPD, in cycle 1. Same core configuration, axial meshes, and initial state of control rods are used. D bank is ejected by 50 steps at this stage, so the reactivity addition is less than that of BOC REA. Simulation was executed for 3 seconds in this case. Reactivity is also less because of the presence of burnt fuel at this stage. All the results in the case of HZP REA MOC show the comparison of 'Only TH' coupled and fully coupled system with dynamic and static gap conductance.
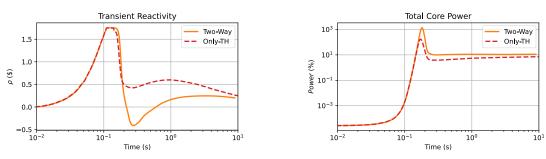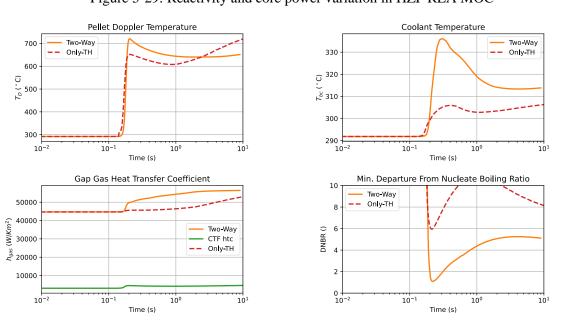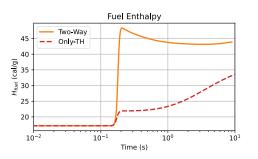

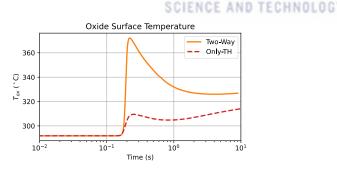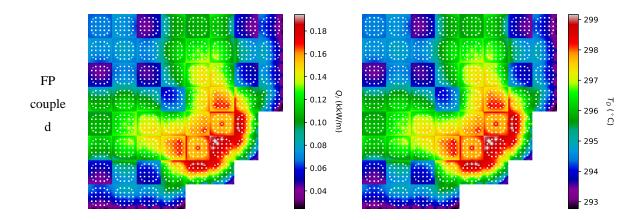Figure 5-29: Reactivity and core power variation in HZP REA MOC

Figure 5-30: Pin-wise peak values of parameters for HZP REA MOC with time

The 'Only TH' coupled system did not include FP code at all so the enthalpy, gap conductance by FRAPTRAN and centerline temperature from FRAPTRAN are not included in the results. It must be mentioned here that in the case of nuclear industrial practice of NK-TH coupling, these parameters are obtained from separate FP calculations. In this study, the sperate study from FP code is not done. But the trend from static and dynamic gap results can estimate where the results would lie in case of FRPATRAN separate modeling. The safety parameters for fuel centerline and enthalpy are reduced by dynamic gap width but the coolant profile is approaching close towards the DNB region. Any shift towards change in boiling regime could have catastrophic results, as the surface temperature gets very high. The safety parameters which look promising to us now would be even worse in the case of DNB. The reason for less peak power is the effective doppler feedback in 'Only TH' model and FP coupled module. At one time (where linear power is almost same) during the transient, the axial integrated radial power and doppler temperature in both cases is given in Figure 5-31. Doppler temperature given by CTF is much higher that leads to more negative reactivity and thus reduces the peak power. CTF solves pseudo transient for solving the time dependent phenomenon while FRAPTRAN is especially designed for rapid transients so the rate of power increase may or may not increase fuel temperature rapidly in FRAPTRAN case.

FP coupled
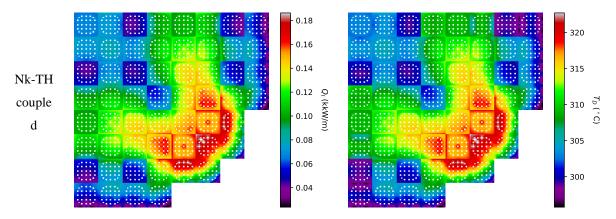
Figure 5-31: Linear power and doppler temperature predicted by CTF and FRAPTRAN

5.4 Computational requirement of each module
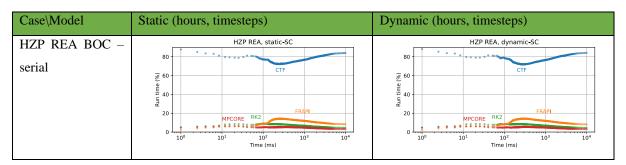
The execution times for the all the cases executed for HZP and HFP are compared in Table 4. The number of time steps is also mentioned for each case. Execution time for HZP and HFP REA at BOC were performed with the serial code. The parallel version is utilized for MOC transient case.

