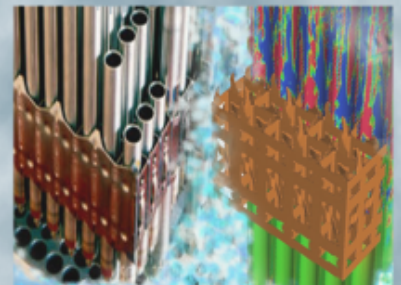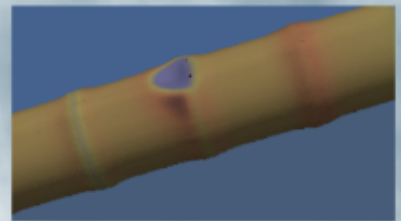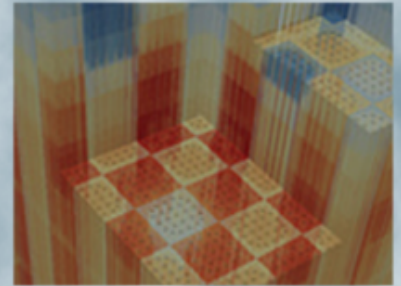# Bison Improvements for Robustness and Speed L2:FMC.P16.01

Benjamin W. Spencer, INL
Richard L. Williamson, INL
Alexander D. Lindsay, INL
Fande Kong, INL
Russell, J. Gardner, INL
Jason D. Hales, INL
Albert Casagranda, INL
Daniel Schwen, INL
Hailong Chen, INL
Naveen Prakash, LANL
Christopher Matthews, LANL
Cetin Unal, LANL

June 29, 2018

U.S. DEPARTMENT OF ENERGY | Nuclear Energy

<Document Name>

REVISION LOG

Revision	Date	Affected Pages	Revision Description
0	06/29/2018	All	Initial Release

Document pages that are:

Export Controlled _____NONE_____

IP/Proprietary/NDA Controlled_____NONE_____

Sensitive Controlled_____NONE_____

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Requested Distribution:

To: FMC FA Lead

Copy: CASL PM

Bison Improvements for Robustness and Speed

Benjamin W. Spencer[1], Richard L. Williamson[1], Alexander D. Lindsay[2],
Fande Kong[2], Russell J. Gardner[1], Jason D. Hales[1], Albert Casagranda[1], Daniel Schwen[1],
Hailong Chen[1], Naveen Prakash[2], Christopher Matthews[3], Cetin Unal[4]

[1]Fuels Modeling and Simulation
[2]Modeling and Simulation
Idaho National Laboratory
P.O. Box 1625
Idaho Falls, ID 83415-3840

[2]T-3: Fluid Dynamics and Solid Mechanics
[3]MST-8: Materials Science in Radiation & Dynamics Extremes
[4]W-13: Advanced Engineering Analysis
Los Alamos National Laboratory
PO Box 1663
Los Alamos, NM 87545

June 29, 2018

Introduction

Early in its development, the primary emphasis of efforts to improve the Bison fuel performance code were focused on initial development of new capabilities. As Bison has matured and become adopted by a wider set of users, it has become clear that a number of robustness issues needed to be addressed to ensure that the code can reliably produce a converged solution. This need became particularly evident when a recent effort by Westinghouse Electric Company to simulate all of the fuel rods in a reactor using Bison resulted in approximately 30% of the models failing to run to completion because of lack of convergence or other errors that were encountered.

It is important to emphasize that there was no evidence that Bison generated results that were incorrect for the set of models exercised. In many cases, issues such as those encountered could be addressed by adjusting solver-related parameters. However, there is no question that for Bison to become widely adopted for production work, it needs to be inherently far more robust than it was. Because of this, a concerted effort was undertaken to improve Bison's robustness, or its ability to reliably obtain converged solutions under a wider variety of conditions.

As a result of this work, a number of high-priority areas for improvement for modeling light water reactor (LWR) fuel rods have been identified and addressed. As a result of this work, the number of failing Bison analyses in the aforementioned set of simulations for a full core has been reduced to a fraction of a percent. In addition, the robustness of Bison models using the VERA core simulator has been significantly improved. This is important because all of the Bison models representing individual fuel rods must run to completion successfully for the full core model to run to completion.

This report provides summaries of the identified sources of robustness issues in Bison and the development that was done to resolve these issues. It also summarizes the addition of new capabilities to address deficiencies in Bison's

ability to include needed aspects of LWR fuel behavior, and improvements that were made to improve Bison's solution efficiency.

1 Summary of Robustness Issues and their Resolution

1.1 Water Properties

Issue:

A library in MOOSE has been used in Bison to compute thermodynamic properties of water. This library has had occasional convergence problems. The coolant channel model that uses this library has also lacked sufficient robustness and error checking. As an example of the issues seen, some of the more complex coolant channel model regression tests were not consistent across computing platforms (Mac OS vs. Linux).

In addition, the water/steam library in MOOSE was already set to be replaced with a more complete library that will be used across MOOSE applications. This was to be done to improve consistency and ensure that licensing concerns are simplified.

Resolution:

As of mid-November 2017, all known robustness and error checking issues had been resolved. The process of moving to a new water/steam library was begun. The new library relies more on a table lookup than on iteration, which should eliminate iteration convergence issues and improve speed. The switch to the new water/steam library was begun, including evaluating differing regression test results.

In December, effort was taken to understand why three coolant channel regression tests gave different results (outside of the tolerance level) on Max workstations compared to on INL's Falcon machine. The reason for the difference was found to be a bit of logic that is used to determine whether the coolant is in the liquid, two-phase, or gas region. On one platform, one region was selected, but due to very small numerical differences, a different region was selected on a different platform. This in turn led to larger drift in the solutions to the point that the tests were not consistent across the two platforms. An improvement to the faulty logic was developed, and testing of that new logic was begun.

Just after the start of the new year, it was shown that consistent behavior for two of the three problematic tests was obtained with improved logic. The third test continued to show differences. In addition, progress was made on adopting the new water/steam library. However, much work remained to ensure the interface to the new library was correct and to review the inevitable small differences in regression tests.

Later it was decided to suspend further work on the logic that that was leading to inconsistency in determining the water/steam region. This was due to the fact that use of the new water/steam library would eliminate that logic entirely.

Today, the switch to the new water/steam library is complete. The new library has entirely replaced the previous one, giving very similar results.

It should be noted that no convergence issues with the previous water/steam library, or with the coolant channel model more generally, have been reported in many months.

