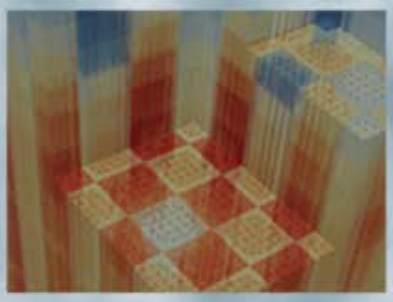
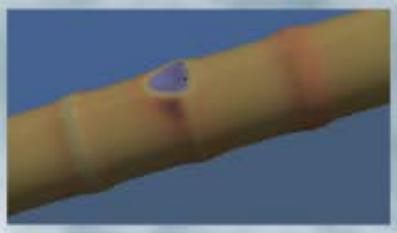
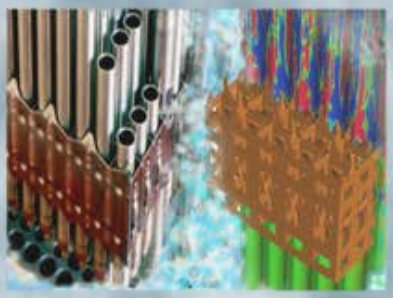
CASL-U-2015-0036-000

Design of a High Fidelity Core Simulator for Analysis of Pallet Clad Interaction







Kevin Clarno, Thomas Evans, and Benjamin Collins
Oak Ridge National Laboratory

Roger Pawlowski
Sandia National Laboratories

Roger Montgomery
Pacific Northwest National Laboratory

Brendan Kochunas
University of Michigan—Ann Arbor

Derek Gaston
Idaho National Laboratory

April 19-23, 2015

CASL-U-2015-0036-000

ANS MC2015 - Joint International Conference on Mathematics and Computation (M&C), Supercomputing in Nuclear Applications (SNA) and the Monte Carlo (MC) Method · Nashville, Tennessee · April 19–23, 2015, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2015)

DESIGN OF A HIGH FIDELITY CORE SIMULATOR FOR ANALYSIS OF PELLET CLAD INTERACTION

R. Pawlowski*[1], K. Clarno[2], R. Montgomery[3], R. Salko[2], T. Evans[2], J. Turner[2], and D. Gaston[4]

[1] Sandia National Laboratories†

[2] Oak Ridge National Laboratory‡

[3] Pacific Northwest National Laboratory

[4] Idaho National Laboratory

ABSTRACT

The Tiamat code is being developed by CASL (Consortium for Advanced Simulation of LWRs) as an integrated tool for predicting pellet-clad interaction and improving the high-fidelity core simulator. Tiamat is a large-scale parallel code that couples the the multi-dimensional Bison-CASL fuel performance code on every fuel rod with the COBRA-TF (CTF) sub-channel thermal-hydraulics code and either the Insilico or MPACT neutronics codes. Tiamat solves a transient problem where each time step is subcycled using Picard iteration to converge the fully-coupled nonlinear system. This report discusses the solution algorithms and software design of the simulator. Results are shown for a five assembly cross and compared against a separate core simulator developed by CASL. Tiamat has demonstrated that it can compute quantities of interest for analyzing pellet-clad interaction.

Key Words: Multiphysics, Pellet Clad Interaction, Bison, Insilico, MPACT, Cobra-TF, Tiamat

1 INTRODUCTION

Pellet-Cladding interaction (PCI) is a local cladding failure mechanism that can occur in various fuel rod locations in the reactor core depending on the core operating strategy, the power maneuvering approach, and the fuel rod design characteristics [1]. An important capability for the assessment of PCI failure is the ability to calculate a small set of quantities such as the fuel rod clad hoop stress and maximum centerline temperature as a function of position within the reactor core. These quantities can identify a candidate subset of rods in the reactor core for performing more detailed simulations. The Bison-CASL single-rod, fuel performance code can calculate these quantities using a 2-D axi-symmetric, or 3-D, geometric representation of the fuel rod [2, 3]. A high-fidelity core simulator, named Tiamat, is being developed to couple the Bison-CASL fuel performance calculation to a high-resolution representation of the fission density and coolant

*Corresponding Author: rppawlo@sandia.gov; Albuquerque, NM

†Sandia National Laboratories is a multiprogram laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under Contract DE-AC04-94-AL85000.

‡Oak Ridge National Laboratory is managed by UT-Battelle LLC under Contract No. DE-AC05-00OR22725 with the US Department of Energy

conditions for each fuel rod within the core [4]. This coupling provides a high-fidelity, integrated approach for estimating rods that are susceptible to PCI and can also be used to drive improvements in the core simulator.

VERA (Virtual Environment for Reactor Analysis) is a suite of simulation capabilities being developed to address the CASL (Consortium for Advanced Simulation of Light Water Reactors) Challenge Problems [5]. The VERA-CS (Core Simulator) is an integration of VERA components to provide all of the functionality of a traditional core simulator (multi-cycle analysis of power, temperature, and flow distributions), but with a much higher fidelity [4, 6]. Multiple top level drivers exist in VERA-CS that couple specific application components for certain analyses. Prior to Tiamat, VERA-CS contained two drivers that coupled a neutronics code to the COBRA-TF (CTF) sub-channel thermal-hydraulics and fuel heat transfer models [7]. One driver is based on the MPACT neutronics code [8,9] and the other driver uses the Insilico neutronics code [10]. Tiamat replaces the traditional internal CTF fuel heat transfer models with the high fidelity fuel performance capability of Bison-CASL. This manuscript compares the Tiamat simulation results to the VERA-CS coupled driver "CTF+Insilico." Tiamat performance analysis, along with the MPACT neutronics version, is documented in these proceedings in Reference [11].

