NUCLEAR TECHNOLOGY · VOLUME 207 · iii–viii · JULY 2021

DOI: https://doi.org/10.1080/00295450.2021.1915487

The MOOSE Multiphysics Computational Framework for Nuclear Power Applications: A Special Issue of *Nuclear Technology*

Richard C. Martineau

Argonne National Laboratory, Lemont, Illinois 60439

This special issue of *Nuclear Technology* centers on the Multiphysics Object-Oriented Simulation Environment (MOOSE) framework[1] and the MOOSE-based multiphysics simulation tools created for nuclear power applications. MOOSE is a development and runtime high-performance computing (HPC) framework that solves systems of partial and ordinary differential equations and is in use for simulation of nuclear power, geophysics and geochemistry, multiscale materials, advanced manufacturing techniques, and other physics applications. The complexities of parallel computing, spatial and temporal discretization and integration techniques, and nonlinear solvers are encapsulated in the MOOSE framework software package, leaving the computational scientist or engineer to focus upon the required physics. MOOSE is worldwide and used by many laboratories and universities and in industry. For this special issue, we concentrate on the applications of MOOSE-based tools created primarily under the Department of Energy's (DOE's) Office of Nuclear Energy (DOE-NE) Nuclear Energy Advanced Modeling and Simulation (NEAMS) program.

Before establishing Idaho National Laboratory (INL) as the DOE-NE laboratory in 2005, INL had no history of dedicated HPC resources and software development efforts. DOE-NE's 2003 request for proposals (RFP) required that the selected prime contractor "establish a world-class capability in the modeling and simulation of advanced systems such as Generation IV nuclear energy systems." In particular, the RFP called for the prime contractor to "develop the capability to model and simulate advanced nuclear systems from the microscopic to the macroscopic level, enabling advanced experimentation involving Generation IV technologies."

Despite the stated mandate, federal budget constraints for fiscal year (FY) 2005 did not provide for funding a "world-class capability in modeling and simulation" effort. Therefore, in 2005, INL management invested Laboratory Directed Research and Development (LDRD) funds and allowed me to begin developing INL's "capability to model and simulate advanced nuclear systems." Kathryn McCarthy, then director of INL's Nuclear Science and Engineering (NS&E) Division, made two substantial directed LDRD projects available to me, the first in October 2004 and the second in October 2007. In 2006, I recruited several top computational scientists, including Dana Knoll from the Los Alamos National Laboratory (LANL) T-3 Group. With Dana as lead, we organized the Computational Multiphysics Analysis Group in the Advanced Nuclear Energy Systems Department of INL's NS&E Division. The group's primary purpose was to investigate nonlinear Newton approaches to solve complex nuclear power applications problems. Along the way, we acquired Christopher Newman from Sandia National Laboratories (SNL) and Ryosuke Park from Georgia Insitute of Technology.

I first proposed developing an HPC development and runtime computational framework software package in one of the first Global Nuclear Energy Partnership meetings, held at Lawrence Livermore National Laboratory on December 15, 2005. While certainly not the first time a multiphysics framework was proposed, the desire to construct a multiphysics computational framework for nuclear power applications was motivated by the difficulty of traditional "code-coupling" strategies for multiphysics software. Traditional approaches can be composed of lengthy handwritten physics codes of one million or more source code lines employing various spatial discretization schemes and design philosophies with different languages, input syntax, data structures, and library dependencies. These

codes are then "coupled" together with complex interfaces to provide a multiphysics simulation capability. Even installation and compilation of a code-coupled approach can present particular difficulties.