1.2 Element Aspect Ratio Limits

Issue:

By their nature, LWR fuel performance models inherently have very large aspect ratios. LWR fuel rods are roughly 4m long and 1cm in diameter, which means that the fuel form being modeled has an aspect ratio of approximately 1:400. It is generally preferred to construct finite element models such that their aspect ratios are as close to 1:1 as possible. Because of the inherently large aspect ratios in LWR fuel rods, constructing the computational models to have aspect ratios close to 1:1 would lead to unreasonably high computation costs. Also, because of the fact that the gradients of solution variables such as temperature are typically strongly oriented in the radial direction, which is the direction

where LWR models are typically most refined, the element aspect ratios for LWR fuel models can be significantly higher than what would be acceptable in general.

Nevertheless, using excessively high element aspect ratios can still lead to accuracy and robustness issues. Because Bison fuel performance models account for finite strain mechanical deformation, elements with extremely high aspect ratios are more susceptible to becoming highly distorted or even inverted during the process of performing nonlinear solution iterations, which can have have a severe adverse effect on solution robustness.

During the process of debugging a set of problematic Bison axisymmetric 2D LWR fuel models with smeared representations of the fuel, it was found that those models had aspect ratios on the order of 1:1000. Simply increasing the axial refinement to reduce the aspect ratios to a more reasonable value eliminated the convergence issues that were encountered. Based on this finding, it was clear that limits should be imposed on the element aspect ratio allowed in a Bison simulation.

Resolution:

As a result of this finding, limits have been imposed on the aspect ratios allowed in a Bison analysis.

A new class called the `ElementQualityChecker` has been developed in MOOSE for this purpose. This object can be used to check any of the element quality metrics supported by the libMesh mesh library used by MOOSE. If a user wants to control an element quality metric, they can specify the metric to be checked, it upper and lower bounds, and specify whether it generates a warning or an error. The following is an example of the input for such a quality check:

```
[./elem_quality_check]
    type = ElementQualityChecker
    metric_type = STRETCH
    failure_type = WARNING
    upper_bound = 1.0
    lower_bound = 0.5
  [../]
[]
```

This is useful if a user explicitly wants to ensure that their mesh meets certain quality standards. To ensure that Bison models are always subject to a certain minimal mesh quality standards, an Action has been added to Bison to automatically generate an `ElementQualityChecker` object that generates an error if the element aspect ratio exceeds 1:200.

1.3 Non-Repeatable Solution Convergence

Issue:

In some cases, it was found that Bison models did not have repeatable solution behavior. In other words, re-running a model using the same input file on the same computer using the same number of processors did not result in an identical solution history. Not only was the convergence history different, but in some cases, the model would successfully run to completion, when in other cases it would fail due to convergence issues.

Resolution:

After this issue was reported by Bison users, a significant effort was undertaken to investigate this problem to understand which conditions must be present for this problem to occur. It was found that for a typical Bison model, this behavior is readily observable when both of the following conditions were met:

- The model is run in parallel using more than 4 processors

- A direct solver is used either for preconditioning (if the preconditioned Jacobian-free Newton Krylov method is used) or as the linear solver within a Newton method.

Bison uses the PETSc solver library, which provides access to a variety of state-of-the-art solvers and is widely used in the scientific computing community. The two parallel direct solvers commonly used in Bison are SuperLU_dist and MUMPS, and both exhibited very similar non-repeatable behavior. This behavior was much less evident when the available iterative solvers such as the algebraic multigrid (AMG) and additive Schwartz method (ASM) were used to apply the Jacobian for preconditioning.

Although the aforementioned iterative solvers did not suffer from these non-repeatability issues, the solver performance for a typical moderately-sized Bison model is far better using a parallel direct solver, so simply switching away from direct solvers is not currently a viable solution. A detailed investigation was performed on the SuperLU_dist solver to understand the reason for non-repeatable behavior.

In SuperLU_dist, the matrix is decomposed into several blocks, and each processor handles a few of those blocks. A processor usually needs some data updates from neighboring blocks owned by different processors for solving its own blocks. A calculation will proceed immediately once the required data is received. This data is communicated between processors using nonblocking communication, which means that the other processors do not need to wait for that data to be transmitted before they proceed with other tasks. This has the benefit of improving efficiency, but it also means that the order that the data is received by processors is non-deterministic, and can change due to a variety of factors related to the load on the computer and the networking system. If messages are received in a different order, then the floating point calculations performed using that data will be performed in a different order. It is well known that changing the order of mathematically associative operations can have an effect on the result when these are done using floating point representations. The sensitivity of the floating point result to the order of operations is higher for poorly conditioned operations.

Consequently, different parallel runs will have slightly different results due to changes in roundoff corresponding to changes in floating point operations. Other parallel preconditioners such as ASM and MUMPS also share this feature, and they have been confirmed to behave in a non-repeatable way when used with a sufficiently large number of processes. One has to run with a higher number of processes to see this behavior with ASM because it involves less parallel communication than SuperLU_dist.

The non-deterministic behavior due to differences in the order of floating point operations is always present in a parallel computing environment. It is an issue that cannot be easily eliminated, but can be managed. The roundoff error should not impact results unless the matrix is singular or highly ill-conditioned, where a small perturbation can be significantly amplified. The effects of ill-conditioning can be minimized by improving the matrix, reformulating the original physics, or improving the quality of the mesh.

For a user of Bison, it is important to emphasize the following:

- **Although convergence is not strictly repeatable, as long as the solution has converged within numerical tolerances it is acceptable.** The use of iterative methods is inherent to the solution of nonlinear problems. The solution is never exactly solved, but instead, iterations are performed until the residual, or imbalance in the solution, is small enough that is within tolerances selected by the user to be acceptable for the problem being solved. There is not a single unique solution that meets this criterion. Rather, an infinite set of solutions meets this criterion, but they are all close enough to each other that for engineering purposes, they can be regarded as being the same. Although it can be unnerving to see a change in the convergence history between two runs, as long as both solutions are converged, they are both acceptable within the convergence tolerances.

- **The non-repeatability only presents a practical issue if the solution is not robust.** The issue of non-repeatability only became apparent when users found that when a simulation was repeatedly run, sometimes it failed to run to completion, while other times it did not. There are no known cases where a simulation produced results that differed by an amount larger than that admitted by the solution tolerances. This statement is not being made to diminish the significance of this issue – it is clearly unacceptable for a model to not run to completion due to convergence issues, and this should be addressed. However, in determining where to allocate resources to address this problem, it is important to realize that the non-repeatability of the solution is not the underlying cause of the problem. Rather, attention should be focused on the aspects of the model that made it ill-conditioned or unreliable in the first place.