Table 5: Time comparison for cases with different models

Case\Model	Static (hours, timesteps)	Dynamic (hours, timesteps)
HZP REA BOC – serial	22.7 (416)	22.3 (411)
HFP REA BOC – serial	19.2 (172)	19.1 (269)
HZP REA MOC	2.4 (313)	2.6 (330)
Case\Model	One-way	Two-way
Cycle 1	45.8 (37)	36.8 (37)
Cycle 2	29.6 (25)	33.2 (25)
Cycle 3	23.2 (25)	25.1 (25)
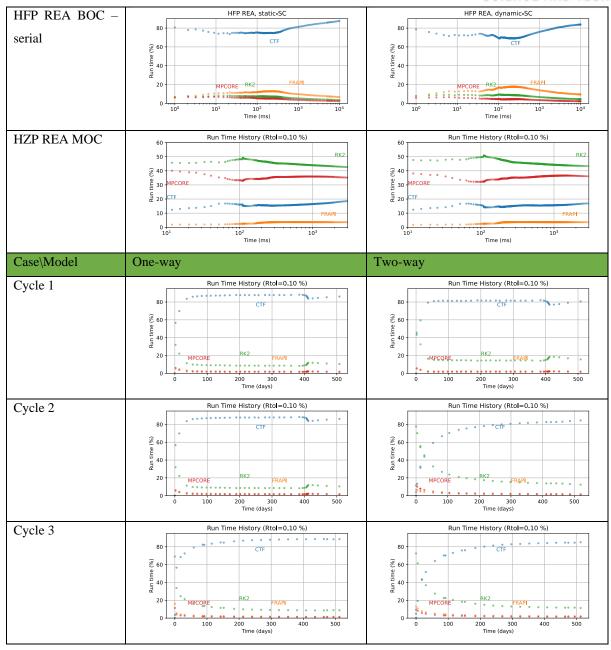
Share of each module in these cases are given below:

Table 6: Share of each module

Case\Model	Static (hours, timesteps)	Dynamic (hours, timesteps)
HZP REA BOC – serial		

HFP REA BOC – serial	HFP REA, static-SC	HFP REA, dynamic-SC
HZP REA MOC	Run Time History (Rtol=0.10 %)	Run Time History (Rtol=0.10 %)
Case\Model	**One-way**	**Two-way**
Cycle 1	Run Time History (Rtol=0.10 %)	Run Time History (Rtol=0.10 %)
Cycle 2	Run Time History (Rtol=0.10 %)	Run Time History (Rtol=0.10 %)
Cycle 3	Run Time History (Rtol=0.10 %)	Run Time History (Rtol=0.10 %)

6 Uncertainty Quantification

The best estimate code is developed and tested for the whole core cases. The input parameters uncertainty quantification part is developed in MPCORE as well that can do random sampling or use Latin Hypercube Sampling. For testing of this UQ capability single assembly case and 3*3 mini-core case (using quarter symmetry so model 4 assemblies).

6.1 Single Fuel Assembly

Reactivity Initiated Accident (RIA) is performed for fresh fuel assembly of VERA benchmark with 2.11% enrichment. It is a 17×17 fuel assembly with 25 guide tubes and 264 fuel rods (Westinghouse design). Hot zero power RIA is modeled by change in boron concentration which was reduced by 60 ppm in 0.1 sec.

6.1.1 Modeling

Uncertainty quantification with perturbed input is performed such that each pin diameter, clad radius can be different. Fuel rod fabrication data uncertainty contains the mean value of parameters (diameter, density, length etc.) and tolerance values allowed during manufacturing. Each fuel rod can have a slightly different value of parameter within the tolerance range; therefore, different uncertainty value is applied to each input parameter of all the fuel rods. This work uses simple random sampling for all the parameters. Latin hypercube sampling is used for mini core problem future to assess the implications of the sampling technique. Since all the rods have different random values, it is nearly impossible that any two assembly inputs are the same. Power and inlet temperature values are taken at the assembly level. Benchmark (mean) value and perturbed values of input are given in Table 7. Normal distribution occurs often in nature where sample distribution is influenced by large number of tiny, random disturbances, hence, current work focuses on normal distribution for the input parameters (Uniform distribution may be taken for some parameters in future). Maximum tolerance of each parameter is $\pm 3\sigma$ but taking the normal distribution means that more samples occur near the mean value. Perturbation variance considered for uncertainty quantification in input data are taken from [60]. RAST-K uses the same cross-sections for all the cases so cross section co-variance effect is not analyzed in this study.

Table 7: Perturbed Input Values

Parameter	Mean Value	Perturbation (σ)
Power (MW)	3411/193	$\pm\ 3411/193 \times 0.015$
Inlet temperature (K)	565	$\pm\ 565 \times 0.03$
Pellet diameter (cm)	0.8192	$\pm\ 0.0013/3$
Cladding outer diameter (cm)	0.9500	$\pm\ 0.0040/3$
Cladding inner diameter (cm)	0.8360	$\pm\ 0.0040/3$
Fuel density (g/cc)	94.0147	$\pm\ 0.91/3$

| Length of plenum (cm) | 19.340 | ± 1.14/3 |
| Pin Pressure (MPa) | 2.0 | ± 0.07/3 |

Homogeneous spacer grid is assumed during the cross-section generation in STREAM. For this work, RAST-K employs 50 axial nodes while FRAPTRAN and CTH/CTF employ 10 axial nodes. Linear power for 10 axial meshes of FRAPTRAN and CTH/CTF is obtained by interpolating the data obtained from RAST-K. For the FRAPTRAN model, 5 radial nodes per fuel pellet and 3 radial nodes per cladding have been used.

