Engineering 3 (2017) 518–526

Contents lists available at ScienceDirect

Engineering

journal homepage: www.elsevier.com/locate/eng

Research
Clean Energy—Article

Computational Tools for the Integrated Design of Advanced Nuclear Reactors

Nicholas W. Touran, John Gilleland*, Graham T. Malmgren, Charles Whitmer, William H. Gates III

TerraPower, LLC, Bellevue, WA 98005, USA

ARTICLE INFO

Article history:
Received 16 March 2017
Revised 7 July 2017
Accepted 12 July 2017
Available online 3 August 2017

Keywords:
Simulation
Nuclear energy
Electricity generation
Advanced reactor
Traveling wave reactor

ABSTRACT

Advanced nuclear reactors offer safe, clean, and reliable energy at the global scale. The development of such devices relies heavily upon computational models, from the pre-conceptual stages through detailed design, licensing, and operation. An integrated reactor modeling framework that enables seamless communication, coupling, automation, and continuous development brings significant new capabilities and efficiencies to the practice of reactor design. In such a system, key performance metrics (e.g., optimal fuel management, peak cladding temperature in design-basis accidents, levelized cost of electricity) can be explicitly linked to design inputs (e.g., assembly duct thickness, tolerances), enabling an exceptional level of design consistency. Coupled with high-performance computing, thousands of integrated cases can be executed simultaneously to analyze the full system, perform complete sensitivity studies, and efficiently and robustly evaluate various design tradeoffs. TerraPower has developed such a tool—the Advanced Reactor Modeling Interface (ARMI) code system—and has deployed it to support the TerraPower Traveling Wave Reactor design and other innovative energy products currently under development. The ARMI code system employs pre-existing tools with strong pedigrees alongside many new physics and data management modules necessary for innovative design. Verification and validation against previous and new physical measurements, which remain an essential element of any sound design, are being carried out. This paper summarizes the integrated core engineering tools and practices in production at TerraPower.

1. Introduction

Computing environments provide all engineering design efforts with vast virtual laboratories in which ideas can mature into sophisticated, well-optimized systems. Leveraging such virtual facilities has been and remains particularly essential to progress in the nuclear industry due to the associated complex physics and high experimental costs. Indeed, modern computing and nuclear technology were born together, as the first digital computer's first task was to solve a set of partial differential equations modeling deuterium-tritium behavior for Edward Teller in 1945 [1].

The first nuclear reactors were designed with analytic methods informed by experiments. Soon, radiation transport problems were being solved numerically on early computers to supplement shielding and core design activities. In 1949, submarine reactor designers at Knoll's Atomic Power Laboratory obtained the first known computer solution of the neutron diffusion equation on an IBM 604 [2]. Increasingly powerful computers, as well as the advent of the Fortran programming language on the IBM 704 in 1957, enabled more sophisticated physics to be treated numerically. Concurrent advances in storage technology enabled the inclusion of extended nuclear data libraries, a key input to reactor simulations representing reaction probabilities. To this day, however, new reactor design concepts are generally simulated physically, often by zero-power critical arrangements of fuel assemblies in experimental core mock-up facilities.

From the 1970s through the 1990s, reactor simulation software became increasingly precise for isolated physics problems, covering the neutronics, heat transfer, fuel performance, transient analysis, and mechanical fields, among others. These codes were largely

* Corresponding author.
E-mail address: johng@terrapower.com

http://dx.doi.org/10.1016/J.ENG.2017.04.016

developed independently from one another, with individual subject matter experts being responsible for understanding their range of applicability and operation. In many cases, design teams created standard interface files to transfer data, allowing full system treatment in a somewhat manual, though effective fashion. Many commercial tools now perform some essential physics coupling (e.g., neutronics and thermal/hydraulics in water-cooled reactors), but much modeling focus in the advanced reactor regime has been on increasingly high-fidelity physics models, taking advantage of modern computer architectures to minimize approximations. Today's ongoing high-fidelity and multiphysics simulation efforts are a scientific endeavor aimed at reducing uncertainties, enhancing understanding, and enabling predictive capabilities. However, they are not meant to be directly usable by the industry yet.

In 2006, TerraPower set out to develop sustainable, scalable, and low-carbon energy in the Traveling Wave Reactor (TWR) program, featuring a fourth-generation, sodium-cooled, metal-fueled reactor unique in that it uses a once-through deep-burn fuel cycle to achieve many fast reactor capabilities (natural safety, reduced waste, reduction and eventual elimination of enrichment, and high fuel and thermal efficiency) without requiring reprocessing [3]. High-fidelity (but decoupled) physics models demonstrated the fundamental feasibility of the TWR design. As the organization grew, new software was developed and procured to support the evolving reactor design. In June 2009, development of the Advanced Reactor Modeling Interface (ARMI) code system began with the intent of incorporating new and existing physics modeling tools with data management and automation routines into a consistent reactor design toolbox. The relatively clean slate and initially small team provided the impetus for applying modern design patterns and programming practices to the challenge of highly efficient, scalable, and integrated reactor design. As the framework and data management developed, subject experts focused on creating new physics modules or adapters to high-quality external physics solvers. As the tools were integrated, each member of each team could seamlessly run the entire system analysis, from specifying the pin dimensions and tolerances to computing the system cost and peak cladding temperature during design-basis transients. Detailed and meaningful design, innovation, and sensitivity studies could be done with ease. Such a system has allowed TerraPower to develop its designs with aggressive timescales and small, agile teams.

2. Architecture

The ARMI framework comprises two key entities: the *reactor model* and the *interface stack*. The reactor model is responsible for maintaining a self-consistent state representing the full physical description of the reactor, including the geometry, material properties, power, temperatures, and so forth. The interface stack is responsible for performing various physics modeling activities using and updating the reactor model. This object-oriented architecture allows the coupling of reactor physics and analysis tools while promoting the decoupling of the source code itself, which is ideal, given the breadth of advanced nuclear design tools and the specialization of engineers with expertise in operating them. The diverse specialization of the ARMI contributors (who typically have advanced degrees in engineering) suggests the use of a high-level programming language for data management that can be learned relatively quickly (i.e., Python[†] for the majority of the code).

