

Mixed Potential Model for Used Fuel Dissolution - Fortran Code

Used Fuel Disposition

***Prepared for
U.S. Department of Energy
Used Fuel Disposition Campaign***

James Jerden¹

Jacqueline M. Copple¹

Kurt E. Frey²

William Ebert¹

¹Argonne National Laboratory

²University of Notre Dame

January 12, 2015

FCRD-UFD-2015-000159



This work was supported by the US Department of Energy, Office of Nuclear Energy. The report was prepared at Argonne National Laboratory as part of the Used Fuel Disposition (UFD) Campaign.

Government License Notice: The submitted manuscript has been created by UChicago Argonne, LLC, Operator of Argonne National Laboratory (“Argonne”). Argonne, a U.S. Department of Energy Office of Science laboratory, is operated under Contract No. DE-AC02-06CH11357. The U.S. Government retains for itself, and others acting on its behalf, a paid-up nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government.

DISCLAIMER

This information was prepared as an account of work sponsored by an agency of the U.S. Government. Neither the U.S. Government nor any agency thereof, nor any of their employees, makes any warranty, expressed 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. References herein to any specific commercial product, process, or service by trade name, trade mark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the U.S. Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the U.S. Government or any agency thereof.

This work is being performed as part of the DOE NE Used Fuel Disposition (UFD) Campaign's Argillite and Crystalline Disposal R&D work packages: FT-15AN080601 and FT-15AN080701 respectively. This document and associated materials meets milestones: M3FT-15AN0806014 and M3FT-15AN0807014: Mixed Potential Model for Used Fuel Dissolution - Fortran Code.

Summary

This report documents the completion of the Fortran coding of a preliminary version of the Mixed Potential Model (MPM V.2) for used fuel degradation that is being used to develop the interfaces between the MPM and the Generic Disposal System Analysis (GDSA). Issues identified during the initial discussions with researchers developing the GDSA and the approaches being taken to resolve them are summarized.

Introduction

The MPM for used fuel degradation is an electrochemical reaction/diffusion model that is being developed and implemented with four main objectives:

- Calculate the dissolution rate of used fuel based on the corrosion potential and redox reaction currents of electrochemical reactions (most importantly the anodic reaction releasing UO_2^{2+} into solution) that are established at the fuel/solution interface. The fuel dissolution rate includes UO_2 released by chemical dissolution and UO_2^{2+} released by oxidative dissolution.
- Produce surface area-scalable results that can be used to define a fractional fuel degradation rate to calculate radionuclide source terms for use in performance assessment models.
- Quantify the effects of oxidants (primarily O_2 and radiolytic H_2O_2), interfacial redox reaction kinetics on the fuel dissolution rate, including the catalytic efficiencies of the noble metal particles (NMP) present within the fuel.
- Quantify the mitigating effects of H_2 on fuel oxidation reactions on the fuel dissolution rate.
- Quantify the effects of the evolution of fuel burnup and the related temperature and dose histories of the fuel on the fuel dissolution rate.

The MPM, which is described in Jerden et al. 2014, extends the model developed in Canada by including additional processes affecting fuel dissolution. Most importantly, the MPM includes hydrogen oxidation reactions that affect other redox reactions and can decrease the fuel dissolution rate by several orders of magnitude. Specifically, MPM V.2 includes separate UO_2 and NMP phases to distinguish their catalytic effects on several redox reactions, a more sophisticated radiolysis model, and a modified diffusion model for more efficient calculations over long disposal times. Other modifications planned for future versions include adding a steel corrosion module to quantify hydrogen generation, varying the catalytic efficiency of NMP to represent the effects of catalytic poisons present in the groundwater (e.g., bromide) and corrosion, and evolving the reactive surface area of the fuel as it corrodes. Figure 1 provides the context of MPM V.2 with regard to the continuing development of the MPM and planned changes in MPM V.3.

