Neutron Kinetics and Dynamics in Liquid-Fueled Nuclear Reactors

SIAM CSE 2019 Minisyposium MS187: Computational Methods for Linear Kinetic Transport Equations

Kathryn Huff
Advanced Reactors and Fuel Cycles Group

University of Illinois at Urbana-Champaign

February 27, 2019

ILLINOIS

Outline

Advanced Reactors and Fuel Cycles group (PI: Kathryn Huff)

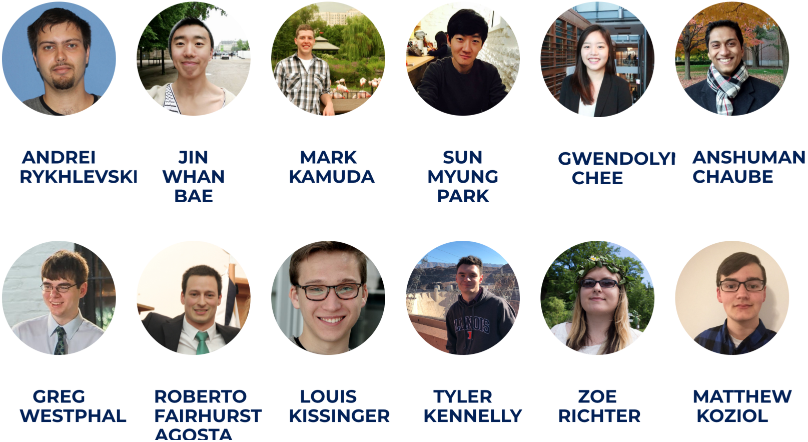


Figure: Current undergraduate and graduate students.

Advanced Reactors and Fuel Cycles group (PI: Kathryn Huff)

Figure: Past ARFC Group members who contributed to this work.

Fission

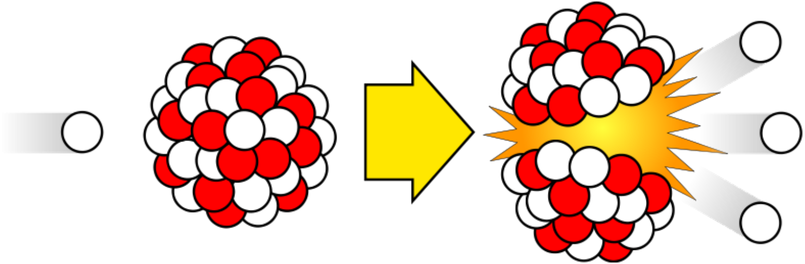


Figure: Cross sections: $\sigma(E, \vec{r}, \hat{\Omega}, T, x, i)$

Fission Chain Reaction

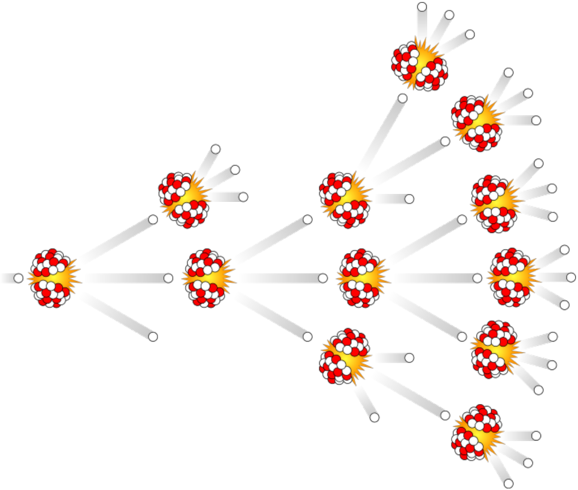
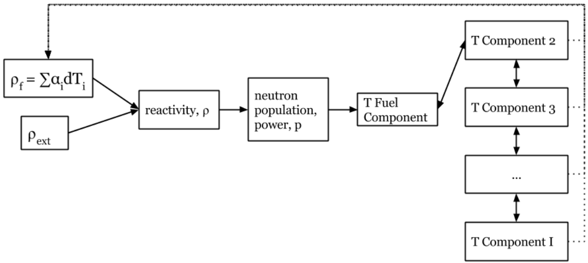


Figure: Criticality: $k = 1$

Reactivity

$$
\begin{aligned}
k &= \text{"neutron multiplication factor"} \\
&= \frac{\text{neutrons causing fission}}{\text{neutrons produced by fission}} \\
\rho &= \frac{k-1}{k} \\
\rho &= \text{reactivity}
\end{aligned}
$$

Feedback


$\rho_f = \sum \alpha_i dT_i$
ρ_{ext}
reactivity, ρ
neutron population, power, p
T Fuel Component
T Component 2
T Component 3
...
T Component I

Kinetics with Delayed Neutrons

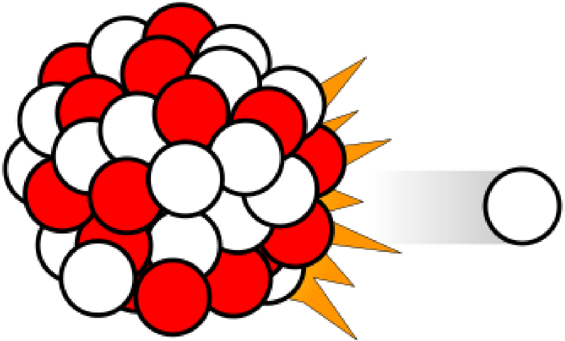


Figure: Delayed neutron fraction, β_i, and corresponding decay constant, $\lambda_{d,i}$.