2.1. Reactor model

The reactor model in the ARMI code system is a composite pattern [4] mirroring physical components of a nuclear reactor core, as depicted in Fig. 1. The *reactor* object comprises a number of *assemblies*, which are made of subsections known as *blocks*, and which in turn contain shaped *components* such as fuel pins, cladding, and coolant. Each component is linked to a *material* with temperature- and composition-dependent properties. Data management in the ARMI code system simply involves reading and writing the state to the reactor model via the defined model interface. Standardizing this eponymous reactor model interface is key to the multiphysics interoperability of the ARMI code system, as any data passed through it lose their code-specific character and become globally accessible.

An application programming interface (API) provides users and programmers with intricate access to the entire reactor model. Weighted averages of any state variable over any arbitrary selection of the reactor are immediately available, and all-encompassing state changes (e.g., changing coolant density) can be programmed in a single statement.

A consistent and programmatically accessible material properties database is essential for integrated reactor design. Material objects have been created with various correlations embedded for thermo-mechanical properties: mass density, thermal expansion, heat capacity, viscosity, thermal conductivity, Young's modulus, yield strength, and so forth. Physics modules that in the past may have queried for *sodium* properties at a region's current temperature now only query for *coolant* properties. Thus, with a single line change to the input, a module that had been computing the sodium density coefficient of reactivity instantly computes the fluoride salt density coefficient of reactivity. A feature to automatically evaluate the ranges of the correlations and print them to a report for offline use has also proven valuable. In addition, the intricate link to the materials library allows for automatic and consistent thermal expansion from cold, as-manufactured dimensions to hot, full-power dimensions, or anywhere in between.

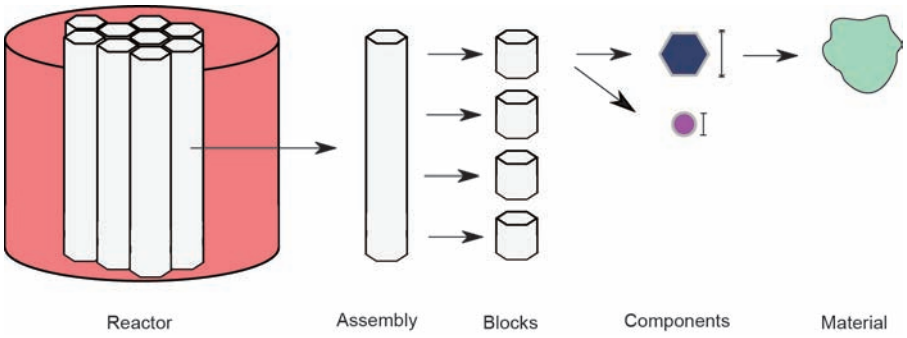

Fig. 1. The ARMI reactor model composite pattern in hexagonal geometry.

[†] A widely used programming language created by Guido van Rossum in 1991.

Physics solvers typically offer a variety of geometric approximations. Accordingly, the ARMI code system can convert the reactor state to and from various geometric representations. The converters perform automated (and tested) mass-conserving transformations eliminating the possibility of manual conversion errors. This feature can convert three-dimensional (3D) hexagonal cases into equivalent one-dimensional (1D), or cylindrical models (e.g., for ultra-high energy resolution, low spatial resolution simulations) if required for nuclear interaction probability preparations in certain regions or conditions, as depicted in Fig. 2. The possibility of maintaining 3D geometry for cross-section generation also remains.

2.2. Interface stack

The interface stack consists of physics modules and bookkeeping routines that maintain a self-consistent reactor model. The order and specific modules enabled during an individual ARMI simulation are dependent upon user input and the desired engineering evaluation. The interface stack invokes the user-specified modules at various simulation times (e.g., beginning-of-life, beginning-of-cycle, middle-of-cycle, end-of-cycle, and end-of-life). Ordering is set by physical interdependencies (e.g., between thermal power production and required coolant flow); however, the proper ordering is ill-defined in many coupled physics problems.

The design of the interface stack (depicted in Fig. 3) allows, for example, a module for nuclear depletion to be swapped out for a different module of higher fidelity or with an independent methodology. The other physics modules will interact with any new module without modification. Such swaps are frequently used for sensitivity studies, independent verification, and major upgrades, and for problems outside of a module's domain. A high-level listing of the modules and adapters in the ARMI code system is provided in Table 1.

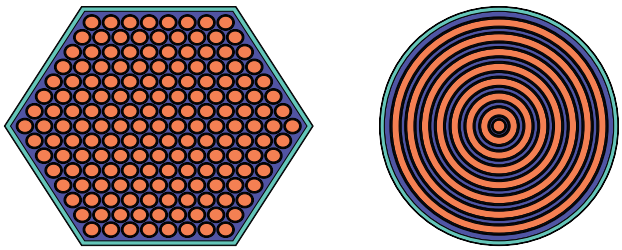

Fig. 2. An as-designed fuel assembly with its ARMI-generated 1D equivalent representation used in ultra-high energy resolution lattice physics simulations. Automatic conversions such as these are essential when modeling complex cores containing multiple different assembly designs.

3. Physics modules

A large number of physical interactions take place inside a nuclear core, spanning many orders of magnitude, both spatially and temporally. A neutron with a lifetime of 3×10^{-7} s may cause a fission that releases its decay heat over the following 3×10^{12} s. Gamma rays from the decaying fission products may asymmetrically heat a shield structure, causing it to mechanically bow and pinch the core assemblies, challenging fuel management. Coolant emerging from neighboring assemblies with different powers may mix turbulently, inducing thermal striping on instrumentation above the core. Many such coupled interactions must be considered together in order to effectively design a nuclear reactor.