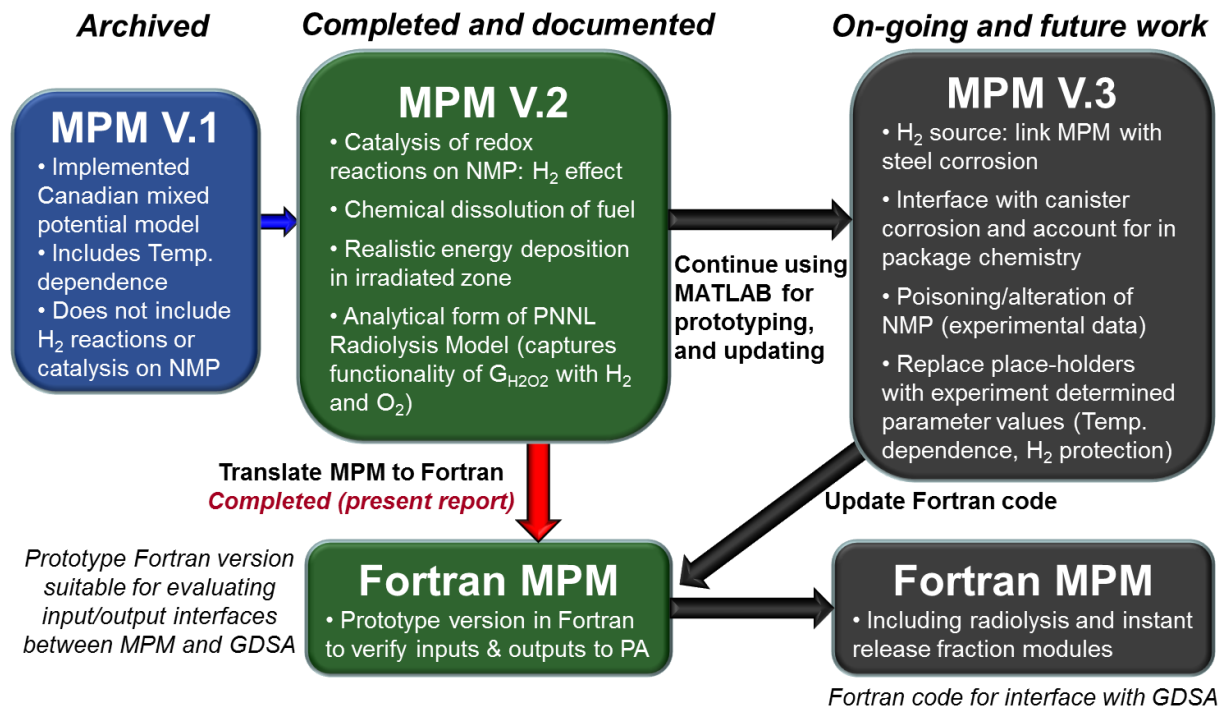


Figure 1. Schematic flow diagram highlighting progress made in the development of the MPM and future priorities based on sensitivity results from MPM V.2.

The MPM was converted from MATLAB, the language used for model development and testing, to Fortran 90 to identify data interfaces, formats, and time scales for integration with the PLFLOTTRAN code, which is the central component of the GDSA. The MPM calculates the dissolution rate (due to both by oxidation and chemical processes) of used fuel exposed to groundwater within a breached waste package on a mass uranium per area per time basis. The MPM dissolution rates will be combined with information on radionuclide inventories and assumptions about the effective surface area of the dissolving fuel to provide the radionuclide source term for Performance Assessment (PA) calculations (i.e. GDSA argillite and crystalline rock repositories) as mass fraction radionuclide per time. The conceptual layout of the GDSA, showing the context for the MPM and other source term process models is provided in Figure 2 (adapted from Figure 2-6 of Sevougian et al., 2014).