2 APPLICATION CODES

In order to accurately model PCI, the Tiamat code couples three physics application codes together. All neutronic aspects of the problem (cross-section calculation, neutron transport, power generation) for this manuscript are solved using the Insilico code suite [10, 12], but the MPACT code can be used as well [11]. The conjugate heat transfer duties are split between solving for energy conservation and mechanics in the fuel rods by Bison-CASL [2, 3] and energy conservation and flow in the coolant by CTF [7].

2.1 COBRA-TF (CTF)

CTF is a thermal-hydraulic simulation code designed for Light Water Reactor (LWR) analysis [7]. CTF includes a wide range of thermal-hydraulic models important to LWR safety analysis including flow regime dependent two-phase wall heat transfer, inter-phase heat transfer and drag, droplet breakup, and quench-front tracking. CTF is a time-dependent, control volume code that enforces conservation between volumes and is being used within Tiamat in a steady-state mode. Internally, CTF uses a pseudo-transient solver to converge towards a steady-state for a particular flow/power state. CTF also includes several internal models to help facilitate the simulation of actual fuel assemblies. These models include spacer grid models, a rod conduction model, and built-in material properties. CTF uses a two-fluid, three-field representation of the two-phase flow.

2.2 Insilico (Denovo/XSProc)

Insilico is one of the neutronics solvers in the VERA Core Simulator and is part of the SCALE nuclear analysis code suite that leverages the Exnihilo transport suite, both of which are in active

development at ORNL [13]. For this manuscript, the Denovo deterministic transport code was used to solve for the flux and eigenvalue solutions for the 3D problem using the Simplified Legendre (SPN) angular discretization [10, 12, 14]. Insilico uses XSProc to provide problem-dependent, microscopic and/or macroscopic multi-group cross sections that account for the shielding of the resonances. XSProc was used to process a variety of cross section libraries (from an 8 energy group test library to 252 energy-group production library) using the intermediate-resonance approximation with full-range Bondarenko factors with spatial-homogenization, along with (in some cases) energy-collapsing to a coarse energy-group structure, through a one-dimensional (1D) discrete ordinates transport calculation. For the calculations presented in this manuscript, the fine energy group structure leveraged the 252-group library collapsed to 23-groups.

2.3 Bison-CASL

The Bison-CASL fuel performance code is being developed to provide a single-rod, fuel performance modeling capability to assess safety margins, and the impact of plant operation and fuel rod design on the thermo-mechanical behavior, including Pellet-Cladding Interaction (PCI) failures in PWRs [2, 3]. PCI is controlled by the complex interplay of the mechanical, thermal and chemical behavior of a fuel rod during operation, and therefore modeling PCI requires an integral fuel performance code to simulate the fundamental processes of this behavior. Bison-CASL is built upon INL's MOOSE/Bison packages [15, 16]. This code architecture uses the finite element method for geometric representation and MOOSE uses a Jacobian-free Newton-Krylov (JFNK) scheme to solve systems of partial differential equations [15]. Bison-CASL is a time-dependent code because the material models and geometry (deformations) are strongly influenced by the time history. The Bison-CASL framework consists of a numerical representation of the heat conduction and the equilibrium mechanics equations, which are coupled via the temperature and displacement variables. Material property and constitutive model libraries provide thermal, mechanical, and chemical property models and irradiation effects models, such as fission product-induced swelling or irradiation creep. Bison-CASL can capture the different temporal regimes associated with fuel performance, such as burnup accumulation over several years followed by a rapid power ramp over several minutes. For this manuscript, the fuel rod geometric representation in Bison-CASL was a 2D azimuthally-symmetric (R-Z), smeared-pellet model. To model a set of fuel rods, Tiamat uses the MOOSE "MultiApp" interface [17] and every fuel rod is modeled as a separate MOOSE "App."

3 COUPLED PHYSICS DESIGN

The application codes are coupled to create a single-executable with full feedback between components using two VERA infrastructure packages: Physics Integration KErnels (PIKE) [18, 19], a unified interface with basic implementations for parallel, coupled application mathematics, and the Data Transfer Kit (DTK), for transferring data between physics applications [20]. The design philosophy of VERA [21, 22] is to develop and utilize library components or "packages" for assembling the coupled application. This "package" concept is implemented in the TriBITS build and testing environment [23] which is built on CMake. Each application in Tiamat is wrapped as a "package" that implements the coupling interfaces. The application packages can then be reused for other couplings within CASL and for other projects.