Significant simplification of the software applications confronting computational scientists is required if complex multiphysics applications for nuclear power simulation are to be made usable. In November 2007, I recruited Derek Gaston, an outstanding computer scientist, from SNL's Sierra Framework Group, and he began working at INL in May 2008. Derek's academic experience with Professor Graham Carey of the University of Texas at Austin (UT Austin) provided the basis for the MOOSE HPC development and runtime framework. MOOSE is primarily composed of the libMesh finite element library,[2] developed at UT Austin, and the Portable Extensible Toolkit for Scientific Computation (PETSc), a highly scalable mathematical parallel nonlinear solver library developed at Argonne National Laboratory (ANL) under the DOE Office of Science (DOE-SC). PETSc (Ref. 3) is known for its many awards and user prizes, including five Gordon Bell Prizes and a 2009 R&D 100 Award. The collaboration between the ANL PETSc Group and UT Austin produced the software foundation for MOOSE.

Derek made rapid progress in developing the MOOSE framework. After two months, Derek gave his first MOOSE training to six of us at INL. After three months, in July 2008, the team demonstrated parallel computations on three-dimensional finite element domains, which resulted in the BISON nuclear fuel performance code.[4] BISON received seed funding from the Fuel Cycle Research & Development (FCRD) program and was simultaneously developed with MOOSE. Preliminary three-dimensional BISON results demonstrating strongly coupled nonlinear thermomechanics and oxygen nonstoichiometry served as a proof of concept for MOOSE's multiphysics viability. By May 2009, Michael Tonks, a postdoctoral computational materials scientist at INL, had successfully stood up the MOOSE-based MARMOT microstructure material simulation tool.[5] MARMOT is a phase-field finite element code for modeling irradiation-induced grain structure evolution, including the effects of radiation damage on microstructure evolution, void nucleation and growth, fission gas bubble growth, grain boundary migration, and gas diffusion and segregation. Also, MARMOT calculates the microstructure evolution effect on various bulk material properties, including thermal conductivity and porosity.

Dana Knoll, Derek Gaston, Cody Permann (who transferred to the MOOSE team in 2008), and Michael Tonks first developed and implemented a novel implicit approach for strongly coupling the MARMOT and BISON codes under the MOOSE framework. With MARMOT providing grain-scale calculations to predict thermomechanical material property evolution, the macroscale BISON simulation provided a lower-length-scale-informed material simulation of nuclear fuel. Later on, David Andersson of LANL's Materials Science and Technology Division implemented science-based materials capabilities by feeding atomistic and molecular dynamics predictions into MARMOT. The addition of first-principles information provides a science-based predictive capability for multiscale simulation of material evolution in nuclear fuel under irradiation, the first of its kind, "atoms to pins," predictive simulation capability.

Early external use of MOOSE and MOOSE codes began in 2012. Both the Electric Power Research Institute and Knolls Atomic Power Laboratory began evaluating MOOSE, BISON, and MARMOT. The Consortium for Advanced Simulation of Light Water Reactors (CASL) chose BISON as its nuclear fuel simulator. CASL also funded Michael Short of the Massachusetts Institute of Technology (MIT) to develop a MOOSE-based microscale CRUD code called MAMBA-BDM. Dmitry Karpeyev and colleagues from ANL stood up the Ferret, Catfish, and Condor MOOSE-based codes. Funded by LDRD, Ferret simulates the microstructure's evolution of ferroelectric materials under elastic strain. Catfish, also funded by LDRD, simulates the translocation of long chains of charged beads (modeling DNA molecules) through nanochannels. Condor was a DOE-SC SciDAC (Scientific Discovery through Advanced Computing) project that simulates the behavior of superconducting materials with inclusions (particles and columnar defects) to find the optimal inclusion geometries that result in the highest efficiency of the superconductor. The model is a coupled system of partial differential equations, including complex variable Ginzburg-Landau equation, overdamped Maxwell's equation, Poisson's equation, and temperature diffusion equation.