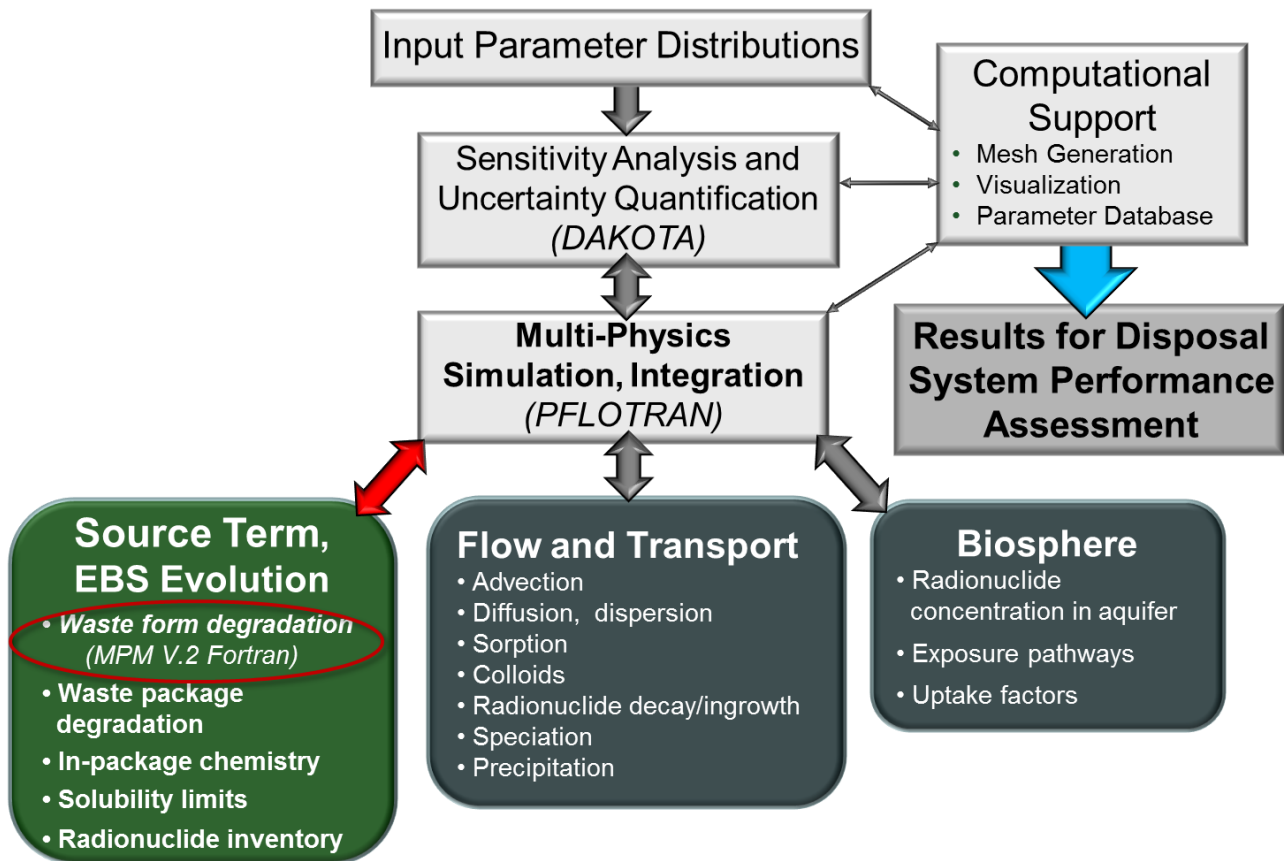


Figure 2. Schematic diagram showing the conceptual layout of the Generic Disposal System Analysis (GDSA) adapted from Figure 2-6 of Sevougian et al., 2014.

Figure 3 is another conceptual diagram summarizing the flow of specific types of information between the MPM and other components of the GDSA. The Fortran version of MPM V.2 is being used to identify the data to be exchanged between the models and coordinate the exchanges in the coupled models, such as memory allocation.

