

Verification and Validation Strategy for Implementation of Hybrid Potts-Phase Field Hydride Modeling Capability in MBM

Jason Hales
Veena Tikare

April 2014



The INL is a U.S. Department of Energy National Laboratory
operated by Battelle Energy Alliance

Verification and Validation Strategy for Implementation of Hybrid Potts-Phase Field Hydride Modeling Capability in MBM

**Jason Hales
Veena Tikare¹**

¹Sandia National Laboratories

April 2014

**Idaho National Laboratory
Idaho Falls, Idaho 83415**

<http://www.inl.gov>

**Prepared for the
U.S. Department of Energy
Office of Nuclear Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517**

Verification and Validation Strategy for Implementation of Hybrid Potts-Phase Field Hydride Modeling Capability in MBM

Jason Hales, Idaho National Laboratory
Veena Tikare, Sandia National Laboratories

Introduction

The Used Fuel Disposition (UFD) program has initiated a project to develop a hydride formation modeling tool using a hybrid Potts-phase field approach. The Potts model is incorporated in the SPPARKS code from Sandia National Laboratories. The phase field model is provided through MARMOT from Idaho National Laboratory.

Technical overview

Hydride formation/reorientation

In LWRs, zircaloy cladding experiences oxidation on its outer surface through contact with the coolant. One consequence of this oxidation is the migration of hydrogen into the cladding. The solubility of hydrogen in zircaloy is quite low at operating temperatures, resulting in the formation of hydrides in the cladding wall. These hydrides typically form in a circumferential direction.

Following its useful life in the reactor, the fuel is moved to cooling pools where decay heat can be carried away. In preparation for long-term storage, the fuel is vacuum dried. In that process, the cladding temperature increased dramatically. The hydrogen picked up while in the reactor goes into solution. As the cladding cools, the hydrogen again forms hydrides. However, the stress state in the cladding is much different while in the reactor compared to outside. This different stress state (positive hoop stress due to internal rod pressure) may cause the hydrides to form in a radial direction.

Hydride effect on cladding

Hydrides are far more brittle than zircaloy. Thus, their presence in cladding is cause for concern. However, hydrides in a circumferential direction have a reasonably low effect on cladding mechanical properties. The same is not true for hydrides oriented radially. In that case, the ductility of the cladding is significantly reduced and the potential for cracking is much higher.

For these reasons, it is important to control the temperature and stress history of fuel rods in order to ensure that failure-inducing radial hydrides are not created.

Value of meso-scale modeling

Computer modeling of fuel rods, including the processes of hydride formation, is attractive since physical experiments of irradiated material is expensive and time consuming. Constitutive models of cladding strength in the presence of hydrides are available. However, these require hydride concentrations or fraction as well as hydride orientation. Mesoscale modeling (modeling physical properties at the grain level) can provide a way to determine the hydride concentration and orientation.

With a mesoscale model that predicts hydrides and an engineering-scale model that calculates cladding strength, it is possible to build a computational tool that will allow designers to find a set of operating conditions that reduce the likelihood of cladding cracking while in storage.

Hybrid Potts-phase field model

Potts model

The Potts model represents microstructure as an ensemble of particles populating a lattice. Each lattice site is identified by an integer value, a spin, that determines its membership in a particular grain, phase, and/or microstructural feature such as a precipitate. The energy of each particle is given by its bulk and interfacial energies. The microstructure evolves by using a statistical method that attempts to change the spin at each lattice site to reduce the system's total free energy.

Strengths of the Potts model are that it is computationally efficient, requiring no linear solve, and is easily parallelizable. Its weakness is that the discrete particles cannot represent smoothly varying continuous characteristics such as concentration.

Phase field model

The phase field model uses continuous fields to represent microstructure using a set of order parameters that designate membership in particular microstructural features such as grains, phase, precipitates, etc. These order parameters vary continuously so that at any given point partial membership in different features is possible. The energy of the system is the integral bulk and interfacial energies over the simulation space. Microstructural evolution is simulated using the Cahn-Hilliard equation, so that order parameters evolve to minimize the system's total free energy. The evolution equations are often implemented through the finite element method.

A strength of the phase field model is that it can treat complex microstructural and composition evolution phenomena; however, it may become prohibitively computationally expensive with the additional complexity.

