*M&C 2017 - International Conference on Mathematics & Computational Methods Applied to Nuclear Science & Engineering, Jeju, Korea, April 16-20, 2017, on USB (2017)*

# MIMIC: Michigan Interface for Multi-state In-memory Coupling with STAR-CCM+ for Nuclear Applications

D. J. Walter, V. Petrov, N. Adamowicz, A. Manera

*Nuclear Engineering and Radiological Sciences Department, University of Michigan*
*2355 Bonisteel Blvd., Ann Arbor, MI 48109*
*djwalter@umich.edu*

**Abstract** – *The high computational power available for nuclear reactor analysis has allowed the coupling of high fidelity physics, where sub-pin level resolution of neutronics, fuel structure behavior, fluid chemistry, heat transfer, and turbulent fluid flow is realized. The primary objective of such efforts includes using the simulation and modeling capabilities to improve the operation and future design of nuclear reactors. Specifically, the elimination of fuel rod failures is a principal focus of the nuclear reactor research and analysis community. MIMIC is a computational tool that seeks to provide a framework for single pin cell and sub-assembly analysis with computational fluid dynamics at its foundation. The turbulent fluid flow—resulting from spacer grid mixing vanes—and the conjugate transfer of heat from the fuel rod to the fluid has been shown to be the primary driver for several in-core fuel rod phenomena, including CRUD deposition and cladding hydriding. Therefore, the robust and versatile software STAR-CCM+ is coupled with the direct-solution neutronics solver MPACT, the coolant and CRUD chemistry solver MAMBA, and the fuel performance solver BISON. The unique challenges associated with in-memory coupling to STAR-CCM+ are discussed and addressed, which should offer valuable insights that will support other similar coupling efforts with STAR-CCM+.*

## I. INTRODUCTION

Computational fluid dynamics (CFD) has increasingly become a useful tool in high fidelity nuclear reactor physics analysis. STAR-CCM+ is a commercial CFD tool that provides robust fluid flow predictions, as well as heat transfer solutions. And, its popularity within the nuclear reactor analysis community is growing rapidly. The present work focuses on:

1) the coupling of various physics to enable high fidelity fuel rod analysis,
2) and the unique coupling aspects associated with the closed source STAR-CCM+ software.

Of course, there are other CFD tools available, such as the ANSYS distributions of Fluent and CFX, and the open source tool OPENFOAM, which are not included in the scope of this work.

The exchange of energy at the solid/fluid interface is known as conjugate heat transfer (CHT). On pressurized water reactor (PWR) fuel rods, heat transfer between the heat-generating fuel rods and the working fluid, which is light water, is of high importance from a safety and efficiency standpoint.

MIMIC is a computational tool that facilitates in-memory coupling with STAR-CCM+ by leveraging the STAR-CCM+ Java API and so-called *user code* functionality. It is specifically designed for nuclear fuel rod array analysis, but may be used for other applications. Currently, MIMIC supports in-memory two-way coupling with the direct-solution neutronics suite MPACT [1] and the Chalk River Unidentified Deposit (CRUD) deposition solvers MAMBA-3D and -1D [2,3]. A file-based, one-way coupling with the fuel performance code BISON [4] is also supported.

MIMIC coordinates the coupled cycle simulation, where multiple STAR-CCM+ states may be converged throughout a reactor cycle simulation. MIMIC adopts the operator-split coupling approach, which treats each unique physics code independently and couples them through their boundary conditions.

Moreover, MIMIC gives the user control over the temporal coupling as well as coupled physics convergence through the use of fixed-point iteration. User input parameters give control over which physics are coupled, and whether their coupling is one- or two-way; this allows convenient sensitivity studies and analyses. In addition, coupled physics software can be run simultaneously with MIMIC, or instead, pre-generated physics values can be read in through file-based coupling.

MIMIC is capable of surface- and volume-based solution transfer between non-conformal meshes. Moreover, for coupling with MPACT, both STAR-CCM+ and MPACT may be uniquely spatially decomposed for efficient parallel execution. MIMIC handles all of the processor communication and solution transfer.

## II. MOTIVATION

Fuel rod phenomena, such as corrosion product deposition, cladding oxidation, and cladding hydriding are strongly influenced by the local temperature and heat flux distributions on the cladding surface. Recently, there have been several advances in the development of computational frameworks for modeling these phenomena [5,6,7,8]. CFD

provides a high fidelity tool that can resolve these thermal hydraulic distributions in the complex fluid flow regions downstream of spacer grids and mixing vanes.

Because CFD simulations of this scale and fidelity are inherently computationally expensive, MIMIC supports in-memory coupling. This allows STAR-CCM+ models, which may contain upwards of hundreds of millions of finite volume cells, to efficiently perform solution transfers to another physics domain and solver. For example, a 5x5 fuel rod array model with a hexahedral mesh, which fully resolves three spacer grids with mixing vanes, has approximately 145 million cells [8]. To couple such a model with a CRUD deposition or neutronics solver requires the solution fidelity to be reduced by 1-2 orders of magnitude. Out of memory coupling of a CFD simulation of this magnitude would require significant time spent on I/O operations.

## III. METHODOLOGY

### 1. High Fidelity Fuel Rod Analysis

There are a handful of physics-based parameters that are relevant to fuel rod analysis, including—but not limited to—power density, temperature, isotopic composition, stress/strain, and neutron flux. Recently, the research community has shown great interest in fuel rod phenomena that require high fidelity modeling and simulation. Often accurate predictions are obtained only through consideration of the relevant parameters' coupled interaction.