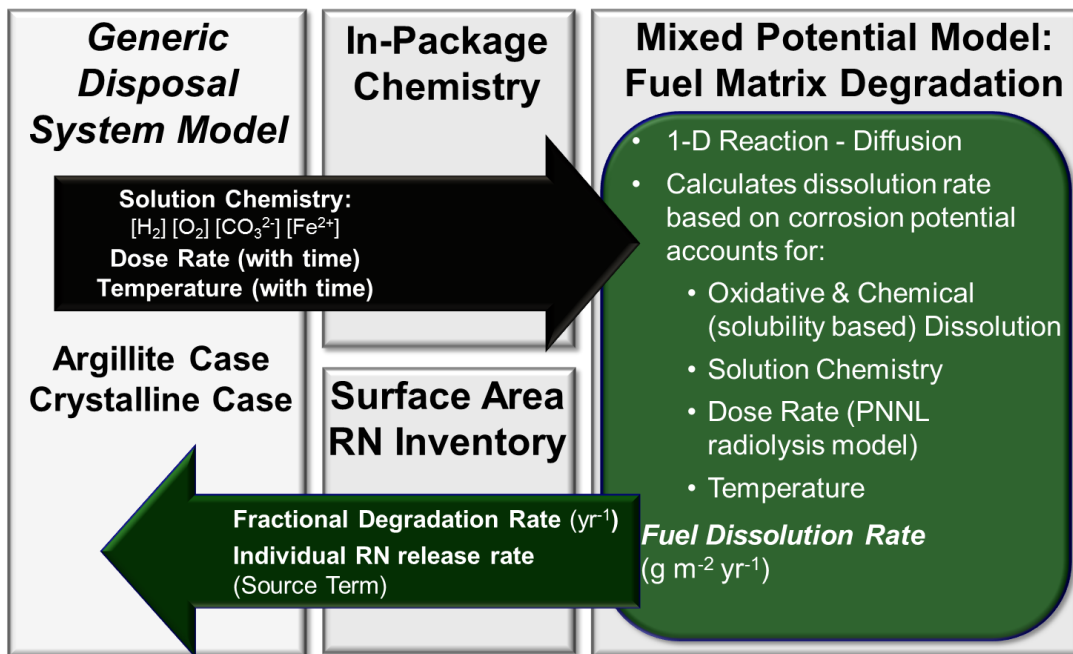


Figure 3. Schematic diagram showing the flow of information between the MPM and other parts of the GDSA.

Integration of MPM V.2 into GDSA

The GDSA for the generic salt repository (Sevougian et al., 2014) made use of fuel degradation rates that were determined by the probabilistic sampling of a range of values derived from Kienzler et al. (2012). This range of fuel degradation rates is not applicable to Argillite or Crystalline rock disposal cases because it is specific to salt brine environments. Furthermore, the statistical sampling approach does not account for important feedback processes between the chemistry of the solution interacting with the used fuel and the fuel dissolution rate.

In the generic PA models for the Argillite and Crystalline rock cases the MPM will replace this statistical sampling approach with a more realistic fuel degradation rate that is sensitive to the burn-up and temperature of the fuel as well as the chemistry of the in-package groundwater solution. The integration of the MPM into GDSA was explored by a collaborative team of scientists from Argonne, Sandia and PNNL (including developers of the PFLOTRAN code) and it was determined that the most promising approach was to configure MPM for direct communication with PFLOTRAN (convert to Fortran) so that the full MPM can be run as needed by during a PA calculation. This approach captures the full functionality of the MPM (e.g. fuel dissolution rate dependence on temperature, dose rate and local solution chemistry) and allows updated versions of the MPM to be easily incorporated into GDSA (modular approach).

This model integration approach was developed based on discussions held at the UFD campaign meetings in 2013 and 2014 as well as a technical meeting held at Sandia on 12/4/2013. A

prototype of the MPM V.2 Fortran version was completed and communicated to other team members on 10/15/2014 (Jerden et al., 2014b). This was followed by another technical meeting held at Sandia on 11/19/2014. The discussions at this most recent meeting identified a number of technical action items that will improve the integrated MPM-PFLOTRAN model. These action items, which are summarized below, are currently being addressed by the Argonne-Sandia-PNNL team.

Summary of action items for implementing and optimizing the combined MPM-PFLOTRAN model for use in the GDSA: The first 5 deal with the values and ranges of key data and the rest with the implementation of the transfers or calculated values.

- Define the relation between the used fuel burn up (average values identified for waste packages in PFLOTRAN) and the initial surface dose rate of the fuel so that the appropriate alpha radiation energy deposition curves can be selected within the MPM based on PFLOTRAN supplying either the fuel burn up or the initial surface dose rate as input to the MPM.
- Evaluate the temperature difference between the waste package surface (calculated by PFLOTRAN based on average burnup) and the used fuel surface. Assess the need to account for this temperature off-set (probably varies with time).
- Define the values of total dissolved carbonate for the ambient system to be used as feeds to the MPM passed by PFLOTRAN. Initial assumption is that the ambient system will be imposing this constituent concentration throughout the post-closure period.
- Define hydrogen and oxygen constraints:
 - Define H₂ and O₂ feeds for PFLOTRAN to pass to MPM.
 - Define ambient H₂ and O₂ fugacity constraints on the system, reflecting equilibrium with groundwater solution.
 - Define fugacity of H₂ generated from corrosion of (a) waste package barrier metal and (b) internal metals.
- Define Fe(II) aqueous concentrations based on the above ambient and corrosion driven systems.
- Define description of evolving surface area of fuel pellets to be used to convert MPM output to fractional degradation rate.
- Define Storage of auxiliary variables within integrated code (MPM- PFLOTRAN).
- Evaluate Recursion within MPM and optimize MPM.
- Develop regression testing for ensure MPM is not broken once built-into PFLOTRAN.
- List explicit table of input variables that go into the MPM.

