Nuclear Engineering and Design 449 (2026) 114790

Contents lists available at ScienceDirect

Nuclear Engineering and Design

journal homepage: www.elsevier.com/locate/nucengdes

Multi-Physics Benchmark for a Thermal Molten Salt Reactor

P. Pfahl [a,*], B. Kędzierska [b], I. Kim [d], A. Chambon [a], L. Fischer [c], L. Bureš [c], J. Groth-Jensen [c], Y. Kim [d], A. Rineiski [b], B. Lauritzen [a]

[a] Technical University of Denmark (DTU), Center for Nuclear Energy Technology, Frederiksborgvej 399, 4000 Roskilde, Denmark
[b] Karlsruhe Institute of Technology (KIT), Institute for Neutron Physics and Reactor Technology, Hermann-von-Helmholtz-Platz 1, 76344 Eggenstein-Leopoldshafen, Germany
[c] Saltfoss Energy (former Seaborg Technologies), Titangade 11, 2200 Copenhagen, Denmark
[d] Korea Advanced Institute of Science and Technology, 291 Daehak-ro, Daejeon, Republic of Korea

ARTICLE INFO

Dataset link: https://github.com/philipJFpfahl/Thermal_Molten_Salt_Reactor_Benchmark

Keywords:
Benchmark code-to-code molten salt reactor transient

ABSTRACT

Verification of nuclear codes is an important step in licensing nuclear reactors. For molten salt reactors, the involved physics phenomena are strongly coupled and include those introduced by the movement of liquid fuel that are not present at nominal conditions in solid fuel reactors. This movement of fuel inside and outside the core poses new simulation challenges. In this paper, a benchmark for a graphite-moderated molten salt reactor with a simplified out-of-core model is proposed and studied. The benchmark addresses both neutronics and thermal-hydraulics phenomena, including the delayed neutron precursor drift inside and outside of the active core region, as well as the temperature feedback. As for the thermal-hydraulics, a laminar flow field with conjugate heat transfer, delayed neutron precursor movement, and a simplified heat exchanger is modeled. The benchmark is investigated with the MOOSE tools Griffin and Squirrel, coupled with the MOOSE internal thermal-hydraulics abilities, the Monte Carlo code iMC coupled with OpenFOAM, Nek5000 with a custom point kinetics solver, the coupled neutronics and fluid dynamics code SIMMER with capabilities for severe accident simulations, and the Modelica-based library TRANSFORM. By employing a variety of high- and low-fidelity modeling approaches, a robust comparison across different codes is ensured. OpenMC and Serpent are employed as reference codes to verify the correct implementation of the neutronics. This paper provides a comprehensive comparison of the strengths and weaknesses of the codes and their underlying modeling assumptions. It examines how modeling assumptions affect the steady-state solution and how they propagate into the transient analysis.

1. Introduction

The interest in building new Molten Salt Reactors (MSRs) (LeBlanc, 2010) is rapidly growing and has created a demand for validated codes to assess their performance, transient behavior, and potential accidents (Diamond et al., 2018). The Molten Salt Reactor Experiment (MSRE) (Haubenreich and Engel, 1970) is currently the only source of full experimental validation. Even though the MSRE provided valuable data (Haubenreich et al., 1964) during its operational period, the data is often noisy, and the available transient information is limited (Steffy, 1970). Additionally, the MSRE is challenging to model due to the high level of detail needed. This additional barrier makes code-to-code comparison difficult. Therefore, simplified MSRE-based benchmarks have been proposed (Delpech et al., 2003a).

In previous studies, the MSRE has been used to validate simulation data against experimental data. At zero power, the reactor did not produce any heat, and the temperature feedback could be neglected. The critical eigenvalue problem with stationary fuel has been benchmarked by Shen et al. (2021) and Clarno et al. (2023). A pure investigation of the Delayed Neutron Precursor (DNP) drift caused by the flowing fuel is presented in Shen et al. (2018). At full power, the temperature feedback has to be considered. Delpech et al. (2003b) investigated the pump-start, coast down, and the natural circulation.

Lots of efforts to investigate the transient behavior have been made using different codes to investigate the MSRE (Singh et al., 2018; Fei et al., 2024; Ponce Tovar, 2021). During the SAMOFAR (2025) and SAMOSAFER (2025) projects, the pool-type Molten Salt Fast Reactor has been developed (Brovchenko et al., 2013). Within these projects, the neutronics (Fiorina et al., 2012), the thermal-hydraulics (Yamaji et al., 2014), and their coupled multi-physics behavior (Vieira et al., 2024) are investigated. This reactor is particularly sensitive to the fluid dynamics modeling assumptions (Laureau et al., 2022), making it difficult to compare the neutronics behavior in a multi-physics

* Corresponding author.
E-mail address: philip.j.f.pfahl@gmail.com (P. Pfahl).

https://doi.org/10.1016/j.nucengdes.2026.114790
Received 26 April 2025; Received in revised form 15 December 2025; Accepted 15 January 2026
Available online 28 January 2026

context. Additionally, the transient behavior of the MSFR has been studied (Fiorina et al., 2014; Laureau et al., 2017).

Simulation benchmarks from the past and the one studied in the following offer an opportunity to investigate the behavior and individual physics phenomena present in MSRs in a simplified design. Tiberga et al. (2020) benchmarked different deterministic neutronics codes for a molten salt fast reactor in a pool-type configuration. The fuel was heated in a 2×2 m^2 2D cavity and circulated by a moving lid. The entire reactor was homogeneous. The heat transfer coefficient was constant throughout the core. This benchmark, which has been repeated several times (Abou-Jaoude et al., 2021; Pfahl et al., 2025; Groth-Jensen et al., 2021; Kędzierska et al., 2025), enables a detailed comparison between codes used for fast-spectrum reactors. This allows us to investigate the ability of the code to evaluate the different physics present in an MSR. However, this benchmark lacks an out-of-core region, so it cannot evaluate possible recirculation-dependent and time-delayed phenomena occurring with an out-of-core model. Additionally, it neglects the interaction between flowing fuel and a solid moderator or any internal structure.