For example, the fuel rod cladding temperature and heat flux distributions are strongly influenced by both the fuel rod power density distribution and the fluid flow turbulence distribution. As a result, MIMIC has been built on top of a CFD tool, and it provides an interface for extracting thermal- and hydraulic-related parameters from the CFD solution, so that other physics tools can improve their solutions by using the CFD-computed parameters.

In this section, a physics operator approach will be taken to elucidate the input and output parameter relationships between respective physics domains used in MIMIC. Each physics domain is assigned a unique operator which maps the inputs required by that domain through the solution process to the outputs produced. All aspects of non-conformal mesh mapping are ignored in the simplified description presented here. As such, it is assumed that all physics values are specific to a single volumetric or surface mesh cell and that mapping of inputs and outputs directly transfer to or from the same spatial mesh cell in other physics domains. All values described are scalar quantities for a single mesh cell, except the nuclide concentration $N$ and neutron interaction cross sections, $\sigma$, which denote vector quantities specific to a mesh cell.

#### *A. Power density distribution*

The calculation of the power density distribution is performed using the neutronics solver by predicting the distribution of the neutron flux. The neutronics operator is denoted $\mathcal{N}$ and represents the solution to the Boltzmann neutron transport equation. Inputs to $\mathcal{N}$ include the nuclide concentrations, $N$, nuclide microscopic cross sections, $\sigma$, nuclide temperature, $T$, and nuclide (coolant) density, $\rho_{cool}$. Outputs include the neutron scalar flux, $\phi$, and critical coolant boron concentration, $N_B^{cool}$:

$$(\phi, N_B^{cool}) = \mathcal{N}(N, \sigma, T, \rho_{cool}). \quad (1)$$

Within the inputs to the neutronics operator, the cross section values, $\sigma$, are temperature dependent quantities due to Doppler broadening.

Power density is extracted from the neutronics solution using the power operator $\mathcal{P}$ which scales the calculated neutron flux distribution, $\phi$, to a volumetric heat generation rate $q'''$:

$$q''' = \mathcal{P}(\phi). \quad (2)$$

In order to obtain accurate power density distributions during the depletion cycle, the transmutation of isotopes must be considered. Nuclide depletion is represented by the physics operator $\mathcal{D}$, and the inputs include the neutron scalar flux, $\phi$, neutron interaction cross sections, $\sigma$, and nuclide densities, $N$. The output of the neutron depletion is a new nuclide density, $N^{updated}$, which is calculated from the Bateman equations supplemented with terms for nuclide production and destruction from neutron interactions:

$$N^{updated} = \mathcal{D}(\phi, \sigma, N). \quad (3)$$

In practice, the coupling of the neutron flux and the nuclide concentration changes due to depletion is realized via a predictor-corrector algorithm. For each depletion time step, two steady state neutron flux calculations and two time-dependent nuclide depletion calculations are performed, which allow increased time step lengths without sacrificing accuracy.

#### *B. Temperature and heat flux*

The CFD portion of the multiphysics coupling in MIMIC is composed of two physics operators. The fluid operator, $\mathcal{F}$, denotes the solution to the Navier-Stokes equations in the reactor pin subchannels. The solution to the heat equation is satisfied using the heat transfer operator, $\mathcal{H}$. When combined, these operators specify the solution to the conjugate heat transfer problem. Input to $\mathcal{F}$ is solely comprised of surface heat flux $q''$ exiting the cladding

surface. Outputs of $\mathcal{F}$ include the coolant temperature, $T_{cool}$, coolant density, $\rho_{cool}$, and turbulent kinetic energy $k_{CFD}$:

$$(T_{cool}, \rho_{cool}, k_{CFD}) = \mathcal{F}(q''). \quad (4)$$

The operator $\mathcal{H}$ takes as input the fuel power density, $q'''$, coolant temperature, $T_{cool}$, and a thermal resistance $\Gamma$ (e.g. from CRUD or oxidation on the water-side cladding surface). The heat transfer operator produces solid temperature, $T_{solid}$, as well as the outer cladding heat flux, $q''$:

$$(T_{solid}, q'') = \mathcal{H}(q''', T_{cool}, \Gamma). \quad (5)$$

*C. CRUD and oxide thermal resistance*

The CRUD chemistry and physics are represented by the operator $\mathcal{C}$. This operator takes inputs of cladding heat flux, CRUD/coolant interface temperature, TKE, coolant boron concentration and neutron reaction rate of boron-10, $R_{B10}$. In turn, it produces CRUD/oxide thermal resistance and CRUD nuclide composition $N_{CRUD}$:

$$(N_{CRUD}, \Gamma) = \mathcal{C}\left(q'', T_{clad/cool}, k_{CFD}, N_B^{cool}, R_{B10}\right). \quad (6)$$

*D. Cladding hydriding*

The cladding hydriding physics is represented by the operator $\mathcal{W}$. This operator takes the cladding surface temperature as input and outputs the stress/strain of the cladding, S:

$$S = \mathcal{W}\left(T_{clad/cool}\right). \quad (7)$$

In the current implementation, the stress/strain of the cladding is only used within BISON, and no additional information is provided to STAR-CCM+. Therefore, only a one-way coupling, STAR-CCM+ to BISON, exists.

## 2. STAR-CCM+ Coupling

Because STAR-CCM+ is closed source, the in-memory coupling approach is unique to the functionality that STAR-CCM+ provides to the user. Consequently, MIMIC leverages two functionalities provided by STAR-CCM+:

1) a Java API, which gives the user complete control of the STAR-CCM+ software from Java source code,
2) and user code, which gives the user access to STAR-CCM+ solution data stored in memory.

Fig. 1 illustrates the various components of MIMIC, including STAR-CCM+, Java driver package, user code, and a coupled input file. User code allows a high-level programming language, such as C/C++ or Fortran, to be compiled as a shared object library and linked to at runtime. The MIMIC user code is currently written in Fortran 90. All coupled codes must be compiled as static libraries and are linked within this shared object library.

