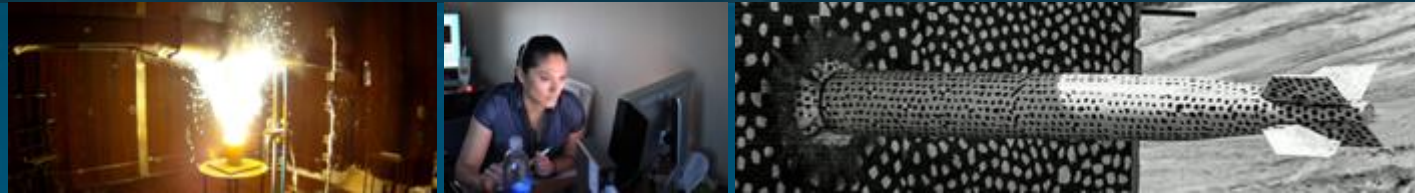


MELCOR Best Practices and Revised Input Structures for HTGR and FHR



The 13th Meeting of the European MELCOR and MACCS User Group

April 27-29, 2021

PRESENTED BY

Bradley Beeny, Sandia National Laboratories



Introduction

- Background: MELCOR for HTGRs and FHRs
- Purpose of “best practices” discussion and input structure revisions

Best practice guidelines for...

- Run-time strategy and user input formatting
- COR
 - Components and materials
 - Core support
 - Thermal hydraulics (conduction, convection, oxidation)
 - Nodalization
- Fission product release model (COR, DCH, RN1)
 - Fuel modeling
 - DCH/RN1 mapping

Conclusions



Timeline of HTGR/FHR MELCOR model development:

Prior to 2008 – few provisions for PMR/PBR or FHR, e.g. users adapted LWR models for PBRs

2008 – CSARP presentation outlines initial development plans/efforts

- New COR package reactor types, components (reflector, RF), and conduction/convection models
- Revisited/updated CVH/NCG capabilities (helium properties)
- Beginnings of fuel fission product release models, graphite oxidation model, and point kinetics model

2009 to 2010 - Miscellaneous improvements, accelerated steady-state, early evaluation model

2011 to 2017 - Limited development activity, some activity from the code user base

2018 to present – Non-LWR modeling initiative

- Revamped/streamlined the HTGR fission product release model (COR) input processing
- Revisited methods and strategies...added, subtracted, reconfigured as necessary
 - MELCOR accident/transient run-time strategy
 - Larger evaluation model picture (cooperation/coordination with SCALE/ORIGEN)
- Added new physics models and modified existing ones, making the code considerably more robust
- Built on existing PBR-400 model for a demonstration source term calculation
- New hydrodynamic materials available via expanded EOS capabilities (e.g. FLiBe salt)
- Built an FHR model for a demonstration source term calculation

Introduction – Purpose of “Best Practices” Discussion



Raise awareness about the present state of the HTGR and FHR modeling methodology

- 10+ year development history at this point
- Last 2 years have been busy with a renewed development effort
- Modeling methodology writ large has come a long way recently

Encourage user base to implement the new in place of the old

Outline a few noteworthy aspects of current best practices

- How to leverage streamlined input structures to access new capabilities?
- How is a given transient/accident analysis configured from start to finish?
- How should COR components be situated to ensure proper thermal hydraulics function?
- How is the fission product release model properly configured?
 - How are diffusion release calculations set up?
 - How do the COR radionuclide inventories correspond to DCH and RN1?



Previously, a series of external text files configured the fission product release model and executed it in sequence:

- An accelerated steady-state stage
- A steady-state diffusion stage
- A steady-state transport stage
- A transient diffusion/transport stage

Currently, COR MELGEN input records accomplish the same task more conveniently and more compactly

- Seamlessly navigate the four stages
- MELCOR internal database manages fission product release model data
- More maintainable and portable input decks

Several extraneous inputs from the old input format have been eliminated (including since EMUG '21)