Types of Molten Salt Reactors

Stationary Fuel

- Prismatic graphite block with TRISO fuel and coolant channels (e.g. FHR DR, TMSR-SF1). Clean salt coolant.
- Stationary TRISO pebble matrix (e.g. TMSR-SF)

Mobile Fuel

- Mobile solid fuel elements, such as pebbles. Clean salt coolant. (e.g. PB-FHR/Kairos)
- Non-circulating fuel salt, "can-type". (e.g. Terrapower MCFR)
- Circulating fuel salt "pool-type". (e.g. MSRE, MSBR, MSFR, Terrestrial MSR, TAP MSR, etc.)

Stationary Solid Fuel

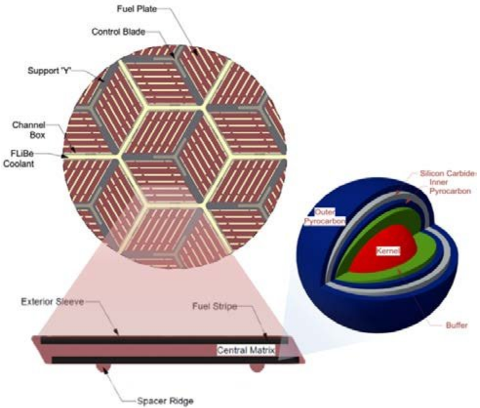


Figure: The AHTR [4] is an example of a fluoride salt cooled reactor design fueled by a **stationary**, **solid** prismatic graphite TRISO compacts, and cooled by clean fluoride salt. Image source [6].

Mobile Solid Fuel

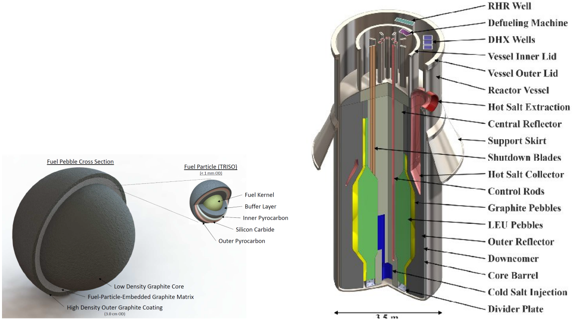


Figure: The PB-FHR is an example reactor design fueled by **solid**, **mobile** graphite pebbles, with TRISO particles embedded in them. Image source [1].

Mobile, Non-Circulating, Liquid Fuel

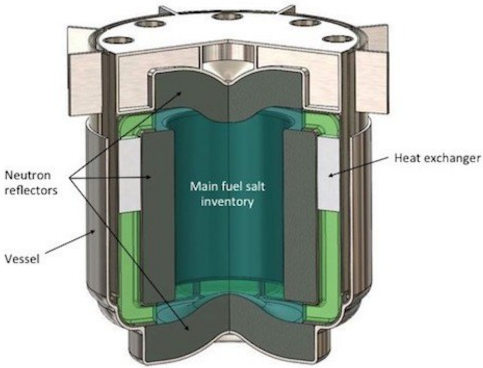


Figure: The MCFR from TerraPower is an example reactor design with **liquid**, **mobile**, **non-circulating** chloride salt fuel. Image source [14, 2].

Mobile, Circulating, Liquid Fuel

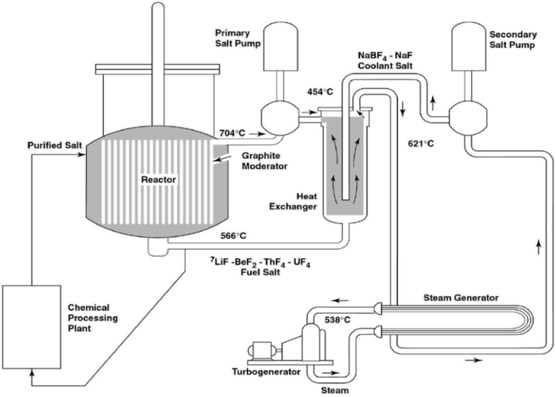


Figure: The MSBR [9] is an example reactor design with **liquid**, **mobile**, **circulating** fluoride salt fuel, including breeding behavior due to varying channel shapes and sizes. Image source [10].

Why Molten Salt Reactors?

Main advantages of liquid-fueled Molten Salt Reactors (MSRs) [3]

1. High coolant temperature (600-750°C).
2. Various fuels can be used (^{235}U, ^{233}U, Thorium, U/Pu).
3. Increased inherent safety.
4. High fuel utilization $\Rightarrow$ less nuclear waste generated.
5. Online reprocessing and refueling.

Main advantages of MSBR [9]

1. Produces more fissile material than it consumes (breeding ratio 1.06).
2. Thorium cycle limits plutonium and minor actinides.
3. Could transmute spent fuel from existing Nuclear Power Plant (NPP).

Challenges in Liquid-Fueled Reactor Simulation

1. Contemporary burnup codes cannot treat fuel movement.
2. Neutron precursor locations drift before neutron emission.
3. Operational and safety parameters change during reactor operation.
4. Neutronics and thermal hydraulics are tightly interdependent.

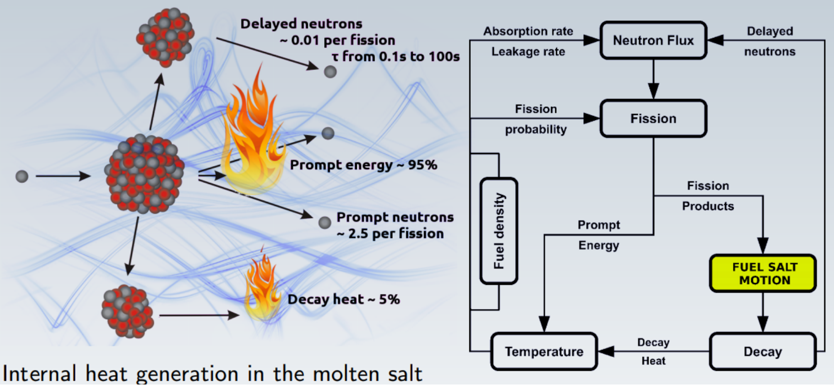


Figure: Challenges in simulating MSRs (Image courtesy of Manuele Aufiero, 2012).