The reactor studied in this work is a graphite-moderated thermal MSR with Fluoride–Uranium–Sodium–Potassium (FUNaK) salt as its fuel. A single fuel channel is modeled, reducing the burden on fluid dynamics and thermal-hydraulics modeling while still capturing the relevant physics, namely the temperature feedback and the DNP feedback when moving inside, leaving, and reentering the core. The out-of-core model is as simple as possible, accounting for the external heat removal and the DNP decay. Overall, the simplicity should reduce the modeling requirements, allowing more codes and developers to participate without a significant computational and development overhead. Since most neutronics codes are not designed for a tight coupling with fluid dynamics, the flow is assumed to be laminar even at high velocities.

The benchmark includes several steps increasing in complexity, combining the neutronics, the thermal-hydraulics, and the out-of-core model in a progressive manner. The benchmark begins with a steady-state single-physics investigation of neutronics and thermal-hydraulics, followed by a steady-state multi-physics evaluation with changing velocities and temperature profiles. The complexity increases to a transient multi-physics assessment of an Unprotected Loss Of Flow (ULOF).

In step 1.1, the neutronics is evaluated by calculating the reactivity of the reactor and reactivity feedback coefficients. Additionally, the heating of the fuel and moderator is calculated. In step 1.2, the thermal-hydraulics is tested, using the heating from the previous step to calculate the temperature in the fuel channel and the graphite. For the multi-physics simulations, the neutronics and thermal-hydraulics are initially loosely coupled. More complexity is added, including a tighter coupling and an out-of-core model. In step 2.1, the effect of a changing temperature field on reactivity is examined. In step 2.2, the impact of DNP advection on reactivity is tested for different out-of-core times. Step 2.3 assesses the effect a changed velocity has on the DNP and temperature reactivity feedback. Finally, in step 2.4, we investigate the steady-state critical reactor. In step 3.1, a ULOF is simulated considering the tight coupling tested in step 2.3.

This benchmark allows the comparison of neutronics codes coupled with thermal-hydraulics solvers at different fidelity levels. The modeling approaches represent a range from high to low fidelity. Combining these different approaches within this benchmark was done to improve the understanding of the implications each approach has when modeling a thermal MSR. The benchmark is intentionally similar to the CNRS benchmark by Tiberga et al. (2020) in structure and scope, while focusing on a thermal reactor and a wider range of codes. These approaches include the high-fidelity Monte Carlo code iMC (Kim and Kim, 2021) coupled with the high-fidelity thermal-hydraulics code OpenFOAM (Weller et al., 1998), the severe accident code SIMMER (Yamano et al., 2003), the coarse-mesh system code TRANSFORM (Greenwood, 2017), and the MOOSE Navier–Stokes module (Lindsay et al., 2023) coupled to two neutronics codes: a point-kinetics solver Squirrel (Pfahl et al., 2025), and the deterministic diffusion code Griffin (Wang et al., 2025). The spectral fluid dynamics code Nek5000 (Fischer et al., 2007) is used as a high-fidelity reference code for single physics thermal-hydraulics; furthermore, using a custom point kinetics solver, Nek5000 is used to interrogate the multi-physics steps of the benchmark as well. For the eigenvalue and heating calculations in the reactor Serpent and OpenMC is used (Leppänen et al., 2025; Romano et al., 2015a).

In Section 2, the benchmark is described in detail. The geometry is described, and the boundary conditions are introduced. The thermophysical parameters are defined, including the chemical composition, heat capacity, density, viscosity, and thermal conductivity of the salt and the moderator. To allow neutronics results to be directly compared, the kinetics parameters and the cross-sections are provided. Reference heating and flux profiles are also provided.

In Section 3, the different codes are presented. A short description of the involved equations and the code capabilities is given. Furthermore, the utilization of the particular code in the benchmark is explained.

The Section 4 describes the individual steps of the benchmark. The description of the assumed parameters and the measured data points follows the general description of the steps.

In Section 5, the results of the individual steps are shown. The simulation results of each code are shown, compared to the other codes, and the underlying modeling assumptions are discussed.

In Section 6, conclusions and ideas for future work are provided.

2. Description of the reactor parameters

The main goal of the benchmark is to enable different participants to investigate their modeling assumptions. Therefore, the geometry consists of a fuel channel and a moderator block. This simple geometry enables the results of 1D, 2D, and 3D codes to be compared. The out-of-core region is also modeled. It allows for the reentering of the fuel with a time delay. The heat and the DNPs are advected with the fuel. In the out-of-core region, heat is removed, and the DNPs decay. The modeling details of the out-of-core region are left to the participants due to code-specific restrictions.

The thermophysical model of the reactor is intended to be accessible to low- and high-fidelity codes without neglecting the relevant physics phenomena involved. This enables the validation of coarser models against higher-fidelity models.

2.1. Geometry description

The reactor layout is illustrated in Fig. 1. Note that the illustration is not to scale. It features a slab reactor with a fuel channel on the left side and a moderator on the right side. The moderator is shaded gray, while the fuel channel is colored white. Black arrows indicate the direction of the salt flow. The numbers (**1-6**) mark the boundaries, while **7** indicates the graphite-fuel interface. A summary of the boundary conditions is provided in Table 1. The parameters used for thermophysical modeling are detailed in Section 2.3, and those for neutronics modeling are provided in Section 2.4.

The reactor has a width of 0.23 m and a height of 1.5 m. These dimensions are summarized in Table 2. The total power output is 350 kW. The inlet flow velocity v_{inlet} is uniformly 0.2 ms^{-1}. The out-of-core length is 2 m, resulting in an out-of-core time of 10 s. The out-of-core time changes with the velocity. These parameters are detailed in Table 3. The chemical composition of the fuel salt is provided in Table 4, and the resulting atomic fractions are provided in Table 7.