Hybrid Approach

Veena Tikare at Sandia National Laboratories has developed a hybrid Potts-phase field model that is appropriate for analyzing hydride evolution. The hybrid combines elements of the Potts model with the phase field model so that the computational efficiency of Potts is retained while enabling treatment of the continuous compositional field by phase field. Details of the approach are in E. R. Homer, V. Tikare, and E. A. Holm, "Hybrid Potts-phase field model for coupled microstructural-compositional evolution," *Computational Materials Science*, 69, 2013, p. 414-423.

The thrust of this project is to create a hybrid model using a Potts model in the SPPARKS code from Sandia National Laboratories and a phase field model in MARMOT from Idaho National Laboratories.

SPPARKS and the Potts model

SPPARKS, an acronym for stochastic parallel particle kinetic simulator, is a Sandia-developed open-source code that can simulate many microstructural evolution processes. SPPARKS is a kinetic Monte Carlo (KMC) code that has algorithms for both rejection-free KMC and rejection KMC, which is sometimes called Metropolis Monte Carlo (Plimpton, S., Thompson, A. and Slepoy, A. (2009) "SPPARKS kinetic Monte Carlo simulator", available at <http://www.cs.sandia.gov/~sjplimp/spparks.html>).

SPPARKS is distributed as an open source code under a GNU-like license. It is highly versatile, supports a number of different applications, can be extended to add new functionalities and is able to run in serial or in parallel. The parallel version uses the message passing interface (MPI) to perform concurrent computations over all processors while minimizing communication overhead between processors. To accomplish this, the space of 3D particles (domain) is partitioned between the processors, and Monte Carlo dynamics is calculated concurrently in each processor. See Figure 1(a). Note that 1-D and 2-D simulation domains are similarly partitioned by SPPARKS. Information about border sites whose neighborhood could belong to other processors is kept locally in ghost sites. See Figure 1(b). While each processor can change the state or spin of its owned sites, it cannot change the state or spin of the ghost sites; it can only access the information contained at the ghost sites. Communication between processors is used to update the state of the ghost sites. Communication is also required to collect some of the statistical information that can be computed while simulating the evolution of the system at hand.

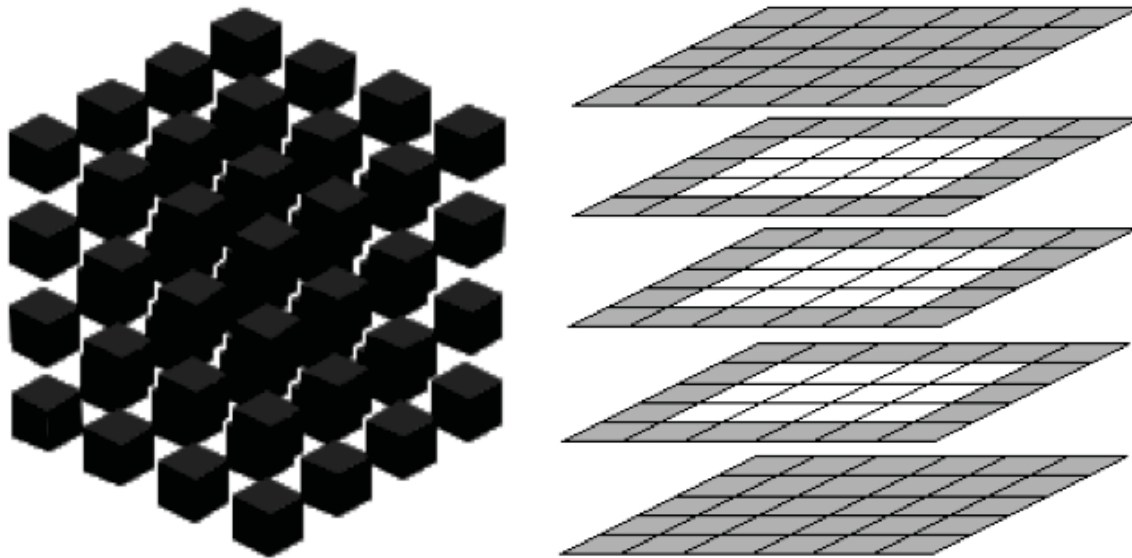


Figure 1. (a) 3D domain partitioned among processors. (b) Caricature of local site structure. Central (owned) sites in white are surrounded by a layer of ghost sites in grey owned by neighboring processors.