Approaches

Point Reactor Kinetics

PyRK [7], for example, is only appropriate for stationary or nearly stationary fuels.

Multiphysics simulation of MSR (Moltres)[8]

1. Steady-state and transient coupling of neutron fluxes, precursor drift, and thermal-hydraulics.
2. Incorporates advective movement of delayed neutron precursors.
3. 2D axisymmetric and 3D geometries supported.

Simulation of online reprocessing and depletion (SaltProc)[11, 13]

1. Create high-fidelity full-core neutronics model of the core neutronics can be necessary for reducing compounding error.
2. SaltProc wraps SERPENT monte carlo neutron transport for simulation of liquid fuel reprocessing.
3. Enables day-to-day rsolution off neutronics and reprocessing modeling over many decades of depletion and fuel cycle performance.

Outline

PyRK: Python for Reactor Kinetics

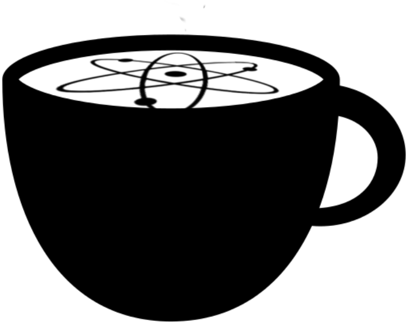


Figure: Special purpose reactor kinetics python tool (https://github.com/pyrk/pyrk) [7]. Research software for simple PRKE: *caveat emptor.*

- Multiple precursor groups (j groups)
- Multiple decay heat groups (k groups)
- Lumped Parameter thermal hydraulics model
- Optional 1-D conduction in pebble fuel compacts
- Object-oriented, geometry and material agnostic framework

Point Reactor Kinetics

$$p = \text{ reactor power} \tag{1}$$
$$\rho(t, T_{fuel}, T_{cool}, T_{mod}, T_{refl}) = \text{ reactivity} \tag{2}$$
$$\beta = \text{ fraction of neutrons that are delayed} \tag{3}$$
$$\beta_j = \text{ fraction of delayed neutrons from precursor group j} \tag{4}$$
$$\zeta_j = \text{ concentration of precursors of group j} \tag{5}$$
$$\lambda_{d,j} = \text{ decay constant of precursor group j} \tag{6}$$
$$\Lambda = \text{ mean generation time} \tag{7}$$
$$\omega_k = \text{ decay heat from FP group k} \tag{8}$$
$$\kappa_k = \text{ heat per fission for decay FP group k} \tag{9}$$
$$\lambda_{FP,k} = \text{ decay constant for decay FP group k} \tag{10}$$
$$T_i = \text{ temperature of component i} \tag{11}$$

Point Reactor Kinetics

$$
\frac{d}{dt}\begin{bmatrix} p \\ \zeta_1 \\ . \\ \zeta_j \\ . \\ \zeta_J \\ \omega_1 \\ . \\ \omega_k \\ . \\ \omega_K \\ T_i \\ . \\ T_I \end{bmatrix} = \begin{bmatrix} \frac{\rho(t,T_i,\cdots)-\beta}{\Lambda} p + \sum_{j=1}^{j=J} \lambda_{d,j}\zeta_j \\ \frac{\beta_1}{\Lambda} p - \lambda_{d,1}\zeta_1 \\ . \\ \frac{\beta_j}{\Lambda} p - \lambda_{d,j}\zeta_j \\ . \\ \frac{\beta_J}{\Lambda} p - \lambda_{d,J}\zeta_J \\ \kappa_1 p - \lambda_{FP,1}\omega_1 \\ . \\ \kappa_k p - \lambda_{FP,k}\omega_k \\ . \\ \kappa_{kp} - \lambda_{FP,k}\omega_k \\ f_i(p, C_{p,i}, T_i, \cdots) \\ . \\ f_I(p, C_{p,I}, T_I, \cdots) \end{bmatrix} \tag{12}
$$

Lumped Parameter Heat Transfer

The heat flow out of body i is the sum of surface heat flow by conduction, convection, radiation, and other mechanisms to each adjacent body, j:

$$
\begin{aligned}
Q &= Q_i + \sum_j Q_{ij} && (13)\\
&= Q_i + \sum_j \frac{T_i - T_j}{R_{th,ij}} && (14)\\
\dot{Q} &= \text{total heat flow out of body i } [J \cdot s^{-1}] && (15)\\
Q_i &= \text{other heat transfer, a constant } [J \cdot s^{-1}] && (16)\\
T_i &= \text{temperature of body i } [K] && (17)\\
T_j &= \text{temperature of body j } [K] && (18)\\
j &= \text{adjacent bodies } [-] && (19)\\
R_{th} &= \text{thermal resistence of the component } [K \cdot s \cdot J^{-1}]. && (20)
\end{aligned}
$$

PB-FHR Example

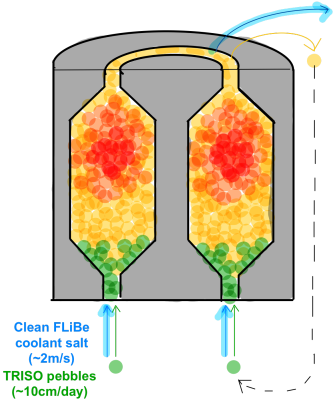


Figure: The pebble fuel can be assumed approximately stationary, as their movement is not comparable to the longest precursor decay times.