COR global inputs COR_DIFF and COR_DIFFGX

- COR_DIFF to configure steady-state diffusion stage timing
- COR_DIFFG1 for the steady-state burn-up/irradiation time
- COR_DIFFG2 for total numbers of models (all TRISO plus matrix) and tracked species
- COR_DIFFG3 and COR_DIFFG4 for globally-applicable fuel parameters
- COR_DIFFG5 for globally-applicable tracked species parameters declaration
- COR_DIFFG6 for declaration of matrix model number and failed TRISO model number
- COR_DIFFG7 for area scaling if applicable

COR model-wise block inputs COR_DIFFMX...applicable to each model in the problem

- COR_DIFFM1 and COR_DIFFM2 for characteristics of the model (name, number, geometry, materials)
- COR_DIFFM3 for model grid geometry/noding definitions
- COR_DIFFM4 for zone-wise initialization and information
- COR_DIFFM5 for zone-wise diffusion coefficient Arrhenius parameter definitions
- COR_DIFFM6, COR_DIFFM7, and COR_DIFFM8 for boundary condition data

COR_XPRT for configuring the steady-state transport phase of the calculation

COR_DIFFT, COR_DIFTG1, and COR_DIFTG2 for configuring transient diffusion timing and for toggling options for the analytic release model



Begin an accident/transient analysis from a thermal steady-state

- EXEC_SS accelerated steady-state capability
- Looking for settled and unchanging COR, CVH, HS temperatures
- MELCOR can print a file full of MELGEN input (EXEC_WRT)
 - COR and HS initialization (COR_CIT, HS_NOD)
 - For same initial condition and nuclear system, don't have to repeat accelerated steady state calculation

Then, if HTGR fission product release model is configured, recommend running through all three stages in every instance of a calculation

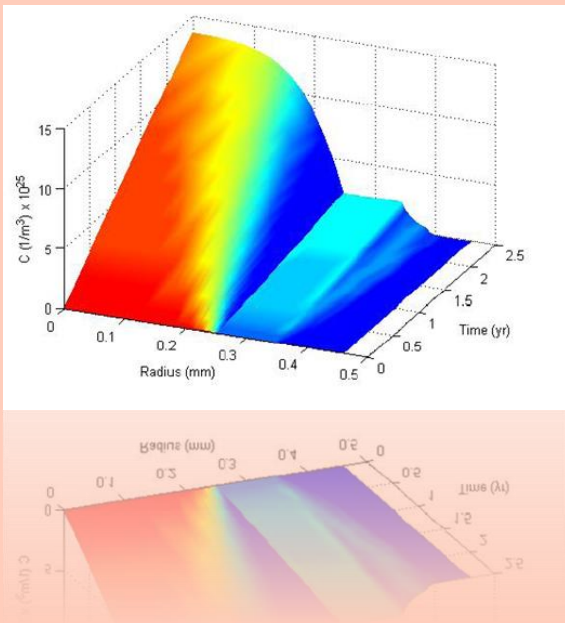
- Steady-state diffusion (SS DIF), executed within a time-step at a prescribed problem time
 - Get steady, end-of-burnup radionuclide distributions within COR
 - Get steady, end-of-burnup radionuclide releases from COR components
- Steady-state transport (SS XPRT), executed over a stretch of “pseudo time”
 - Hold SS DIF conditions fixed, let CVH/HS radionuclide characteristics settle to steady-state
 - Scaling of CV contents and HS deposition is possible at conclusion of SS XPRT
- Transient diffusion/transport (TRANS DIF/XPRT) kicks off transient analysis