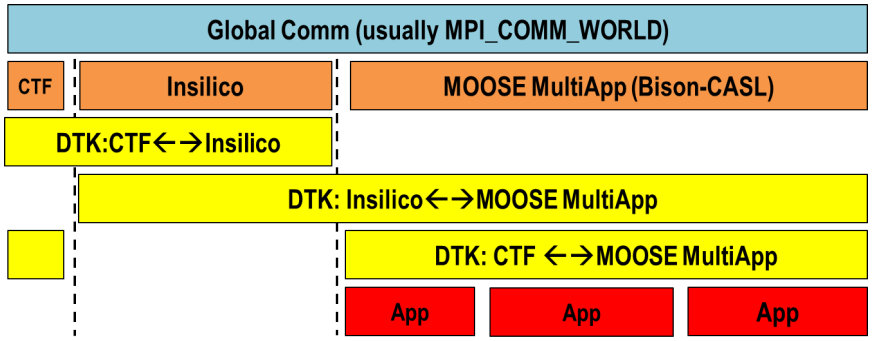


Figure 1. MPI communication layers in Tiamat; global executable comm (blue), application comms (orange), data transfer comms (yellow), and MOOSE "App" comms (red).

3.1 Parallel Distribution

Tiamat is designed for parallel distributed memory architectures and requires MPI. Figure 1 shows the layout of MPI communicator layers. The first layer is the global communicator for the coupled executable (typically MPI_COMM_WORLD). Tiamat splits the global communicator into three separate communicators, one for each application (second level colored in orange) using communicator tools in the PIKE package. While PIKE and Tiamat have the capability to overlap applications on MPI processes, for this manuscript each application is run in a disjoint set of MPI processes. In the past, depending on the problem size, overlapping some or all of the applications on the same MPI nodes has exhausted all the machine memory resulting in simulation failure. Work is currently underway to assess and reduce application memory usage to allow for overlap. Future work will explore overlapping MPI process layout schemes where possible. The third level of communicators (colored in yellow) are for data transfers between applications. These communicators are formed as the union of the two individual application communicators involved in the particular data transfer. This results in 3 additional communicators that are used by DTK: Insilico to/from Bison-CASL, Bison-CASL to/from CTF, and CTF to/from Insilico. Each fuel rod in Bison-CASL is modeled as a single MOOSE application or "App," through the MOOSE MultiApp interface [17]. Depending on the number of MPI processes assigned to MOOSE, multiple fuel rods can be grouped on the same MPI process, each rod could have it's own MPI process or each rod could be split across multiple MPI processes. This fourth level of MPI communicators are internal to the MOOSE MultiApp and not handled, or created, by Tiamat.

3.2 Coupled Solution Algorithm

Tiamat performs a transient simulation where each time step solves a strongly coupled nonlinear problem by fixed-point iteration. Bison-CASL performs a transient simulation, while the CTF and Insilico codes are treated as being in pseudo-steady-state for each solve within the fixed-point iteration loop. The time step is determined by querying the applications for preferred and bounded step sizes. For a given time step, Tiamat uses one of two iteration procedures available through PIKE: Gauss-Seidel (GS) or Jacobi. An example of the Tiamat nonlinear fixed-point algorithm for GS is shown in Algorithm 1 using the nomenclature found in [18]. x are the application solution vectors,

Algorithm 1: Gauss-Seidel Nonlinear Solve for Tiamat

Given x_B^0, x_C^0, and x_I^0.
for $k = 1, 2, \ldots$ *until converged* **do**
  Transfer power from Insilico to Bison-CASL, $z_{B,I}^k = r_{B,I}(x_I^{k-1})$
  if $k > 1$ **then**
    Damp the transferred power, $z_{B,I}^k = \alpha(z_{B,I}^k - z_{B,I}^{k-1}) + z_{B,I}^{k-1}$
  end
  Transfer clad temperature from CTF to Bison-CASL, $z_{B,C}^k = r_{B,C}(x_C^{k-1})$
  Solve Bison-CASL, $f_B(x_B^k, z_{B,I}^k, z_{B,C}^k) = 0$ for x_B^k
  Transfer heat flux from Bison-CASL to CTF, $z_{C,B}^k = r_{C,B}(x_B^k)$
  Solve CTF, $f_C(x_C^k, z_{B,C}^k) = 0$ for x_C^k
  Transfer fuel rod temperature from Bison-CASL to Insilico, $z_{I,B}^k = r_{I,B}(x_B^k)$
  Transfer fluid temperature and density from CTF to Insilico, $z_{I,C}^k = r_{I,C}(x_C^k)$
  Solve Insilico, $f_I(x_I^k, z_{I,B}^k, z_{I,C}^k) = 0$ for x_I^k
end

f are the application model equations in residual form, r are the application response functions that include damping algorithms, z are the application input parameters transferred from other codes, and α is the damping parameter. The subscripts B, C, and I represent Bison-CASL, CTF and Insilico, respectively. The power damping in the fixed-point iteration is required for algorthimic stability. Note that the stopping criteria must account for damping to ensure the algortihm obtains the requested tolerance. Recommended damping parameter values are discussed in [24, 25]. A comparison of Tiamat using the GS and Jacobi solvers for a small problem are discussed in [11].