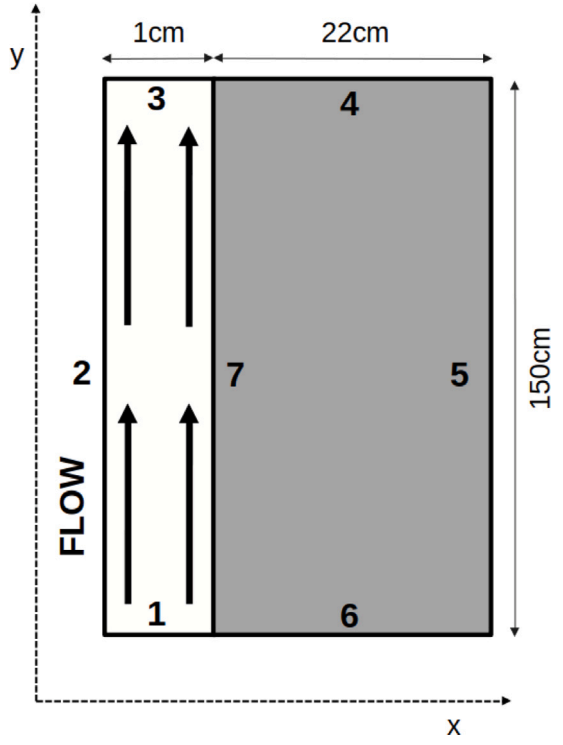

Fig. 1. Layout of the reactor. Note: **Not to scale**.

Table 1
Boundaries and Interfaces.

Number	Thermal hydraulics	Neutronics
1	Inlet	Vacuum
2	Symmetric	Reflective
3	Outlet	Vacuum
4	Adiabatic	Vacuum
5	Symmetric	Reflective
6	Adiabatic	Vacuum
7	Conjugate heat transfer/ No slip	Material interface

Table 2
Reactor layout.

Height	Total width	Fuel width	Moderator width
150 cm	23 cm	1 cm	22 cm

Table 3
Model parameters.

Average velocity	Out-of-core time	Inlet temperature	Power
0.2 ms^{-1}	10 s	873.15 K	3.5×10^5 W

Table 4
Chemical composition of the fuel salt.

Component	UF_4	NaF	KF
Molar composition (%)	13.1736	55.6886	31.1377

2.2. Out of core model

The out-of-core model accounts for the recirculation of the fuel, including the transport and the decay of the DNPs and the removal of heat from the fuel salt via a heat exchanger. Outside of the core, perfect radial mixing is assumed, so that the problem collapses to a 1D problem. While the fuel is outside of the core, no fission is assumed to occur, and the decay heat is neglected. The heat exchanger is modeled

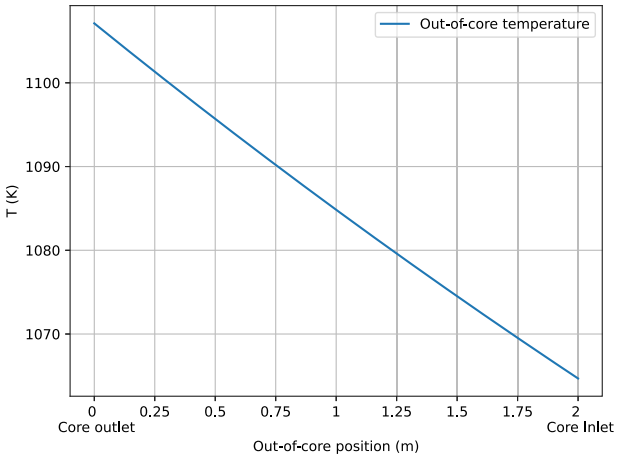

Fig. 2. Salt temperature in the out-of-core region for $T_{out} = 1107.1$ K and $U = 0.2$ ms^{-1}.

by removing heat over the entire length of the out-of-core region, assuming a constant heat transfer coefficient $\gamma = 0.02$ s^{-1} and a constant temperature of the secondary side $T_H = 873.15$ K.

The time- and space-dependent temperature is described by

$$\frac{\partial T(t,l)}{\partial t} = -U(t)\frac{\partial T(t,l)}{\partial l} - \gamma(T(t,l) - T_H), \tag{1}$$

with the one-dimensional out-of-core velocity $U(t)$ and the position l along the out-of-core region. The out-of-core velocity $U(t)$ should not be confused with the inlet boundary condition v_{inlet}. In Fig. 2, the time-independent temperature is shown corresponding to the core outlet temperature of $T = 1107.1$ K and a constant fuel velocity of $U = 0.2$ ms^{-1}.

The time and space-dependent DNP concentration is modeled as

$$\frac{\partial C_I(t,l)}{\partial t} = -U(t)\frac{\partial C_I(t,l)}{\partial l} - \lambda_I C_I, \tag{2}$$

with the decay constant for group I being λ_I.

The temperature and the DNP concentration reentering the core are equal to $T(t,l)$ and $C_I(t,l)$ at $l = 2$ m respectively.

2.3. Fluid dynamics model

The fuel enters the core from the bottom and leaves at the top. The salt flow in the channel is always assumed to be laminar, even for high Reynolds numbers, reducing uncertainties due to different turbulence handling or no turbulence models within the codes. This reduces the heat transfer with the graphite. In the used model, the graphite temperature is therefore higher than for turbulent flow. The physical relevance of the solution is still given, since the graphite has a large thermal mass and relatively little power is deposited in the graphite. For fluid dynamics, the boundary **1** is the inlet, and the boundary **3** is the outlet of the reactor. The fuel flow is symmetric along boundary **2**. At the graphite-fuel interface **7**, a no-slip boundary condition is applied. The DNPs are modeled to reenter perfectly mixed at boundary **1**. At boundary **7** a no-flux boundary condition is applied.

The assumed thermophysical properties for the salt are shown in Table 5 for $T = 873.15$ K. The density of the fuel salt does not change in the fluid dynamics model. The salt density is 4126.85 kg m^{-3}. Buoyancy effects are neglected, and the density is assumed to be constant for the thermophysical modeling. This allows for the benchmark to be accessible for a wider variety of codes, ensuring a good agreement in the fluid dynamics simulations. Note: the density does change in the neutronics modeling.

The moderator is comprised of pure graphite. The assumed thermophysical properties of graphite are shown in Table 6. The graphite is

Table 5
Thermophysical properties of fuel salt.

Density (kg m^{-3})	Dynamic viscosity (kg m^{-1} s^{-1})	Specific heat (J kg^{-1} K^{-1})	Thermal conductivity (W m^{-1} K^{-1})
4126.85	17×10^{-3}	1000	1

Table 6
Thermophysical properties of graphite.

Density (kg m^{-3})	Dynamic viscosity (kg m^{-1} s^{-1})	Specific heat (J kg^{-1} K^{-1})	Thermal conductivity (W m^{-1} K^{-1})
1800	Solid	1760	60

Table 7
Isotopic fuel composition in atomic fractions.