3.1. Neutronics

The nuclear chain reaction occurring in the core serves as the heat source for the plant. The nuclear simulations must determine the spatial distribution of the neutrons, their speed, their direction of travel, and the rate with which they are undergoing various interactions (i.e., scattering, capture, fission, etc.) with the surrounding nuclei. The fission and capture rates determine the heat generation and fuel burnup rate, while the scattering interactions are strongly

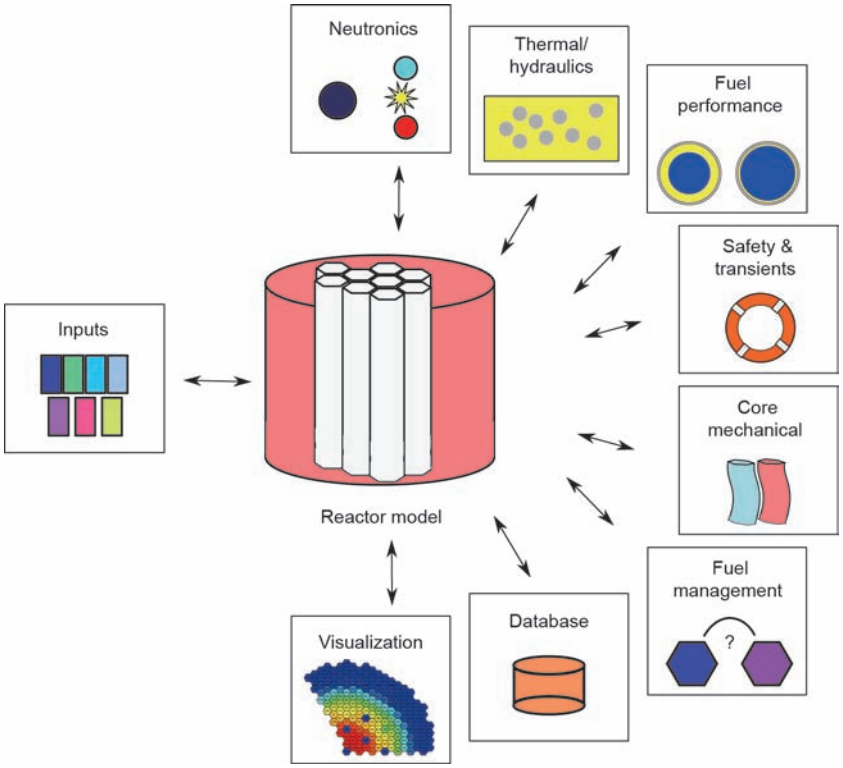

Fig. 3. A depiction of the interface stack operating on the reactor model.

Table 1
Overview listing of ARMI modules.

Module type	Module name
Framework	Reactor model
	Material library and thermal expansion
	Loose and tight physics coupling
	History tracking and summaries
	Database storage of results
	Multi-objective optimization
	Nonlinear regressions
Neutronics adapter	MC^2/DIF3D/REBUS[a]
	MCNPX[a]
Neutronics	Fuel-shuffling design and optimization
	Fast equilibrium fuel cycle iteration
	Parallelized depletion solver
	Microscopic cross-section manager
	Fission product models
	Fuel cycle economics
	Pin-level flux reconstruction
	Reactivity effects of distortions
	Sensitivity coefficients
	Uncertainty quantification
	Intrinsic source
	Gamma source generation
	λ-eigenmode expansion perturbation theory
Thermal/hydraulics adapter	COBRA[a] (communicating subchannel)
	MONGOOSE[b] (communicating subchannel)
Thermal/hydraulics	Subchan (non-communicating subchannel)
	Thermo (simple 1D)
	Flow-orificing system
Fuel performance adapter	FEAST[a]
	ALCHEMY[b] (detailed)
Fuel performance	CRUCIBLE (simple)
Core mechanical adapter	NUBOW[a]
Core mechanical	OXBOW
Transient analysis adapter	SASSYS/SAS4A[a] (plant & transients)
	DIF3D-K[a] (spatial kinetics)
	RELAP5[a]
Transient analysis	Reactivity coefficients
	Control rod worth, shutdown margin
	Frequency stability margin
Visualization	XTVIEW

[a] Adapter to off-the-shelf third-party code.
[b] Adapter to in-house TerraPower code.

correlated to radiation dose and material damage. The ARMI code system is flexible in the type of reactor that can be modeled, but because most TerraPower designs involve fast neutrons, the ARMI code system contains external code adapters for the Argonne National Laboratory (ANL)'s steady-state fast reactor physics suite, including MC^2 [5], DIF3D/VARIANT [6], and REBUS [7], to simulate these events in the nuclear chain reaction under various numerical approximations at a snapshot in time. Given the basic details of the neutron and photon populations, ARMI neutronics modules perform various calculations chosen from Table 1 according to the analysis under consideration.

In judging the merits of advanced reactor designs, it is necessary to understand the fleet performance of a plant as well as the early performance of a first-of-a-kind reactor. The fuel cycle equilibrium state, in which the charge and discharge material of a reactor becomes fully repetitive decades into operation, provides a useful representation of the *n*th-of-a-kind fleet performance, and is thus a target for designers to approach with explicit cycle-by-cycle treatments. To support this essential mode, the ARMI code system contains a new module based on the REBUS methodology [8] to compute the equilibrium fuel cycle implicitly, allowing rapid analysis and iteration.