Although user code provides a powerful avenue for access to data stored in memory, it does not provide a means to control the solution state of STAR-CCM+, i.e. the user code API does not include functions to begin the STAR-CCM+ solution iteration. However, the Java API provides complete control of the STAR-CCM+ solution state, model properties, and a method for executing user code functions as needed. Therefore, the union of the Java API and the user code is the essence of MIMIC and its powerful capability for performing multi-state in-memory cycle simulations.

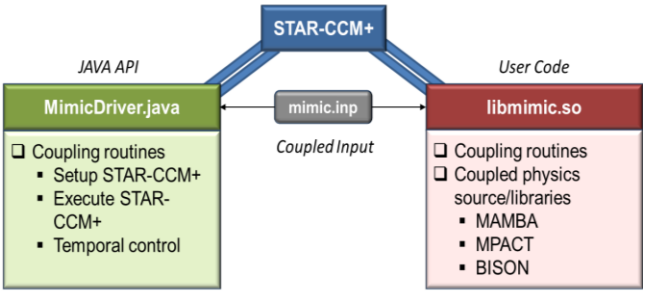


Fig. 1. MIMIC components, including STAR-CCM+, Java driver package, user code, and a coupled input.

Refer to TABLE I in Appendix A for a summary of the most important capabilities of MIMIC.

*A. STAR-CCM+ data structure*

The organization of STAR-CCM+ data in memory is based on the organization of the geometry within the model. A typical three-dimensional STAR-CCM+ model is composed of *regions*, *boundaries*, and *interfaces*. Within regions, the finite volumes are referred to as *cells*; and, on a boundary, the finite surfaces are referred to as *faces*.

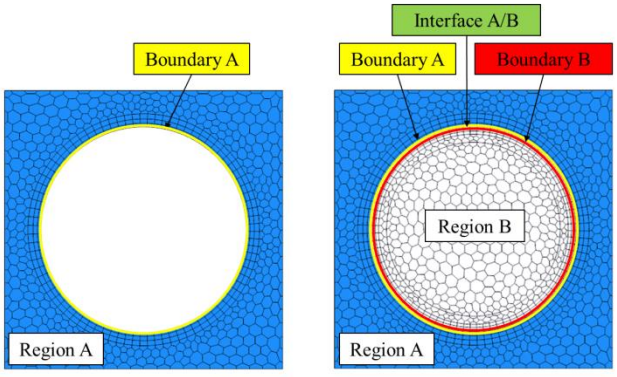


Fig. 2. STAR-CCM+ regions, boundaries, and interfaces.

Consider Fig. 2, which shows two representations of a pin cell model. On the left, the solid fuel rod is not modeled; this will be referred to as Case 1; and, on the right, the fuel rod is modeled, which is referred to as Case 2.

Case 1 represents a fixed heat flux or temperature boundary condition on Boundary A, where Region A represents the flowing coolant. From a data structure standpoint, the data associated with Region A and Boundary A are separate in memory. For example, there is an array containing cell-wise temperatures for Region A. Similarly, for Boundary A, there is another array containing face-wise temperatures.

Case 2 represents a conjugate heat transfer model, where the fuel Region B has a non-zero volumetric heat source, and the heat removal is coupled with the fluid flow in Region A. To model these physics, STAR-CCM+ requires an interface to be created between the different regions, so that the respective physics' data may be exchanged between the boundaries. In Fig. 2 (right), this interface is called Interface A/B and both Boundary A and Boundary B belong to this interface. Moreover, the in-memory data for Boundary A and Boundary B are stored separately. Consider the situation where a thermal resistance is assigned to Interface A/B, the temperature distribution on Boundary A will be different than Boundary B.

The details of how the STAR-CCM+ data are organized in memory are crucial to the design of MIMIC. Specifically, each user code function must be polymorphic in handling either region (cell) or boundary (face) data arrays.

### *B. Executing user code*

In this section, details of executing the user code functions are provided. It is assumed that all user code functions are of the type *ScalarFieldFunction*, which simply return a scalar quantity for each cell/face. Depending on the way that the user code field function is used, the actual result returned by the function may or may not be significant.

An example of a situation where the result *is* significant: when the user code field function is meant to provide the thermal resistance, as computed by MAMBA, for each face on a coupled boundary to STAR-CCM+ so that it may be assigned as a contact resistance on that boundary.

An example of a situation where the result *is not* significant: when the user code field function accesses the temperature of a specific region (via the native field function $Temperature), performs a solution transfer to the MAMBA-3D mesh, and stores the transferred data in memory, i.e. data is not explicitly returned to STAR-CCM+.

In the latter case, the result is not significant and MIMIC simply returns an arbitrary value of 0.5 for all cells/faces to STAR-CCM+. Fig. 3 illustrates the interaction of the user code field functions with the STAR-CCM+ data stored in memory; both situations where the user code returns an insignificant result (evaluated through a report) and where the user code returns significant data (through a physics assignment) are shown.

### *C. Lessons learned*

Unfortunately, documentation for the STAR-CCM+ user code is minimal; and, specific implementation details are even sparser. As a result, several lessons have been learned during years of working with the user code functionality.

*Lesson 1:*

The Java API is critical to the design of a generic STAR-CCM+ based coupling tool. The user code provides direct access to data stored in memory and, therefore, allows efficient numerical calculations using a compiled code. However, it is the Java API that allows flexibility in the execution sequence of STAR-CCM+ and the user code functions.

*Lesson 2:*

So-called *Reports* provide a convenient way to execute user code functions. The Java API should be used in concert to execute the reports to run the user code as needed throughout the execution sequence.