Isotope	Atomic fraction (%)	Atomic density (atoms/barn.cm)
^{23}Na	23.25	1.67099×10^{-2}
^{39}K	12.123	8.71331×10^{-3}
^{40}K	0.0015	1.09316×10^{-6}
^{41}K	0.8749	6.28818×10^{-4}
^{234}U	0.0024	1.75228×10^{-6}
^{235}U	0.2727	1.96045×10^{-4}
^{238}U	5.2235	3.75420×10^{-3}
^{236}U	0.0012	8.97981×10^{-7}
^{19}F	58.25	4.18648×10^{-2}

only cooled through the fuel-graphite interface **7**. The boundaries **4**, **5**, and **6** are adiabatic, so there is no heat flux out of the reactor through the graphite.

2.4. Neutronics model

The used nuclear data library is ENDF/B-VII.1 (Chadwick et al., 2011). All the elements are in natural isotopic composition, except for uranium, which is enriched to 4.9%wt. The isotopic fuel composition is given in Table 7.

The moderator, made out of pure graphite, is assumed to have a natural carbon composition and is modeled under the consideration of the S(α, β) scattering.

It is assumed that the reactor is infinite in the x and z directions. Boundaries **2** and **5** have reflective boundary conditions. Neutrons leave the top and bottom of the reactor unhindered and do not interact with the outer core. Boundaries **1**, **3**, **4**, and **6** have vacuum boundary conditions, and the outer core has no contribution.

For the neutronics modeling, fuel salt density is assumed to vary with temperature so as to capture temperature effects on reaction cross-sections due to both Doppler and density effects. The temperature-dependent salt density for the neutronics is $5000 - 1 \cdot T(\mathrm{K})$ (kg m^{-3}).

Since some codes involved in the benchmark are multi-group deterministic codes, the reference macroscopic cross-sections for the salt are shown in Table 8 and for the graphite in Table 9. Due to the very soft spectrum of the system shown in Fig. 3 only two energy groups are used, with an energy boundary at 6.19 eV. The values between the temperature data points are obtained using linear interpolation. The neutron diffusion coefficient is equal to $1/(3\Sigma_{tr})$. The cross-sections were generated using OpenMC for the geometry presented in Section 2.1, at 873.15 K and 1373.15 K. Anisotropy of scattering is taken into account during the input preparation in the transport cross-sections. The transport approximation was employed, i.e., the transport cross-sections are computed by taking into account the mean cosine of scattering angle, so that

$$\Sigma_{tr} = \Sigma_t - \mu\Sigma_s. \tag{3}$$

The kinetics parameters with 6 delayed neutron groups used in the benchmark are shown in Table 10. The use of these values ensures consistency, since different methods of calculating the kinetics parameters are available. The parameters have been computed using OpenMC. Note that the delayed neutron fraction values are not adjoint-weighted (not $\beta_{effective}$). The total delayed neutron fraction is $\beta = 0.006505$ and the prompt neutron generation time $\Lambda = 0.00024$ s. Additionally, the mean number of neutrons per fission is $\nu = 2.4$, and the total released energy per fission $Q = 202.65$ MeV. The neutron inverse velocity is 4.1588×10^{-6} ms^{-1} for group 1 and 1.64813×10^{-4} ms^{-1} for group 2.

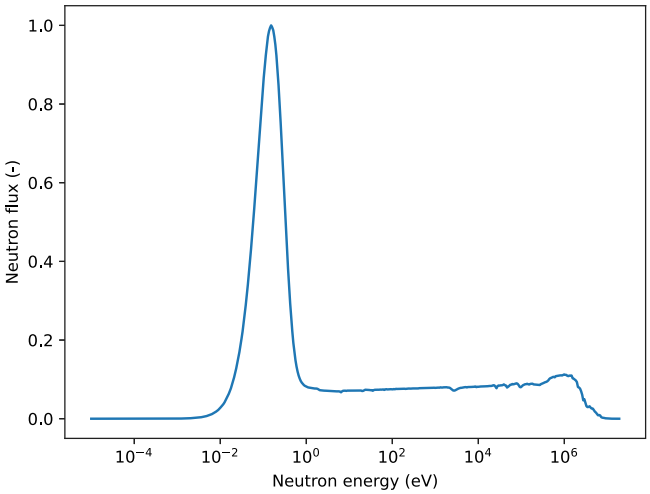

Fig. 3. Neutron energy spectrum obtained with OpenMC, normalized for the maximum neutron flux to be 1.

Since the flux and the heating profiles are essential for most of the steps of the benchmark, the authors give a reference profile in the form of a fit function below. The heating $H(\mathbf{r})$, with $\mathbf{r} = (x, y)$ being the 2D spatial vector, is described by a sine function in the y direction; in the x direction, an exponential increase in the fuel and a decrease in the moderator is used. The heating function is written as

$$H(\mathbf{r}) = A\left(\Omega_y + \sin\left(\frac{\pi y}{h}\right)\right)\left(\Omega_x + \exp(\delta x)\right), \tag{4}$$

with the variables defined in Table 11.

The flux $\phi(\mathbf{r})$ is described by a sine function in the y direction and a second-order polynomial in the x direction. The fit function is written as

$$\phi(\mathbf{r}) = A\left(\Omega_y + \sin\left(\frac{\pi y}{h}\right)\right)\left(1 + ax + bx^2\right), \tag{5}$$

with the variables defined in Table 12.

3. Code description

In this section, the different codes are described. Not all codes can solve all steps, and these are coupled to other codes, so that it is specified which codes solve each step (see also Table 14). Additionally, the coupling between different codes is also explained. The involved codes use either a fine or coarse discretization for the neutronics and the thermal hydraulics. In Table 13, the fidelity level of the coupled codes is shown. There are six multi-physics codes or coupled codes involved in all steps.

The first group of codes contains SIMMER and TRANSFORM. SIMMER is a coupled code developed and applied in Japan, Germany, France, and other EU countries. It is primarily applied to the liquid-metal (sodium and lead) fast reactor severe accident modeling, but has been extended to MSR, including thermal-spectrum applications (Yamano et al., 2003). The TRANsient Simulation Framework Of Reconfigurable Modules (TRANSFORM) (Greenwood, 2017), developed by Oak Ridge National Laboratory (ORNL) using the Modelica programming

Table 8
Macroscopic cross-sections for fuel salt (cm^{-1}).