Fuel management represents a very large problem space, especially in certain advanced reactors such as the TWR. The challenge of starting a core on enriched ^{235}U and transitioning through the breed-and-burn process into a ^{239}Pu fuel while maintaining relatively constant power, reactivity, and cooling distribution is only practical to meet with a vast set of computations. The internally developed adaptive fuel management routines in the ARMI code system leverage our supercomputer facility to simulate hundreds of potential fuel management choices in parallel for a 60-year plant lifetime. In fact, this was the first task that the ARMI code system was applied to in the TerraPower program. Human and machine intelligence are relied upon (mostly heuristically) to determine the choices to be evaluated in the current implementation, but the problem is ripe for advances in machine learning.

Like the other physics modules, the nuclear simulations depend on the physical layout and composition of the core. Uniquely, however, evaluated nuclear data are required to properly model the nuclear interactions. Difficult measurements of neutron-nuclide interaction rates versus incident neutron energy have been supplemented with nuclear physics models by various research institutions and national laboratories worldwide, resulting in data libraries such as the Evaluated Nuclear Data File [9]. To consider sensitivities and tolerances related to these data, the ARMI code system contains an uncertainty quantification (UQ) module that processes and propagates uncertainties based on the work of the Organisation for Economic Co-operation and Development/Nuclear Energy Agency (OECD/NEA) Subgroup 33 (SG33) [10]. A comparison of our implementation with the SG33 benchmark defined in Ref. [10] is shown in Fig. 4.

A TerraPower-modified version of the MCNPX radiation transport code [11] is used for some verification tasks as well as for certain reference criticality and shielding calculations [12]. The modifications made focused on runtime optimization, fuel management, fuel performance coupling, and mass conservation, and led to the ability to run roughly 10^6 independent depletion regions efficiently on 10^3 processors. The ARMI code system can write detailed input files for this code, allowing the rapid creation of very complex models. Like all physics solvers, the ARMI code system can be extended to automate other Monte Carlo tools as well.

3.2. Thermal/hydraulics

The heat released from the fissioning nuclei must be transported

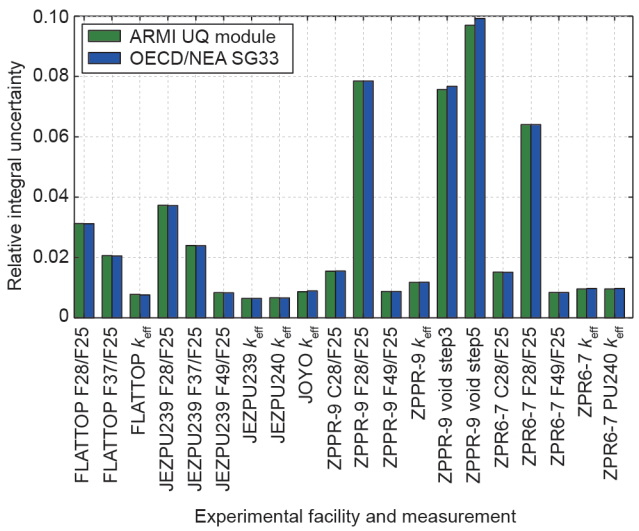

Fig. 4. The ARMI nuclear data UQ module benchmarked against OECD/NEA SG33 results for various experimental facilities and measurements.

to the power conversion system at the rate it is produced. Flowing coolants such as water, liquid metal, gas, or molten salt are typically employed for this purpose. The coolant flow characteristics determine the specifications of major equipment such as pumps and heat exchangers.

The fuel-to-coolant area ratio inside the fuel assemblies is chosen carefully, with tradeoffs between the nuclear chain reaction and flow rate being considered. Because coolant pressure is proportional to the square of flow velocity, designs pushing for high-speed coolant must have stronger (i.e., thicker) structural members, which impede aspects of the nuclear chain reaction.

Coupled reactivity effects between the nuclear chain reaction and the coolant can be very strong. When coolant is heated, thermal expansion decreases its density, and neutron moderation by scattering is diminished. In water-cooled slow-neutron reactors, this is a negative feedback, as the coolant is depended upon as a neutron moderator. Neutrons that slow down in the coolant, protected from parasitic captures in ^{238}U, can efficiently fission ^{235}U fuel upon reentering the pins. Conversely, the coolant density feedback in fast-neutron reactors is generally positive: Faster neutrons release more secondary neutrons per fission and are less likely to be parasitically captured.

Subchannel analysis (i.e., analysis of coolant temperature and flow between fuel pins) is performed in the ARMI code system in a variety of levels of fidelity, depending on the use case. A simple non-communicating subchannel module provides rapid scoping results, while multi-assembly communicating subchannels with inter-assembly heat transfer cases may be run for higher-fidelity analysis. The high-fidelity treatment involves passing continuous pin-level power distributions from the ARMI code system to a TerraPower-developed single-phase subchannel code called MONGOOSE. Computational fluid dynamics models and experiments are used to generate the correlations used in the subchannel codes. MONGOOSE currently features correlations specific to sodium-cooled fast reactors and validation efforts specific to the TWR, although single-phase correlations for other fluids can be supplied.

The fuel management challenge in the first-generation TWR design is constrained by fixed coolant orificing below each assembly position. Detailed analysis requires a two-pass simulation over the plant lifetime. On the first pass, orifice settings are allowed to vary at each cycle to bring the peak cladding temperature (including 2-sigma uncertainty) to its design limit in each assembly. The maximum flow rate in each position is recorded in preparation for the second pass, which optimizes several discrete orifice zones and settings such that no assembly will surpass its limiting temperature at any point in the plant life. Due to the burden of doubling the simulation work, orificed cases are typically only performed after large design sweeps have settled in on a preferred set of shuffles. By their modular design, the flow-orificing routines in the ARMI code system may be activated during implicit equilibrium fuel cycle cases, providing realistic orifice settings to begin a fuel management design iteration.

3.3. Fuel performance

The nuclear fuel system, comprising the fuel and the surrounding cladding, experiences the most extreme thermal, radiation, and chemical environment in the plant. Changes in the fuel system must be well understood in order to meaningfully design any reactor.