The main benefit of MOOSE is its simplification of multiphysics algorithmic coupling. Because all MOOSE-based software packages utilize the same programming interfaces, following identical software design and library dependencies, there is a high degree of "cohesiveness" between the MOOSE-based applications. Because of this cohesiveness, MOOSE provides a simplified path to tightly couple physics through a unique data transfer system designed explicitly for multiscale, multiphysics simulations employing multiple software codes. This data transfer approach, called MOOSE MultiApps and

Transfers, has been extensively used in the numerical investigation of multiscale phenomena where the tightly coupled physics varies in space and time by many orders of magnitude and is coupled to reactor physics, radiation transport, and thermal fluids. DireWolf is a MOOSE driver application that controls the coupling of Griffin for reactor physics, BISON for fuel performance, and the Sockeye heat pipe code to simulate heat pipe–cooled microreactors. Preliminary results for DireWolf are presented in this special issue. The BISON–MARMOT multiscale nuclear fuel performance application relies heavily upon MOOSE MultiApps and Transfers for tightly coupled calculations. Daniel Schwen of INL replaced the older process of coupling BISON and MARMOT with MOOSE MultiApps and Transfers. Furthermore, Daniel rewrote and optimized the finite element solution method for MARMOT's phase-field equations, resulting in magnitude improvement in runtime speed.

Over time, the MOOSE team grew from a single person to 11 people in 2020. Key people were added to the team starting in late 2008 with Cody Permann, who now leads the MOOSE team. An exceptional find was David Andrs, a postdoctoral student from the University of Nevada, Reno. David made significant software design and structure contributions in MOOSE and as a software lead for several MOOSE-based codes. Alexander Lindsay and Fande Kong have significantly increased the MOOSE framework's features, functionality, and performance through advanced optimization and preconditioning techniques. MOOSE team members, including Robert Carlson and Logan Harbor, also spend significant time supporting the MOOSE-based physics tools and acting as an interface between the tools and the framework. The MOOSE team has been the recipient of multiple INL Laboratory Director Awards and two Presidential Early Career Award for Scientists and Engineers (PECASE) awards. Derek Gaston received INL's first PECASE award in 2012, and Michael Tonks, who had since moved to the University of Florida, received his PECASE award in 2017 for his work on MARMOT. The MOOSE framework received an R&D 100 Award in July 2014. INL obtained an open-source (LGPL 2.1) license for MOOSE in February 2014.

In October 2013, I wrote another LDRD proposal and was awarded $1.35 million over three years to develop a MOOSE-based radiation transport simulation capability called Rattlesnake (Ref. 6). I quickly turned over the project to individuals who had the expertise: Mark DeHart, a reactor physicist, became project lead, and Yaqi Wang, an expert in radiation transport, was the lead software developer. The Rattlesnake radiation transport code's primary design goal is to tightly couple with the MOOSE-based multiscale nuclear fuel performance capability, BISON–MARMOT, to resolve high-burnup structures (HBSs) in all forms of nuclear fuel. Building this simulation capability to resolve the fuel's material state is especially important when considering accident tolerant fuel concepts. HBS material states are where the fuel's most damaged condition exists and define the fuel's ability to withstand harsh environments without failing. In addition to Rattlesnake, the DOE-NE NEAMS point of contact, Trevor Cook, assigned INL an additional $400 000 in October 2013 to develop the MOOSE-based MAMMOTH reactor physics capability for cross-section generation, burnup, depletion, and isotopic distribution. Javier Ortensi, an INL reactor physicist, has been the leader of MAMMOTH since its inception. In August 2019, Tanju Sofu of ANL and I negotiated a joint effort for ANL and INL to combine their separate and disparate reactor physics and radiation transport code efforts, MAMMOTH/Rattlesnake and MC^2-3/PROTEUS (Ref. 7), respectively, into an improved, unified capability integrated into a new MOOSE-based reactor physics and radiation transport software package called Griffin. ANL and INL split Griffin copyright ownership and the dedicated NEAMS funding for development.