Point Reactor Kinetics

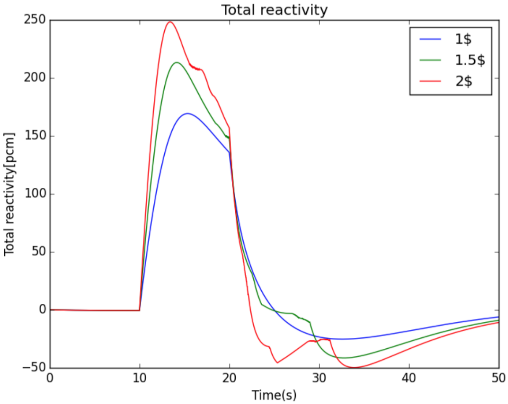


Figure: Total reactivity during ramped reactivity insertion as a function of inserted reactivity [15].

PB-FHR Example

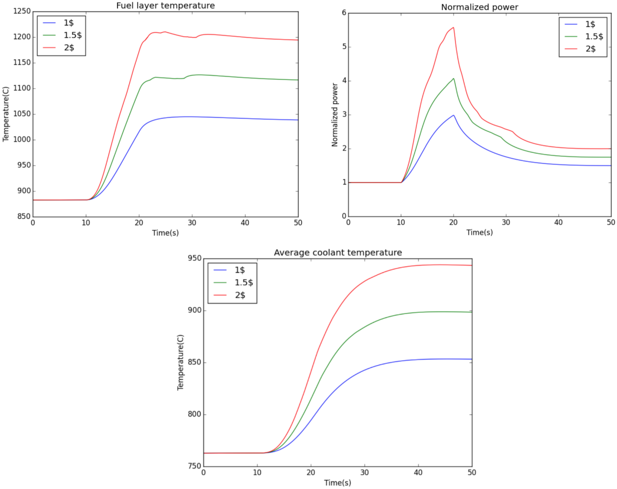


Figure: Average fuel temperature (left) and average normalized core power (right) during a ramp reactivity insertion in the PB-FHR [15].

Point Reactor Kinetics

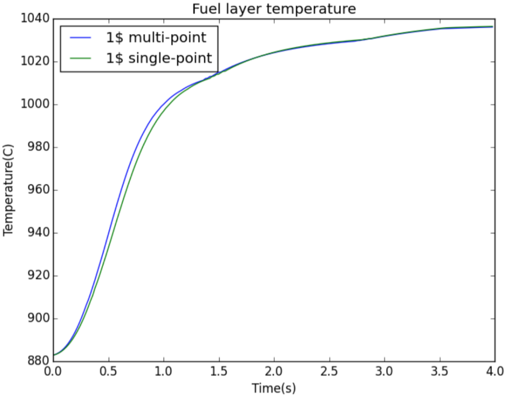


Figure: Fuel temperature rise following 1$ ramp reactivity insertion, calculated with multipoint and single point kinetics in PyRK [15].

Outline

MOOSE Framework

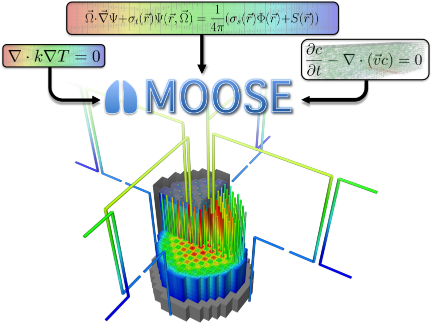
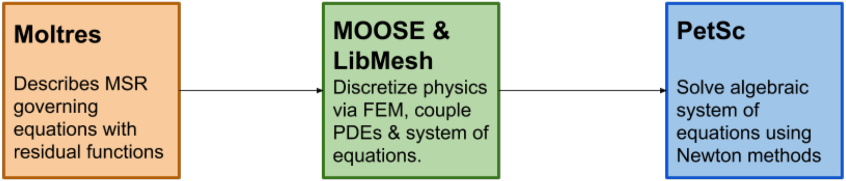


Figure: Multi-physics Object-Oriented Simulation Environment (MOOSE).

- From Idaho National Lab (Gaston et al. [5])
- Fully-coupled, fully-implicit multiphysics solver
- MOOSE interfaces with libMesh to discretize simulation volume into finite elements
- Residuals and Jacobians handed off to PetSc which handles solution of resulting non-linear system of algebraic equations
- Automatically parallel (largest runs $>$100,000 CPU cores!)
- Built-in mesh adaptivity
- Intuitive parallel multiscale solves

Moltres (Coupling in MOOSE)



Intro to Moltres

- Fluid-fuelled, molten salt reactors
- Multi-group diffusion (arbitrary groups)
- Advective movement of delayed neutron precursors
- Navier-Stokes thermal hydraulics
- 3D unstructured
- 2D axisymmetric
- 3D structured
- Initial developer: Alexander Lindsay [8]
- Continued use and development ongoing at UIUC.