TerraPower handles fuel performance through two tools: ALCHEMY and CRUCIBLE [13]. ALCHEMY is a highly detailed finite-element-based mechanistic model of one fuel pin, informed by a database of historical irradiated fuel examinations. The ARMI code system provides lifetime power and coolant temperature histories as boundary conditions, and ALCHEMY determines internal porosity, fission gas release, temperature, stress, and strain.

ALCHEMY runs are intensive, and are therefore not directly coupled into the primary ARMI loop. Instead, the CRUCIBLE module was created as a low-order surrogate, informed by ALCHEMY results. It runs at every time step in the interface stack, updating the burnup-dependent cladding strain, axial fuel strain, fission gas release, thermal bond movement (if applicable), cladding corrosion, and fuel thermal conductivity. These state variables have appreciable effects on reactivity and the temperature field, and more generally on the overall feasibility of a design; therefore, they must be coupled into any advanced reactor design tool.

3.4. Mechanical and seismic analysis

Mechanical interactions of core assemblies and their support structures are coupled strongly into reactor development efforts. Mechanical calculations are required in core design for the following reasons: ① Assemblies must maintain hold-down during normal operation, given upward forces from coolant flow; ② the pressures exerted by the coolant cause inelastic deformations of the assemblies, which can lock up the core and cause long (and expensive) delays in fuel management during an outage; and ③ radial/bowing expansion reactivity feedback is an important component of fast reactor safety and handling loads. As the interacting load-pads above the core heat in a transient, thermal expansion pushes them apart. Changes in fuel positioning relative to the neutron flux gradients are complex, and can vary to the degree that corresponding reactivity effects switch signs at different power-to-flow ratios during startup. This behavior is also highly dependent on core restraint design (i.e., whether the design is of the free flowering or limited free bow configuration).

Integrated reactor design necessitates a close coupling with mechanical analysis. In the ARMI code system, this is accomplished analogously to the fuel performance. A high-fidelity, long-running finite element analysis mechanical code called OXBOW [14] has been developed that receives the input temperature, dose, and flow histories of one of the collections of assemblies from the ARMI code system. Detailed evaluations are performed over a design space, where it computes stresses, elastic/plastic strains, and geometric distortions as functions of power, flow, and irradiation history. Any arbitrary distortions from a power ascent or steady state, or even resulting from an earthquake, can be exported back into the ARMI code system for a follow-on analysis that uses new and sophisticated methods to compute the reactivity effects. A fast-running, correlation-based distortion module based on OXBOW results runs during normal ARMI runs, providing indications of distortions for design optimization.

3.5. Safety and transient analysis

A variety of analyses are required to understand the operational and off-normal behavior of the plant, such as how it responds when a pump or turbine trips. Safety and transient analyses are a primary focus of regulatory and licensing activities, and are therefore absolutely essential in an integrated design tool.

In the ARMI code system, transient analysis features the SASSYS code [15], which was originally developed by ANL in support of the US fast reactor program. The ARMI reactivity coefficient module automatically computes kinetics parameters required for coupled neutronic-thermal/hydraulic transient analysis, including the delayed neutron fraction; prompt neutron lifetime; radial expansion coefficient of reactivity; and 3D spatial distributions of fuel, structure, coolant, Doppler, and voided Doppler reactivity coefficients. Combined with the reactor model assembly dimensions, fuel composition, and flow characteristics, the ARMI code system directly writes the core sections of the SASSYS input. The plant model of SASSYS is not currently created by the ARMI code system, but is

appended to the input from a user-created file. We plan to continue to expand the ARMI code system to animate aspects of the plant as well as the core in the future.

With a complete SASSYS input file on hand, the ARMI code system executes one or more SASSYS cases. It can execute various design-basis or beyond-design-basis transients (e.g., protected or unprotected loss of flow, loss of heat sink, transient overpower) as well as a series of cases at varied power/flow ratios to compute the power defect. Sweeps of cases perturbing reactivity with varied-frequency oscillations are used to numerically compute the frequency-domain full-power transfer function of the plant, which is then fed into the frequency stability ARMI module to estimate the gain and phase stability margins—a key design parameter for any amplifier-like device.

Statistical sampling of the reactivity coefficients and other core inputs is used to run hundreds of SASSYS cases in parallel on a high-performance computer to determine target values of input uncertainty. This provides the intelligence that drives uncertainty targets, which in turn determine design choices, tolerance specifications, and desired nuclear data uncertainties.

4. Design optimization

Multi-objective design optimization (MDO) is in broad use in many engineering fields. The integration and automation of physics capabilities in the ARMI code system enables MDO on advanced nuclear reactor design.

4.1. Parameter sweeps

In conceptual and preliminary design, engineers produce a series of hypotheses aimed at improving product performance. The ARMI parameter sweep capabilities offer a powerful tool for analyzing and evaluating such hypotheses. The user specifies the bounds of a multidimensional region of interest in terms of dimensions, compositions, fuel management parameters, power level, or any other input. The ARMI code system then samples the design space and runs full analyses at many points.

For the TWR design, the ARMI code system is capable of computing the critical equilibrium fuel cycle (with coupled fuel performance, etc.), generating reactivity coefficients, and running a set of design-basis or beyond design-basis transients. This systems-level degree of automation turns out to be extremely valuable when used in conjunction with a supercomputer.

4.2. Surrogate model

As expected due to nuclear reactor complexity, ARMI simulations contain many independent and dependent variables. To evaluate various tradeoffs (e.g., in a generic function minimization routine), a fast-acting surrogate model trained on the results of a parameter sweep is necessary. For this, the ARMI code system uses the alternating conditional expectation (ACE) [16], a nonlinear, non-parametric regression model. ACE is a purely statistical model that is capable of teasing out dependencies that other forms of regression struggle with. In addition to producing accurate and fast regression models for optimization, the intermediate transform functions of the independent variables provide analysts with deep intuition about the shape and relative magnitudes of, for example, how changes in fuel dimensions (fuel smear density) and feed enrichment affect the critical equilibrium cycle length in a TWR design versus a reduction of the radial core size (number of assemblies), as demonstrated in Fig. 5.