An expansion of nuclear power simulation tools developed on the MOOSE framework took place under DOE-NE's Light Water Reactor Sustainability (LWRS) program, where Kathryn McCarthy also served as technical integration officer. Three MOOSE-based codes, designed to support the Risk-Informed Safety Margin Characterization (RISMC) Pathway of the LWRS program, were initiated to help predict the current light water reactor fleet's life extension. The first was the next-generation systems analysis code, called RELAP-7 (Ref. 8). The most significant development goal of RELAP-7 is to take advantage of the previous 30 years of advancements in computer architecture, software design, numerical methods, and physical models to provide capabilities needed to support nuclear power plant (NPP) safety analysis. ANL has also built an NPP analysis capability upon MOOSE for nearly incompressible single-phase liquid-cooled reactor concepts, called Systems Analysis Module (SAM) (Ref. 9). SAM is currently being used to analyze NPPs with sodium fast reactors and fluoride salt–cooled high-temperature reactors (FHRs). Being MOOSE-based, both RELAP-7 and SAM are easily coupled through the MOOSE MultiApps and Transfers system to other MOOSE-based codes, such as the BISON fuel performance code to analyze

nuclear fuel in a plant setting. SAM has also received a 2019 R&D 100 Award.

The second LWRS effort is the Grizzly structural mechanics code designed to account for component aging and damage evolution due to irradiation and corrosion effects.[10] Benjamin Spencer, who came to our team from SNL, has led the Grizzly effort from the beginning. Ben also created an open-source version of Grizzly called BlackBear for simulating concrete degradation issues associated with concrete containment vessels. Further, Ben delivered MOOSE's discrete fracture mechanics capability and introduced the Extended Finite Element Method (XFEM) discretization scheme into the MOOSE framework. Grizzly's fracture mechanics models are used in reactor pressure vessel (RPV) simulations verified against analytical solutions. Grizzly's RPV steel embrittlement model is based on an extensive experimental data set, and its integral RPV capability has been benchmarked against another code. Grizzly and BlackBear concrete models have been validated against laboratory experiments of concrete subjected to accelerated aging. Grizzly now enjoys NEAMS funding for support.

The third LWRS code, complimentary to Grizzly, is MASTODON (Ref. 11), a MOOSE-based application that analyzes three-dimensional soil–structure system response to natural hazards such as earthquakes and floods. This effort began with Justin Coleman as project lead and Swetha Veeraraghavan as lead MASTODON developer. MASTODON has the capability to perform extensive "source-to-site" simulations, including earthquake fault rupture, nonlinear wave propagation, and nonlinear soil–structure interaction analysis. The unique capability of MASTODON is that as a MOOSE-based application, it seamlessly couples with other physics-based applications, such as Grizzly. This allows for modeling the NPP's structural response to earthquake scenarios. MASTODON stochastically models and simulates natural hazards and phenomena at NPPs and other critical nuclear infrastructure facilities. MASTODON now enjoys several funding sources, including the National Nuclear Security Administration.

The Pronghorn (Ref. 12) nuclear reactor full-core conjugate heat transfer (CHT) simulator began under the MOOSE LDRD in September 2008 with the idea of simulating high-temperature gas-cooled reactors (HTGRs). Pronghorn's design goal is a coarse-mesh multidimensional reactor simulator that will execute rapidly on high-end workstations. Resolving computationally expensive lower-length-scale phenomena, such as boundary layer theory, is avoided, thus relying on closure relations to model lower-length-scale physics. Further simplification of physics is achieved by homogenization, in which each computational cell represents a mixture of solid-state material (fuel, moderator, or core internals) and reactor coolant. Pronghorn bridges the spatial length scales between high-resolution lower-length-scale calculations and plant-scale systems calculations. Pronghorn's homogenized CHT approach resolves large-scale flow features with tightly coupled solid-state heat transfer and homogenizes small-scale flow features and solid-state heat conduction as a mixture.