3.3 Convergence Criteria

Convergence in Tiamat is assessed at the local and global level. Global convergence refers to the convergence of the entire coupled system. Convergence metrics of this form usually require that the update/change in a quantity of interest (QOI) between fixed-point iterations to be below a certain tolerance. Local convergence refers to the convergence of a single solve of an individual application. Each application has its own internal metrics for local convergence. Global convergence of Tiamat is declared successful if all of the following metrics are met:

1. **Power (Neutronics):** The difference of the power L_2 norm between fixed-point iterations, scaled by the L_2 norm of the power must be less than a user-defined tolerance scaled by the damping factor.
2. **Eigenvalue (Neutronics):** The change in k-eff between fixed-point iterations must be less than a user-defined tolerance scaled by the damping factor.
3. **Temperature (CTF):** The change in the maximum clad temperature and maximum coolant temperature between fixed-point iterations must be less than a user-defined tolerance.
4. **Temperature (Bison-CASL):** The change in the max fuel temperature across all apps in the MultiApp must be less than a user-defined tolerance.

5. **Local Convergence:** CTF, Insilico and Bison-CASL must report local convergence success.

The global convergence of the coupled code is strongly affected by the accuracy of each individual component. If the tolerances of the solvers in the application codes are too loose compared to the global convergence criteria, the code can fail to converge due to oscillations in the monitored QOI. The CTF local convergence criteria is based on five parameters that measure the convergence of the approximate steady-state solution: global energy and mass balance, fluid and solid energy storage, and global mass storage. We have found that CTF does not locally converge if the "balance" tolerances are set at or below 10^{-7}, because of the solution algorithm and boundary conditions for the multiphase equations. In this manuscript, the balance equations are converged to 10^{-6} and the other terms are converged to 0.005%.

Insilico local convergence, when used to solve the SPn equations, is controlled by the Krylov solver "tolerance." The tolerance can be safely reduced to a value of 10^{-12} without affecting the convergence internal to the Denovo solver. For convergence of the global system, a tolerance of 10^{-6} or tighter is typically needed. XSProc solves for the temperature- and density-dependent cross sections to a very tight tolerance (direct solve) on each iteration.

The local convergence in Bison-CASL is controlled by two primary tolerances, the absolute (10^{-10}) and relative (10^{-4}) norms for the nonlinear residual equations. In the clad solid mechanics calculation, there is an internal, point-wise, nonlinear solve, to a given absolute (10^{-12}) or relative (10^{-3}) tolerance, during the residual evaluation that can impact convergence. In this manuscript, the values used for each parameter are shown in parentheses when described. Additionally, the number of elements in the mesh can introduce discretization error and the time step size used in the ramping phase can introduce temporal error that can both impact convergence of the global system.

3.4 Data Transfer Between Codes

The data transfers between applications leverage the Data Transfer Kit (DTK) to achieve efficient and scalable transfers [20]. All transfers are performed directly in memory with point-to-point communication between MPI processes (i.e. there is no reading or writing to disk during transfers).

Conservation of transferred quantities is enforced for the transer of power in the fuel rods from Insilico to Bison-CASL and the transfer of energy via heat flux across the cladding outer surface from Bison-CASL to CTF. The data transfers in VERA codes leverage a "coupling mesh" to facilitate conservation. This was a VERA design choice dictated by application components and not on the coupling infrastructure. The coupling mesh represents the coarsest application component mesh in the simulation. For this work, it is used for both the discretization of the XSProc cross section library in Insilico and the CTF axial mesh. The XSProc mesh consists of cylindrical fuel rod control volumes that can be refined axially. Insilico's use of XSProc requires a constant radial temperature in the control volume and Insilico provides a constant power within each control volume. While the application codes use finer meshes internally, the data must be averaged over the control volumes for interacting with XSProc and CTF. Coupling meshes that can be refined radially

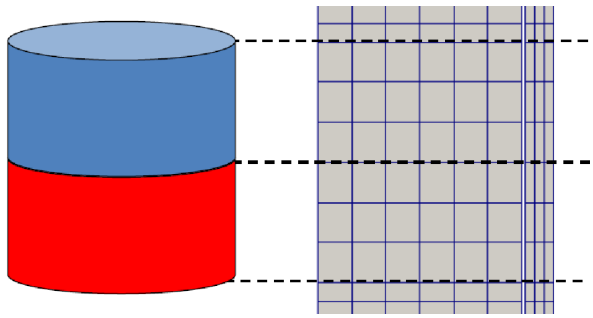


Figure 2. Coupling mesh control volume boundaries (left) are aligned with Bison-CASL finite element mesh (right).

and azmuthally are planned.

For the power transfer, Insilico provides DTK with the average power in non-overlapping cylindrical control volumes; Bison-CASL requests from DTK the power at nodal basis points on the finite-element mesh. Transferring power to the nodal basis in Bison-CASL (instead of the quadrature points) causes a subtle issue in conservation. At the control volume boundaries of the coupling mesh (Figure 2), a set of nodes exist on the intersection of each of the adjoining control volumes. These nodes can be assigned the values of either bordering control volume. The value chosen depends entirely on the DTK search algorithm and the ordering of geometric data when the DTK map is established. These boundary nodes make it impossible to enforce conservation in each individual control volume since since they contribute to multiple control volumes (i.e. the boundary nodes would be scaled multiple times, once for each adjacent control volume). Therefore, power conservation is enforced across an entire rod instead of each individual control volume. In the future, we plan to correct this by transferring source terms directly to the quadrature points, or cell-centroids, in Bison-CASL instead of the nodal locations.