*Lesson 3:*

So-called *halo* (ghost) cells/faces exist during parallel execution, care must be taken of these halo cells/faces when executing user code in parallel. A native field function called $IsHalo exists to identify these halo cells/faces; this field function is not documented within the STAR-CCM+ manual.

*Lesson 4:*

When using user code to assign data to a so-called *Interface*, both boundaries associated with the interface will, by default, call the user code function. This complicates the assignment of data if the interface mesh is non-conformal.

*Lesson 5:*

Use the *–mpi* flag to tell STAR-CCM+ which MPI distribution to use. In the case of MIMIC, the OPENMPI distribution is used to compile all of the user code functions and other coupled codes, e.g. MPACT, MAMBA, etc. Be sure to maintain consistency in the compilers used for all libraries linked to STAR-CCM+.

## 3. MIMIC Execution Sequence

The MIMIC execution sequence is detailed in this section. The coupled physics of CRUD deposition, via MAMBA, and neutronics, via MPACT, are considered. The execution sequence is separated into two parts. The first part, as shown in Fig. 4 (left), sets up the coupled simulation. The second part, as shown in Fig. 4 (right), solves the coupled simulation.

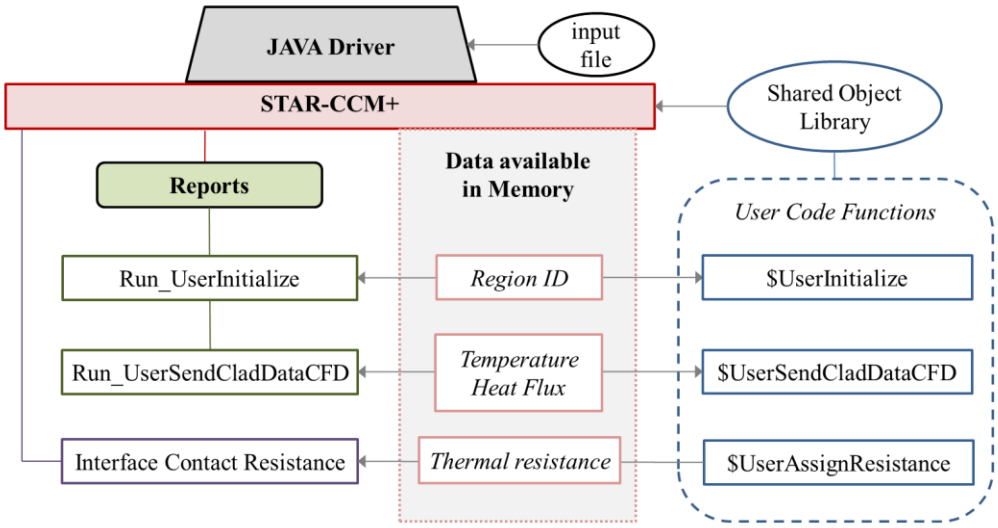


Fig. 3. Interaction of the user code field functions with STAR-CCM+ data stored in memory.

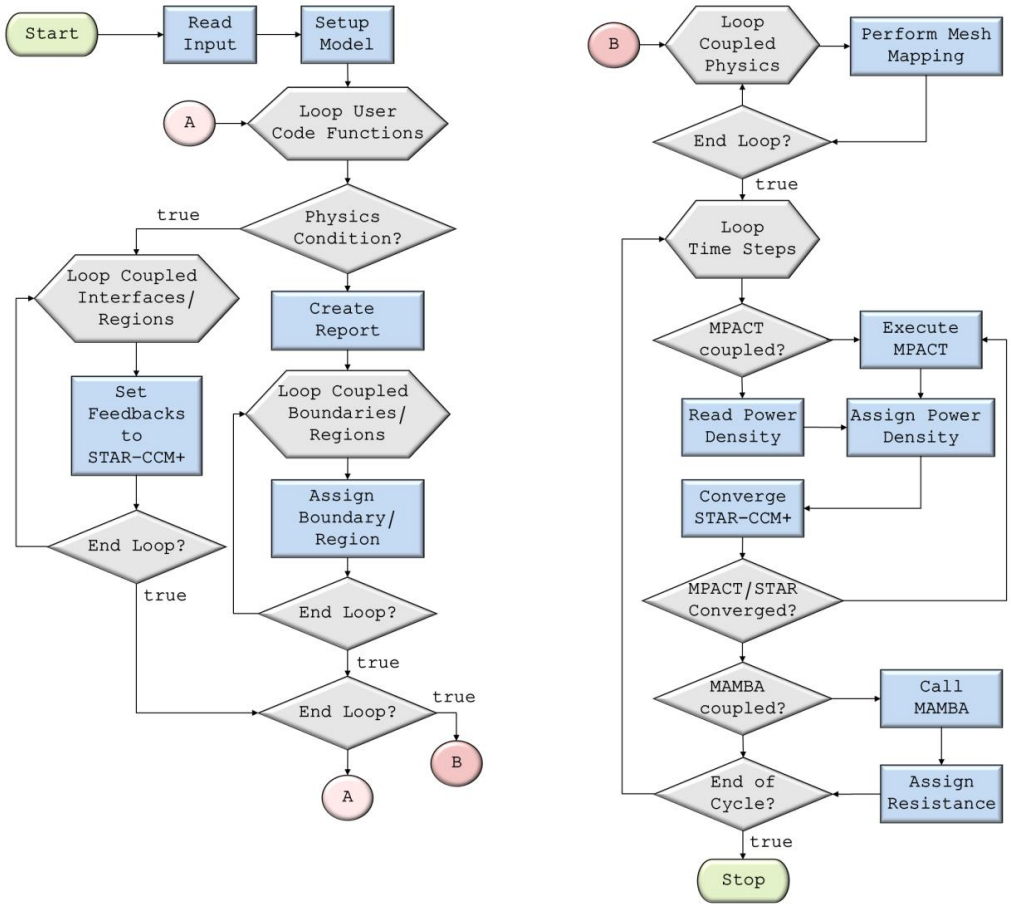


Fig. 4. Execution sequence associated with setting up (left) and solving (right) a coupled CRUD deposition simulation using STAR-CCM+ and MAMBA-3D within MIMIC.