Once the causes of non-repeatability were identified, emphasis was placed on making improvements to the conditioning of the system solved by Bison and making the underlying models more robust, rather than on eliminating the non-repeatability itself. Not only will such improvements minimize the changes in the solution history from run to run, but they will make the code run faster and be less prone to failure due to non-convergence of the nonliner solution. Almost all of the efforts documented in other parts of this report have served to either improve the reliability of models in Bison or provide a more accurate contribution to the Jacobian matrix for individual models, which has improved solution reliability and efficiency.

To specifically address matrix conditioning, it was shown that the condition number can be dramatically improved by scaling the non-linear variables and reducing element aspect ratios. In a nonlinear system of equations that contains multiple variables (such as displacement and temperature), the magnitudes of the terms in the Jacobian matrix corresponding to those variables can differ significantly, which can lead to poor conditioning. To address this issue, MOOSE has long had a capability to scale the contributions of individual variables, but the effects of scaling on Bison models had not been investigated in a significant way. An issue related to the use of scaling with mechanical contact enforcement was addressed. It was found that after improving the matrix condition number through proper selection of scaling parameters on a Bison LWR fuel rod, the non-repeatability of the solution effectively disappeared and the solver was much more efficient. Work is still underway to develop usage guidelines for scaling on a typical Bison model.

1.4 Material Model Convergence

Issue:

The creep and plasticity materials in Bison rely on a nonlinear solution for a single variable (the effective inelastic strain) internal to the material model. It has been a longstanding issue that these internal material iterations occasionally fail to converge. If that happens, the Bison model would exit with an error.

Resolution:

There has been an ongoing effort to improve the robustness of the return mapping algorithm used for these nonlinear iterations. Originally, several models had their own separate implementations of this algorithm based on a pure Newton iteration scheme. In 2017, these models were refactored to rely on a single function for these iterations that offered the following benefits over the previous approach:

- The return mapping iterations are implemented in a base class that any model needing this capability can derive from to reduce code duplication. This class is used by all of the models in both the SolidMechanics and TensorMechanics modules that need this capability.

- This algorithm adds line search and bisection capabilities to the existing Newton iteration scheme to drastically improve the robustness of the return mapping iterations.

- All models that use this functionality now compute residuals in strain increment units for consistency. Previously, some of the return mapping models used stress units for the residual, which made it difficult for a user to determine what the appropriate tolerances should be.

- The elastic strain magnitude is used as the reference quantity for relative convergence (rather than the initial residual, which was previously used). This ensures that the models converge tightly even when the initial residual is large.

- The default convergence tolerances were set to much tighter values than they previously were. They are set to be as tight as can be reasonably achieved on typical problems, and it is now recommended that users do not set those tolerances manually. The vast majority of Bison tests (regression, assessment, and example problems) all use the default tolerances now.

These improvements drastically reduced the number of failures from what was previously seen, but Bison users were still occasionally reporting cases where models exited with errors because the material iterations failed. These cases

were evaluated, and almost all of these failures can be traced back to the zirconium alloy creep model, which computes the creep rate as a function of the hyperbolic sine of the trial stress. Occasionally, the nonliner iteration scheme produces a trial stress that is very large, which causes $\sinh(\sigma_{tr}) \to \infty$ when the trial stress σ_{tr} becomes large.

Several steps were taken as part of the present effort to address this:

- The function that computes $\sinh(x)$ was replaced with one that computes a regularized version of that function, in which it is replaced with a Taylor series expansion of $\sinh(x)$ when x exceeds a certain value. This regularized sinh function is smooth and continuous, and behaves in a non-singular way. Its use does not affect the accuracy of the converged solution, but avoids computing not-a-number (NaN) for the value of the effective inelastic strain.

- The equation for the creep strain rate in the zirconium alloy creep model was re-cast to have a more numerically well-conditioned form, which makes the nonlinear material iterations more robust in cases with high trial stresses.

- The material model was modified to throw a C++ exception rather than generating an error. The exception is handled by cutting back the time step rather than exiting without obtaining a solution. In addition, the handling of exceptions in MOOSE was improved to be more reliable because in some cases in parallel, other errors were encountered in attempting to recover from an exception by cutting back the time step.

- The temperature-dependent Young's modulus used in the zirconium alloy creep model produces unreasonable results at very high temperatures, which can cause convergence problems with this model. The code now throws an exception in those cases, which can occasionally occur during nonlinear iterations. This simply causes the solution to be attempted with a smaller time step.

- During the transition to the new return-mapping algorithm, an option called `legacy_return_mapping` was added to permit usage of the previous iteration scheme. This was necessary because there were still a small set of models in both the Bison and MOOSE test suites that had issues with the new algorithm. All of those issues have now been addressed, so that option is no longer being used in the test suites, has been deprecated, and will be completely removed in the near future.

With these improvements, the likelihood of encountering a failure in the material iterations has been greatly decreased, and in the very rare instances when that still happens, the error is handled by cutting the time step, which reliably allows the analysis to be run to completion.

1.5 Mechanical Contact Convergence

Issue:

Mechanical contact, especially frictional contact, has long presented convergence issues because of its non-smooth function behavior. Newton's method, on which non-linear solution algorithms are typically based, is predicated on application to smooth functions, so convergence issues are not necessarily surprising. Other codes that offer mechanical contact enforcement capabilities encounter similar convergence issues. Efficient solution algorithms and their software implementation are an open area of research in the computational mechanics community.

Resolution:

Multiple concepts, outlined in the following sections, have been pursued to further improve the robustness of mechanical contact enforcement in Bison.

1.5.1 Expanded Testing of Augmented Lagrange Formulation

In 2017, as part of a broader effort to improve Bison's mechanical contact algorithms, an Augmented Lagrange formulation was developed specifically with the goal of improving the robustness of frictional contact enforcement [1]. As part of that prior effort, this formulation was tested on a limited number of Bison fuel rods. In the current effort, the full set of Calvert Cliffs test problems in the Bison assessment test suite was converted from frictionless contact to

use the Augmented Lagrange contact algorithm with a friction coefficient of 0.4. This set of test cases consists of 13 full-length fuel rods.

The Calvert Cliffs test cases had previously been set up to use a previously implemented frictional contact model based on the penalty method, but the nightly testing of these cases was very unreliable, with various rods in that suite intermittently failing. The current Augmented Lagrange contact algorithm is much more robust, however, occasional failures still occur due to the relative lack of experience on full length rod models. There are plans to add additional debugging information from the Augmented Lagrange algorithm, which will be useful in determining the parameter adjustments necessary to correct failing models.