6.1.2   Results

For single assembly case, the neutronics and fuel performance codes have been fixed to RAST-K and FRAPTRAN respectively. The only difference is the use of single channel TH (CTH) code or sub-channel TH code (CTF). First both the models are executed for the benchmark values as input for fuel assembly. Afterwards, they are run for perturbed inputs. The transient is introduced at 0.01 sec, and ends at 0.11 sec. With an initial power of 17.7 MW ± 0.54 (3.1%), critical boron concentration (CBC) at the start of transient in CTH cases was 1374.1 ± 20.8 (1.51%) ppm. This introduced a reactivity addition of \$0.92 ± 0.02 (1.7%). In the case of CTF model initial power was 17.6 MW ± 0.49 (2.8%) and the CBC value came out to be 1379.86 ± 15.78 (1.14%) ppm. Here the reactivity addition was \$0.92 ± 0.01 (1.4%). Different values of gHTC and moderator temperatures are obtained at each axial mesh. Assembly average values are provided to neutronics module. Depending on the perturbed total assembly power value, RAST-K provides linear power at each axial mesh of fuel rods. A total of 220 samples are executed with each of CTH and CTF framework. The standard deviation of parameters reached steady state with this number of samples as visible from Figure 6-1.
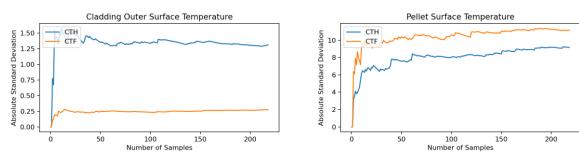

Figure 6-1: Standard deviation with samples

The range of output parameters obtained in both the cases is given in Table 8. It contains minimum departure from nuclear boiling ratio (MDNBR), maximum temperature at different locations (fuel center, fuel surface, coolant, cladding inner surface, cladding outer surface), dynamic gHTC, film HTC and maximum power during transient. The range of values show that the fuel center line temperature and cladding temperatures are higher in case of CTH. Due to higher value of Film HTC in CTF, more

heat transfer results in less temperatures. Hence, DNBR safety margin is higher in case of subchannel thermal hydraulic code.

Table 8: Output values of UQ for single assembly

Parameter	CTF, mean ± abs SD (%Rel. SD)	CTH, mean ± abs SD (%Rel. SD)
MDNBR	3.335 ± 0.298 (8.92)	1.402 ± 0.084 (6.01)
pellet centerline temperature (K)	2280.847 ± 82.654 (3.62)	2280.205 ± 87.263 (3.83)
pellet surface temperature (K)	947.732 ± 11.163 (1.18)	946.489 ± 9.163 (0.97)
coolant temperature (K)	616.64 ± 0.87 (0.14)	617.81 ± 0.36 (0.06)
cladding inner surface temperature (K)	689.26 ± 2.89 (0.42)	699.52 ± 3.19 (0.46)
cladding outer surface temperature (K)	632.89 ± 0.27 (0.04)	646.09 ± 1.31 (0.20)
gap gas heat transfer coefficient (W/K-m2)	26355.1 ± 9362.4 (35.5)	26409.3 ± 9231.6 (35.0)
film heat transfer coefficient (W/K-m2)	68075.4 ± 3062.6 (4.50)	44445.9 ± 2471.3 (5.56)
Fuel Enthalpy (cal/g)	87.29 ± 3.03 (3.46)	87.48 ± 3.14 (3.59)
Peak Power (MW)	137.634 ± 18.0 (13.09)	144.422 ± 26.66 (18.46)

The focus of this study is the difference in safety parameters when sub-channel coolant model is employed instead of single channel code, so here the results for each of the crucial parameters are provided. Figure 6-2 shows the DNBR values for benchmark input and perturbed input. Single channel coolant modeling yields less margin for DNBR while Sub-channel code produces more realistic coolant modeling. The inter-channel mixing of coolant shows that single channel was providing more conservative values. DNBR values obtained by CTF (using W3 correlation) show that reactor is safe in contrast to value given by CTH (using EPRI correlation). As the maximum coolant and cladding temperature obtained via sub channel code are lower than CTH, therefore, the DNBR value is well above the allowed threshold value.

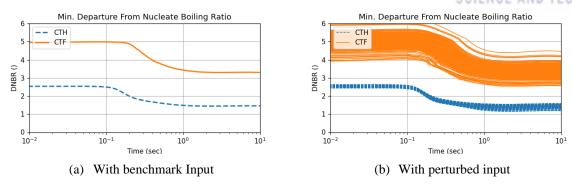(a) With benchmark Input   (b) With perturbed input

Figure 6-2: DNBR comparison for single assembly.

Fuel Enthalpy and peak centerline temperature for CTH and CTF models at the end of transient are compared in Figure 6-3 and Figure 6-4. Like DNBR, Peak centerline temperature and fuel enthalpy too have more safety margin in case of subchannel code.
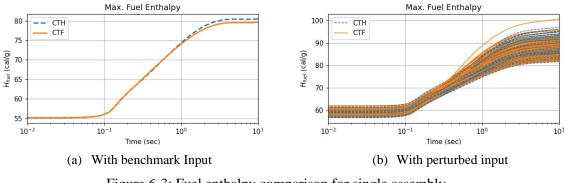

(a) With benchmark Input   (b) With perturbed input

Figure 6-3: Fuel enthalpy comparison for single assembly.