The ARMI code system performs one ACE regression of each of the n dependent variables functions of m independent variables. Once built, this set of regressions represents a continuous surrogate of the parameter sweep. Any provided combination of design inputs instantly yields a full set of reactor performance metrics.

4.3. Physical programming

Given the rapid surrogate model of the design space, the actual MDO process begins. Established optimization methods operate robustly, but decisions must be made to weight the multiple objective functions. Tradeoffs are fundamental to engineering, but specifying meaningful relative weighting of levelized cost of electricity against core damage frequency is not straightforward, although results are highly sensitive to the choice of weights. Accordingly, the ARMI code system implements physical programming [17], an algorithm that translates meaningful engineering preferences into consistent weighting functions. The resulting aggregate objective function may then be optimized using standard methods to identify optimal design tradeoffs.

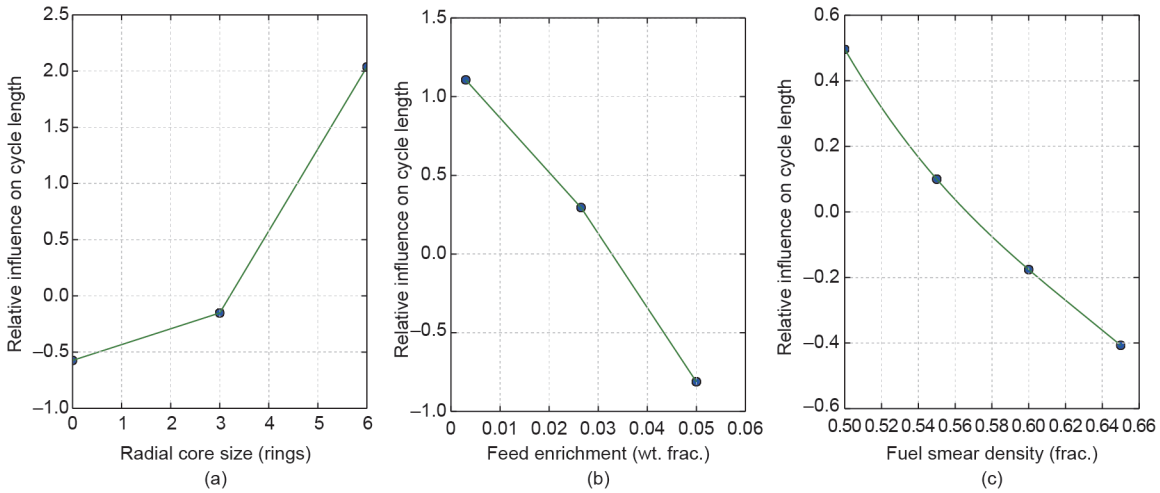

Fig. 5. Results of an ACE run showing the dependence of three design variables on the critical equilibrium cycle length in a TWR design. The images together show that (a) radial core size (number of assemblies) is the dominant variable and more assemblies lead to longer cycle lengths; (b) higher feed enrichment (up to 5% ^{235}U) reduces cycle length, although less rapidly than core size; and (c) increasing fuel smear density reduces cycle length, but only by a fraction of what the other two controls can do. The intermediate ACE transform variables are shown on the y-axis.

4.4. Speed of design iterations

The speed with which reactor designers can complete a design is often limited by the time between a proposed change and the results of the corresponding analysis. The ARMI code system enables rapid full-system feedback and, with it, rapid design progression. The ARMI code system typically provides 3D equilibrium analysis feedback within roughly 2 h and detailed cycle-by-cycle results over a 40-year plant lifetime within 8 h. Thus, major design iterations may proceed on a daily basis as necessary.

5. Quality assurance

Computer code verification and validation (V&V) is a critical step, both in that it demonstrates satisfactory compliance with design requirements during the preliminary phase of reactor design, and in that it supports the definitive safety analysis required for licensing. The ARMI software is controlled under an internal procedure guided by the ASME NQA-1-2008 [18] and NQA-1a-2009 [19] standards.

5.1. Verification tests

The requirements of each ARMI module are proposed and documented based on identified reactor design, analysis, and licensing needs. A test suite comprising unit tests and in-use tests has been developed to demonstrate the satisfaction of each requirement. Each test is explicitly linked to one or more requirements, and an automated requirements traceability matrix is maintained. Identifying regressions due to ongoing development is an essential secondary use of this test suite.

5.2. Experimental validation

The validation of advanced reactor models against neutronics experiments is challenged by the relative shortage of high-quality data. Some of the best data have been compiled by the International Reactor Physics Experiment Evaluation Project (IRPhEP) [20]. The first phase of validation for the TWR is ongoing, and involves the evaluation of various experimental benchmarks from the IRPhEP, supplemented by other experimental data obtained on contract. Key facilities identified as relevant for TerraPower's first reactor are discussed in Ref. [21]. A satisfactory benchmark comparison of the spatial ^{235}U reaction rates in the BFS-73-1 experiment from the IRPhEP is shown in Fig. 6.

Advanced reactors have historically relied upon measurements from dedicated critical mockups of their core. Today, previous experience, high-fidelity modeling, and more complete nuclear data evaluations permit the consideration of relying only upon historical experiments for nuclear validation. To this end, a strong argument must be built demonstrating the applicability of the experiments to the target reactor system. This argument can be built from a nuclear standpoint by comparing sensitivity coefficients between nuclear data and integral measurements, as formalized in the representativity factor [9]. Nuclear data covariance available in the ENDF/B-VII.1 library has progressed to the point that meaningful estimations of data uncertainty can be made in software, although additional development is required.