1.5.2   Improved Algorithm for Updating Contact Constraints

Bison employs the Jacobian-free Newton Krylov (JFNK) solution, method which is a variant of Newton's method that allows developers to omit Jacobian entries for highly complex material models while still retaining the Newton method's desirable convergence properties. JFNK requires an evaluation of the residual function for every linear iteration taken during solution of the linearized governing equations. To maintain the integrity of the linear solve, the active contact constraint set must not be updated during the linear iterations.

To meet this algorithmic criterion, past versions of MOOSE updated the contact constraint set during calculation of the preconditioning matrix, which occurs at the beginning of every non-linear iteration. Because this is done at the beginning of a new non-linear iteration, an undesirable and unintended consequence of this decision was that a solution could be accepted as being converged in some cases when the constraint set should have been updated. The typical manifestation of this issue was that if a node slid from one face to another during the last nonlinear iteration of a time step and the residual after that iteration was below the convergence tolerance, mechanical contact was not enforced on that node, and local regions of the two contact surfaces could have been interpenetrated. During a transient simulation, the negative effects of this inaccuracy were bounded; a constraint that was not enforced on the last time step would have been detected in the first non-linear iteration of the next time step. Nevertheless, this is undesirable behavior, and needed to be addressed to improve the accuracy and robustness of Bison.

To address this issue, it is necessary to know within MOOSE whether the current residual evaluation is part of a nonlinear or a linear iteration to determine whether the contact set should be updated. Recent versions of PETSc, the non-linear solver package that MOOSE uses, allow setting different callback functions for non-linear and linear residual evaluation. This PETSc addition has enabled movement of the constraint determination step from the preconditioner evaluation to non-linear residual evaluation. With this change, a converged solution is guaranteed to have the correct active set of contact constraints, e.g. the solution will never exhibit node penetration. This change was made to MOOSE as part of the present work.

Unfortunately, this guarantee of solution accuracy has come at the cost of increasing the required number of nonlinear iterations in many cases. Typically, this is a relatively small increase for a given model. Because the new method of updating the contact set is algorithmically more consistent and more reliable, the decision was made to go ahead with this change.

1.5.3   Contact Line Search

One strategy to combat the solution difficulty of mechanical contact is to improve the global non-linear solution strategy. Globalization of Newton's method is typically achieved using a line search. Line searches compute a scaling factor applied to the current nonlinear incremental update to the solution to optimize its benefit. They typically use the criterion that the non-linear residual must not only decrease during a given iteration, but must decrease by at least a certain amount. There are multiple line-search algorithms that work to achieve this criterion. Perhaps the most common are back-tracking algorithms that start with the full Newton step and then back-track along the Newton update vector if the attempted step does not adequately reduce the residual.

While typical line search globalization strategies work well for a wide class of non-linear problems, they do not work well for contact problems. Contact often requires a temporary *increase* in the residual during the course of the non-linear solution as nodes move into or out of contact. Abandoning the line search completely is not advisable, however,

because blindly taking the full Newton step at each non-linear iteration can lead to a phenomenon known in the contact literature as zig-zagging or jamming, where the contact set simply oscillates back and forth between states from one nonlinear iteration to the next.

A specialized line search algorithm that is aware of the unique characteristics of mechanical contact problems has been developed and merged into the MOOSE code base. This required the development of a interface within MOOSE that allows for a variety of customized line search methods to be implemented in a modular fashion.

The contact line search allows temporary residual increases while also allowing cuts in the line search direction that attempt to minimize residual increases and maximize residual decreases. It also takes advantage of knowledge of the contact set in order to reduce computational cost. When the contact set is changing during the initial non-linear iterations, the linear solution tolerance is loosened to reduce the number of linear iterations and function evaluations. Once the contact set has been finalized, the linear solve tolerance is tightened and Newton's method is allowed to proceed at its quadratic convergence rate.

In combination with variable residual scaling, the contact line search has shown promising convergence results when applied to the Bison assessment cases.

2   Summary of Capability Development

2.1   UO$_2$ Smeared Cracking

Issue:

Fracture has important effects on the mechanical response of LWR fuel, and ideally, it should be included in all Bison LWR fuel performance models. However, fracture models are inherently nonlinear and introduce significant challenges to obtaining a converged solution in any finite element code, including Bison. Because of longstanding issues with convergence, Bison LWR fuel performance models have typically not included the effects of either cracking or creep in the representation of fuel mechanical behavior, and instead simply employed elastic models. If cracking is not included, the predicted stress distribution within the pellet is not realistic, which causes fuel creep models to predict nonphysical behavior if they are used.

Ideally, all Bison simulations should employ a realistic model that accurately represents the effects of cracking as well as a model for fuel creep. It is expected that the smeared cracking model should be routinely used in engineering calculations, and that it would not cause undue robustness issues.

Resolution:

Under funding from the NEAMS program, several improvements to Bison's smeared cracking model were made during the early stages of the present work. These are documented in detail in [2]. Although this work was supported by a different program, it is of direct relevance to the present CASL effort to improve code robustness, and is summarized here. As part of the NEAMS-supported work, several bug fixes and general improvements to the smeared cracking model based on the TensorMechanics system, which all Bison users are now encouraged to use:

- A bug was found and corrected, in which the crack coordinate system was not being correctly rotated to the current material orientation in finite deformation analyses.

- A number of general cleanup tasks and optimizations of the smeared cracking code were performed to make the code better documented and more readable.

- The output variables used to visualize the extent of cracking were revised to make them easier to understand and more consistent with standard damage mechanics conventions.

- An option was added to allow for separate softening models to be used for cracking in specific directions. It was shown that the effect of radial crack formation on the integral hoop stress-strain response is very different than

that in other directions, so it would be more appropriate to use a different softening model in that direction for axisymmetric simulations.

- The smeared cracking model was successfully tested in conjunction with the fuel creep model on two problems in the Bison test suite: a simplified example problem, and the TSQ002 model in the assessment test suite, and the checked versions of those problems in the Bison test suite now include those models.

After this work was done, there were still some reported issues with the smeared cracking model in another Bison model where it had been used. Upon investigation, it was found that there were simply some issues with the material properties that were being used in that model, and once those were resolved, the model ran very well.

There is still much more work to do to migrate the full Bison assessment test suite to use the smeared cracking and creep models to represent LWR fuel mechanical behavior. However, this model is already being used much more widely than it previously had, with no reported issues related directly to the use of the smeared cracking model.