Initial programmatic funding from the DOE's Next Generation Nuclear Plant project furthered Pronghorn's development in FY 2010 and FY 2011. Pronghorn development then sat dormant until 2016, when April Novak from the University of California, Berkeley, incorporated Pronghorn into her graduate research on "Multiscale Thermal-Hydraulic Methods for Pebble Bed Reactors" funded under DOE-NE's Nuclear Energy University Program (NEUP). April updated Pronghorn to current MOOSE standards and added significant features to simulate FHRs with pebble bed cores. Since then, Pronghorn has obtained significant funding from the NEAMS program and the U.S. Nuclear Regulatory Commission (U.S. NRC). Sebastian Schunert and Paolo Balestra of INL have made significant improvements to Pronghorn for an advanced HTGR reactor plant simulator by incorporating features to model the reactor cavity cooling system, RPV, barrel gaps, reflector heat transfer, and tight coupling to RELAP-7 for modeling one-dimensional flow in the control rod and riser channels inside a three-dimensional reactor domain. The DOE-NE Center of Excellence for Thermal Fluid Applications in Nuclear Energy, led by Elia Merzari of Pennsylvania State University, is funding April Novak and Aidyin Karahan of ANL to provide Pronghorn with a multidimensional subchannel capability. A NEUP funds Elia Merzari to improve the porous flow capability in Pronghorn.

A significant recent advancement in MOOSE multiphysics coupling capability is to tightly couple external (non-MOOSE-based) codes to MOOSE-based codes. Adding this coupling capability was motivated by the availability of numerous high-quality, high-performance physics software packages algorithmically optimized for a given set of governing equations. Cody Permann and Derek Gaston from INL and MIT graduate student Matt Ellis developed a method called MOOSE-Wrapped Apps, which utilizes MOOSE MultiApps and Transfers, along with a minimal application programmer interface, to treat external (non-MOOSE-based) codes as if they were MOOSE-based—in, effectively, a "cohesive-like" manner.

Matt Ellis and April Novak explored several ways to couple OpenMC, an open-source Monte Carlo radiation transport code developed by MIT and ANL, to BISON. Matt also worked with Ronald Rahaman of ANL to couple ANL's state-of-the-art open-source computational fluid dynamics code, Nek5000, a spectral element solver supported under the NEAMS program, to BISON. The Cardinal MOOSE-Wrapped App[13] controls the multiphysics integration between BISON, OpenMC, and Nek5000 to deliver coupled high-resolution multiphysics simulations in pebble bed reactors. OpenMC provides the power distribution within the pebbles, which is then transferred to BISON for accurate fuel performance calculations. The highly turbulent coolant flow field and heat transfer are solved using a low-Mach-number approximation by Nek5000. BISON and Nek5000 are coupled through solution exchange of heat flux and temperature at the boundary. Another significant effort is BlueCRAB, the U.S. NRC's MOOSE-Wrapped App[13] for advanced reactor concepts and advanced fuel analysis. BlueCRAB is composed of several NEAMS tools and the U.S. NRC's light water reactor simulation capability. The U.S. NRC's NPP system and safety code, TRACE (Ref. 14), is coupled with BISON using the MOOSE-Wrapped App approach to investigate accident tolerant fuel performance.

There were many key moments in MOOSE development over the years since 2008, and I will acknowledge a few more here. In 2010, Richard Williamson took over as BISON team lead, where he remained for nearly a decade of brilliant leadership. Steve Novascone, who has been with the BISON team since its inception, is now the BISON project lead. Steve has led metallic fuel capability development, performed significant validation work on BISON, and led much of BISON's software quality effort. In October 2010, I brought in Jason Hales from SNL. Within a few weeks, Jason and Derek Gaston implemented an implicit multibody contact finite element scheme into MOOSE. This algorithmic development allowed BISON to simulate three-dimensional nuclear fuel pellet–cladding interaction, a large step forward for DOE-NE and its nuclear fuel simulation programs. This accomplishment by Jason and Derek solidified MOOSE and BISON in DOE-NE modeling and simulation. In 2012, Giovanni Pastore joined the BISON group and, based on his PhD work,[15] quickly developed powerful, state-of-the-art fission gas modeling capabilities in BISON. Giovanni led the BISON team during 2016–2017 and is responsible for much of the light water reactor accident modeling capability in the code.[16,17] Andrew Slaughter has been instrumental in building MOOSE software quality assurance tools that have had meaningful impacts on Nuclear Quality Assurance (NQA-1)–compliant development as well as the NQA-1 standards themselves. He has authored many other tools assisting with input file generation, visualization, and stochastic analysis. Because of the hard work of Andrew Slaughter, Steve Novascone, and Cody Permann, MOOSE and BISON passed a rigorous audit in February 2020 and are recognized as NQA-1–compliant safety codes.[18] In 2014, then graduate student Andrea Jokisaari (now a member of the MARMOT team) modernized MOOSE's solid-mechanics module to take on a more generalized tensor-based implementation, called Tensor Mechanics. Andrew Wilkins (from Australia's Commonwealth Scientific and Industrial Research Organisation) and Michael Tonks added needed capability for micromechanics and geomechanics. Several years later, Stephanie Pitts and Daniel Schwen optimized its performance and added BISON's capability to begin using the Tensor Mechanics module.