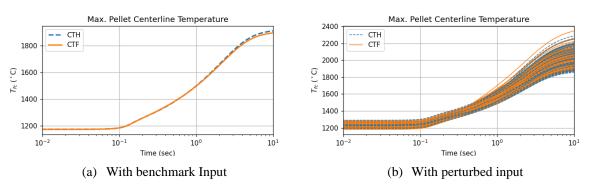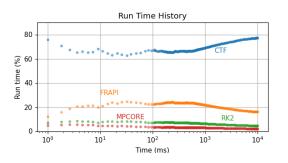(a) With benchmark Input   (b) With perturbed input

Figure 6-4: Peak centerline temperature comparison for single assembly.

The time required by thermal hydraulic module is highest in CTF run model while for CTH run model, most of the time is used by fuel performance code as shown in Figure 6-5.
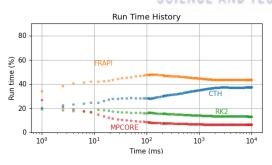
Figure 6-5: Comparison of modules computation time

The results obtained by the present study dictate that using sub-channel code as thermal hydraulics component in multi-physics is more realistic compared to single-channel code. Although, single-channel TH code takes much lesser computation time in a single iteration as well as lesser total number of iterations for overall convergence, however, results are more conservative in nature and likelihood of crossing the safety parameters is higher. In contrast, Sub channel TH code takes longer computation time, but results are more realistic. This comes at the expense of increased number of iterations, perhaps due to slower convergence caused by the sub-channel coolant mixing, as in this case, an increase in power of any rod affects more than just the adjacent channels. Shifting from the conservative approach to BEPU approach requires deep understanding of all the multi-physics processes that interact with each other. Even though adding more modules increases the computational time but the results obtained are in close agreement to the benchmark results.

6.2   3*3 Mini core problem

VERA cycle 1 full core central 3*3 assemblies are taken for verification, performance, and UQ studies of MPCORE. Central assembly has the D bank control rod, so the problem can be modeled using quarter symmetry. D bank withdrawal is considered from 20 steps to 56 steps in 0.1 sec to induce prompt reactivity in the system.

6.2.1   Modeling

Latin Hypercube Sampling (LHS) is a statistical sampling method used to generate a set of samples from a multidimensional distribution. It is particularly useful in situations where the number of parameters or variables to be sampled is large and where it is difficult or impossible to explore the entire range of possible values. In LHS, the range of values for each parameter is divided into equally spaced intervals, and a random value is selected from each interval. The selected values are then arranged in such a way that they form a Latin square, which ensures that no two samples have the same values for any pair of parameters. This reduces the correlation between variables and makes the resulting sample set more representative of the distribution being sampled.

Latin Hypercube sampling is used for input perturbation in min core problem. 124 cases with static and dynamic case are executed with varying inputs following normal distribution. The range of parameters

with normal distribution is provided to experimental design package of python (pyDOE) that generates the required number of samples. The samples are popped out on each next request if the set is not empty. In each case, each fuel pin has the same pin diameter and clad radius. Power and inlet temperature values are also sampled according to normal LHS method. Perturbation variance considered for uncertainty quantification in input data are taken from [60]. RAST-K, in this case as well, uses the same cross-sections for all the cases so cross section co-variance effect is not analyzed in this study. Single fuel assembly study was focused on the difference in 1D and subchannel TH code, while this example is dedicated towards the difference in dynamic and static gap width.

Table 9: Perturbation input values mini core

Parameter	Mean Value	Perturbation (σ)
Power (MW)	159E-6	± 0.02/3
Inlet temperature (ºC)	291.5	± 2/3
Pressure (bar)	155.132	± 0.02/3
Mass flow (kg/s)	765.0	± 0.02/3
Pellet diameter (cm)	0.8192	± 0.0013/3
Cladding outer diameter (cm)	0.9500	± 0.0040/3
Cladding inner diameter (cm)	0.8360	± 0.0040/3
Length of plenum (cm)	19.340	± 1.14/3
Pin Pressure (MPa)	2.0	± 0.07/3

Different cross-sections for spacer grid are assumed during the cross-section generation in STREAM. For this work, RAST-K employs 55 axial meshes while FRAPTRAN and CTH/CTF employ 24 axial meshes. Linear power for 24 axial meshes of FRAPTRAN and CTF is obtained by interpolating the data obtained from RAST-K. For the FRAPTRAN model, 5 radial meshes per fuel pellet and 3 radial meshes per cladding have been used.