Bison-CASL transfers an average heat flux, integrated over the coupling-mesh regions, to CTF. Since the finite element regions in Bison-CASL are axially aligned with the CTF control volumes, this transfer is conservative by construction. A MOOSE LayeredSideIntegral object computes the total heat flux for a coupling mesh control volume by integrating element outer clad surfaces for all elements in the control volume.

3.5 Algorithm for Ramping to Hot Full-Power

During the ramp up to HFP conditions, Bison-CASL requires small time steps to accurately and robustly model the evolution of the fuel rod mechanics and material properties. The solution process in Section 3 could be used to advance the simulation through all time steps in a strongly coupled fashion. However, solving the fully-coupled simulation for this time segment is expensive in the fully-coupled mode and, therefore, an operator-split time integration procedure was developed to speed-up the initialization phase to HFP. The overall multi-step method for computing the coupled physics solution at HFP includes an optional initialization phase to bring all codes to approximate

HFP conditions and then performs the strongly-coupled, fixed-point iterations between the codes until the coupled solution has converged. The following list describes the startup procedure, with Steps 2 and 3 being optional (the impact on accuracy and run-time has not yet been quantified):

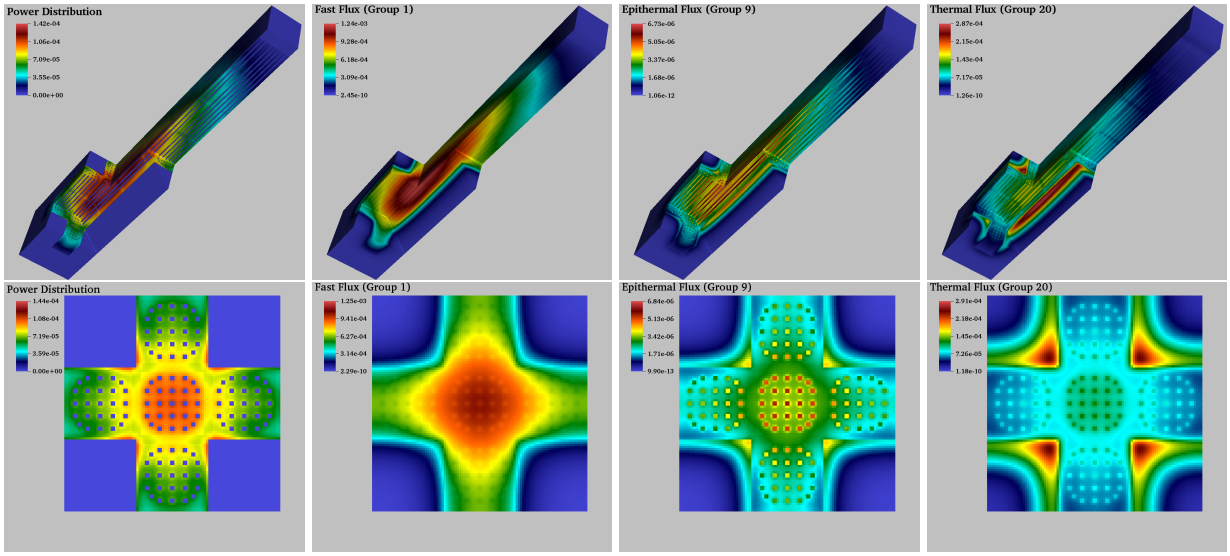
1. Model transition from cold, zero-power (CZP) to hot, zero-power (HZP) in stand-alone Bison-CASL. A transient simulation is run for 100 seconds during which the clad surface temperature is linearly ramped from 20 to 293 C, and the power is zero in every rod.
2. Estimate reactor state at hot, full power (HFP). There are currently three choices to obtain this estimate. The first is to perform several iterations of CTF+Insilico (without Bison-CASL) at HFP conditions; the iterations of CTF+Insilico are decoupled from Bison-CASL by using the CTF rod heat transfer model. The second choice is to restart from an approximate solution using restart data, from either Tiamat or CTF+Insilico, saved on the coupling mesh. The third choice is to use a combination of the first two choices – restart with an approximate solution and then perform a few iterations of CTF+Insilico.
3. Model transition from HZP to estimated HFP running a Bison-CASL transient for 48 hours. During this transient simulation, the clad surface temperature is linearly ramped from 293 C to the estimated HFP value (from Step 2) and the power distribution is linearly ramped from zero to the estimated HFP value.
4. Obtain consistent HFP conditions by running the fully-coupled simulation as described in Algorithm 1 for one or a few short short time steps.

4 DEMONSTRATION OF CAPABILITY

The test problems used in this manuscript are based on the dimensions and state conditions of Watts Bar Unit 1 Cycle 1, with detailed geometric and material specifications for the fuel rods in CASL "Progression Benchmark 6," [5]. Further problem details can also be found in another manuscript included in these proceedings [11].