Nuclear experiments aside, it must be noted that much additional work is required in building experimental databases to complement models. Associated with the V&V activities, TerraPower is carrying out a large array of technology development-testing programs at dozens of institutions. These cover fuel and materials development and testing, full-size fuel bundle manufacturing and sodium flow tests, controls and diagnostics, and many others [3]. For example, the test rig at the TerraPower laboratory, shown in Fig. 7, applies forces and temperature gradients to full-size assemblies and precisely measures the displacements in order to validate the OXBOW module.

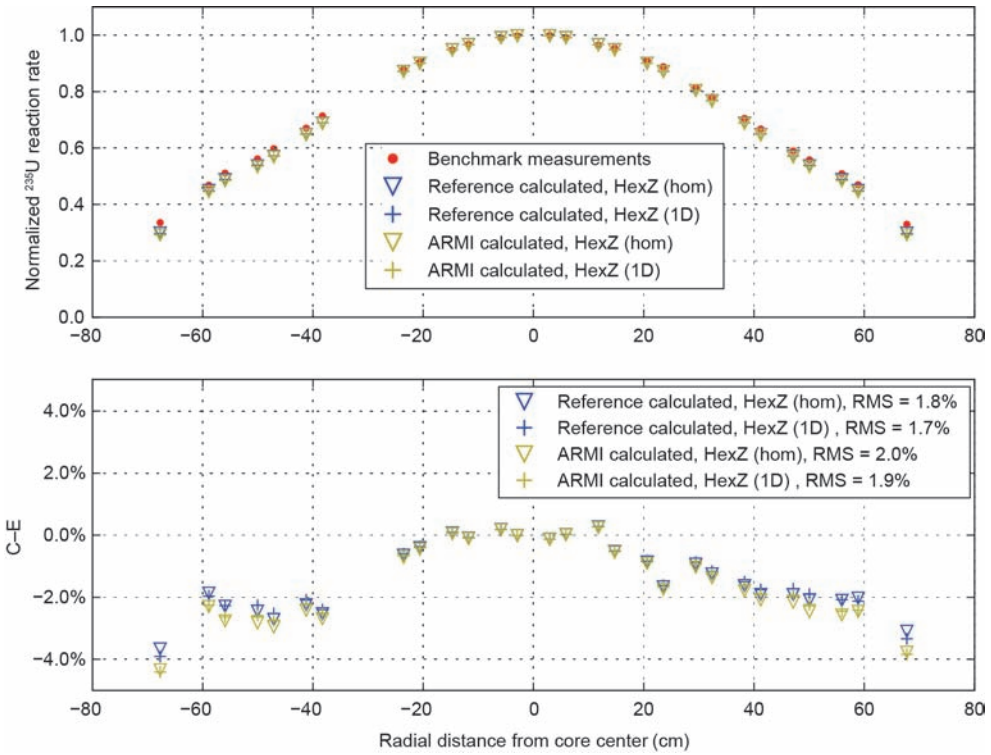

Fig. 6. A comparison of calculated (C) and experimentally measured (E) ^{235}U reaction rates (normalized) in the BFS-73-1 experiment. Calculated results include 0-dimensional homogenized (hom) and one-dimensional (1D) cross-section treatments in hexagonal prism (HexZ) geometry in hand-built reference and ARMI-generated models. Root mean squared (RMS) deviations from the experimental values below 2% were observed.

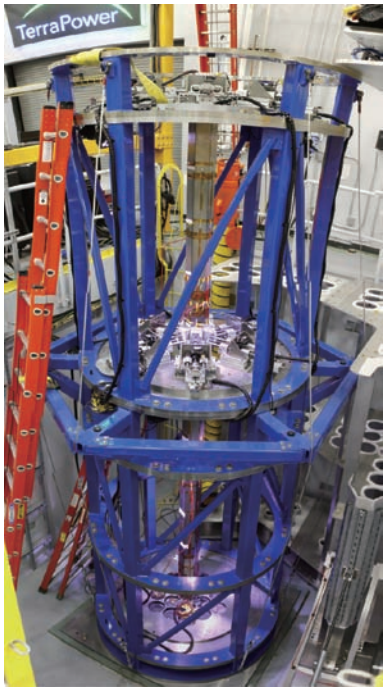

Fig. 7. A test rig at the TerraPower laboratory used for validating OXBOW bowing calculations.

5.3. Continuous integration and policies

The ARMI code system is still under heavy development. It is imperative to both maintain the quality and retain the value of V&V activities done on previous releases. The ARMI development team requires code review of all software changes in the ARMI code system by an appropriately skilled engineer. Our source code repository is hosted internally on a system that facilitates code reviews, access controls, and other policies. The ARMI development team only permits code from the released code branch to be used in formal design analysis. Desired reactor analyses often precede released new features, and cutting-edge or exploratory analysis is permitted on prototypic code branches; however, any results that lead to a design decision must be supported by the released code.

The ARMI development team utilizes continuous integration (CI) techniques, borrowed from the software industry, which perform automated building, testing, and health-checking of the codes that are implemented for this purpose. The CI system automates many tasks, including the execution of tests upon code review request. Once new code is approved by the reviewer and goes live, the CI system executes the full test suite and provides feedback to update a baseline for future changes. Every night, the CI system also executes tests to identify any environment changes. Finally, to ensure that prior V&V activities remain valid, the CI system executes the full validation suite and the current baseline reactor design cases. Various code health and stability metrics are tracked in detail.

6. Conclusions

TerraPower has applied modern software engineering practices to build an integrated reactor design system. This system grew from a fuel management routine into a full-system framework capable of rapid, high-fidelity, and consistent analysis. While many of the physics solvers currently in use are relevant for sodium-cooled fast reactors, numerous general modules of the system are applicable to a wide variety of reactors, from the conceptual phase through the detailed engineering design, and on to operation. Validation and verification work is ongoing, both in software quality and in the evaluation of past and current experiments. The ARMI code system has proven to be an invaluable tool as the TWR design moves toward licensing and construction, and will support TWR plant operation in the future. It is also providing value as new innovative concepts mature.