6.2.2 Results

For mini core 3x3 case, the neutronics and thermal hydraulics have been fixed to RAST-K and CTF respectively. The only difference is the dynamic/static gap conductance in FRAPTRAN. The results obtained are shown in Figure 6-6 that shows the opposite trend as compared to single assembly case. Coolant temperature is higher in case of static gap conductance. This is because static gap conductance stays higher than the maximum dynamic gap conductance till around 1 sec. the standard deviation obtained for different parameters is shown in Table 10 and Figure 6-7. The area covered by the dynamic gap conductance is higher as the changed in fuel rod perturbation effect has clearly been represented by it. Static gap conductance stays the same regardless of the gap width taken.
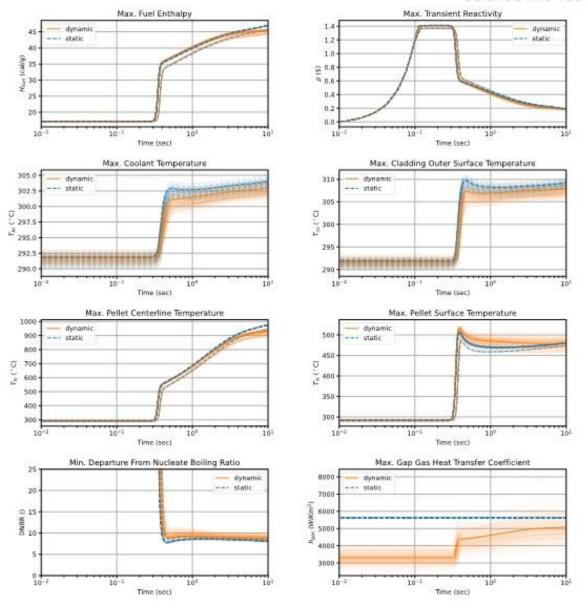
Figure 6-6: Parameters comparison for UQ cases 3x3 mini core

Table 10: Bias of dynamic ('d') and static ('s') gap HTC models compared to STD of multi-physics data peak values

Var	Units	μ_d-μ_s	σ_s	σ_d
ρ	pcm	-255	1847	2016
Q_l	kW/m	-108	735	778
T_{fc}	^{o} C	-44	13	16
T_{fs}	^{o} C	8	9	11
T_{ci}	^{o} C	-4	2	2
T_{co}	^{o} C	-2	1	1

T_{bc}	^{o} C	-1	1	1
DNBR	%	0.92	0.25	0.45

The distribution obtained by dynamic gap conductance shows that the range of safety measures has increased significantly.
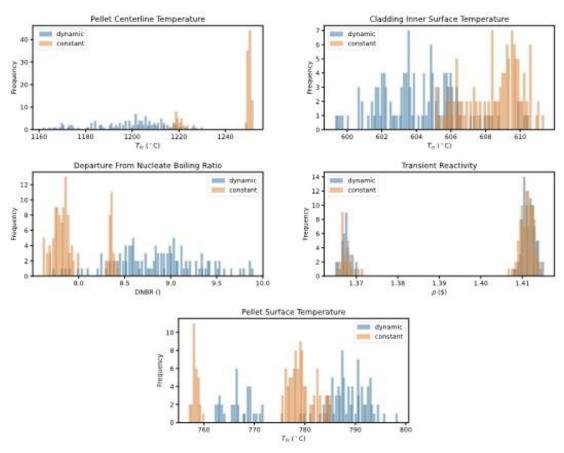

Figure 6-7: Distribution of parameters for staic and dynamic gap conductance models.

7 Discussion

7.1 Effect of coupling FP module

First the NK-TH coupled code is analyzed with the NK-TH-FP coupled code system. The inclusion of FP code gives burnup dependent gap conductance and thus gives good approximation for effective doppler temperature. Burnup dependent fuel and clad thermal conductivity is developed in CTF, but the absence of burnup dependent gap conductivity needs to be modeled correctly by use of lookup tables, AI modules or FP module. The fuel conductivity is validated with FRAPCON but for the transient, the CTF pseudo transient module also predicted higher fuel doppler temperature thus keeping the power peak low.

7.2 Effect of crossflow on transient calculations

Crossflow effect is compared for single fuel assembly with perturbed input (section 6.1), HZP REA BOC(section 5.3.2) and HFP REA BOC (section 5.3.3) case. In the assembly case, CTH1D results were compared with the CTF subchannel code results. The difference between the two models for single assembly was much higher but that difference was not only due to the cross flow. CTH1D codes solve single fluid mass and energy conservation equations. CTF solves 8 conservation equations for mass, energy, and momentum conservation for two fluid three field. Thus, that difference has really emphasized the use of crossflow model in transient cases.

But later in whole core transient cases, it was observed that crossflow does not make results much different. In whole core cases, CTF is used as 1D code (by reducing the channel gap area) and subchannel code. The difference between both models is observed but not as significant as in single assembly case. It was realized that crossflow option in CTF could give good estimates and increase the safety margin of some critical parameters. But it could also decrease the safety margin like the increase in fuel enthalpy case. As the current work focuses on the development of the best estimate code for steady state and transient analysis, in future simulations only crossflow enabled simulations are executed with CTF.

The difference of crossflow option should not be calculated by taking the difference between subchannel code and simple thermal hydraulic 1D codes. 1D codes do not solve the momentum conservation equations and take simplified heat correlations. Subchannel code, on the other hand, not only takes the crossflow but also solves the conservation equation of momentum considering the pressure difference in axial and radial channels.

7.3 Effect of dynamic gap conductance on steady state and transient calculations

Dynamic gap conductance difference is observed in all the cases, and it has been realized that with the burnup, the requirement for dynamic gap conductance increases. Considering the same gap conductance models with the depleted fuel give greater safety margin, however, considered conservative. The

conservative models deal with a single physics phenomenon and modeling one physics conservatively always gives risk on the other physics phenomena. Considering static gap conductivity of small value gives low safety margin for fuel centerline temperature and higher safety margin for DNBR. High static gap conductivity gives a higher safety margin for fuel centerline temperature and low safety margin for DNBR. Thus, there is not a single value that can be used for conservative modeling of all the parameters.