MOOSE-BISON-MARMOT (MBM)

Idaho National Laboratory is developing a modern computational simulation framework called MOOSE. MOOSE is based on the finite element method, runs on a single CPU or on a massively parallel computer, and leverages software packages that provide, among other things, advanced numerical solvers. MOOSE itself is not built to solve any particular physical problem but provides interfaces that allow applications to be built with relative ease. These applications are tailored for a particular field of interest. Two examples are MARMOT, a meso-scale modeling application, and BISON, a nuclear fuel performance analysis tool.

Since 2008, Idaho National Laboratory (INL) has been developing a next-generation nuclear fuel modeling capability known as BISON. BISON is a massively parallel, finite element-based software product that solves the coupled nonlinear partial differential equations associated with nuclear fuel performance. BISON supports the use of one-, two-, and three-dimensional meshes and uses implicit time integration, important for the widely varied time scales in nuclear fuel simulation. An object-oriented architecture is employed which minimizes the programming required to add new material and behavior models.

BISON is written in a general way such that it may be applied to a variety of fuels and geometries. Though primarily used for LWR fuel analysis, BISON has been used to analyze TRISO-coated particle fuel and metal fuel in rod and plate form, design and interpret irradiation experiments, and investigate novel fuel concepts.

MARMOT is a multiphysics mesoscale code focused on modeling irradiation-induced microstructure evolution in reactor materials. It is based on the MOOSE framework and uses the basic phase field classes that are built into the phase field module in MOOSE (MOOSE-PF). The phase field method is used to model the microstructure evolution and it is coupled to finite deformation mechanics and heat conduction to capture the multiphysics nature of the microstructure evolution. The impact of the evolution on the material properties is quantified using advanced homogenization approaches to take the local material properties at the mesoscale and determine a single effective property over the entire mesoscale domain. The phase field method is used in MARMOT due to its flexibility in modeling many coupled categories of microstructure evolution. In the phase field model, microstructural features are represented by continuous variables, which are evolved to minimize a function describing the free energy of the system. In MARMOT, the phase field model has been directly applied to model radiation damage in UO₂ and zircaloy cladding. The phase field model is coupled to linear elasticity to model the impact of stress on the fuel behavior, and to finite strain mechanics to model the impact of deformation in zirconium alloys. The phase field models are also regularly coupled to heat conduction, where the local thermal conductivity varies locally across the microstructure.

Together, MOOSE, BISON, and Marmot are sometimes referred to as MBM.

Implementation Approach

The approach taken to accomplish the coupling described above is to use MARMOT as the driver for the coupled code. MARMOT will handle time stepping, invoking data transfers, and calling the Potts model as needed.

The first step toward this end is including SPPARKS as a library within the MARMOT tool. This required working out licensing questions library build and link requirements. This stage is complete.

The next critical task was to ensure that data computed in SPPARKS could be passed to MARMOT and vice versa. This was complicated by the fact that when running with multiple processors, the decomposition of the computational domain in SPPARKS would not match that in MARMOT. Code has been written to accommodate this complication, and it is possible to pass data in both directions.

The remaining primary development task is to settle on the coupling approach for driving SPPARKS via MARMOT. The approach is to begin each timestep in MARMOT with a call to SPPARKS. Before calling SPPARKS, MARMOT data is passed to SPPARKS. Following the call to SPPARKS, SPPARKS data is passed to MARMOT. This is repeated until the final simulation time is reached.

All of this may be run on multiple processors if desired.

Verification and Code Comparison

Verification is the process of demonstrating that the implementation of a mathematical model in computer code is correct. This is done through tests of the software. Tests are developed with well-defined inputs such that the expected output can be determined independently of the software. If the software computes the expected output for a given test, that correct result is evidence that the model has been implemented correctly. Tests with analytic solutions are good verification tests.

Code comparisons may add value if an analytic solution to a given problem is not known.

This section outlines a set of verification tests and code comparison tests for the coupled SPPARKS/MBM tool. These tests are found in `elk/tests/spparks/` and `marmot/tests/TwoComp/` within the MBM repository.