TempL. salt (K)	Temp.mod. (K)	Group 1: 6.19 eV–20 MeV					
		$\Sigma_t \times 10^{-1}$	$\Sigma_{tr} \times 10^{-1}$	$\Sigma_a \times 10^{-3}$	$\Sigma_f \times 10^{-3}$	$\Sigma_{vf} \times 10^{-3}$	$\Sigma_{1->2} \times 10^{-4}$
873.15	873.15	3.42405	3.11192	8.22290	1.77462	4.44516	3.61515
873.15	1373.15	3.42404	3.11206	8.24322	1.77774	4.45277	3.63746
1373.15	873.15	3.02736	2.75006	7.68625	1.57007	3.93486	3.17927
1373.15	1373.15	3.02750	2.75019	7.70570	1.57229	3.94026	3.19884
Temp. salt (K)	Temp.mod. (K)	Group 2: 0 eV–6.19 eV					
		$\Sigma_t \times 10^{-1}$	$\Sigma_{tr} \times 10^{-1}$	$\Sigma_a \times 10^{-2}$	$\Sigma_f \times 10^{-2}$	$\Sigma_{vf} \times 10^{-2}$	$\Sigma_{2->1}$
873.15	873.15	3.34718	3.27519	6.70748	4.25143	10.3595	0
873.15	1373.15	3.22774	3.15538	5.64244	3.54751	8.64422	0
1373.15	873.15	2.96064	2.89788	5.89589	3.73883	9.11041	0
1373.15	1373.15	2.84975	2.78634	4.95437	3.11645	7.59385	0

Table 9
Macroscopic cross-sections for graphite (cm^{-1}).

Temp. salt (K)	Temp.mod. (K)	Group 1: 6.19 eV–20 MeV					
		$\Sigma_t \times 10^{-1}$	$\Sigma_{tr} \times 10^{-1}$	$\Sigma_a \times 10^{-6}$	Σ_f	Σ_{vf}	$\Sigma_{1->2} \times 10^{-3}$
873.15	873.15	3.80970	3.54721	9.57536	0	0	4.46236
873.15	1373.15	3.81023	3.54780	9.59858	0	0	4.48199
1373.15	873.15	3.80500	3.54254	9.67557	0	0	4.45157
1373.15	1373.15	3.80552	3.54314	9.69148	0	0	4.47138
Temp. salt (K)	Temp. mod. (K)	Group 2: 0 eV–6.19 eV					
		$\Sigma_t \times 10^{-1}$	$\Sigma_{tr} \times 10^{-1}$	$\Sigma_a \times 10^{-4}$	Σ_f	Σ_{vf}	$\Sigma_{2->1}$
873.15	873.15	4.38617	4.28512	1.62219	0	0	0
873.15	1373.15	4.39796	4.25572	1.32941	0	0	0
1373.15	873.15	4.38673	4.28604	1.62882	0	0	0
1373.15	1373.15	4.39850	4.25633	1.33398	0	0	0

Table 10
Delayed Neutron parameters.

	1	2	3	4	5	6
β_I (–)	0.00022773	0.00117549	0.00112222	0.00251611	0.00103157	0.00043212
λ_I (s^{-1}	0.013336	0.032739	0.12078	0.30278	0.84949	2.853

Table 11
Fit parameters of the heating function defined in Eq. (4).

	Moderator heating	Fuel heating
A ($W\ m^{-3}$)	3.192828×10^{5}	1.092355×10^{5}
Ω_x (–)	1.703478×10^{-1}	2.818206×10^{2}
Ω_y (–)	6.001240×10^{-2}	4.450372×10^{-2}
δ (m^{-1})	-1.907400×10^{1}	3.285263×10^{2}
h (m)	1.5	1.5

Table 12
Fit parameters of the flux function defined in Eq. (5).

	Flux in the moderator	Flux in the fuel
A (s^{-1})	4.26519742	9.69975810×10^{1}
Ω_y (–)	4.925683×10^{-2}	5.068259×10^{-2}
a (m^{-1})	5.206809×10^{-1}	0
b (m^{-2})	−1.444232	0
h (m)	1.5	1.5

Table 13
Fidelity level of the coupled codes.

	Thermal-Hydraulics	Neutronics
SIMMER	Coarse	Deterministic
TRANSFORM	Coarse	Point kinetics
Squirrel and MOOSE-NS	Fine	Point kinetics
Griffin and MOOSE-NS	Fine	Diffusion
iMC and OpenFOAM	Fine	Monte Carlo
Nek5000	Fine	Point kinetics

language, is an open-source component library for simulating dynamic thermal-hydraulic and multi-physics systems.

The codes in the second group are based on the Multi-physics Object Oriented Simulation Environment (MOOSE) framework (Giudicelli et al., 2024). The MOOSE framework combines codes dedicated to analyzing advanced reactors. It is currently developed at Idaho National Laboratory and combines finite element and finite volume solvers. It utilizes libmesh (Kirk et al., 2006) for the mesh generation and translation, and the Portable, Extensible Toolkit for Scientific computation (PETSc) (Balay et al., 2001) for numerically solving time-dependent partial differential equations. The necessary multi-physics capabilities are achieved by allowing for a data transfer between different sub-apps. For this benchmark, the neutronics codes Griffin (Wang et al., 2025) and Squirrel (Pfahl et al., 2025) are used. These codes are coupled to the thermal-hydraulics (Hansel et al., 2024) solver in MOOSE via the multi-app system (Gaston et al., 2015). For the initial Monte Carlo simulation, OpenMC (Romano et al., 2015b) and the MOOSE app Cardinal (Novak et al., 2022) are used.

The third group contains the iMC code and Nek5000. iMC is a Monte Carlo neutron transport code developed at the Korean Advanced Institute of Science and Technology (KAIST). The code provides a highly accurate solution by utilizing a continuous-energy cross-section for both neutrons and photons. It is combined with the open-source CFD solver OpenFOAM by transferring a heat distribution and obtaining the corresponding temperature and fuel flow profile. Note that neutronics results from the iMC code are statistically obtained, which are followed by uncertainty. For further analysis, 1-sigma uncertainties are presented in the tables and the graphs. Nek5000 is a spectral finite element thermal-hydraulics code currently developed at the Argonne National

Laboratory (Fischer et al., 2007). Using a custom point kinetics solver, Nek5000 is used to calculate the multi-physics steps of the benchmark.