Finally, the recent development of a powerful new MOOSE feature, an automatic differentiation (AD) capability, dramatically simplifies and speeds up MOOSE-based code development. An efficient solution for nonlinear equations using Newton's method requires the accurate computation of the Jacobian matrix. However, obtaining analytical derivatives of the discretized residuals can be time-consuming and error-prone. To reduce the burden on application developers, MOOSE has implemented forward-mode AD, which correctly handles chain rule differentiation with an arbitrary number of variables. Using this capability, application developers implement only residual statements, reducing the amount of code and time needed to implement a new physics object. Sharing physics code objects among applications can now be accomplished with greater confidence. In this way, AD enhances the modularity of MOOSE-based applications. Derek Gaston and Roy Stogner (libMesh lead) initiated the effort in 2016. Alexander Lindsay matured and optimized the AD software packages and implemented AD for finite volume applications. Daniel Schwen took advantage of AD and created the ability to describe equations in a text file that runs on native hardware. LANL's Topher Mathews tested and verified MOOSE's AD against highly nonlinear thermomechanical analysis and pushed it into MOOSE modules.

To summarize MOOSE's success, both national and international laboratories and universities have stood up more than 60 known MOOSE-based applications and over 70 000 known MOOSE framework build package downloads since 2011. MOOSE-based software applications are in development for nuclear power (radiation transport, reactor physics, nuclear plant safety, and systems analysis; CRUD growth and effects; and multiscale nuclear fuels performance), materials (fundamental materials

development; effects of corrosion, damage, and aging evolution; and irradiated material analysis), structural dynamics, multiphase flow, waste analysis, geophysics (seismic, geothermal, geochemistry, and isotope transport), and advanced digital manufacturing (laser welding and spark plasma sintering). Success can always be attributed to forward-thinking leadership. The NEAMS program was blessed with Shane Johnson, deputy assistant secretary for DOE-NE reactor fleet and advanced reactor deployment, and Daniel Funk, DOE-NE director of advanced modeling and simulation. These two federal program managers were fully integrated into the NEAMS program's technical efforts by LANL's Christopher Stanek, national technical director for NEAMS; INL's Steve Hayes, who led the NEAMS fuels product line; and ANL's Tanju Sofu, who led the NEAMS reactors product line.

The following 13 papers in this special issue are dedicated to MOOSE-based software development efforts for nuclear power applications under DOE-NE's NEAMS program.