2.2 UO$_2$ Creep

Issue:

A model for combined secondary thermal creep and irradiation creep of UO$_2$ fuel, based upon MATPRO [3], has been available in Bison for several years. However, as described above, because of longstanding convergence issues LWR fuel performance models have typically not included the effects of either cracking or creep in the representation of fuel mechanical behavior, and instead simply employed elastic models. With regards to applying creep models, if cracking is not included, the predicted stress distribution within the pellet is not realistic, which causes fuel creep models to predict unphysical behavior.

Resolution:

Improvements to the smeared cracking model, as described above, provide realistic fuel stresses and thus allow routine use of the fuel creep model. This model combines secondary thermal creep and irradiation creep, with the creep rate modeled as a function of time, temperature, effective stress, density, grain size, fission rate, and oxygen to metal ratio (O/M). The constitutive relation is taken from the MATPRO FCREEP material model ([3]) and given as

$$
\begin{aligned}
\dot{\epsilon} = & \frac{A_1 + A_2 \dot{F}}{(A_3 + D)G^2}\sigma \cdot exp\left(\frac{-Q_1}{RT}\right) + \frac{A_4}{(A_6 + D)}\sigma^{4.5} \cdot exp\left(\frac{-Q_2}{RT}\right) \\
& + A_7 \dot{F}\sigma \cdot exp\left(\frac{-Q_3}{RT}\right)
\end{aligned}
\tag{1}
$$

where $\dot{\epsilon}$ is the creep rate (1/s), σ is the effective (von Mises) stress (Pa), T is the temperature (K), D is the fuel density (percent of theoretical), G is the grain size (μm), $\dot{F}$ is the volumetric fission rate (fissions/m^3-s), Q_i are the activation energies (J/mol), and R is the universal gas constant (8.3143 J/mol-K).

The third term in Equation (1) species the irradiation creep, which dominates fuel creep strains at low fuel temperatures. As the MATPRO model was tested, however, it became clear that this term produced unrealistically high strain rates at the higher temperature end of the irradiation creep region. As the equation indicates, the original MATPRO formulation includes exponential temperature dependence in this term. After a careful review of the MATPRO documentation and supporting literature, it became clear that this temperature dependency was based on a single experimental study [4] involving only a very limited number of experiments. Additionally, this early study specified a relatively narrow applicability range for the correlation (523 < T(K) < 773), which was not enforced in the MATPRO implementation. Several other studies, both early ([5–8]) and more recent ([9–11]) did not observe temperature dependency for low temperature irradiation creep, reporting creep rate variation as a function of only stress and fission rate. The most extensive data for irradiation creep of UO$_2$ has been generated at the Halden Research Reactor, resulting in the following correlation ([10]):

$$
\dot{\epsilon}_{irr} = A_7 \dot{F}\sigma
\tag{2}
$$

where $A_7 = 7.78x10^{-37}$.

The new default model in BISON, referred to as the MATPRO-Halden model, replaces the third term in Eqn (1) with Eqn (2), thus removing temperature dependency. Figure 1 compares the MATPRO-Halden model to experimental data in the low-temperature irradiation creep region. There is obviously wide scatter in the data. Note that the original temperature dependent irradiation creep formulation can still be specified simply by providing the original MATPRO material constants as input parameters.

The MATPRO-Halden model has been committed to the Bison repository, including appropriate regression tests. The Bison documentation has also been updated. There have been no convergence issues reported with the UO_2 creep model.
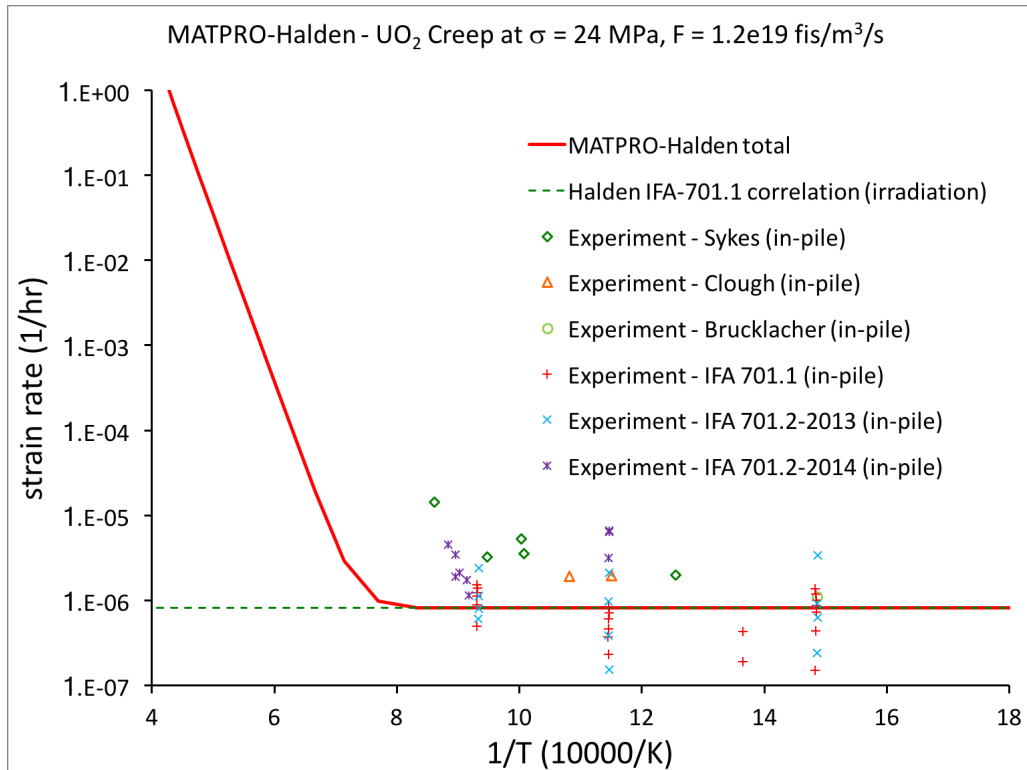

Figure 1: Comparison of the MATPRO-Halden model to experimental data in the low-temperature irradiation creep region.

2.3   Ability to specify multiple blocks within internal (smeared) mesh generator

Issue:

The internal mesh generator in Bison can automatically generate meshes for smeared LWR fuel rod models, which greatly facilitates the modeling process, but it can only generate a single element block to represent the fuel. This makes it harder to include features such as insulator pellets in the model, which currently require the use of an external mesh generator.

Resolution:

A capability for generating multiple element blocks to represent the fuel has been added for layered 1D models. This is the primary type of model currently being employed for CASL full core simulations. This capability is particularly

important for layered 1D models because these models are almost always generated using the internal mesh generator.