## IV. MULTIPHYSICS DEMONSTRATIONS

Three different physics relevant to nuclear reactor analysis will be used to demonstrate MIMIC's capabilities: neutronics, CRUD deposition, and cladding hydriding.

### 1. Power Density Distribution

The time-dependent, three-dimensional (3-D) power density distribution is crucial to accurate fuel rod analysis. The axial power density distribution within a single fuel rod undergoes significant changes during a depletion cycle, especially for fresh fuel. Whether the axial distribution is flat or peaked in certain location strongly influences the axial distribution of the temperature and heat flux leaving the cladding surface.

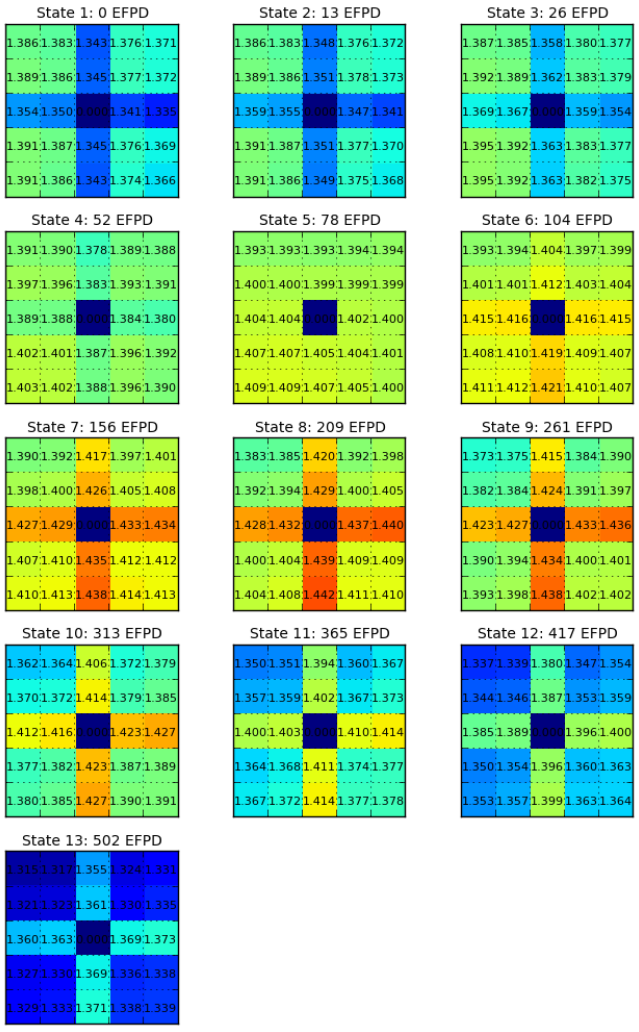


Fig. 5. Relative pin powers of 5x5 sub-assembly predicted by MPACT at discrete depletion steps during simulation of the Seabrook reactor core of cycle 5.

MIMIC facilitates coupling with the direct-solution neutronics solver MPACT. Both in-memory and file-based coupling are supported, which provides flexibility in the type of simulation performed. For example, the CASL core simulator, VERA, simulated several cycles of the Seabrook reactor. This simulation included coupled MPACT and COBRA-TF models. Subsequently, the time-dependent power density distribution computed during that simulation were fed into a MIMC simulation that coupled STAR-CCM+ and MAMBA-3D for a 5x5 pin cell array within a Seabrook assembly. Fig. 5 shows the pin-wise relative power at discrete depletion steps, and Fig. 6 shows the axial power distribution for a select fuel rod as it depletes.

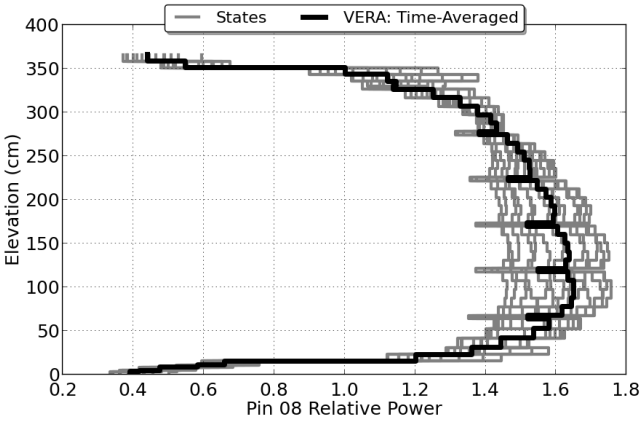


Fig. 6. Relative axial power distribution predicted by VERA for a specific fuel rod within the Seabrook reactor core of cycle 5.

### 2. CRUD-Induced Localized Corrosion

The Seabrook Station Nuclear Power Plant employs a Westinghouse designed PWR with a four loop steam supply system producing a gross electrical power output of 1998 MW. The Seabrook plant was chosen by the Consortium for Advanced Simulation of Light Water Reactors (CASL) as a testbed for the computational simulation of CRUD-induced power shift (CIPS) and CRUD induced localized corrosion (CILC). The Seabrook reactor displayed both of these phenomena in cycle five of operation, which lasted from December 1995 to May 1997. Five fuel rods collectively from four different assemblies failed during December 1996 due to the effects of CILC.