Stage 1: Steady State Diffusion Calculation

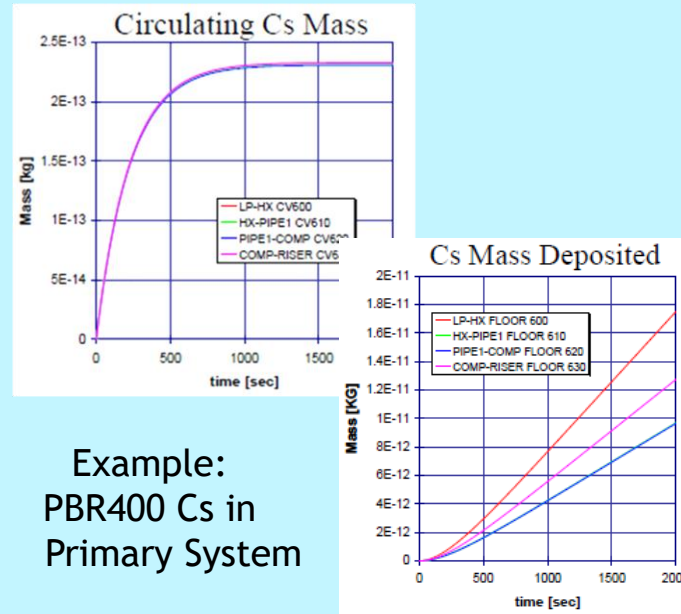
Establish steady state distribution of radionuclides in TRISO particles (and matrix) for normal operation

** assumes a thermal steady-state has been established*



Stage 2: Steady State Transport Calculation

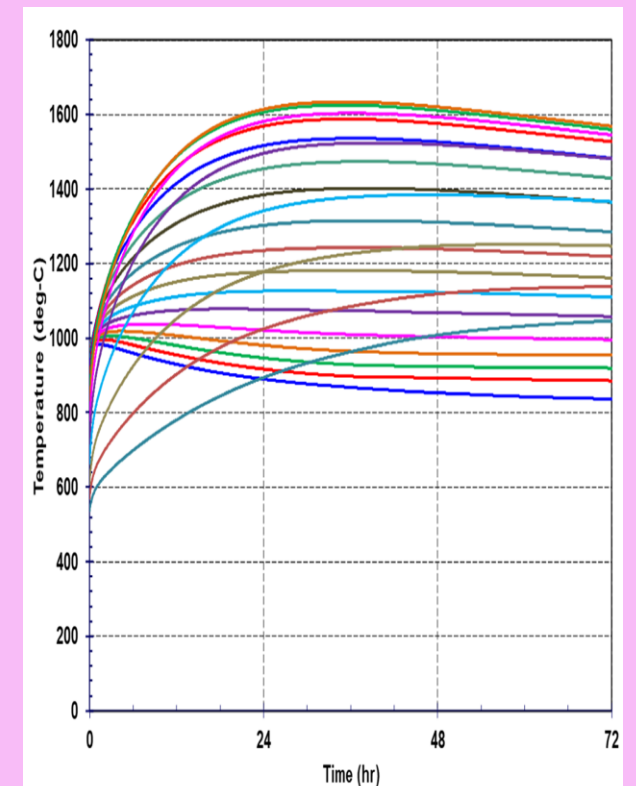
Calculate steady state distribution of radionuclides throughout system during normal operation (deposition on surfaces, convection through flow paths, etc.)



Example:
PBR400 Cs in
Primary System

Stage 3: Transient Diffusion & Transport calculation

Calculate accident progression and radionuclide release



All steps performed in one run with data passed transparently between stages



Reflector (RF) is non-supporting, two-sided, multiple possible orientations

- Radial (inner or outer) and axial (upper or lower)
- Primary material is typically graphite

Matrix (MX) is non-supporting, one-sided, “thick” with thermal resistance

- Ties to HTGR fission product release model and participates in radionuclide transport
- Best practice: Use for non-TRISO bearing graphite in fuel elements
- Primary material is typically graphite
- Effective conductivity (Tanaka-Chisaka, Zehner-Schlunder-Bauer) models

Fuel (FU)

- Ties to HTGR fission product release model and participates in radionuclide transport
- Best practice: Use for TRISO and TRISO bearing graphite in fuel elements
 - UO₂ or UCO for primary material
 - Graphite for “extra” material (FUXM)



RF and MX are non-supporting and ultimately rely on SS for support while conveying weight load to like components below

Some support problems arise naturally from HTGR core layout

- Top reflector (RF) that hangs over an active core (MX)
- Active core (MX) that sits atop bottom reflectors (RF)
- Easily remedied with incorporation of graphite-bearing SS between RF and MX

COR_SS allows definition of SS support rules with graphite as the material

Generally, best practice is to avoid mixing SS with MX-bearing or RF-bearing COR cells