This capability still needs to be developed for 2D models, but the need is not as pressing because Bison's external meshing scripts already provide this capability.

3 Summary of Performance Improving Development

3.1 Nonlinear Material Jacobian for Creep Models

Issue:

One of the major benefits of the JFNK solution strategy employed by Bison is that it permits the use of an approximation, rather than an exact form, of the Jacobian matrix for use in preconditioning. As the preconditioning matrix approaches the exact Jacobian matrix, performance can significantly improve. However, solution performance is often acceptable with a rough approximation of the Jacobian, and correctly forming that matrix for a given physical model can often be quite challenging.

Bison makes widespread use of creep models to describe the mechanical behavior of both fuel and cladding materials under the complex stress states that are developed due to non-homogeneous expansion and swelling during irradiation. Until very recently, it simply used the elasticity tensor rather than the nonlinear material tangent matrix for preconditioning.

Resolution:

At the same time the CASL-funded development work documented here was underway, an effort funded by the Department of Energy's Advanced Fuels Campaign (AFC) was underway at Los Alamos National Laboratory to improve Bison's ability to represent creep behavior in metal fuels, which have significantly higher creep rates than do LWR fuels. This work developed the analytic expressions for the material tangent matrix for creep models and an implementation of the computation of that matrix in MOOSE. The present CASL project supported this effort by assisting with develop the interface for this code with MOOSE, peer-reviewing this code during the iterative process of its design, and ultimately merging it into the MOOSE code-base. The details of the derivation and implementation of this material tangent for creep are described below:

3.1.1 Derivation

The full derivation of the tangent modulus for creep, plastic deformation, and volumetric swelling is derived in [12, 13]. What follows is the formulation pertaining to creep strain only. Assuming small strains and a linear constitutive model, the constitutive equation at time step $n + 1$ can be written in terms of the bulk and shear modulus as,

$$\sigma_{n+1} = 3K \, vol(\varepsilon_{n+1}^{el}) + 2\mu \, dev(\varepsilon^{el}). \tag{3}$$

The creep strain is assumed to be purely deviatoric in nature such that Equation 3 can be written as,

$$\sigma_{n+1} = K \, tr(\varepsilon_{n+1})\mathbf{1} + 2\mu(e_{n+1} - \varepsilon_{n+1}^{cr}), \tag{4}$$

where the creep strain can be written as

$$d\varepsilon_{n+1}^{cr} = d\varepsilon_n^{cr} + d\Delta\varepsilon_{n+1}^{cr} \tag{5}$$

Equation 4 can be decomposed into an elastic step and an inelastic correction using Equation 5,

$$d\sigma_{n+1} = \mathbb{C} : d\varepsilon_{n+1} - 2\mu \left(d\Delta\varepsilon_{n+1}^{cr} \right) \tag{6}$$

Rewriting this equation in terms of the total strain,

$$d\boldsymbol{\sigma}_{n+1} = \left[\mathbb{C} - 2\mu \frac{\partial \Delta \boldsymbol{\varepsilon}_{n+1}^{cr}}{\partial \boldsymbol{\varepsilon}_{n+1}} \right] : d\boldsymbol{\varepsilon}_{n+1} \tag{7}$$

$$= [\mathbb{C} - \boldsymbol{A}] : d\boldsymbol{\varepsilon}_{n+1} \tag{8}$$

The creep strain increment can be written in terms of the magnitude of creep strain and the direction of flow as,

$$\Delta \boldsymbol{\varepsilon}_{n+1}^{cr} = \Delta \tilde{\varepsilon}_{n+1}^{cr} \left(\tilde{\sigma}_{n+1}, T_{n+1}, t_{n+1} \right) \sqrt{\frac{3}{2}} \boldsymbol{n}_{n+1} \tag{9}$$

Differentiating this expression,

$$d\Delta \boldsymbol{\varepsilon}_{n+1}^{cr} = d \left(\Delta \tilde{\varepsilon}_{n+1}^{cr} \left(\tilde{\sigma}_{n+1}, T_{n+1}, t_{n+1} \right) \sqrt{\frac{3}{2}} \boldsymbol{n}_{n+1} \right) \tag{10}$$

$$= \Delta \tilde{\varepsilon}_{n+1}^{cr} \sqrt{\frac{3}{2}} d\boldsymbol{n}_{n+1} + \sqrt{\frac{3}{2}} \boldsymbol{n}_{n+1} d\Delta \tilde{\varepsilon}_{n+1}^{cr} \tag{11}$$

Substituting Equation 11 in Equation 6,

$$d\boldsymbol{\sigma}_{n+1} = \mathbb{C} : d\boldsymbol{\varepsilon}_{n+1} - 2\mu \left(\Delta \tilde{\varepsilon}_{n+1}^{cr} \sqrt{\frac{3}{2}} d\boldsymbol{n}_{n+1} + \sqrt{\frac{3}{2}} \boldsymbol{n}_{n+1} d\Delta \tilde{\varepsilon}_{n+1}^{cr} \right) \tag{12}$$

Writing in terms of the total strain,

$$d\boldsymbol{\sigma}_{n+1} = \left[\mathbb{C} - 2\mu \sqrt{\frac{3}{2}} \Delta \tilde{\varepsilon}_{n+1}^{cr} \frac{\partial \boldsymbol{n}_{n+1}}{\partial \boldsymbol{\varepsilon}_{n+1}} - 2\mu \sqrt{\frac{3}{2}} \boldsymbol{n}_{n+1} \otimes \frac{\partial \Delta \tilde{\varepsilon}_{n+1}^{cr}}{\partial \boldsymbol{\varepsilon}_{n+1}} \right] : d\boldsymbol{\varepsilon}_{n+1} \tag{13}$$

Assuming that the direction of creep and the creep strain increment depend on the current stress, the derivatives in Equation 13 can be written as,

$$\frac{\partial \boldsymbol{n}_{n+1}}{\partial \boldsymbol{\varepsilon}_{n+1}} = \frac{1}{\|\boldsymbol{s}_{n+1}\|} \left[\mathbb{I}^{sym} - \boldsymbol{n}_{n+1} \otimes \boldsymbol{n}_{n+1} \right] \mathbb{I}^{dev} \frac{\partial \boldsymbol{\sigma}_{n+1}}{\partial \boldsymbol{\varepsilon}_{n+1}} \tag{14}$$

and,

$$\frac{\partial \Delta \tilde{\varepsilon}_{n+1}^{cr}}{\partial \boldsymbol{\varepsilon}_{n+1}} = \frac{\partial \Delta \tilde{\varepsilon}_{n+1}^{cr}}{\partial \|\boldsymbol{s}_{n+1}\|} \boldsymbol{n}_{n+1} \frac{\partial \boldsymbol{\sigma}_{n+1}}{\partial \boldsymbol{\varepsilon}_{n+1}} \tag{15}$$