The purpose of the present simulation campaign is to better understand CRUD deposition and oxide formation mechanisms, as well as demonstrate the current state-of-the-art of light water reactor multiphysics analysis. Simulations were partitioned into two separate steps.

First, a quarter core model of the Seabrook plant was simulated using the CASL developed VERA-CS computational toolkit, specifically MPACT, COBRA-TF, and MAMBA-1D. The purpose of this simulation was to

model CIPS and deplete nuclides in the Seabrook core in order to calculate pin power distributions for cycle five. A detailed description of the procedure used for this simulation is given by Ref. 9.

The second phase of the Seabrook CIPS / CILC simulation campaign focused on an assembly, G70, containing multiple failed fuel rods, as shown in Fig. 7 (top). Within G70, a 5x5 rod array is used as the model domain, which contains one failed rod in position 8 as a result of CILC; see Fig. 7 (bottom). MIMIC was used during this analysis to couple STAR-CCM+ to MAMBA-3D and made use of the pin power profiles provided by phase one of the simulation, i.e. axial power density distributions computed by MPACT. Details of the CFD and CRUD simulation parameters used including computational resources employed in this study are available in Ref. 8.

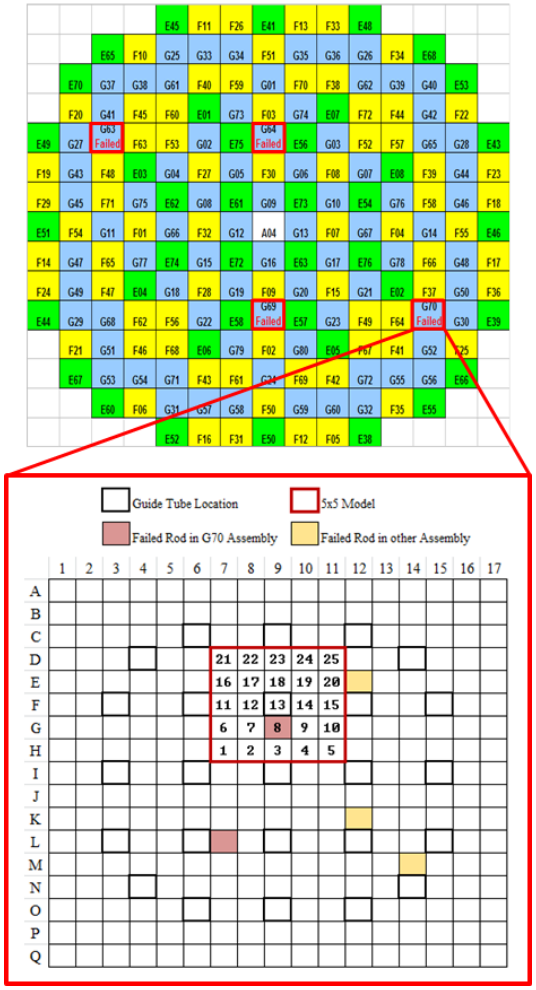


Fig. 7. Seabrook reactor core layout (top) with assembly G70 showing 5x5 model domain (bottom).

Final results of the simulation give excellent agreement with both measurements and visual observations performed on Seabrook cycle five fuel rods; refer to Ref. 10 for additional details. Key among the visual observations was that there was no preferred CRUD orientation with location of thimbles, and a vertical striping or gradual spiraling CRUD deposit pattern moving along the rod axial direction. These observations were replicated in simulation as shown in Fig. 8. Measurements of both non-failed and failed rod oxide thicknesses, including that of G70 rod 8, were performed using eddy current methods in Ref. 10. It should be noted that the eddy current methods employed may not be particularly accurate as these methods are calibrated to zirc oxide standards which do not include CRUD. Nonetheless, Fig. 9 shows excellent agreement in simulated CRUD and oxide thickness compared to values obtained through eddy current methods. These results represent a new state-of-the-art capability for prediction of CRUD and oxide deposition patterns.

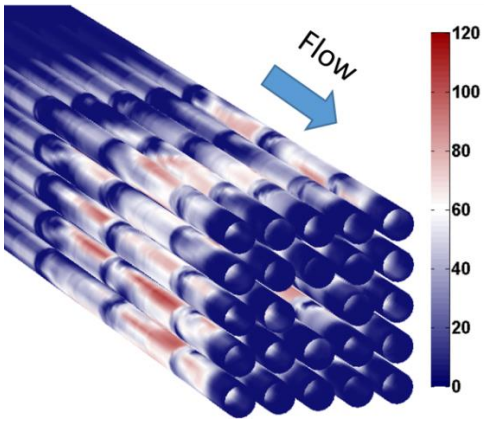


Fig. 8. CRUD thickness at end of cycle predicted by coupled MAMBA-3D and STAR-CCM+ using MIMIC.

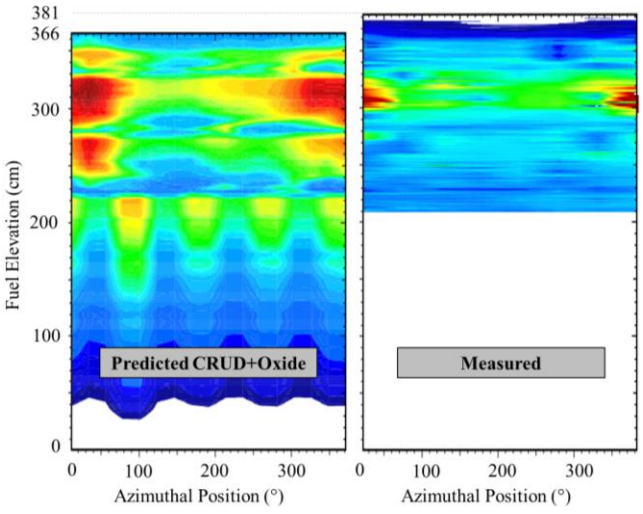


Fig. 9. Contour plot of predicted CRUD+oxide thickness (left), and measured data (right).