Currently, analyses using Tiamat have been completed on single rod, single assembly, and a set of 5 assemblies, which replicates the complexity of a full assembly with a radial reflector. A comprehensive parametric study for the single rod case was used to benchmark the code and is found in these proceedings in [11]. Results for all simulations are documented in [26]. For this manuscript, the 5-assembly results will be shown. The assemblies are a standard 17x17 Westinghouse fuel design with uniform fuel enrichment. There are no axial blankets or enrichment zones in the first cycle. Each assembly has 264 fuel rods, 24 guide tubes, and a single instrument tube in the center. There are no control rods or removable burnable absorber assemblies in this problem. The assemblies were arranged in a cross pattern with bypass coolant in the outer regions, and the CASL production neutron cross section library (252 groups collapsed to 23) was utilized.

The problem was run on 1735 cores of the Oak Ridge Leadership Computing Facility "Eos" system. One MPI process was assigned to CTF, 289 MPI processes to Insilico and the remaining 1445 MPI processes to Bison-CASL. This distribution allowed Bison-CASL to assign each fuel rod to a separate MPI process.



(a) (b) (c) (d)

Figure 3. Reactor plots and slice planes at power peak height at HFP for (a) power distribution, (b) fast flux, (c) epithermal flux and (d) thermal flux.

Figure 3 depicts the power distribution and three of the 23-group neutron fluxes at hot full power (HFP) as computed by Tiamat. Because of the higher density and increased moderation at the bottom of the core, the power distribution is significantly lower-peaked. The Tiamat results are compared against the separate VERA code CTF+Insilico. In terms of globally integrated quantities, the peak power computed by Tiamat was 296 W/cm while it was 297 W/cm for CTF+Insilico. The average fuel temperature was 625 C for Tiamat and 680 C for CTF+Insilico. Figure 4 shows a comparison of the linear power averaged over all fuel rods in the reactor for Tiamat and the CTF+Insilico code. The horizontal line segments represent the constant value over each control volume section on the coupling mesh. Tiamat shows good overall agreement with CTF+Insilico. Note that the CTF+Insilico and Tiamat results should not be expected to agree since CTF uses different material and conduction models for the fuel rods.

An important function of the Tiamat code will be the ability to identify regions of interest within the core for more detailed PCI analysis of failure potential using 3-D fuel rod representation. One method to identify this region of interest is by evaluating the maximum cladding hoop stress and maximum rod centerline temperature in each rod. The maximum cladding hoop stress and the maximum centerline temperature parameters are shown in Figure 5. The cladding hoop stress results are consistent with the full power conditions at zero burnup that were simulated with Tiamat, where the coolant pressure is the primary contributor to the stress state in the cladding. The guide tube and instrumentation tube locations are depicted by the white squares.

Timing results for the full core problem are shown in Tables I, II and III. The code was restarted from a converged initial guess from a previous CTF+Insilico simulation. This explains the small number of iterations needed for CTF+Insilico. Tables II and III break down the simulation into the

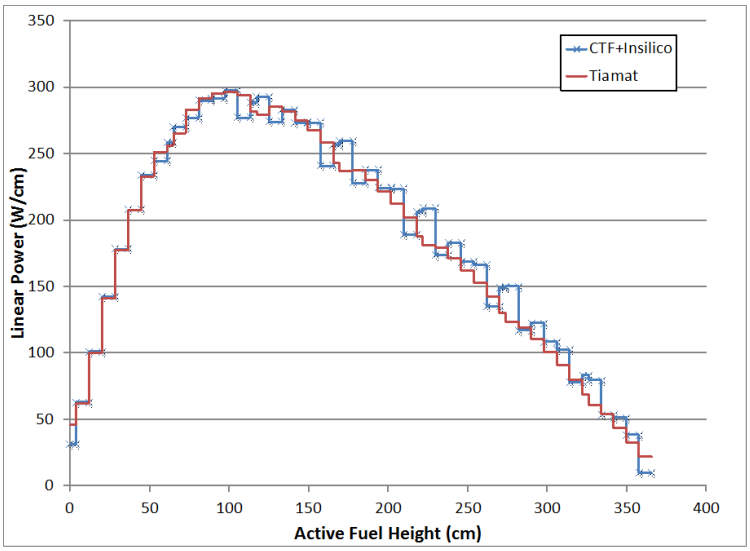


Figure 4. Linear power for coupling mesh control volumes as a function of reactor height.