Substituting for $\frac{\partial \boldsymbol{n}_{n+1}}{\partial \boldsymbol{\varepsilon}_{n+1}}$ and $\frac{\partial \Delta \tilde{\varepsilon}_{n+1}^{cr}}{\partial \boldsymbol{\varepsilon}_{n+1}}$ in Equation 13 and rearranging, we get the equation

$$\left[\mathbb{I}^{sym} + 2\mu \sqrt{\frac{3}{2}} \Delta \tilde{\varepsilon}_{n+1}^{cr} \frac{1}{\|\boldsymbol{s}_{n+1}\|} \left(\mathbb{I}^{sym} - \boldsymbol{n}_{n+1} \otimes \boldsymbol{n}_{n+1} \right) \mathbb{I}^{dev} + 2\mu \sqrt{\frac{3}{2}} \frac{\partial \Delta \tilde{\varepsilon}_{n+1}^{cr}}{\partial \|\boldsymbol{s}_{n+1}\|} \boldsymbol{n}_{n+1} \otimes \boldsymbol{n}_{n+1} \right] : d\boldsymbol{\sigma}_{n+1}$$
$$= \mathbb{C} : d\boldsymbol{\varepsilon}_{n+1} \tag{16}$$

where $\mathbb{I}^{sym}$ is the fourth order symmetric identity tensor. This can be simplified as

$$\left[\mathbb{I}^{sym} + 2\mu \sqrt{\frac{3}{2}} \Delta \tilde{\varepsilon}_{n+1}^{cr} \frac{1}{\|\boldsymbol{s}_{n+1}\|} \left(\mathbb{I}^{dev} - \boldsymbol{n}_{n+1} \otimes \boldsymbol{n}_{n+1} \right) + 2\mu \sqrt{\frac{3}{2}} \frac{\partial \Delta \tilde{\varepsilon}_{n+1}^{cr}}{\partial \|\boldsymbol{s}_{n+1}\|} \boldsymbol{n}_{n+1} \otimes \boldsymbol{n}_{n+1} \right] : d\boldsymbol{\sigma}_{n+1}$$
$$= \mathbb{C} : d\boldsymbol{\varepsilon}_{n+1} \tag{17}$$

If the term in the square brackets on the left hand side is written at $\tilde{\boldsymbol{A}}$, then Equation 17 can be written as

$$\tilde{\boldsymbol{A}} : d\boldsymbol{\sigma}_{n+1} = \mathbb{C} : d\boldsymbol{\varepsilon}_{n+1} \tag{18}$$

and the differential change in stress can be written as,

$$d\boldsymbol{\sigma}_{n+1} = \tilde{\boldsymbol{A}}^{-1}\mathbb{C} : d\boldsymbol{\varepsilon}_{n+1} \tag{19}$$

where the tangent modulus is $\tilde{\boldsymbol{A}}^{-1}\mathbb{C}$ in which $\tilde{\boldsymbol{A}}$ is given by,

$$\tilde{\boldsymbol{A}} = \left[\mathbb{I}^{sym} + 2\mu\sqrt{\frac{3}{2}}\Delta\tilde{\varepsilon}_{n+1}^{cr}\frac{1}{\|\boldsymbol{s}_{n+1}\|}\left(\mathbb{I}^{dev} - \boldsymbol{n}_{n+1} \otimes \boldsymbol{n}_{n+1}\right) + 2\mu\sqrt{\frac{3}{2}}\frac{\partial\Delta\tilde{\varepsilon}_{n+1}^{cr}}{\partial\|\boldsymbol{s}_{n+1}\|}\boldsymbol{n}_{n+1} \otimes \boldsymbol{n}_{n+1} \right] \tag{20}$$

3.1.2 Implementation

The application of the tangent operator using Equation 19 to the relevant Jacobian formulation is already handled by MOOSE. What remained was the implementation of the combination and formulation of Equation 20. Given that the inverse of $\tilde{\boldsymbol{A}}$ is required, combination of tangent operators should occur before the inversion,

$$\tilde{\boldsymbol{A}} = \left[\mathbb{I}^{sym} + \alpha_1 + \alpha_2 + ... \right] \tag{21}$$

As a consequence of Equation 21, each creep model supplies only the "partial" contribution α to the full tangent operator. For all radial return methods, the formulation of Equation 20 is contained within `RadialReturnStressUpdate`. The only model-specific remaining component of this equation is $\frac{\partial\Delta\tilde{\varepsilon}_{n+1}^{cr}}{\partial\|\boldsymbol{s}_{n+1}\|}$, which is calculated for all radial return creep models within `RadialReturnCreepStressUpdateBase`.

It should be noted that this formulation is fundamentally different if plastic deformation is involved, primarily due to the application of the tangent operator to the strain instead of the stress. This requires in an analogous but different starting point than Equation 8, resulting in a tangent operator combination. This has not yet been implemented.

Also, the nonlinear material tangent has been implemented only for the TensorMechanics system, so as models are migrated to this system from the legacy SolidMechanics system, they will be able to take advantage of these improvements.

4 Impact of Development and Future Work

4.1 Summary of New Development

Development has been performed in a number of areas to address identified issues with the robustness, and efficiency of Bison for solving LWR fuel performance models, as well as limitations in Bison's capabilities that hampered its ability to realistically model LWR fuel behavior. This work included:

- Migrating Bison to use a new library for water/steam properties used by the coolant channel model.

- Adding mandatory checks to ensure that finite element aspect ratios are within acceptable limits to avoid convergence issues that have been found to result from using poor quality meshes.

- Performing a thorough investigation of the causes of identified non-repeatability of nonlinear solution and taking steps to improve the conditioning of the equation system to manage this behavior.

- Addressing lingering issues with convergence of the internal material iterations used by creep and plasticity models and gracefully handling failure of those iterations in very rare cases by cutting the time step.

- Improving mechanical contact convergence by expanding testing of the Augmented Lagrange algorithm for frictional contact enforcement, addressing a long-standing issue that could admit solutions where contact is not enforced at some nodes, and implementing a contact-specific line search algorithm.

- Expanding the usage of the smeared cracking model to work toward the goal of using a fracture model in all standard Bison LWR simulations.

- Improving the UO_2 creep model to more realistically represent recently-obtained irradiation creep data.