### 3. Cladding Hydriding

Cladding hydriding occurs at high temperature when the zirconium alloy oxidizes and leads to hydrogen production via the reaction,

$$Zr + 2H_2O \rightarrow ZrO_2 + 2H_2. \quad (8)$$

The absorbed hydrogen redistributes within the cladding based on temperature gradients, concentration gradients, and stress gradients. Hydrogen migrates down temperature gradients and, at a high enough concentration, it precipitates as a hydride, which may embrittle the cladding and increase the likelihood of rupture in an accident scenario. The physics of hydrogen migration has been implemented within the BISON code [11].

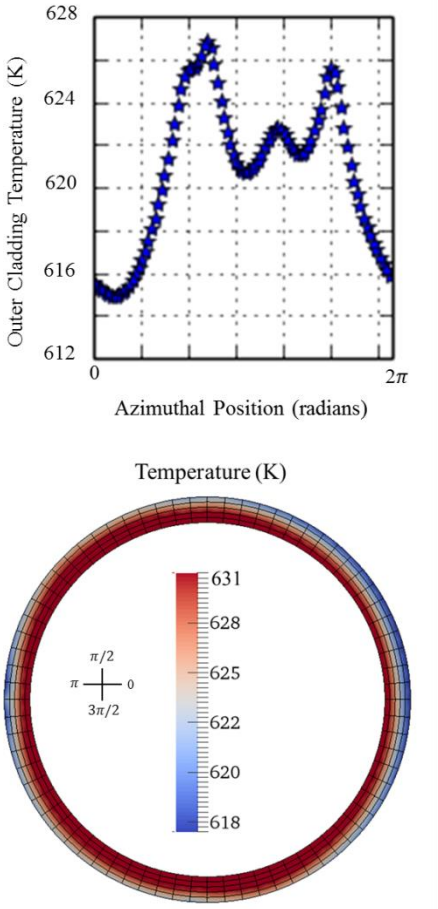


Fig. 10. Outer cladding temperature distribution predicted by STAR-CCM+ (top), and cladding temperature distribution predicted by BISON (bottom).

Because of the strong dependence of hydrogen distribution on temperature, CFD provides the high resolution data necessary to capture the effects. Fig. 10 (top) shows the outer cladding temperature distribution predicted by STAR-CCM+ at an elevation downstream of a spacer grid within mixing vanes. MIMIC is used to extract this distribution and write it into the appropriate format to be read by BISON. Subsequently, BISON assigns the temperature distribution as the Dirichlet boundary condition on the outer cladding surface. Then, the solution of the heat conduction (shown in bottom of Fig. 10), material stresses, and hydrogen migration are computed. Fig. 11 shows the distributions of the in-solution hydrogen (top), and the precipitated hydrides (bottom) within the cladding [12].

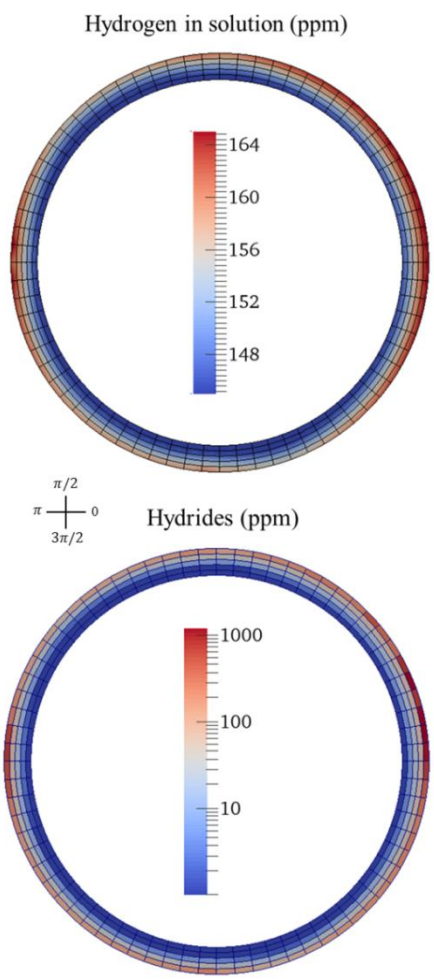


Fig. 11. Hydrogen in solution (top) and hydride (bottom) distribution predicted by BISON based on the outer cladding temperature provided by STAR-CCM+.

### V. CONCLUSIONS

A computational tool, called MIMIC, has been developed for nuclear reactor fuel rod analysis. This simulation framework is built on top of the commercial CFD code STAR-CCM+. To date, MIMIC has been coupled with the direct-solution neutronics suite MPACT, the CRUD deposition tools MAMBA-3D/1D, and the fuel performance code BISON.

MIMIC has tackled several challenging fuel rod phenomena, including CRUD induced power shift, CRUD induced localized corrosion, and cladding hydriding. Using computational fluid dynamics coupled with heat transfer within the fuel rod enables sufficient solution resolution to predict these strongly coupled physics.

The results predicted by MIMIC have demonstrated the high importance of resolving the heat flux and temperature distributions downstream of spacer grids and mixing vanes within light water reactors. It is the authors' hope that the DOE CASL program continues its success with these multiphysics demonstrations by incorporating MIMIC's capabilities within the VERA computational framework.