The last group consists of the two Monte Carlo codes for neutron and photon transport. The open-source codes OpenMC and Serpent, developed at VTT (Technical Research Centre of Finland) (Romano et al., 2015a; Leppänen et al., 2025). Both codes are used as references for the neutron flux, the eigenvalue, and the graphite heating. Additionally, OpenMC is the code used for the cross-section generation.

3.1. SIMMER

SIMMER (Sn Implicit Multi-field Multi-component Eulerian Recriticality) is a severe accident code primarily used for liquid metal fast reactor application (Yamano et al., 2003). The SIMMER-III version is owned by the JAEA and co-developed by the Karlsruhe Institute of Technology (KIT) and other EU partners for two-dimensional safety analyses. It couples three modules: fluid flow and heat transfer module, structure model, and neutron transport solver. The fluid flow and heat transfer part employs a macroscopic approach to the flow and heat transfer problems, originating from the AFDM code developed at Los Alamos National Laboratory (Bohl et al., 1990). The module can treat various materials, including molten salts, with a phase transition between solid, liquid, and gas. Each material is represented by a full or simplified analytic equation-of-state model (Morita and Fischer, 1996). The heat transfer in the structure has been improved by a multi-node can-wall model, employed to include the radial heat conduction and obtain a 2D temperature field in the structure.

The convective heat transfer at the solid/liquid interface is determined by a correlation. The selected correlation, originally proposed for the laminar Graetz problem (Graetz, 1882) in the context of convective heat transfer, is based on work in Bennett (2019). The correlation is formulated in terms of the Graetz number Gz, which is defined as follows:

$$Gz = RePr\frac{D_h}{L}, \tag{6}$$

where Re is the Reynolds number, Pr the Prandtl number, D_h the hydraulic diameter, which equals twice the gap width for a parallel plates duct, and L represents the length of the channel. An average Nusselt number Nu_L over the entire channel length is found to have the correlation:

$$Nu_L = \left(12.74Gz^{1.235} + 2243\right)^{0.2698} + 0.215, \tag{7}$$

and finally, the local Nusselt number Nu_x correlation is found to be:

$$\frac{Nu_{L=x}}{Nu_x} = 1 + \frac{\left(Gz_{L=x}^{1.235}/3\right) / \left(Gz_{L=x}^{1.235} + 176\right)}{1 + 0.1082/\left(Gz_{L=x}^{1.235} + 176\right)^{0.2698}}, \tag{8}$$

This correlation takes into account the growth of thermal boundary layers into a prescribed velocity distribution over the entry region of a heated duct. Using the local Nusselt number and the thermal conductivity k, the local heat transfer coefficient h_x can be determined as follows:

$$h_x = k \cdot Nu_x / D_h. \tag{9}$$

The full equation-of-state model is based on the MOSART salt, with all the coefficients fit so that the fluid thermophysical properties are as in 5. The dynamic viscosity and thermal conductivity were adjusted directly, while the density was approximated by a function with an error below 0.1% for the range between 873.15 and 1373.15 K. The heat capacity is not available as an equation-of-state parameter but other coefficients were tuned to set the internal energy temperature derivative.

The neutronics module of SIMMER calculates time- and space-dependent reactivity and power by an improved quasi-static kinetics model, and updates the neutron flux shape by Sn method as implemented in the TWODANT code (Buckel et al., 1999). Meshes used for the of the fluid flow and neutronics part are partly independent, with neutronics mesh obtained by subdivision of the fluid flow cells. SIMMER uses CCCC format of cross-section data (O'Dell, 1977) which requires an input of isotope-wise microscopic cross-sections in isotxs and brkoxs files.

A correction set was written in the past for the MSRE benchmark (Delpech et al., 2003a) and was recently updated to read an external file with macroscopic cross-sections for two regions. For computing the power distribution, the macroscopic fission cross-sections for two groups g (Tables 8 and 9) and fractions of energy released in graphite, $fkerma$, are used

$$\Sigma_{salt,f,g}(t) = \Sigma_{salt,f,g}^{OpenMC}(t) \cdot (1 - fkerma), \tag{10}$$

$$\Sigma_{graphite,f,g}(t) = \Sigma_{salt,f,g}^{OpenMC}(t) \cdot fkerma. \tag{11}$$

The heating from the step 1.1 is projected on the fluid dynamics mesh for steps requiring only the fluid dynamics module (1.2 and part of 2.3), resulting in the same rescaled sine shape of axial power and two values for radial power distribution (one for salt and one for graphite). It means that there is a uniform radial heating profile in the graphite. These factors refer to the power and are the same for all the runs.

For SIMMER, a correction set accounting for DNPs movement was developed in the past (Rineiski et al., 2005) and has been recently updated and benchmarked (Kędzierska et al., 2025). The DNPs are produced on the neutronics mesh and are mapped onto the fluid dynamics mesh in the fluid dynamics time steps. They are advected on the fluid dynamics mesh and decay also in the out-of-core region, without producing any power.

The out-of-core model is different for the DNP transport and for the heat exchanger. For the DNPs, a 1D channel on the top of the core is added where the DNPs decay and are returned to the core inlet. This model uses a real out-of-core time resulting from the variable inlet velocity. The heat exchanger modeled with the analytical solution of Eq. (1):

$$T_{in}(t) = T_H + (T_{out}(t) - T_H) \cdot e^{-\gamma\tau(t)}, \tag{12}$$

where τ is out-of-core residence time. Since an analytical solution for the out-of-core time does not exist for the exponential velocity decrease (Diniz et al., 2022), the model was simplified by a solution for a linear velocity drop for time from $t = 0$ to $t = 20$ s. Every fluid particle leaving the core is moved along the 1D channel used for the DNP drift with no temperature change, allowing to know the temperature at which the salt left the core at time $t-\tau$, and the inlet temperature is then calculated by a piecewise solution for a linear velocity decrease (Diniz et al., 2022). It was shown that the inlet temperature approximated by this simplified model and a full model by another code did not differ more than 10 K.