8   Conclusion and Future work

Generalized architecture for the coupling of different physics modules has been developed and tested for depletion and transient calculations. Modules interaction with MPCORE for data transfer, mesh overlay, time step update and global solution convergence has made the modules independent from each other. Replacing 1D TH code with subchannel TH code requires no change for NK and FP module/interface update. Thus, MPCORE can serve as a good coupling tool that can relate different physics phenomenon together, yet, keeping them separate from one another. Depletion studies have revealed the fact that with burnup the gap and fuel conductance change. They usually change in the opposite direction, meaning gap conductance increases while fuel conduction decreases with burnup. Therefore, in simplified fuel conduction solvers burnup dependent fuel conductivity correlations are not used. In current study, it has been shown that two-way modeling of FP and TH uses burnup dependent correlations for coolant calculations and results in different safety limits. Transient studies also reveal that partial rod ejection accidents with best estimate codes lower the safety criteria as proposed by the conservative approaches. Thus, conservative approaches used may give conservative results for some parameters but for overall safety limits calculations, best estimate codes should be used.

However, there are some improvements that are proposed based on this research and future developers may think more than the proposed changes.

8.1   Pin by pin temperature feedback

FP and TH module in the study give feedback on pin level while node average parameters are transferred to NK code. NK code does nodal calculation followed by pin power reconstruction to calculate the linear power at each axial level. Pin power construction, however, depends on the form functions generated by the lattice physics code. Single temperature on node level is considered and cross sections are read at the average fuel and moderator temperature. This modeling approach works fine for the depletion calculations and has been used in industrial practice. In transient calculations, however, the difference between transport and nodal codes is much higher than dynamic/static gap transfer or 1D/subchannel code results. So, in future work, pin-by-pin feedback is proposed to be used so that the immediate increase in fuel and moderator temperature for the peak fuel pin should be feedback for doppler and thermal feedback else the effect is averaged out and results in delay of the feedback.

8.2   Coupling with high performance NK codes

MPCORE framework, as already described, was created with the aim to add Multiphysics modules in all the neutronics modules owned by CORE lab, UNIST. Lattice transport code, STREAM, and Monte Carlo code, MCS, has been developed in CORE lab and transient capability is present in both the codes. Running time for high performance code with inbuilt TH1D code is very high so they are not yet

integrated with MPCORE. After fast computation of these codes is completed, they can be coupled with MPCORE framework.

8.3 Machine learning driven FP and TH modules

Multiphysics codes are required for the BEPU approach that is an alternative to conservative approach and accepted by the regulatory commissions. The developed model used nodal code but the FP and subchannel TH code used most of the computation time. An alternative machine learning model is proposed for dynamic gap heat conductance. Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN) trained models can be used that can model the time and spatial dependent behaviors. Validated machine learning trained models for the reactor system can be used for the UQ studies by coupling them with MPCORE framework.

8.4 Convergence algorithm options

Damped Picard and Anderson acceleration algorithm are included in the convergence study. The Jacobian free Newton Krylov method or other advanced convergence algorithm can be used in future. Partial convergence of component modules is also proposed to work fine for the Multiphysics coupling studies [61].

9   References

1. Agency, N.E. *Expert Group on Multi-physics Experimental Data, Benchmarks, and Validation (EGMPEBV)*. 2017-12-18   2022-08-11]; Available from: https://www.oecd-nea.org/science/egmpebv/mpcmiv/.

2. Demazière, C., *Nuclear Reactor System Modelling: Past, Present, and Future*, in *Nordic-Gen4 Seminar*. 2012: Risø, Denmark.

3. Basualdo, J.R., *Development of a Coupled Neutronics/ThermalHydraulics/Fuel Thermo-Mechanics Multiphysics Tool for Best-Estimate PWR Core Simulations*, in *Mechanical Engineering*. 2020: Karlsruhe Institute of Technology.

4. Targa, A., *Development of multi-physics and multi-scale Best Effort Modelling of pressurized water reactor under accidental situations*. 2017, Université Paris-Saclay.

5. Delipei, G.K., et al., *High to Low pellet cladding gap heat transfer modeling methodology in an uncertainty quantification framework for a PWR Rod Ejection Accident with best estimate coupling*. EPJ Nuclear Sciences & Technologies, 2020. **6**(56).

6. Stimpson, S., et al., *Coupled fuel performance calculations in VERA and demonstration on Watts Bar unit 1, cycle 1*. Annals of Nuclear Energy, 2020. **145**(107554).

7. Kochunas, B., et al., *VERA Core Simulator Methodology for Pressurized Water Reactor Cycle Depletion*. Nuclear Science and Engineering, 2017. **185**(1): p. 217-231.

8. Tuominen, R. and V. Valtavirta, *BEAVRS pin-by-pin calculations with Ants-SUBCHANFLOW-SuperFINIX code system*. Annals of Nuclear Energy, 2023. **180**.

9. Yu, J., et al., *MCS based neutronics/thermal-hydraulics/fuel-performance coupling with CTF and FRAPCON*. Computer Physics Communications, 2019. **238**: p. 1-18.