- Adding an option to the internal mesh generator for layered 1D models to use multiple element blocks to represent LWR fuel, enabling it to represent insulator pellets.

- Supporting integration of code to compute the nonlinear material tangent for all TensorMechanics-based creep models used by Bison, with significant performance benefits.

4.2 Impacts

These developments have been demonstrated to significantly benefit individual test cases that were used in the testing process. They have also resulted in improved solution robustness for multiple users:

Westinghouse Models Prior to this work, an effort by Westinghouse to model all of the fuel rods in a reactor using Bison resulted in model failures of around 30% of the rods due to convergence issues. As a result of some of the early developments in this effort, the number of model failures was reduced to a fraction of a percent of those cases.

VERA Core Simulator Models The VERA core simulator depends on all of the Bison models representing individual fuel rods running to completion successfully for the core model to run to completion. Prior to this work, that effort encountered serious difficulties due to convergence failures in Bison, both in the global nonlinear iterations as well as in local material iterations.

The combined effect of the development efforts documented here has significantly improved the robustness of these full quarter-core VERA simulations, which has allowed them to run much farther in time than was previously possible.

Bison Assessment Tests In conjunction with the efforts to improve code robustness, an effort has been underway to improve the robustness of the suite of assessment problems, which is run nightly and primarily tests LWR fuel problems. These improvements have included adding reference solutions that the model results are compared against for many of the test cases. Before this work, the test cases were run nightly, and plots were generated of quantities of

interest, but no checking against reference solutions was done as part of the running of the tests to give an immediate notification that a code change resulted in a difference in test behavior.

Reference solutions are used for all of the Bison regression tests, but because of their size and complexity, the assessment problems, they presented unique challenges in this regard. Rather than checking the solution at every degree of freedom in a model, it is better to check a small set of quantities of interest against a reference solution. Some infrastructure changes needed to be made to the testing system to permit that type of testing. Those changes were made, and reference solutions were added for many of the assessment problems. Currently 120 of the cases are now compared against reference solutions for a set of quantities of interest at the last time step of the analysis.

In addition, a significant effort has been taken to standardize the solver options used in these models. Because many Bison users use these problems as a starting point for setting up new models, it is important for these models to use the recommended set of options.

4.3 Acknowledgments

4.4 Future Work

Although major improvements to Bison robustness have been made, it is important to emphasize that significant work still remains in making it into a robust tool that will always obtain converged solutions in a production environment for modeling LWR fuel.

This work has shown that improving the Jacobian matrix can drastically improve solution convergence rates. This has been done for some of the models used in Bison, but similar efforts should still be taken for other models as well. Models relevant to LWR fuel known to use approximations of the Jacobian that can be improved upon include combined creep-plasticity models, coupling terms for thermal and mechanical contact, and hot-pressing models.

The work documented here significantly improved the reliability of the Bison models run as part of the VERA core simulation. However, the full quarter-core model still has persisting problems traceable to Bison. A small subset of the Bison models are impeding the solution because they struggle to converge and are cutting the time step back to very small values. This issue is being actively investigated. This is a very demanding use case for Bison because it involves a large number of fuel rod models that must all converge reliably, so any further improvements made to Bison to address this issue will benefit all Bison users.

The Bison test suite, including example and assessment problems, is a very important component of Bison's code quality strategy, and still needs significant efforts to migrate it to models based on the TensorMechanics module, standardize the options used in the models, and compare all cases against reference solutions.

Although there is still more work to be done, it is clear from the work documented here that the remaining issues can be addressed if a concerted effort is taken to resolve robustness issues.

References

[1] B. W. Spencer, J. W. Peterson, W. Jiang, Y. Liu, S. Veeraraghavan, and A. Casagranda. BISON contact algorithm improvements in support of pellet cladding mechanical interaction modeling. Technical Report INL/EXT-17-43399, Idaho National Laboratory, September 29 2017.

[2] Benjamin W. Spencer and Russell J. Gardner. Improved LWR fuel rod mechanics models. Technical Report INL/EXT-18-44749, Idaho National Laboratory, February 2018.

[3] C. M. Allison, G. A. Berna, R. Chambers, E. W. Coryell, K. L. Davis, D. L. Hagrman, D. T. Hagrman, N. L. Hampton, J. K. Hohorst, R. E. Mason, M. L. McComas, K. A. McNeil, R. L. Miller, C. S. Olsen, G. A. Reymann, and L. J. Siefken. SCDAP/RELAP5/MOD3.1 code manual, volume IV: MATPRO–A library of materials properties for light-water-reactor accident analysis. Technical Report NUREG/CR-6150, EGG-2720, Idaho National Engineering Laboratory, 1993.

[4] D. Brucklacher, W. Dienst, and F. Thummler. Creep behavior of oxide fuels under neutron irradiation, Paper 251. In *Proceedings REAKTORTAGUNG 1973 des Deutschen Atomforums/KTG held in Karlsruhe*, pages 413–416, Apr 10–13, 1973.

[5] D. Brucklacher and W. Dienst. Creep behavior of ceramic nuclear fuels under neutron irradiation. *Journal of Nuclear Materials*, 42:285–296, 1972.

[6] J. S. Perrin. Effect of irradiation on creep of UO2-PuO2. *Journal of Nuclear Materials*, 42:101–104, 1972.

[7] A. A. Solomon. Radiation induced creep of UO2. *Journal of the American Ceramic Society*, 56:164–171, 1973.

[8] W. Dienst. Irradiation induced creep of ceramic materials. *Journal of Nuclear Materials*, 65:1–8, 1977.

[9] K. Sakai, H. Hanakawa, and T. Tverberg. Investigation of fission induced creep of UO2 and Cr-doped fuel in IFA-701. Technical Report HWR-1006, OECD Halden Reactor Project, 2011.

[10] K. Sakai. The fuel creep test IFA-701: results after four irradiation cycles. Technical Report HWR-1039, OECD Halden Reactor Project, 2013.

[11] R. Szoke and T. Tverberg. Update on in-pile results from the fuel creep test IFA-701. Technical Report HWR-1092, OECD Halden Reactor Project, 2014.

[12] Naveen Prakash, Christopher Matthews, and Daniele Versino. Stress and tangent update equations for creep and plasticity in an implicit hypoelastic formulation . Technical Report LA-UR-18-23619, LANL, April 2018.

[13] Naveen Prakash, Christopher Matthews, and Daniele Versino. Stress and tangent update equations for time hardening creep effects in an implicit hypoelastic formulation . Technical Report LA-UR-18-23541, LANL, May 2018.