Nuclear reactor design, construction, and operation have relied heavily on computer systems and will continue to do so. Software models started simply, gained sophistication, and are now acting as truly integrated full-system tools. Future systems will undoubtedly feature artificial intelligence, leveraging integrated models such as the ARMI code system to a level at which physically consistent conceptual designs can be fully automated, given user-input basic characteristics and performance targets. It is possible to envision the operating plants that will use such models being coupled to instrumentation and control systems in order to ensure smooth, efficient, and safe operations.

Acknowledgements

The authors would like to acknowledge their colleagues for providing numerous technical and editorial contributions to this paper, including Zhiwen Xu, Mark Werner, Mark Onufer, Pavel Hejzlar, Robert Petroski, Jesse Cheatham, Bao Truong, Sam Miller, Chris Gross, Dustin Langewisch, Evan Albright, Peter McNabb, and others.

Compliance with ethics guidelines

Nicholas W. Touran, John Gilleland, Graham T. Malmgren, and Charles Whitmer are employed by TerraPower, LLC, which is working to commercialize traveling wave reactor technology. William H. Gates III is the TerraPower chairman.

Nicholas W. Touran, John Gilleland, Graham T. Malmgren, Charles Whitmer, and William H. Gates III declare that they have no conflict of interest or financial conflicts to disclose.

References

[1] Haigh T, Priestley M, Rope C. ENIAC in action: Making and remaking the modern computer. Cambridge: MIT Press; 2016.
[2] Greenspan H, Kelber CN, Okrent D, editors. Computing methods in reactor physics. New York: Gordon and Breach Science Publishers; 1968.
[3] Gilleland J, Petroski R, Weaver K. The traveling wave reactor: Design and development. Engineering 2016;2(1):88–96.
[4] Gamma E, Helm R, Johnson R, Vlissides H. Design patterns: Elements of reusable object-oriented software. Boston: Addison-Wesley Longman Publishing Co., Inc; 1995.
[5] Lee C, Yang WS. MC^2-3: Multigroup cross section generation code for fast reactor analysis. Nuclear Sci Eng. Epub 2017 Jun 30.
[6] Derstine KL. DIF3D: A code to solve one-, two-, and three-dimensional finite-difference diffusion theory problems. Report. Illinois: Argonne National Laboratory; 1984 Apr. Report No.: ANL-82-64.
[7] Olson A. A users guide for the REBUS-PC code, version 1.4. Illinois: Argonne National Laboratory; 2001 Dec. Report No.: ANL/RERTR/TM-32. Contract No.: W-31-109-ENG-38.
[8] Hosteny RP. ARC system fuel cycle analysis capability, REBUS-2. Report. Illinois: Argonne National Laboratory; 1978 Jan. Report No.: ANL-772-1.
[9] Chadwick MB, Herman M, Obložinský P, Dunn ME, Danon Y, Kahler AC, et al. ENDF/B-VII. 1. Nuclear data for science and technology: Cross sections, covariances, fission product yields and decay data. Nucl Data Sheets 2011;112(12):2887–996.
[10] Salvatores M, Palmiotti G, Aliberti G, Archier P, De Saint Jean C, Dupont E, et al. Methods and issues for the combined use of integral experiments and covariance data: Results of a NEA international collaborative study. Nucl Data Sheets 2014;118:38–71.
[11] Pelowitz DB. MCNPX user's manual version 2.5.0. Los Alamos: Los Alamos National Laboratory; 2005 Apr. Report No.: LA-CP-05-0369.
[12] Ellis T, Petroski R, Hejzlar P, Zimmerman G, McAlees D, Whitmer C, et al. Traveling-wave reactors: A truly sustainable and full-scale resource for global energy needs. In: Proceedings of ICAPP'10; 2010 Jun 13–17; San Diego, USA. Illinois: American Nuclear Society; 2010. p. 10189.
[13] Miller SJ, Latta R. Fuel performance modeling of traveling wave reactor fuel elements [poster session]. In: Materials Modeling and Simulation for Nuclear Fuels 2013 Workshop; 2013 Oct 14–16; Chicago, IL, USA. 2013.
[14] Cohen M, Werner M, Johns C. Mechanical model of a TerraPower, traveling wave

reactor fuel assembly duct. In: 22nd International Conference on Structural Mechanics in Reactor Technology; 2013 Aug 18–23; San Francisco, CA, USA. Red Hook: Curran Associates, Inc.; 2014.
[15] Dunn FE, Prohammer FG. The SASSYS LMFBR systems analysis code. Math Comput Simul 1984;26(1):23–6.
[16] Breiman L, Friedman JH. Estimating optimal transformations for multiple regression and correlation. J Am Stat Assoc 1985;80(391):580–98.
[17] Messac A. Physical programming-effective optimization for computational design. AIAA J 1996;34(1):149–58.
[18] American Society of Mechanical Engineers. Quality assurance requirements for nuclear facility applications. NQA-1-2008. New York: American Society of Mechanical Engineers; 2008.
[19] American Society of Mechanical Engineers. Addenda to ASME NQA-1-2008. Quality assurance requirements for nuclear facility applications. NQA-1a-2009. New York: American Society of Mechanical; 2009.
[20] OECD Nuclear Energy Agency. International handbook of evaluated reactor physics benchmark experiments. Paris: OECD Nuclear Energy Agency; 2009.
[21] Xu Z, Onufer M, Touran N. Reactor physics benchmark suite development for nuclear analysis codes and methods validations in travelling wave reactor applications. In: Proceedings of the Physics of Reactors Conference: Unifying Theory and Experiments in the 21st Century (PHYSOR 2016); 2016 May 1–5; Sun Valley, ID, USA. Illinois: American Nuclear Society; 2016. p. 362–76.