Acquiring Moltres

```
git clone https://github.com/arfc/moltres
cd moltres
git submodule init
git submodule update
```

Diffusion in Moltres

$$\frac{1}{v_g}\frac{\partial \phi_g}{\partial t} - \nabla \cdot D_g \nabla \phi_g + \Sigma_g^r \phi_g = \tag{21}$$

$$\sum_{g \neq g'}^{G} \Sigma_{g' \to g}^s \phi_{g'} + \chi_g^p \sum_{g'=1}^{G} (1-\beta)\nu \Sigma_{g'}^f \phi_{g'} + \chi_g^d \sum_{i}^{I} \lambda_i C_i \tag{22}$$

v_g = speed of neutrons in group g

ϕ_g = flux of neutrons in group g

t = time

D_g = Diffusion coefficient for neutrons in group g

Σ_g^r = macroscopic cross-section for removal of neutrons from group g

$\Sigma_{g' \to g}^s$ = macroscopic cross-section of scattering from g' to g

χ_g^p = prompt fission spectrum, neutrons in group g

G = number of discrete groups, g

ν = neutrons produced per fission

Σ_g^f = macroscopic fission cross section due to neutrons in group g

χ_g^d = delayed neutrons in group g

I = delayed neutron precursor groups

β = delayed neutron fraction

λ_i = average decay constant of delayed neutron precursors in group i

C_i = concentration of delayed neutron precursors in precursor group i

.

Moltres Delayed Neutrons

$$\frac{\partial C_i}{\partial t} = \sum_{g'=1}^{G} \beta_i \nu \Sigma_{g'}^{f} \phi_{g'} - \lambda_i C_i - \frac{\partial}{\partial z} u C_i \tag{23}$$

$G =$ number of discrete groups, g

$I =$ delayed neutron precursor groups

$C_i =$ concentration of delayed neutron precursors in precursor group i

$.u =$ vertical fluid velocity

$\lambda_i =$ average decay constant of delayed neutron precursors in group i

$\beta =$ fraction of delayed neutron precursors in group i

Moltres Fuel Temperature

$$\rho_f c_{p,f} \frac{\partial T_f}{\partial t} + \nabla \cdot (\rho_f c_{p,f} \vec{u} \cdot T_f - k_f \nabla T_f) = Q_f \tag{24}$$

$$\rho_f = \text{density of fuel salt} \tag{25}$$
$$c_{p,f} = \text{specific heat capacity of fuel salt} \tag{26}$$
$$T_f = \text{temperature of fuel salt} \tag{27}$$
$$\vec{u} = \text{velocity of fuel salt} \tag{28}$$
$$k_f = \text{thermal conductivity of fuel salt} \tag{29}$$
$$Q_f = \text{source term} = \sum_{g=1}^{G} \epsilon_{f,g} \Sigma_{f,g} \phi_g \tag{30}$$

Moltres Moderator Temperature

$$\rho_g c_{p,g} \frac{\partial T_g}{\partial t} + \nabla \cdot (-k_g \nabla T_g) = Q_g \tag{31}$$

$$\tag{32}$$

$$\rho_g = \text{density of graphite moderator} \tag{33}$$

$$c_{p,g} = \text{specific heat capacity of graphite moderator} \tag{34}$$

$$T_g = \text{temperature of graphite moderator} \tag{35}$$

$$k_g = \text{thermal conductivity of graphite moderator} \tag{36}$$

$$Q_g = \text{source term in graphite moderator} \tag{37}$$

$$\tag{38}$$

Where does the data come from?

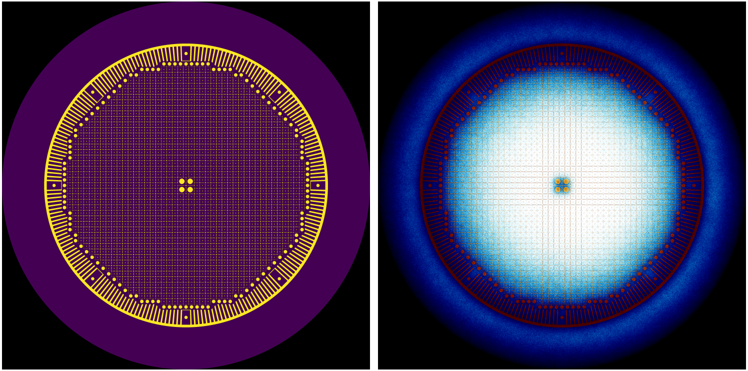
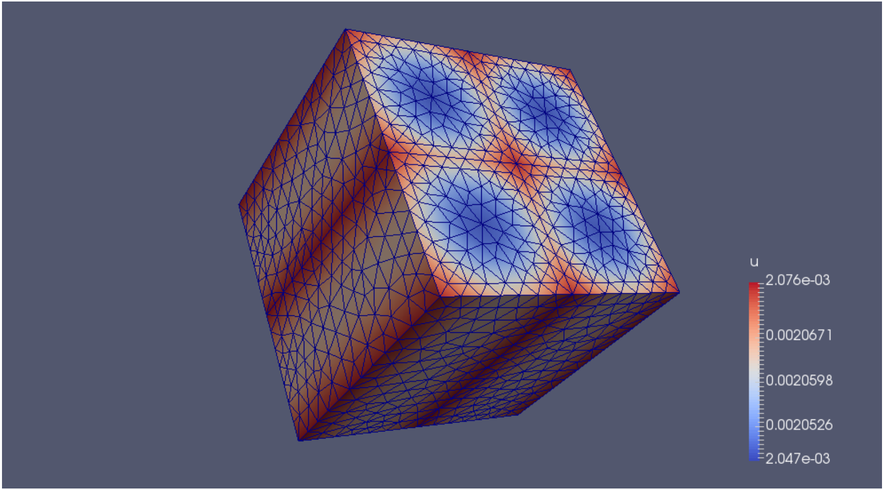
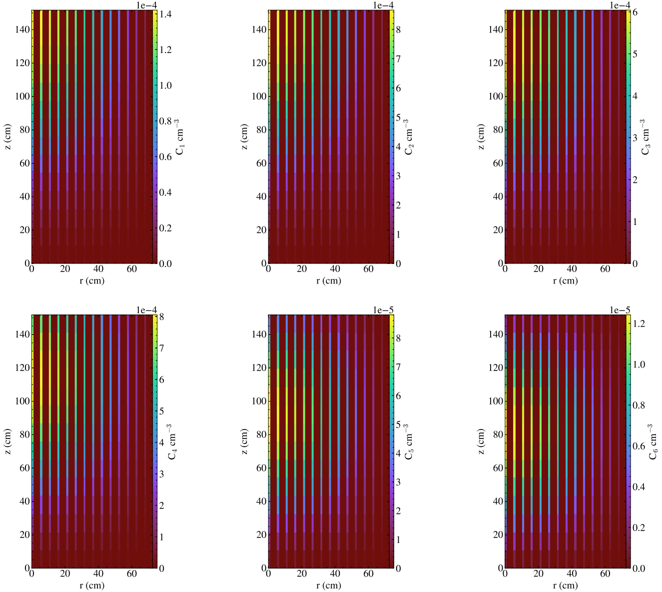