10. Xu, X., et al., *Neutronics/thermal-hydraulics/fuel-performance coupling for light water reactors and its application to accident tolerant fuel*. Annals of Nuclear Energy, 2022. **166**(108809).

11. García, M., et al., *A Serpent2-SUBCHANFLOW-TRANSURANUS coupling for pin-by-pin depletion calculations in Light Water Reactors*. Annals of Nuclear Energy, 2020. **139**(107213).

12. Ma, Y., et al., *RMC/CTF multiphysics solutions to VERA core physics benchmark problem 9*. Annals of Nuclear Energy, 2019. **133**: p. 837-852.

13. Delipei, G.K., et al., *CTF-PARCS Core Multi-Physics Computational Framework for Efficient LWR Steady-State, Depletion and Transient Uncertainty Quantification*. Energies, 2022. **15**(14): p. 5226.

14. Perin, Y. and K. Velkov, *CTF/DYN3D multi-scale coupled simulation of a rod ejection transient on the NURESIM platform*. Nuclear Engineering and Technology, 2017. **49**: p. 1339-1345.

15. Elsawi, M.A. and A.S.B. Hraiz, *Benchmarking of the WIMS9/PARCS/TRACE code system for neutronic calculations of the Westinghouse AP1000™ reactor.* Nuclear Engineering and Design, 2015. **293**: p. 249-257.

16. Safavi, A., et al., *Application of a new neutronics/thermal-hydraulics coupled code for steady state analysis of light water reactors.* Nuclear Engineering and Technology, 2020. **52**(8): p. 1603-1610.

17. Bennett, A., M. Avramova, and K. Ivanov, *Coupled MCNP6/CTF code: Development, testing, and application.* Annals of Nuclear Energy, 2016. **96**: p. 1-11.

18. Ferraro, D.E., *Monte Carlo-based multi-physics analysis for transients in Light Water Reactors.* 2021: Karlsruher Institut für Technologie (KIT).

19. Romano, P.K., et al., *A Code-Agnostic Driver Application for Coupled Neutronics and Thermal-Hydraulic Simulations.* Nuclear Science and Engineering, 2021. **195**(4): p. 391-411.

20. Vazquez, M., et al., *Coupled neutronics thermal-hydraulics analysis using Monte Carlo and sub-channel codes.* Nuclear Engineering and Design, 2012. **250**: p. 403-411.

21. Jung, Y.S., et al., *Practical numerical reactor employing direct whole core neutron transport and subchannel thermal/hydraulic solvers.* Annals of Nuclear Energy, 2013. **62**: p. 357-374.

22. Liu, Z., et al., *An internal parallel coupling method based on NECP-X and CTF and analysis of the impact of thermal-hydraulic model to the high-fidelity calculations.* Annals of Nuclear Energy, 2020. **146**(107645).

23. Park, J.K., *Coupling between subchannel module of CUPID and fuel code FINIX for multi-physics transient analysis of PWR core*, in *Department of Energy System Engineering*. 2021, Seoul National University.

24. Davies, S., et al., *CTF and FLOCAL Thermal Hydraulics Validations and Verifications within a Multiscale and Multiphysics Software Development.* Energies, 2021. **14**.

25. Bielen, A.S., *Sensitivity and Uncertainty Analysis of Multiphysics Nuclear Reactor Core Depletion*, in *Nuclear Engineering and Radiological Sciences*. 2015, University of Michigan.

26. Kim, H. and Y. Kim, *Unstructured Mesh–Based Neutronics and Thermomechanics Coupled Steady-State Analysis on Advanced Three-Dimensional Fuel Elements with Monte Carlo Code iMC.* Nuclear Science and Engineering, 2021. **195**(5).

27. Turner, J.A., et al., *The Virtual Environment for Reactor Applications (VERA): Design and architecture.* Journal of Computational Physics, 2016. **326**: p. 544-568.

28. Choe, J., et al., *Verification and validation of STREAM/RAST-K for PWR analysis.* Nuclear Engineering and Technology, 2018.

29. Jang, J., et al., *Analysis of Rostov-II Benchmark Using Conventional Two-Step Code Systems.* Energies, 2022. **15**(9).

30. Choi, S., et al., *Impact of inflow transport approximation on light water reactor analysis.* Journal of Computational Physics, 2015. **299**: p. 352-373.

31. Choi, S., C. Lee, and D. Lee, *Resonance treatment using pin-based pointwise energy slowing-down method.* Journal of Computational Physics, 2017. **330**: p. 134-155.

32. Choi, S., et al., *On the diffusion coefficient calculation in two-step light water reactor core analysis.* Journal of Nuclear Science and Technology, 2017. **54**(6): p. 705-715.

33. Zheng, Y., S. Choi, and D. Lee, *A new approach to three-dimensional neutron transport solution based on the method of characteristics and linear axial approximation.* Journal of Computational Physics, 2017. **350**: p. 25-44.

34. Park, J., et al., *Development Status of Nodal Diffusion Code RAST-K v2.2*, in *Transactions of the Korean Nuclear Society Spring Meeting*. 2019: Jeju, Korea.

35. Hu, H., et al. *Boiling and quenching heat transfer advancement by nanoscale surface modification.* Scientific reports, 2017. **7**, 6117 DOI: 10.1038/s41598-017-06050-0.