SIMMER was used in all steps from 1.1 to 3.1, with a fluid dynamics mesh of 50 axial cells and 2 radial cells (1 in the salt and 1 in the graphite), with an internal structure multi-node option (20 equidistant nodes) for radial temperature profile in the structure. For neutronics, a mesh of 50 axial and 15 radial cells (5 in the salt and 10 in the graphite) was used.

3.2. TRANSFORM

TRANSFORM,[1] developed at ORNL, is an open-source Modelica-based library to model dynamic thermal-hydraulic and multi-physics systems (Greenwood, 2017). It provides a collection of reusable and adaptable components for the rapid development of models, with a focus on nuclear applications (Greenwood et al., 2020; Rader et al., 2019; Cetiner et al., 2016). The model presented in this study follows a methodology similar to that described in Fischer and Bureš (2024).

[1] https://github.com/ORNL-Modelica/TRANSFORM-Library.

The one-dimensional fluid dynamics equations are solved using the lumped-parameter approach with the finite-volume method and a staggered grid scheme for momentum balances at each node along the z-axis of a duct geometry. For a hydrodynamically fully developed laminar flow and thermally developing flow between parallel plates with a constant wall heat flux, the local Nusselt number correlation in Eq. (8) was implemented in TRANSFORM.

The model uses the heating power $H(\mathbf{r})$ and the neutron flux $\phi(\mathbf{r})$, as defined in Eqs. (4) and (5), but normalized so that their integral is constrained to unity; temperature feedback coefficients are obtained from the OpenMC simulation. A more detailed description of the underlying equations for the modified point kinetics can be found in Fischer and Bureš (2024). In general, the change in reactivity due to the drift of DNPs, assuming self-adjointness, is calculated with the help of $\phi(\mathbf{r})$, which determines the importance of neutrons emitted at a given location. The final governing equation for the time-dependent power $P(t)$ is of the form,

$$\frac{dP(t)}{dt} = \frac{\rho(t) - \beta_{flow} + \rho_{temp}(t)}{\Lambda} P(t) + \frac{Q}{\nu\Lambda} \sum_i \lambda_i \frac{\int \phi(\mathbf{r}) C_i(\mathbf{r},t)\, d\mathbf{r}}{\int H(\mathbf{r})\phi(\mathbf{r})\, d\mathbf{r}}, \tag{13}$$

where $\rho(t)$ refers to the reactivity, ν is the number of neutrons emitted per fission, $C_i(\mathbf{r})$ number of DNPs, β_{flow} represents the effective delayed neutron fraction considering the drift of DNPs and λ_i is the decay constant of group i, Λ is the mean neutron generation time and Q is the usable energy per fission. The temperature feedback $\rho_{temp}(t)$ is calculated as follows

$$\rho_{temp}(t) = \frac{\int H(\mathbf{r})\phi(\mathbf{r})\Delta T(\mathbf{r},t) d\mathbf{r}}{\int H(\mathbf{r})\phi(\mathbf{r}) d\mathbf{r}} \Gamma, \tag{14}$$

where $\Delta T(\mathbf{r},t)$ is the local temperature difference to a given reference temperature and Γ is the global reactivity coefficient of temperature. The temperature feedback of graphite is determined using the heating profile within the material as the shape function. TRANSFORM accounts for the DNP circulation using scalar transport equations in which the diffusion term is neglected. For the one-dimensional approach, the balance equation for the flowing fuel is given by

$$\frac{\partial C_i(\mathbf{r},t)}{\partial t} = \frac{\nu\beta_i}{Q} P(t) \cdot H(\mathbf{r}) - \lambda_i C_i(\mathbf{r},t) + \dot{c}_i^{in}(\mathbf{r},t) - \dot{c}_i^{out}(\mathbf{r},t), \tag{15}$$

where $\dot{c}_i^{in}(\mathbf{r},t)$ is the incoming and $\dot{c}_i^{out}(\mathbf{r},t)$ the outgoing DNP flow rate of group i determined by the fluid mass and momentum balances using an upwind discretization scheme. The out-of-core was modeled as a pipe with a length corresponding to the specified residence time. The fluid flow in the excore pipe model terminates at a boundary condition, while the temperature and precursor concentration at the outlet are recycled as input parameters during transients to simulate a flow circuit. Conversely, at steady state, an analytical expression is applied for the DNP decay in this region to mitigate errors arising from coarse ex-core discretization.

TRANSFORM was used in all steps from 1.2 to 3.1; calculations were performed on a computational mesh consisting of 50 equidistantly spaced control volumes along the y-axis and 26 control volumes along the x-axis. One of these volumes corresponds to the fuel channel component, while the remaining 25 volumes are uniformly distributed within the moderator domain.

3.3. Squirrel

Squirrel is a point kinetics solver for the transient analysis of molten salt reactors developed at the Technical University of Denmark (DTU) within the MOOSE framework (Pfahl et al., 2025). Squirrel is able to evaluate the change in reactivity caused by the DNP advection by weighting the spatial importance of the DNPs with the adjoint of the flux. The reactivity change due to a spatial temperature change is calculated by weighting the spatial difference in temperature with the adjoint.

Squirrel uses modified point kinetics equations. A derivation can be found in the literature (Mattioli et al., 2021; Akcasuh, 2012). It is assumed that the time dependent neutron flux $\Phi(t,\mathbf{r})$ can be separated into a product of the amplitude $N(t)$ and the shape function $\psi(\mathbf{r})$, implying that the time in which the flux changes is slow compared to the amplitude change. The importance of the delayed neutrons is weighted with the adjoint shape function $\psi^*(t,\mathbf{r})$. Essentially, it measures the spatial difference between the adjoint shape and the delayed neutron shape. The temporal change of the amplitude is given by

$$\frac{dN(t)}{dt} = \frac{\rho_{\text{insertion}}(t) + \rho_{\text{temp}}(t) - \beta_{\text{flow}}}{\Lambda} N(t) + \sum_i \lambda_i \frac{\langle \psi(t,\mathbf{r}) | C_i(t,\mathbf{r}) \rangle}{\langle \psi(t,\mathbf{r}) | \nu\Sigma_f \psi(t,\mathbf{r}) \rangle}, \tag{16}$$