Do-nothing test

The foundational verification exercise is to execute a SPPARKS model from within MBM and check that the results match those computed by a stand-alone SPPARKS run.

Check transfer to MARMOT

Another worthwhile test is one in which the computed values from SPPARKS are transferred to and written out by MARMOT. If done correctly, the output from MARMOT should match that the native output from SPPARKS.

Check transfer to SPPARKS

A similar test is one in which data from MARMOT is sent to SPPARKS. The data in SPPARKS should match that in MARMOT.

The following verification tests are based on examples from E. R. Homer, V. Tikare, and E. A. Holm, "Hybrid Potts-phase field model for coupled microstructural-compositional evolution," *Computational Materials Science*, 69, 2013, p. 414-423.

Equilibrium of binary, two-phase interface

Characterize the equilibrium configuration of a binary, two-phase interface and compare to the SPPARKS solution. Allow an α -grain shaped like a rectangle or cube (in 3D) to come to equilibrium with the same-shaped β -grain so that they share a flat interface. The starting composition for both grains is the equilibrium composition of that phase. When the composition across the interface no longer changes, characterize the interface structure and composition change. See Section 3.1 in the paper.

Diffusion kinetics of binary, two-phase diffusion couple

Characterize the diffusion kinetics of a binary, two-phase diffusion couple and compare to SPPARKS solution and to analytical solution. Form a diffusion couple of α -grain shaped like a rectangle or cube (in 3D) with the same-shaped β -grain so that they share a flat interface. The composition of both is to be off-set from equilibrium so that the total volume of the two phases does not change. (This is done by choosing the compositions of the two phase, so that the lengths from the overall composition to the free energy of the two phases does not change as the composition of the two phases change.) Allow the two components to diffuse across the interface and compare the diffusion kinetic results to the analytic solution for this configuration. See Section 3.2 in the paper.

Diffusion-controlled grain growth

Simulate diffusion-controlled grain growth in a binary, two-phase system and compare to SPPARKS results. Start with 50% α -phase and 50% β -phase by area (or volume in 3D) under two different starting compositions; start with composition of both phases at 1. the equilibrium compositions for both phases and non-equilibrium compositions for both so that the equilibrium phase mixture is not 50% α and 50% β . Characterize grain growth kinetics and topological results to SPPARKS. See Section 3.3 in the paper.

Hydride precipitation

Simulate hydride precipitation in zircaloy under the same conditions done in SPPARKS. Perform simulations of hydride precipitate nucleation and growth in a single crystal and a polycrystalline aggregate from a supersaturated solution. Compare microstructure and compositional evolution.

Validation

Validation is the process of demonstrating that an analysis tool is capable of modeling real-world phenomena. Performing validation of a computer program is showing that its output matches data from physical experiments.

Hydride precipitate growth under lab conditions

1. Simulate conditions used by Blythe Clark et al. to perform hydrogen diffusion from an aqueous solution at the sample surface at 95 °C. Hydrogen diffuses in at the surface and forms hydride precipitates that grow as lathes from the surface inward.
2. Diffuse H from hydrogen-gas environment at higher temperatures. Use existing data from ORNL hydriding experiments and other sources to simulate hydrogen diffusing into a zircaloy sample, distributing itself fairly uniformly and nucleating precipitates at

energetically favorable sites. (The validation exercise will need to identify specific experiments or phenomenon for setting up simulations and comparing results.)

Hydride dissolution and re-precipitation

Simulate hydride dissolution and re-precipitation under conditions of no applied stress and uni-axial stress ranging up to 85 MPa. KB Colas et al. characterized the extent of dissolution as a function of temperature while heating the sample and then the extent of precipitation during cool down. Furthermore, they characterized the extent of re-orientation under stress as the fraction of hydrides oriented at different angles. Simulation of hydride precipitation under these conditions will be performed, and the result will be compared to experimental results. See <http://dx.doi.org/10.1016/j.actamat.2010.07.018>.

Future Plans

In order to increase the accuracy of the Potts-phase field simulations, it will be important to include other effects. Perhaps the most important of these is the effect of stress on the microstructural evolution. Including stress will be a high priority in future work.

It is also expected that further experimental data for validation will be required. Details will become clearer as the verification and validation work progresses.