## VI. FUTURE WORK

MIMIC has proven to be a successful tool in accurately modeling CRUD deposition and oxidation on reactor fuel cladding. Because of the high computational cost of computational fluid dynamics, pin resolved neutronics, sub-pin structural mechanics, and CRUD/oxide growth kinetics, special attention will be given to the convergence criteria. A generalized method based on control theory has shown promise for controlling the time stepsize associated with loosely coupled physics [7,13,14].

Future work aims to extend MIMIC's capabilities as a tool for the development of algorithms specific to the reactor multiphysics coupling problem. In doing this, MIMIC will move towards the goal of being a software suite which can couple arbitrary multiphysics packages for reactor applications.

In addition, the new STAR-CCM+ co-simulation API, which is in active development, will be tested in order to see if its capabilities can meet those currently available in MIMIC. Documentation for the co-simulation API is much more extensive compared to that available for the user code functionality MIMIC leverages. Use of this API would benefit MIMIC by more closely following the STAR-CCM+ normally supported workflow.

Moreover, research is currently underway to remove the current deficiency in MIMIC in which STAR-CCM+ is required to be the driver program for all coupled simulation software due to constraints in the user code API. Work with the co-simulation API may remove this constraint, and simultaneously, other avenues are being explored to fix the restriction of STAR-CCM+ being the driver program.

## APPENDIX A: MIMIC CAPABILITIES

TABLE I. Summary of MIMIC capabilities.

| Capability | Enables |
|---|---|
| Multi-state | Cycle simulation with multiple STAR-CCM+ solutions |
| In-memory | Ability to handle hundreds of millions of data points |
| | Efficient solution transfers between non-conformal meshes |
| Time stepping methods | Lagged<br>Fixed-point iteration |
| Multiple physics coupling | Two or more physics may be coupled together |
| Region coupling | Heat source assignment<br>Temperature and density extraction |
| Boundary coupling | Temperature, heat flux, and turbulent kinetic energy extraction |
| Interface coupling | Conjugate heat transfer simulations (solid/fluid interface)<br>Thermal resistance assignment |
| Feedback mechanism selection | Sensitivity studies of feedback effects<br>On-the-fly control of whether to include feedback |
| Parallel partitioning | Efficient parallel solutions |
| Halo cell/face treatment | Accurate solution transfer for simulations using MPI |
| Centroid-based mapping | Generalized non-conformal mesh mapping |
| 1-D interpolation | Coarse-to-fine mesh solution transfer<br>Extruded mesh solution transfer to finer mesh |
| Power normalization | Total power conservation |
| Data post-processing | Extraction of averaged quantities to compare to lower resolution tools |

## REFERENCES

1. Kochunas, B., Collins, B., Downar, T.J., Martin, W., "MPACT: Michigan Parallel Advanced Characteristics Transport," M&C 2013, Sun Valley, ID (2013).
2. Kendrick, B., "CRUD Modeling: MAMBA (FMC Advanced Model for Boron Analysis) Development and Applications." University of Tennessee, Knoxville. 14 Apr. 2016. Lecture.
3. Kendrick, B., "MAMBA Theory Manual", Los Alamos National Laboratory, March 31, 2016.
4. Hales, J. D., "BISON Users Manual," Idaho National Laboratory (2013).
5. Walter, D. J., Kendrick, B.K., Petrov, V., Manera, A., Collins, B., Downar, T.J., "Proof-of-principle of high-fidelity coupled CRUD deposition and cycle depletion simulation", Annals of Nuclear Energy, 85, 1152-1166, 2015.
6. Petrov, V., Kendrick, B.K., Walter, D. J., Manera, A., Secker, J., "Prediction of CRUD deposition on PWR fuel using a state-of-the-art CFD-based multi-physics computational tool", Nuclear Engineering and Design 299, pp. 95-104, 2016.

7. Walter, D. J., “A high fidelity multiphysics framework for modeling CRUD deposition on PWR fuel rods,” PhD thesis, University of Michigan (2016).
8. Manera, A., Walter, D. J., Petrov, V., Kendrick, B. K., Wirth, B., Sweet, R., Rizk, J., Kucukboyaci, V., Secker, J., “Qualification of CFD-based PWR CILC capabilities for the identification of high risk fuel rods,” L1.CASL.P13.04 (2016).
9. Kucukboyaci, V., Secker, J., Salazar, D., “VERA-CS with MAMBA Model of Seabrook”, CASL-X-2016-1196-000.
10. Secker, J.R., “Seabrook cycle 5 report from WEC to CASL: CE-11-285 and CE-11-198, Rev 1 in support of CASL AMA milestone: AMA.CHALNG.07.Y1”, March 29, 2011.
11. Courty, O., Motta, A. T., Hales, J. D., "Modeling and Simulation of Hydrogen Behavior in Zircaloy-4 Fuel Cladding," Journal of Nuclear Materials, no. 452, pp. 311-320, 2014.
12. Mankosa, M., Walter, D. J., Ivanob, K., Avramova, M., Manera, A., Petrov, V., Williamson, R. L., Novascone, S. R., “Three-Dimensional Modeling of Hydrogen and Hydride Distribution in Zirconium Alloy Cladding using High-Fidelity Multi-Physics Simulations,” PHYSOR 2016, Sun Valley, ID (2016).
13. Walter, D. J., Manera, A., “Adaptive burnup stepsize selection using control theory for 2-D lattice depletion simulations,” *Progress in Nuclear Energy*, **88**, pp. 218-230 (2016).
14. Walter, D. J, Petrov, V., Manera, A., “A Control Theory Approach to Adaptive Stepsize Selection for Coupled CFD and CRUD Chemistry Simulations,” *Proc. NURETH-16*, Chicago, IL, August 19 – September 4, 2015.