Figure: Above, full MSBR core neutronics simulation in Serpent (Rykhlevskii et al. 2019 [12]). Left: geometry. Right: Monte Carlo Neutron Transport scattering and fission.

Moltres (coupling in MOOSE) (Lindsay et al. 2018 [8])



Moltres Precursor Drift (Lindsay et al. 2018 [8])



Multiphysics simulation results (3D)

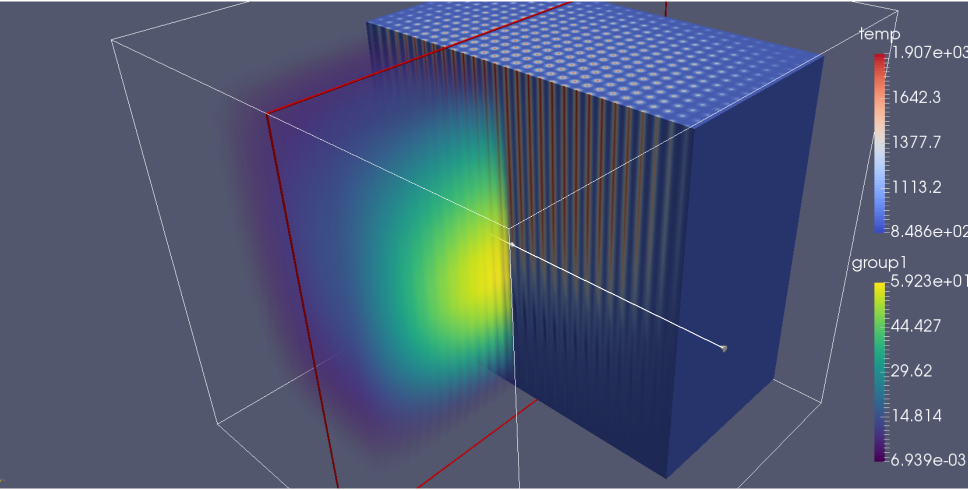


Figure: Cuboidal MSR steady-state temperature and fast neutron flux tests by Gavin Ridley.

Conclusions

Ordinary tools cannot capture kinetics in mobile fuels or long term fuel cycle performance of liquid-fuelled reactors.

Moltres

- New tool **Moltres** was developed for modeling coupled physics in novel molten salt reactors.
- 2D-axisymmetric and 3D multiphysics models are presented.
- **Moltres** demonstrated strong parallel scaling (up to 384 physical cores) but further optimization required.
- Over 55,000 node-hours were consumed on **Blue Waters** to perform this research.

Also: SaltProc

- New tool **SaltProc** was developed to simulate fuel depletion in MSRs.
- **SaltProc** was tested for the MSBR conceptial design, equilibrium fuel salt composition was found and verified against recent studies.

Acknowledgements

- This research is part of the Blue Waters sustained-petascale computing project, which is supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) and the state of Illinois.
- Andrei Rykhlevskii is supported by the Department of Nuclear, Plasma, and Radiological Engineering.
- Kathryn Huff is additionally supported by the NRC Faculty Development Program, the NNSA (awards DE-NA0002576 and DE-NA0002534), and the International Institute for Carbon Neutral Energy Research (WPI-I2CNER).
- The authors would like to thank members of Advanced Reactors and Fuel Cycles research group (ARFC) at the University of Illinois at Urbana Champaign who provided valuable code reviews and proofreading.
- Alex Lindsay (Idaho National Laboratory), Gavin Ridley (University of Tennessee-Knoxville).

References I

[1] C. Andreades, A.T. Cisneros, J.K. Choi, A.Y.K Chong, David L. Krumwiede, Lakshana Huddar, Kathryn D. Huff, M.D. Laufer, Madicken Munk, Raluca O. Scarlat, Jeffrey E. Seifried, Nicolas Zwiebaum, Ehud Greenspan, and Per F. Peterson.
Technical Description of the 'Mark 1' Pebble-Bed, Fluoride-Salt-Cooled, High-Temperature Reactor Power Plant.
Thermal Hydraulics Group UCBTH-14-002, University of California, Berkeley, Department of Nuclear Engineering, Berkeley, CA, September 2014.

[2] DOENE.
Southern Company and TerraPower Prep for Testing on Molten Salt Reactor, August 2018.

[3] Badawy M. Elsheikh.
Safety assessment of molten salt reactors in comparison with light water reactors.
Journal of Radiation Research and Applied Sciences, 6(2):63–70, October 2013.