Flow/convection

- MX conduction resistance factors into MX-to-coolant heat transfer (parallel)
- Nusselt number correlations (PBR or PMR) accessible by SC
- Use packed bed pressure drop (flow resistance, K-loss) formulations for PBR
- Channel/bypass flow area allocations for RF

Core conduction

- Intracell conduction (FU-to-MX) is “on” automatically, non-negligible MX resistance
- Intercell conduction (MX-to-MX, MX-to-RF, MX-to-SS, RF-to-SS, RF-to-RF, SS-to-SS) whether axial or radial ought to behave properly (discourage MX, RF, SS collocation)
- CORSC1507 disables cell component volume fraction scaling for radial conduction
- Physically impossible conduction pathways are logically excluded, e.g. FU-to-FU intercell axial/radial conduction for PBRs
- Boundary conduction to radial boundary heat structures should be utilized

Oxidation of graphite primary materials in MX, RF, or SS ought to proceed naturally in the presence of oxygen (air) or steam



General guidelines:

- Use RF only in cells that comprise reflectors (besides SS graphite in cells for core support)
- Use FU and MX only in active core cells (if it isn't clear by now, CL is deprecated for PBR/PMR)
- Use SS for actual core support structures
- Use SS on an as-needed basis to hold up MX and/or RF and avoid support logic issues
- Avoid convoluting the COR conduction picture unnecessarily
- Choose radial boundary HS's judiciously in view of boundary conduction model

Previous limitations forced users into certain input tricks that should no longer be necessary under any circumstances to achieve modeling goals

- User-defined materials (UDM) and generic oxidation modeling (GOM) capabilities
- Creative use of COR components beyond their intended application space (especially CL/MX)
- Artificial tuning of certain parameters (e.g. component view factors) to impact heat transfer

Best Practices – COR Nodalization: Swapping MX for CL



Formerly, CL physically represented unfueled regions of fuel elements (pebble shells, hex blocks)

- Located CL along with FU in active core cells
- Used COR_KCL to define CL mass by cell
- Associated CL with graphite somehow
 - Primary material (zircaloy) swapped for graphite on COR_CLM and primary material mass specified on COR_KCL, or
 - Used the material slot for graphite on COR_KCL such that CL consisted entirely of graphite

Current best practice (and soon a requirement enforced on user input) is to use MX with PBR or PMR

Current preferred method of incorporating MX in view of new UDM and GOM capabilities:

- Remove COR_KCL, COR_CLM, and any other notion of CL
- Disregard anything seen/heard/used in recent past regarding COR_KMX, COR_MXM, etc.
- Leverage ‘COR-USER-METAL’ (or UDM3) and MX on COR_NMAT to specify cell mass
- Leverage UDM input in MP to set up UDM3 as built-in (or otherwise user-defined) graphite
- GOM should automatically connect graphite oxidation model



Reminders...

Overall goals of the diffusional fission product release model include:

- Prediction of radionuclide distributions within fuel
- Prediction of radionuclide release from fuel elements to coolant

Conventional CVH/HS/RN1 takes over once radionuclide inventory escapes COR

A general configuration of the radionuclide release/transport problem includes:

- **Two or more models** representing all cohorts of the TRISO population (e.g. intact, failed)
- **One model** representing the fuel element matrix
- **One or more tracked fission product species** mapped to radionuclide classes
- Global parameters describing the HTGR (PBR or PMR) core
- A one-dimensional spherical or cylindrical finite volume solution grid for each defined **model**
- Diffusion coefficient data (Arrhenius temperature relations)
- Miscellaneous data (failure, boundary conditions, sorption isotherms, partition coefficients)

Ascertain radionuclide distribution/release with FV diffusion and/or modified Booth



Best practice guidelines:

- Configure models for two TRISO cohorts at minimum: intact TRISO and failed TRISO
- Do not configure models for more TRISO cohorts unless sufficient information is available
- Configure only a single matrix model (more aren't allowed) according to fuel element design (e.g. a conventional PBR pebble vs. a fueled spherical shell pebble)
- Be careful about initial TRISO model population fractions and about those dynamics during the transient diffusion/transport stage of the calculation
- Choose reasonable recoil and contamination fractions
- Don't overdo finite volume diffusion grid noding for each model
- Use the best diffusion coefficient data available...depending on the circumstances, absurd diffusivities can lead to difficulties in the SS DIF stage
- Track as many fission product species as necessary
 - Some species diffusion data is required
 - Consider RN1 mapping and note can scale RN class releases to any given tracked species
- Do not invoke partition coefficients or sorption isotherms unless sufficient data is available (or for sensitivity probing)

Best Practices – FP Release DCH/RN1 Mapping



Use DCH input to:

- sort chemical elements into DCH/RN1 classes as normal (DCH_CL),
- set the total element/class mass inventories for the problem (DCH_EL),
- set the specific decay powers for each chemical element (DCH_EL)

Use RN1 input to:

- apportion DCH class mass totals to FU components of individual COR cells (RN1_FPN)
- coordinate mapping from tracked fission product species to DCH/RN1 classes (RN1_DIFMAP)
- specify any DCH/RN1 class release scaling rules (RN1_DIFMAP)

Use COR FP release input to declare a tracking option (COR_DIFFG5)



HTGR Fission Product Release Model

Recommendations for Global User Inputs (1/2)



* Text in **red** indicates any deprecated MELGEN inputs

* Text in **green** indicates any new information

COR_DIFF

- Generally, make DFLAGS = 1 and start the calculation (EXEC_TSTART) somewhere in problem time prior to the selected time DTIMEONS...this will ensure MELCOR executes the SS DIFF stage at DTIMEONS (**which can be given by CF or constant**)
- Do the steady-state diffusion calculation at a point in problem time DTIMEONS where an accelerated steady-state run (EXEC_SS) has established constant COR, HS, and CVH temperatures
- **Strongly suggest inputs of 'NO' for both MDIFFILE and INITFILE**

COR_DIFFG1

- TBURNUP should reflect amount of time for which the core was operating at steady full power prior to the accident/transient in question

COR_DIFFG2

- NREG is the total number of TRISO models plus one for the matrix model. Minimum recommended level of detail is NREG = 3 so that there is space for three models: intact TRISO, failed TRISO, and matrix
- NSPECIES is the total number of tracked fission product species. Note that these species must be mapped to an RN1 class somehow. The bare minimum is NSPECIES = 1 so that at least one species is being calculated.

COR_DIFFG3

- NPARTPFUEL approximates the number of total TRISO particles (i.e. TRISO particles of all kinds be they intact, failed, etc.) in a single fuel element (a pebble or compact)
- Choose/set FRECOIL and FCONTAM to represent the desired amount of allowed fission product recoil and allowed tramp uranium contamination in the matrix

HTGR Fission Product Release Model

Recommendations for Global User Inputs (2/2)



* Text in **red** indicates any deprecated MELGEN inputs

* Text in **green** indicates any new information

COR_DIFFG4

- Choose/set XFUMASSPFUEL to approximate the mass of fissile material in a single fuel element (pebble or compact)

COR_DIFFG5

- Input a table of length NSPECIES with appropriate decay constants for each tracked species (use 0.0 if no decay)

COR_DIFFG6

- General recommendation of MFAIL = 2 and MMATRIX set to 1 plus the number of TRISO particle models in the problem, e.g. if intact TRISO is model number 1 and failed TRISO is model number 2, set MMATRIX to model number 3

COR_DIFFG7

- AFACTOR is intended to account for deviations in fuel element geometry, so use a scalar other than unity if, e.g., a non-cylindrical fuel compact is being modeled for a PMR. The AFACTOR should then be set to a scalar that makes the cylindrical fuel compact outer surface area equal to the true fuel compact outer surface area

HTGR Fission Product Release Model

Recommendations for Model-wise User Inputs (1/2)



* Text in **red** indicates any deprecated MELGEN inputs

* Text in **green** indicates any new information

COR_DIFFM1

- Enter a descriptive object name for MODNAME and choose a model number NMODEL bearing in mind the global values for MFAIL and MMATRIX
- Use **'BOTH'** for QOPT
- Pick a spherical or cylindrical geometry for NMODEL=MMATRIX according to reactor type, but for all TRISO models of any type, only GEOM='SPH' for spherical geometry is allowed
- Use **'SS'** for TTYPE