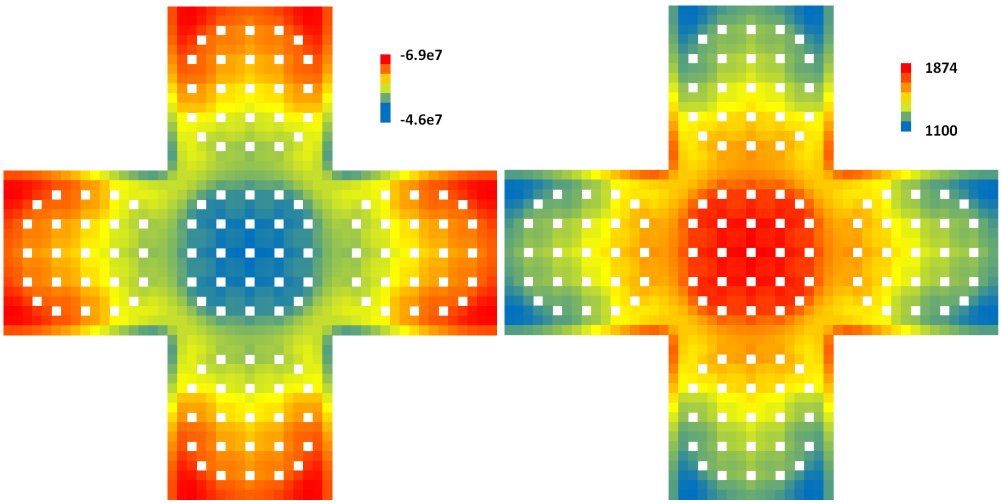


Figure 5. Spatial plot of figures of merit for assessing possible pellet-clad interaction. Left: Maximum cladding hoop stress (Pa). Right: Maximum rod centerline temperature (K).

Table I. Timing Comparison of Tiamat to CTF+Insilico

	Tiamat	CTF+Insilico
Number of Cores	1735	289
Total Time (s)	22840	6919
Object Construction Time (s)	2200	-
Initialization Time (s)	3641	-
Solve Time (s)	16999	-
Number of Fixed-point Iterations	7	2

Table II. Tiamat Application Solve Timings

Application	Phase	Num Cores	Solve Time (s)	Num Solves	Time/Solve (s)
CTF	Init	1	697	1	697
Insilico	Init	289	2316	1	2316
Bison-CASL	Init	1445	96.4	12	8
CTF	Solve	1	963	7	138
Insilico	Solve	289	15600	7	2229
Bison-CASL	Solve	1445	94.48	7	14

Table III. Tiamat Transfer Timings

Transfer	Phase	Time (s)				
		Number	Min	Mean	Max	Mean/Transfer
CTF to Insilico	Init	1	4.40e-1	19.46	1991	19.46
	Solve	7	1.93e-1	1.93e-1	1.93e-1	2.75e-2
CTF to Bison-CASL	Init	1	89.91	89.91	89.91	89.91
	Solve	7	4.92	4.92	4.92	0.702
Bison-CASL to CTF	Init	1	4.67e-1	4.67e-1	4.69e-1	4.67e-1
	Solve	7	1.90e-3	2.36e-3	4.83e-3	3.38e-4
Insilico to Bison-CASL	Init	1	102.3	102.3	102.3	102.3
	Solve	7	332.9	332.9	333.2	47.56
Bison-CASL to Insilico	Init	1	7.24e-2	7.24e-2	7.25e-2	7.24e-2
	Solve	7	1.38e-3	1.47e-2	1.51e-2	2.10e-3

initialization phase (Section 3.5) and a single coupled time step (Section 3.2) at HFP to transition from estimated HFP to a consistent HFP. The total runtime was 4.75 hours, 80% of which was spent in the neutronics solves. The data transfers are shown to be quite efficient. Note that no effort to load balance the application codes was done for the timings in this manuscript. In practice, users can manually change the number of MPI processes assigned to each physics application in the input file to load balance the simulation.

5 CONCLUSIONS

A new high-fidelity core simulation tool has been developed to study pellet-clad interaction. The software design supports parallel, distributed architectures and performs transfers in-memory for efficiency. This capability has been demonstrated for the Insilico-version of Tiamat on a five assembly cross for the core physics progression Problem 6 (17x17 assembly from Watts Bar cycle 1). The timings of the individual components and data transfer between components have been presented and the method for ramping from cold, zero-power to hot, full-power was described. The simulations results for the five-assembly cross, along with the single rod results presented in [11], compare favorably with an independent core simulator in the CASL software suite. The Tiamat code has been demonstrated to be capable of producing the required metrics for assessing potential PCI failure regions. Future work will involve extending the capability to perform full-core simulations and integrating depletion to simulate single and multiple fuel cycles.

6 ACKNOWLEDGMENTS

This research was supported by the Consortium for Advanced Simulation of Light Water Reactors (www.casl.gov), an Energy Innovation Hub (http://www.energy.gov/hubs) for Modeling and Simulation of Nuclear Reactors under U.S. Department of Energy Contract No. DE-AC05-00OR22725.

This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

7 REFERENCES

[1] S. Nesbit, M. Kennard, and S. Yagnik, "Use of Core Analyses in Assessments of Fuel Failure Risk due to Pellet-Cladding Interaction," *Proc. ANS 2009, Topical Meeting ANFM*, pp. 1–20, 2009.

[2] R. Montgomery and et al., "Peregrine: Advanced Modeling of Pellet-Cladding Interaction (PCI) Failure in LWRs," *Proceedings of the TopFuel 2012 Reactor Fuel Performance Conference*, Manchester, United Kingdom, September 2-6, 2012.

[3] R. Montgomery and et al., "Advanced Pellet-Cladding Interaction Modeling using the US DOE CASL Fuel Performance Code: Peregrine," *Transactions of the American Nuclear Society Annual Meeting*, Reno, Nevada, June, 2014.