[4] Charles W. Forsberg and Madeline Feltus.
Fuel requirements for the advanced high-temperature reactor: graphite coated-particle fuel and molten fluoride salt coolant.
Paper Due, 7, January 2004.

References II

[5] D. Gaston, G. Hansen, S. Kadioglu, D. A. Knoll, C. Newman, H. Park, C. Permann, and W. Taitano.
Parallel multiphysics algorithms and software for computational nuclear engineering.
Journal of Physics: Conference Series, 180(1):012012, July 2009.

[6] Cole Gentry, G Maldonado, and Ondrej Chvala.
Burnable Poison Reactivity Control for the Advanced High Temperature Reactor.
November 2015.

[7] Kathryn Huff.
PyRK: Python for Reactor Kinetics, 2015.
https://github.com/pyrk/pyrk.

[8] Alexander Lindsay, Gavin Ridley, Andrei Rykhlevskii, and Kathryn Huff.
Introduction to Moltres: An application for simulation of Molten Salt Reactors.
Annals of Nuclear Energy, 114:530–540, April 2018.

[9] R. C. Robertson.
Conceptual Design Study of a Single-Fluid Molten-Salt Breeder Reactor.
Technical Report ORNL–4541, comp.; Oak Ridge National Lab., Tenn., January 1971.

References III

[10] M. W. Rosenthal, P. R. Kasten, and R. B. Briggs.
Molten-Salt Reactors—History, Status, and Potential.
Nuclear Applications and Technology, 8(2):107–117, February 1970.

[11] Andrei Rykhlevskii, Jin Whan Bae, and Kathryn Huff.
arfc/saltproc: Code for online reprocessing simulation of Molten Salt Reactor with external depletion solver SERPENT, March 2018.

[12] Andrei Rykhlevskii, Jin Whan Bae, and Kathryn D. Huff.
Modeling and simulation of online reprocessing in the thorium-fueled molten salt breeder reactor.
Annals of Nuclear Energy, 128:366–379, June 2019.

[13] Andrei Rykhlevskii, Alexander Lindsay, and Kathryn D. Huff.
Online reprocessing simulation for thorium-fueled molten salt breeder reactor.
In *Transactions of the American Nuclear Society*, Washington, DC, United States, November 2017. American Nuclear Society.

[14] TerraPower LLC.
MCFR TerraPower, December 2018.

References IV

[15] Xin Wang, Kathryn D. Huff, Manuele Aufiero, Per F. Peterson, and Massimiliano Fratoni.
Coupled Reactor Kinetics and Heat Transfer Model for Fluoride Salt-Cooled High-Temperature Reactor Transient Analysis.
June 2016.

Online reprocessing method

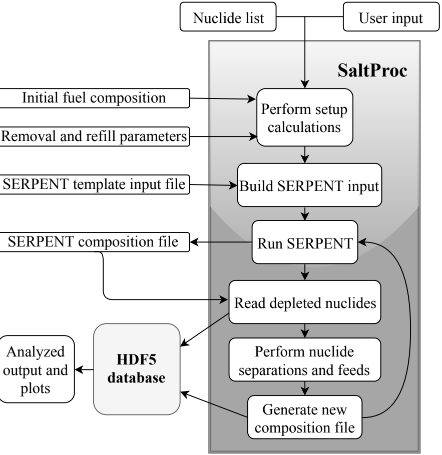


Figure: Flow chart for the SaltProc.

SaltProc capabilities

- Remove specific isotopes from the core with specific parameters (reprocessing interval, mass rate, removal efficiency)
- Add specific isotopes into the core
- Maintain constant number density of specific isotope in the core
- Store stream vectors in an HDF5 database for further analysis or plots
- Generic geometry: an infinite medium, a unit cell, a multi-zone simplified assembly, or a full-core

Online reprocessing method

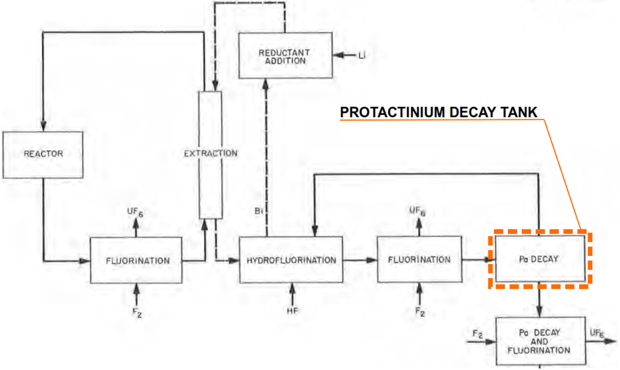


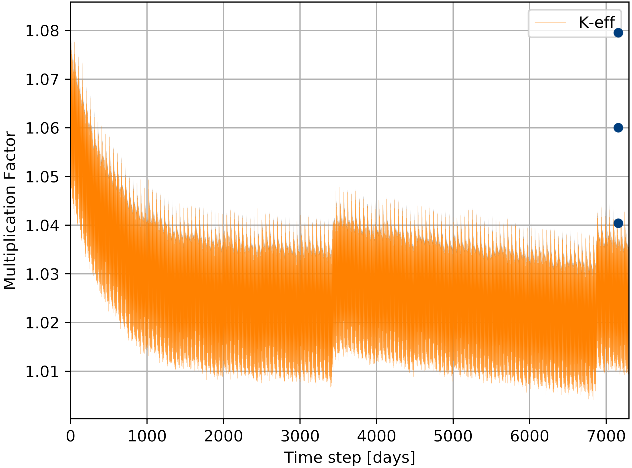
Figure: Protactinium isolation with uranium removal by fluorination [9].