COR_DIFFM2

- For all TRISO models, NKERN = 1 (innermost zone is the first zone which is the fuel-bearing or kernel zone)
- For all TRISO models that have a buffer region, NBUFF ought to be 1+NKERN as the buffer surrounds the kernel
- Total number of nodes must be consistent with subsequent model-wise grid geometry input
- Ensure FRAC population fraction input is consistent with input for other models that all together comprise the total TRISO population. Note that for the failed TRISO model, population fraction data is given elsewhere (**CF or constant can be used for FRAC**)

COR_DIFFM3

- Choose zone-wise material names that are identified in MP package input
- Use **QDOT0** of unity in fueled zones and use **QDOT0** of zero elsewhere, e.g. in a typical intact TRISO model, the kernel zone should have **QDOT0** of 1.0 and all other zones should have a **QDOT0** of 0.0

COR_DIFFM4

- Give a molecular weight for the material **and an initial temperature**

HTGR Fission Product Release Model

Recommendations for Model-wise User Inputs (2/2)



* Text in **red** indicates any deprecated MELGEN inputs

* Text in **green** indicates any new information

COR_DIFFM5

- Use the best data available to configure zone-wise diffusion coefficients, or treat them parametrically
- Only attempt to configure partition coefficients and sorption isotherms if sufficient data is available to round out the description on subsequent input records (or if probing sensitivities)

COR_DIFFM6

- At the “inner” boundary, recommend symmetry (zero flux) conditions for diffusion

COR_DIFFM7

- At the “outer” boundary, recommend zero concentration (MOLE 0.0) for diffusion

COR_DIFFM8

- According to the partition coefficient formulation and/or the sorption isotherm empirical model, round out the description here

HTGR Fission Product Release Model

Recommendations for Other User Inputs



* Text in **red** indicates any deprecated MELGEN inputs

* Text in **green** indicates any new information

COR_XPRT

- Recommend 0 for DFLAGS so that the SS XPRT stage may commence immediately upon the conclusion of the SS DIFF calculation
- **Strongly suggest 'NO' for INITFILE and TRANSFILE**

COR_DIFT

- Recommend 0 for DFLAGT so that the TRANS DIFF/XPRT stage commences upon conclusion of the SS XPRT stage
- DTIMEONT is the problem time where SS XPRT concludes and TRANS DIFF/XPRT begins, so ensure that by this problem time the requisite CVH/HS/RN1 steady-state has been established (**DTIMEONT can be CF or constant**)
- **Strongly suggest 'NO' for INITFILE, TRANSFILE, MDIFFFILE, FAILFILE**
- **GSCALE is a user-imposed scalar on the CVH radionuclide contents and the HS radionuclide depositions upon conclusion of the SS XPRT stage. GSCALE is currently surmised from user input as TBURNUP/(DTIMEONT-DTIMEONS)**
- A CF (**to include a vectorized CF**) can specify the failed TRISO population dynamics, otherwise a built-in temperature curve can be used

COR_DIFTG1

- Optionally set a problem time at which the TRANS DIFF/XPRT stage concludes and diffusion is “frozen”
- Optionally request diffusion time step characteristics that are taken into consideration in view of the COR timestep

COR_DIFTG2

- Configure analytic release model parameters to treat the failed TRISO model (**a few new options to check out**)



Elements of best practice pertaining to MELCOR HTGR/FHR modeling were reviewed

- User input
- Run-time strategy
- COR package
- Fission product release

Recommendations assume availability of a more recent MELCOR version for which the general state of HTGR and FHR modeling capabilities is much improved

As always, feel free to contact the SNL staff with questions and concerns as we are in an active development cycle and user feedback is helpful and constructive (Brad - babeeny@sandia.gov, Larry - llhumph@sandia.gov)

AMUG '21 included an in-depth HTGR modeling workshop, and materials are available though some input structures have since been updated