Fortran Version of MPM V.2

This report documents the completion of the translation of MPM V.2 from MATLAB to Fortran 90 to study aspects of integration with GDSA. However, as shown in Figure 1, the development of the MPM is continuing. Additional modules are being developed for the MPM and MPM V.2 does not represent the final model. Nevertheless, the input/output pieces of the MPM code are sufficiently complete to be beginning designing interfaces between the MPM and PLFLOTRAN code in parallel with continued development of the MPM.

The source code for the Fortran version of MPM V.2 (referred to hereafter as MPM V.2 Fortran) is available to project participants by sending a request to jerden@anl.gov. The code is distributed via an Argonne file transfer site. A distribution list of those who download the code will be used for communicating new version announcements. This source code provides a version of the MPM suitable for evaluating integration with GDSA because it includes the variables to be exchanged between the two models and how they are updated during sequential model runs. It is anticipated that the Fortran MPM V.2 code will go through several iterations (all tracked as new version numbers, e.g., MPM V.2a Fortran) as the code integration work proceeds.

The envisioned role of the MPM within the GDSA will be to calculate the dissolution rate of used fuel to provide source terms for radionuclides released from a breached waste package. The extensive parameter database used by the MPM V2 (Jerden et al. 2014) makes it applicable to a wide range of geochemical settings; however, more experimental data are required to reduce uncertainties in key functionalities such as the temperature dependencies of rate constants.

Table 1 identifies the key variable names within the Fortran code. These variable names can be used to readily locate the functions in which the variables are used.

Notes for Running and Working with the MPM V.2 Fortran Code (updated from Jerden et al., 2014b)

The MPM V.2 was converted from MATLAB to Fortran 90 for incorporation into the PFLOTTRAN code being used in GDSA. This conversion was a line by line translation of the MATLAB code, which greatly increased the number of code lines. The intent of the MPM V.2 Fortran code is to develop the input/output interfaces with PFLOTTRAN code; it has not been optimized for computing run time. The only external library used in the MPM is LAPACK for its linear algebra $Ax=B$ solver “dgesv” [LAPACK 3.5.0 Windows 32-bit static library (from <http://icl.cs.utk.edu/lapack-for-windows/lapack/#libraries>)].

The following lines of code are needed to run the Fortran version of the MPM and are included here to give an example of how the MPM can be called for integration into GDSA. A single call to the MPM V.2 Fortran code requires a call to the subroutine “AMP_step” whose interface is shown below.

```
interface
  subroutine AMP_step ( sTme, conc, initialRun, flux, status )
    real ( kind = 8), intent( in ) :: sTme !time in seconds
    real ( kind = 8), intent( inout ), dimension (:,:) :: conc !concentration
    logical ( kind = 4), intent( in ) :: initialRun ! initial run flag
    real ( kind = 8), intent(out), dimension (:,:) :: flux
    integer ( kind = 4), intent(out) :: status ! zero indicates success
  end subroutine
end interface
```

The first time the code is called, initialRun must be set to “.true.” and the content of the concentration matrix “conc” is ignored. The initial concentrations are defaulted to the values returned by the subroutine “enVals” in file “AMP_class.f90”. In subsequent calls to “AMP_step”, the concentration from the previous AMP_step call must be preserved and used as input to the current AMP_step call.

There are eleven components tracked by the MPM V.2 Fortran code and the order is defined in the file “AMP_compList.Inc”. The first dimension of the concentration matrix and the flux matrix identifies the value for species in this order.

UO2_2p	[symbol for UO_2^{2+}]
UCO3_2n	[symbol for $UO_2(CO_3)_2^{2-}$]
UO2	[symbol for $UO_2(aq)$]
CO3_2n	[symbol for CO_3^{2-}]
O2	[symbol for $O_2(aq)$]
H2O2	[symbol for H_2O_2]
Fe_2p	[symbol for Fe^{2+}]
H2	[symbol for $H_2(aq)$]
UO2_s1d	[symbol for U(IV) oxide precipitate]
UO3_s1d	[symbol for U(VI) oxide hydrate precipitate]
UO4_s1d	[symbol for U(VI) peroxide precipitate]