Online reprocessing approach

- Continuously removes all poisons, noble metals, and gases.
- ^{233}Pa is continuously removed from the fuel salt into a decay tank.

$$^{232}_{90}\mathrm{Th} + ^{1}_{0}\mathrm{n} \rightarrow ^{233}_{90}\mathrm{Th} \xrightarrow[22.3\text{ min}]{\beta^-} ^{233}_{91}\mathrm{Pa} \xrightarrow[26.967\text{ d}]{\beta^-} ^{233}_{92}\mathrm{U}$$

Effective multiplication factor for full-core MSBR model



- Strong absorbers (^{233}Th, ^{234}U) accumulating in the core
- Fissile materials other than ^{233}U are bred into the core (^{235}U, ^{239}Pu)
- The multiplication factor stabilizes after approximately 6 years

Figure: k_{eff} during a 20 years depletion simulation.

Power and breeding distribution

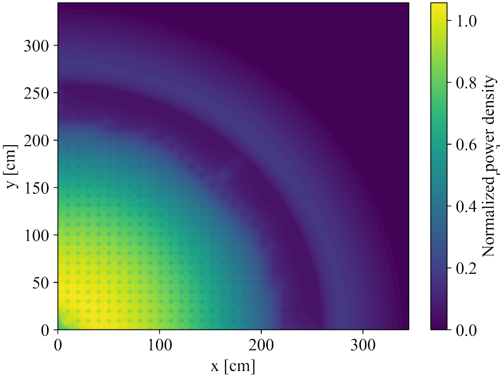


Figure: Normalized power density

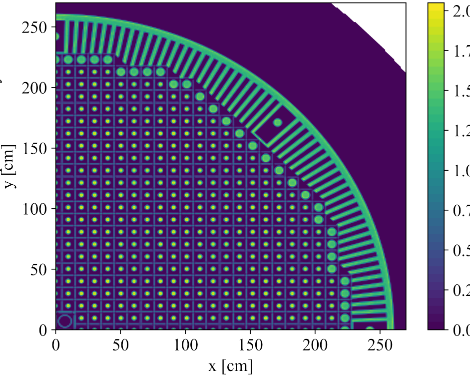


Figure: ^{232}Th neutron capture reaction rate normalized by total flux

^{232}Th refill rate

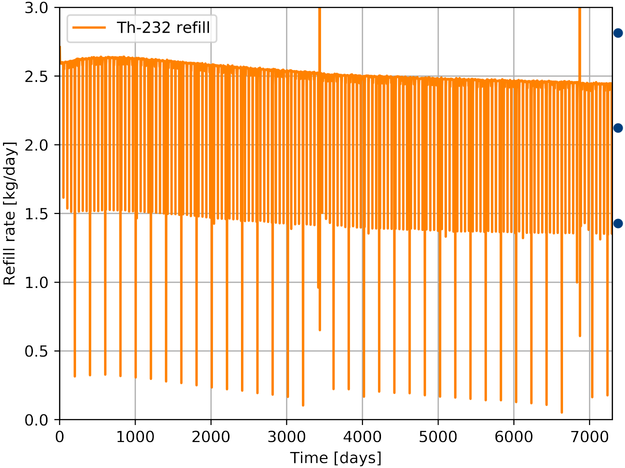
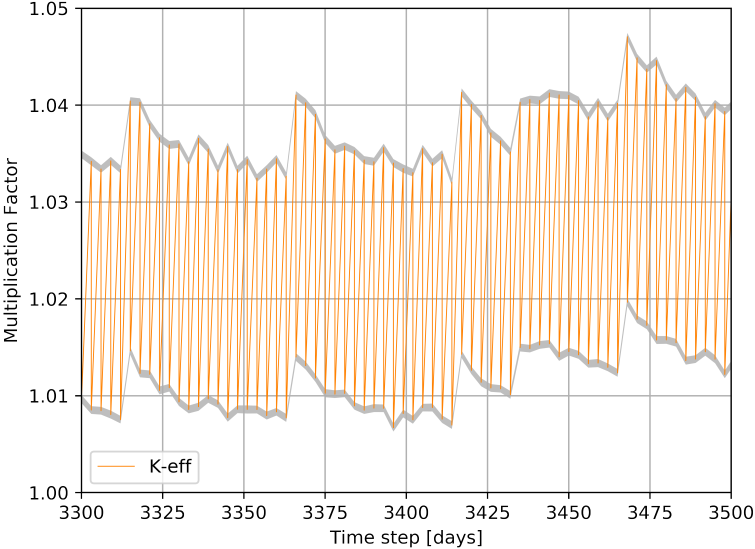


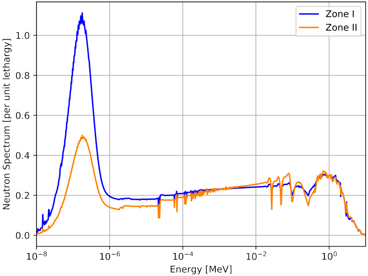
Figure: ^{232}Th feed rate over 20 years of MSBR operation

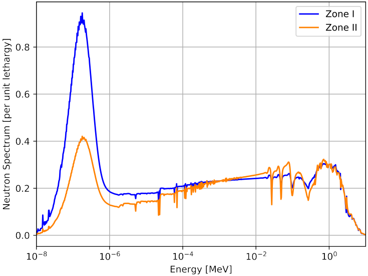
- Fluctuation due to batch-wise removal of strong absorbers
- Feed rate varies due to neutron energy spectrum evolution
- ^{232}Th consumption is 100 g/GWh$_e$

Multiplication factor dynamics during Rb, Sr, Cs, Ba removal (3435days)



MSBR neutron energy spectrum for different regions





Fissile isotopes in the MSBR core