[4] S. Palmtag et al., "Coupled Neutronics and Thermal-Hydraulic Solutions of a Full-Core PWR Using VERA-CS," *Proceedings of PHYSOR 2014*, Kyoto, Japan, September 28–October 3, 2014.

[5] A. T. Godfrey, "VERA Core Physics Benchmark Progression Problem Specifications," CASL-U-2012-0131-004, Oak Ridge National Laboratory (2014), `http://www.casl.gov/docs/CASL-U-2012-0131-004.pdf`.

[6] S. P. Hamilton et al., "Multiphysics Simulations for LWR Analysis," *Proceedings of the International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C 2013)*, Sun Valley, ID, USA, May 5–9, 2013.

[7] M. N. Avramova, "CTF: A Thermal Hydraulic Sub-Channel Code for LWR Transient Analyses, User's Manual," Pennsylvania State University, Department of Nuclear Engineering (2009).

[8] "MPACT Theory Manual," University of Michigan (2013).

[9] B. Kochunas et al., "MPACT Methodology for PWR Cycle Depletion," *Proceedings of the ANS Joint International Conference on Mathematics and Computation (M&C 2015), Supercomputing in Nuclear Applications (SNA) and the Monte Carlo (MC) Method*, Nashville, TN, USA, April 19–23, 2015.

[10] T. M. Evans, A. Stafford, R. Slaybaugh, and K. T. Clarno, "DENOVO: A New Three-Dimensional Parallel Discrete Ordinates Code in SCALE," *Nuclear Technology*, **171**, pp. 171–200 (2010).

[11] K. T. Clarno, R. P. Pawlowski, T. M. Evans, B. Kochunas, and J. Turner, "High Fidelity Modeling of Pellet-Clad Interaction Using the CASL Virtual Environment for Reactor Applications," *Proceedings of the ANS Joint International Conference on Mathematics and Computation (M&C 2015), Supercomputing in Nuclear Applications (SNA) and the Monte Carlo (MC) Method*, Nashville, TN, USA, April 19–23, 2015.

[12] G. Davidson, T. M. Evans, R. Slaybaugh, and C. Baker, "Massively Parallel Solutions to the k-Eigenvalue Problem," *Trans. Am. Nucl. Soc.*, **103** (2010).

[13] "SCALE: A Comprehensive Modeling and Simulation Suite for Nuclear Safety Analysis and Design," ORNL/TM-2005/39, Oak Ridge National Laboratory (2011), Available from Radiation Safety Information Computational Center at Oak Ridge National Laboratory as CCC-785.

[14] S. Hamilton and T. Evans, "Efficient solution of the simplified PN equations," *Journal of Computational Physics*, **284**, pp. 155–170 (2015).

[15] D. Gaston, C. Newman, G. Hansen, and D. Lebrun-Grandie, "MOOSE: A parallel computational framework for coupled systems of nonlinear equations," *Nuclear Engineering Design*, **239**, pp. 1768–1778 (2009).

[16] R. Williamson et al., "Multidimensional multiphysics simulation of nuclear fuel behavior," *Journal of Nuclear Materials*, **423**, pp. 149–163 (2012).

[17] D. R. Gaston et al., "Physics-based multiscale coupling for full core nuclear reactor simulation," *Annals of Nuclear Energy* (2014).

[18] R. P. Pawlowski, R. A. Bartlett, N. Belcourt, R. Hooper, and R. Schmidt, "A Theory Manual for Multi-physics Code Coupling in LIME," Sandia Technical Report SAND2011-2195, Sandia National Laboratories (2011).

[19] 2014, `https://trilinos.org/packages/pike`.

[20] S. Slattery, P. Wilson, and R. P. Pawlowski, "The Data Transfer Kit: A Geometric Rendezvous-Based Tool for MultiPhysics Data Transfer," *Proceedings of the International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C 2013)*, Sun Valley, ID, USA, May 5–9, 2013.

[21] J. A. Turner, "Virtual Environment for Reactor Applications (VERA): Snapshot 3.1," CASL-U-2013-0164-000, Oak Ridge National Laboratory (2013), `http://www.casl.gov/docs/CASL-U-2013-0164-000.pdf`.

[22] C. Baker et al., "Domain Model Specification for VERA: Version 1.0," CASL-U-2011-0169-000, Oak Ridge National Laboratory (2011), `http://www.casl.gov/docs/CASL-U-2011-0169-000.pdf`.

[23] 2014, `https://tribits.org`.

[24] S. P. Hamilton et al., "An Assessment of Coupling Algorithms for Nuclear Reactor Core Physics Simulations," *Journal of Computational Physics*, submitted October 2014.

[25] M. A. Berrill, K. T. Clarno, S. P. Hamilton, and R. P. Pawlowski, "Evaluation of Coupling Approaches," CASL-U-2014-0081-000, Oak Ridge National Laboratory (2014), `http://www.casl.gov/docs/CASL-U-2014-0081-000.pdf`.

[26] R. P. Pawlowski, K. T. Clarno, and R. O. Montgomery, "Demonstrate Integrated VERA-CS for the PCI Challenge Problem," CASL-I-2014-0153-000, Oak Ridge National Laboratory (2014).