Below is example code for a call to AMP_step. Memory must be allocated for the matrices “conc” and “flux”. Also required are the “AMP_compList.inc” file which contains the variable nCMP: the number of components that the MPM tracks. Also required is the “AMP_const.inc” file which contains the values identified in Table 1).

```
program main
  implicit none
  include 'AMP_const.inc'
  include 'AMP_compList.inc'

  interface
    subroutine AMP_step ( sTme, conc, initialRun, flux, status )
      real ( kind = 8), intent( in )  :: sTme
      real ( kind = 8), intent( inout ), dimension (:,:) :: conc
      logical ( kind = 4), intent( in ) :: initialRun
      real ( kind = 8), intent(out), dimension (:,:) :: flux
      integer ( kind = 4), intent(out) :: status
    end subroutine
  end interface

  ! subroutine arguments
  real ( kind = 8)  :: sTme  !time in seconds
  real ( kind = 8), dimension (:,:), allocatable :: conc  !concentration
  from previous time step
  logical ( kind = 4) :: initialRun  ! initial run flag
  real ( kind = 8), dimension (:,:), allocatable :: flux
  integer ( kind = 4) :: status  ! zero indicates success

  !local variables
  integer ( kind = 4) :: AllocateStatus

  ! initialize return status - 0 means code ran successfully
  status = 0

  allocate ( conc( nCmps, gridPts ), stat = AllocateStatus )
  if (AllocateStatus /= 0) then
    status = AllocateStatus ! "*** Not enough memory ***"
  end if
  allocate ( flux( nCmps, 1 ), stat = AllocateStatus )
  if (AllocateStatus /= 0) then
    status = AllocateStatus ! "*** Not enough memory ***"
  end if

  ! time in seconds
  sTme = 0.0
  ! if initialRun is true then values will be set to an initial value
  ! and the values in conc are ignored
  ! if initialRun is false the values in conc are used
  initialRun = .true.
  call AMP_step(sTme, conc, initialRun, flux, status)

end
```

Notes for Ongoing Code Integration Work

- Testing of the Fortran MPM V.2 is on-going and the results from this version of the model have not yet been completely validated against the MATLAB version. Data returned from Fortran MPM V.2 should only be used to test integration of the MPM with the PFLOTRAN code. The interface between the MPM and the PFLOTRAN code is not expected to change in subsequent versions of the MPM.
- Optimization of memory use has not been addressed in this version of the Fortran MPM V.2. Optimization of future Fortran versions of the MPM will speed up execution of the code.
- Exception handling has not been addressed and will cause the code to halt. A method on how to best handle exceptions needs to be discussed with the PFLOTRAN programming team.
- The best method to disseminate the model with the PFLOTRAN programming team needs to be established. The method could be the source code, which needs to be compiled or linked, or a library that only needs to be linked.

Future Work on MPM development (on-going work on MPM V.3)

The MPM for fuel degradation requires further development to improve accuracy (experimental determination of key parameter values) and account for key processes that are not yet included in the model (e.g., constraints on the fugacity of H₂ from steel corrosion). On-going work will lead to the following improvements to the model:

- Link MPM with steel corrosion to constrain H₂ fugacity. Previous MPM runs have demonstrated that the presence of moderate amounts of dissolved H₂ can decrease the fuel dissolution rate by approximately four orders of magnitude (Jerden et al., 2014a). However, the fugacity of H₂ within a breached waste package and the feedback with fuel corrosion has not yet been quantified.
- Interface MPM with waste package corrosion and in package chemistry. The evolution of pH and anion concentrations are known to effect used fuel dissolution rates, but not yet fully built into the MPM.
- Quantify the effects of poisoning/alteration of the Noble Metal Particle (NMP) phase. In the MPM, the effect of H₂ largely depends on reactions catalyzed by the NMP phase on the fuel surface (accounted for in MPM V.2). It is known that halides may counteract the catalytic properties of the NMP; however, more experimental evidence is needed to quantify this process in the MPM.
- Replace place-holders values for key parameters used in MPM V.2 with experimentally determined values (e.g., temperature dependence, H₂ protection)

The model integration approach discussed above will allow the updated MPM (V.3) to be readily incorporated into the GDSA.

Reference

Finch, R.J., Ewing R.C., 1992. The Corrosion of Uraninite Under Oxidizing Conditions, *Journal of Nuclear Materials* 190, 133-156

Jerden, J., Copple, J., Frey, K., Ebert, W., 2014a. *ANL Mixed Potential Model For Used Fuel Degradation: Application to Argillite and Crystalline Rock Environments*. FCRD-UFD-2014-000490.