36. Salko, R.K. and M.N. Avramova, *CTF Pre-processor User's Manual*. 2016, The Pennsylvania State University.

37. Kucukboyaci, V., y. sung, and R.K. Salko, *Cobra-TF Parallelization and Application to PWR Reactor Core Subchannel DNB Analysis*, in *ANS MC2015 - Joint International Conference on Mathematics and Computation (M&C), Supercomputing in Nuclear Applications (SNA) and the Monte Carlo (MC) Method*. 2015.

38. Salko, R.K., 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, 2015. **84**: p. 122-130.

39. Cherezov, A., et al., *Fuel Rod Analysis Programming Interface for Loosely Coupled Multiphysics System*, in *In Proceedings of the American Nuclear Society Annual Meeting*. 2020: Chicago, IL, USA. p. 1331-1334.

40. Geelhood, K., et al., *FRAPCON-4.0: A Computer Code for the Calculation of Steady-State, Thermal-Mechanical Behavior of Oxide Fuel Rods for High Burnup*. 2015, Pacific Northwest National Laboratory.

41. Geelhood, K., et al., *FRAPTRAN-2.0: A Computer Code for the Transient Analysis of Oxide Fuel Rods*. 2016, Pacific Northwest National Laboratory.

42. Stimpson, S., et al., *Standalone BISON Fuel Performance Results for Watts Bar Unit 1, Cycles 1-3*. 2016.

43. Magedanz, J., et al., *High-fidelity multi-physics system TORT-TD/CTF/FRAPTRAN for light water reactor analysis.* Annals of Nuclear Energy, 2015. **84**: p. 234-243.

44. Yilmaz, M.O. and M. Avramova, *Development of Burnup Dependent Fuel Rod Model in CTF*, in *NURETH-16*. 2015: Chicago, IL.

45. Davis, I., et al., *High-fidelity multi-physics coupling for determination of hydride distribution in Zr-4 cladding.* Annals of Nuclear Energy, 2017. **110**: p. 475-485.

46. Courty, O., A.T. Motta, and J.D. Hales, *Modeling and simulation of hydrogen behavior in Zircaloy-4 fuel cladding.* Journal of Nuclear Materials, 2014. **452**(1-3): p. 311-320.

47. Suman, S., *Impact of hydrogen on rupture behaviour of Zircaloy-4 nuclear fuel cladding during loss-of-coolant accident: a novel observation of failure at multiple locations.* Nuclear Engineering and Technology, 2021. **53**(2): p. 474-483.

48. Stimpson, S., et al., *Pellet-clad mechanical interaction screening using VERA applied to Watts Bar Unit 1, Cycles 1–3.* Nuclear Engineering and Design, 2018. **327**: p. 172-186.

49. Novascone, S.R., et al., *Evaluation of coupling approaches for thermomechanical simulations.* Nuclear Engineering and Design, 2015. **295**: p. 910-921.

50. Cherezov, A., et al., *A Multi-Physics Adaptive Time Step Coupling Algorithm for Light-Water Reactor Core Transient and Accident Simulation.* Energies, 2020. **13**(23).

51. Radaideh, M.I. and T. Kozlowski, *Combining simulations and data with deep learning and uncertainty quantification for advanced energy modeling.* International Journal of Energy Research, 2019. **43**(14): p. 7866-7890.

52. Roostaii, B., H. Kazeminejad, and S. Khakshournia, *Including high burnup structure effect in the UO2 fuel thermal conductivity model.* Progress in Nuclear Energy, 2021. **131**(103561).

53. Godfrey, A.T., *VERA core physics benchmark progression problem specifications.* 2014, Oak Ridge National Laboratory: Oak Ridge.

54. Toptan, A., R.K. Salko, and M.N. Avramova. *Review of CTF's Fuel Rod Modeling Using FRAPCON-4.0's Centerline Temperature Predictions*. in *American Nuclear Society*. 2017. San Francisco, California.

55. Nguyen, T.D.C., et al., *MCS/TH1D analysis of VERA whole-core multi-cycle depletion problems.* Annals of Nuclear Energy, 2020. **139**.

56. *Topical Report on Reactivity Initiated Accident: Bases for RIA fuel and Core Coolability Criteria*. 2002, EPRI: Palo Alto, CA.

57. Downar, T., et al., *Transient Methods in MPACT with Internal TH*. 2016.

58. Hursin, M., T.J. Downar, and B. Kochunas, *Analysis of the Core Power Response during a PWR Rod Ejection Transient Using the PARCS Nodal Code and the DeCART MOC Code.* Nuclear Science and Engineering, 2012. **170**(2): p. 151-167.

59. Hursin, M., T.J. Downar, and R. Montgomery, *Impact of improved neutronic methodology on the cladding response during a PWR reactivity initiated accident.* Nuclear Engineering and Design, 2013. **262**: p. 180-188.

60. Lee, J. and Y.-S. Bang, *Factors Affecting Fuel Rod Burst in LOCA Safety Analysis*, in *Transactions of the Korean Nuclear Society Autumn Meeting*. 2018.

61.   Senecal, J.P. and W. Ji, *Approaches for Mitigating Over-Solving in Multiphysics Simulations.* International Journal for Numerical Methods in Engineering, 2017. **112**: p. 503-528.

10 Acknowledgement