References

1. C. J. PERMANN et al., "MOOSE: Enabling Massively Parallel Multiphysics Simulation," *SoftwareX*, **11**, 100430 (2020); https://doi.org/10.1016/j.softx.2020.100430.
2. B. S. KIRK et al., "libMesh: A C++ Library for Parallel Adaptive Mesh Refinement/Coarsening Simulations," *Eng. with Comput.*, **22**, *3–4*, 237 (2006); https://doi.org/10.1007/s00366-006-0049-3.
3. S. BALAY et al., "PETSc Users Manual," ANL95/11, Rev. 3.13, Argonne National Laboratory (2020); https://www.mcs.anl.gov/petsc (current as of Apr. 28, 2021).
4. R. WILLIAMSON et al., "Multidimensional Multiphysics Simulation of Nuclear Fuel Behavior," *J. Nucl. Mater.*, **423**, *1–3*, 149 (2012); https://doi.org/10.1016/j.jnucmat.2012.01.012.
5. M. R. TONKS et al., "An Object-Oriented Finite Element Framework for Multiphysics Phase Field Simulations," *Comput. Mater. Sci.*, **51**, *1*, 20 (2012); https://doi.org/10.1016/j.commatsci.2011.07.028.
6. Y. WANG et al., "Hybrid P_N-S_N with Lagrange Multiplier and Upwinding for the Multiscale Transport Capability in Rattlesnake," *Prog. Nucl. Energy*, **101**, 381 (2017); https://doi.org/10.1016/j.pnucene.2017.03.020.
7. Y. S. JUNG, C. LEE, and M. A. SMITH, "PROTEUS-MOC User Manual," Rev. 0, ANS/NSE-18/10, Argonne National Laboratory (2018); https://doi.org/10.2172/1483947.
8. H. ZHANG et al., "RELAP-7 User's Guide," Rev. 1, INL/EXT-14-33977, Idaho National Laboratory (2018); https://doi.org/10.2172/1466669.
9. R. HU, "A Fully-Implicit High-Order System Thermal-Hydraulics Model for Advanced Non-LWR Safety Analyses," *Ann. Nucl. Energy*, **101**, 174 (2017); https://doi.org/10.1016/j.anucene.2016.11.004.
10. B. SPENCER, W. HOFFMAN, and M. BACKMAN, "Modular System for Probabilistic Fracture Mechanics Analysis of Embrittled Reactor Pressure Vessels in the Grizzly Code," *Nucl. Eng. Des.*, **341**, 25 (2019); https://doi.org/10.1016/j.nucengdes.2018.10.015.
11. J. COLEMAN et al., "MASTODON Theory Manual," INL/EXT-17-41930, Idaho National Laboratory (2017); https://doi.org/10.2172/1376857.
12. A. NOVAK et al., "Validation of Pronghorn Friction-Dominated Porous Media Thermal-Hydraulics Model with the SANA Experiments," *Nucl. Eng. Des.*, **350**, 182 (2019); https://doi.org/10.1016/j.nucengdes.2019.04.037.
13. R. MARTINEAU et al., "Multiphysics for Nuclear Energy Applications Using a Cohesive Computational Framework," *Nucl. Eng. Des.*, **367**, 110751 (2020); https://doi.org/10.1016/j.nucengdes.2020.110751.
14. "TRACE V5.0 Theory Manual: Field Equations, Solution Methods, and Physical Models," U.S. Nuclear Regulatory Commission (2010); http://www.nrc.gov/docs/ML1200/ML120060218.pdf (current as of Apr. 28, 2021).
15. G. PASTORE et al., "Physics-Based Modelling of Fission Gas Swelling and Release in UO_2 Applied to Integral Fuel Rod Analysis," *Nucl. Eng. Des.*, **256**, 75 (2013); https://doi.org/10.1016/j.nucengdes.2012.12.002.
16. G. PASTORE et al., "Analysis of Fuel Rod Behavior During Loss-of-Coolant Accidents Using the BISON Code: Cladding Modeling Developments and Simulation of Separate-Effects Experiments," *J. Nucl. Mater.*, **543**, 152537 (2021); https://doi.org/10.1016/j.jnucmat.2020.152537.
17. G. PASTORE et al., "Analysis of Fuel Rod Behavior During Loss-of-Coolant Accidents Using the BISON Code: Fuel Modeling Developments and Simulation of Integral Experiments," *J. Nucl. Mater.*, **545**, 152645 (2021); https://doi.org/10.1016/j.jnucmat.2020.152645.
18. "MOOSE Framework and BISON Software Quality Assessment—2019," ASMT-2020-0436, Idaho National Laboratory (2019).