Jerden, J., Copple, J., Frey, K., Ebert, W., 2014b. *Prototype Fortran Version of the Mixed Potential Model for Used Fuel Degradation. Milestone Letter Report M4FT-15AN0806012*.

Kienzler, B., Altmaier, M., Bube, C., Metz V., 2012. *Radionuclide Source Term for HLW Glass, Spent Nuclear Fuel, and Compacted Hulls and End Pieces (CSD-C Waste)*, KIT Scientific Publishing, Report-Nr. KIT-SR 7624, Karlsruher Institut für Technologie (KIT), Straße am Forum 2, D-76131 Karlsruhe, www.ksp.kit.edu

King, F. and Kolar, M., 2003. *The Mixed-Potential Model for UO₂ Dissolution MPM Versions VI.3 and VI.4*, Ontario Hydro, Nuclear Waste Management Division Report No. 06819-REP-01200-10104 R00.

Sevougian, S.D., Freeze, G.A., Gardner, W. P. Hammond, G. E., Mariner, P., 2014. *Performance Assessment Modeling and Sensitivity Analyses of Generic Disposal System Concepts*. FCRD-UFD-2014-000320

Tsai, H., 2003. *Nuclear Regulatory Commission Review of ANL High-Burnup Cladding Performance Program*, Argonne National Laboratory, July 16, 2003.

Table 1. Parameter and variable inputs for Fortran MPM V2.

Variables set by user (these will ultimately be inputs from other PA models)	Variable Name in FORTRAN MPM V.2	Notes
Temperature	function temp(t)	The evolution of the waste form temperature will be the output of other models and depend on burnup, fuel age, repository design, etc.
Dose rate	subroutine rad = aPartRad(t, rad)	Dose evolution will be the output of other models accounting for fuel burnup and age.
Environmental concentrations of dissolved oxygen	subroutine envVals(cEnv) cEnv(O2)	This key variable will be determined by a number of interdependent kinetic processes within the waste package and near-field, but the dominant buffer may be the geologic environment: Argillite or Crystalline Rock.
Environmental concentration of dissolved carbonate	subroutine envVals(cEnv) cEnv(CO3_2n)	This is a key variable determined by specifics of the geologic environment: Argillite vs. Crystalline Rock.
Environmental concentration of dissolved ferrous iron	subroutine envVals(cEnv) cEnv(Fe_2p)	This key variable will be determined by a number of interdependent kinetic processes within the waste package and near-field, but the dominant source will be corroding steel components.
Environmental concentration of dissolved hydrogen	subroutine envVals(cEnv) cEnv(H2)	This key variable will be determined by a number of interdependent kinetic processes within the waste package and near-field, but the dominant source will be anoxic corrosion of steel components.
<i>Space and Time Parameters</i>		
Length of diffusion grid in model	constant cellLen	Arbitrary, can be changed by user to represent system of interest.
Number of calculation nodes (points) in diffusion grid	constant gridPts()	Arbitrary, can be changed by user to optimize calculation efficiency.
Duration of simulation	constant simTime()	Arbitrary, can be changed by user to represent duration of

		interest.
<i>Radiolysis Parameters</i>		
Alpha-particle penetration depth	constant penD =	The basis for this value was re-examined in FY 2014, but remains unchanged (confirmed). See section 2.1 for discussion.
Generation value for H ₂ O ₂	Subroutine PNNL_gval(rad,cO2,CH2,Gval,dcO2,dcH2)	this value varies depending on the dose rate and dissolved concentrations of O ₂ and H ₂ within the irradiation zone
<i>Physical Interfacial Parameters</i>		
Surface coverage of NMP	constant props_frac	Based on qualitative examination of photomicrographs of spent fuels of different burnups e.g., Tsai, 2003.
Resistance between UO ₂ and NMP domains	constant props_resist()	Assumed that fuel and NMP are electrically well-coupled
Porosity of schoepite corrosion layer	constant ePor()	Reasonable assumption based on qualitative assessment of photomicrographs of schoepite layers e.g., Finch and Ewing, 1992.
Tortuosity factor of schoepite corrosion layer	constant eTor	Reasonable assumption based on analogy of schoepite layer with compacted clay (King and Kolar, 2003)