with the neutron lifetime Λ and the change in reactivity ρ_{temp} due to a temperature change or $\rho_{\text{insertion}}$ due to an external insertion. β_{flow} is the delayed neutron fraction under consideration of the spatial distribution of the DNPs. The temporal change of the DNP distribution of the ith group, solved by the MOOSE-NS solver, can be written as

$$\frac{\partial C_i(t,\mathbf{r})}{\partial t} = \frac{\beta_i}{\Lambda} \nu\Sigma_f N(t)\psi(\mathbf{r}) - \lambda_I C_i(t,\mathbf{r}) - \mathbf{U}(t,\mathbf{r}) \cdot \nabla C_i(t,\mathbf{r}), \tag{17}$$

with the velocity $\mathbf{U}$, the delay constant of the ith group λ_i, the delayed neutron fraction of the ith group and the fission source $\nu\Sigma_f N\psi$. The effective static delayed neutron fraction with flowing fuel is

$$\beta_{flow} = \frac{\Lambda}{N(t=0)} \sum_i \lambda_i \frac{\langle \psi(t,\mathbf{r}) | C_i(t,\mathbf{r}) \rangle}{\langle \psi(t,\mathbf{r}) | \nu\Sigma_f \psi(t,\mathbf{r}) \rangle}. \tag{18}$$

The difference between the effective delayed neutron fraction and the delayed fraction with flowing fuel is the loss of reactivity due to fuel flow

$$\rho_{loss} = \beta_{eff} - \beta_{flow}. \tag{19}$$

The temperature feedback is evaluated by calculating the temperature difference $\Delta T(t)$ of the current reactor temperature with the steady state reference reactor temperature T_{ref} and weighting that temperature with the normalized adjoint shape function.

$$\rho_{temp} = \frac{\langle \psi | \Delta T \rangle}{\int dr \psi(r)} \Gamma, \tag{20}$$

Γ is the differential of the reactivity with respect to a global temperature change and needs to be calculated beforehand.

Within this benchmark, the flux is assumed to be self-adjoint so that $\psi(r) = \psi^*(r)$. Squirrel retrieves the flux, heating power, and fission source from an initial OpenMC simulation (Romano et al., 2015a). The fields are imported into Squirrel using the MOOSE app, Cardinal (Novak et al., 2022). For this benchmark, Squirrel is coupled with the MOOSE-NS module to receive the temperature field and the DNP field. Squirrel provides the time-dependent fission source density and the heating power to the TH-module. The MOOSE apps exchange the data via the MOOSE multi-app function (Gaston et al., 2015).

Squirrel runs on a structured mesh consisting of 50 × 50 control volumes. The same number of volumes is used to discretize the fuel and the moderator. Squirrel is used in steps 2.1, 2.2, 2.3, 2.4, and 3.1.

3.4. Griffin

Griffin is a deterministic finite element neutronics solver developed at Idaho National Laboratory as a MOOSE-based app (Wang et al., 2025). It is capable of solving the linearized Boltzmann transport equation for steady-state and transient problems in 1D, 2D, and 3D. Griffin is able to analyze fast, and thermal MSR (Jaradat et al., 2024). To account for the movement of the DNPs, Griffin splits the fission source density in the prompt and the delayed contribution. The delayed contribution is constructed from a provided DNP field. The DNP field is provided by the Navier–Stokes module, where the DNP field due to advection is calculated. Additionally, the temperature distribution is provided by

the Navier–Stokes module. With these inputs, Griffin calculates the flux Φ_g by solving the multigroup diffusion approximation of the linearized Boltzmann equation as follows

$$\frac{1}{v_g}\frac{\partial \phi_g}{\partial t} = \nabla \cdot D_g \nabla \phi_g - \Sigma_{t,g}\phi_g + \sum_{g'=1}^{G} \Sigma_{s,g'\to g}\phi_{g'} + \frac{(1-\beta)\chi_{p,g}}{k_{eff}} \sum_{g'=1}^{G} \nu\Sigma_{f,g'}\phi_{g'} + \sum_{i=1}^{I} \chi_{d,g,i}\lambda_i C_i, \tag{21}$$

with standard notation. This equation is solved on the same mesh as Squirrel, with 50 × 50 control volumes. In the model, we apply the two-group cross-sections provided in Tables 8 and 9.

The concentration of the DNPs in the ith group is calculated in the Navier–Stokes module with

$$\frac{\partial C_i}{\partial t} = \frac{\beta_i}{k_{eff}} \sum_{g'=1}^{G} \nu\Sigma_{f,g'}\phi_{g'} - \lambda_i C_i - \mathbf{U}(t,\mathbf{r}) \cdot \nabla C_i, \tag{22}$$

with the velocity $\mathbf{U}(t,\mathbf{r})$ used for advection.

The heating profile is calculated in OpenMC. For transients, it is scaled in Griffin at each time step with the power and then provided to the Navier–Stokes module. The loss of reactivity due to the fuel motion is calculated with the reciprocal difference in the eigenvalue k_{eff} between stationary fuel and flowing fuel so that

$$\rho_{loss} = \frac{1}{k_{stationary}} - \frac{1}{k_{flow}}. \tag{23}$$

Both eigenvalues are calculated with two independent simulations at the same temperature with and without fuel flow. Griffin is used in steps 1.1, 1.2, 2.1, 2.2, 2.3, 2.4, and 3.1.

3.5. MOOSE Navier–Stokes

The MOOSE internal Navier–Stokes (MOOSE-NS) module has been developed to provide general thermal-hydraulics capabilities to the MOOSE framework (Lindsay et al., 2023). It is used in this benchmark to solve the Navier–Stokes equation, the heat and DNP transport in the fluid, heat conduction in the graphite, and the conjugate heat transfer on boundary **7**. The discretization is done using finite volumes. For incompressible fluids, it utilizes the Rhie-Chow interpolation to avoid a decoupling of the pressure and velocity field. The fission source and heating power is provided by the neutronics solver, either Griffin or Squirrel, as inputs. With these inputs, the resulting temperature and DNP fields are calculated.

The MOOSE-NS module solves the incompressible Navier–Stokes equations given by the conservation of mass

$$0 = \nabla \cdot \mathbf{U}, \tag{24}$$

by the conservation of momentum

$$\frac{\partial \mathbf{U}}{\partial t} = -(\mathbf{U}\cdot\nabla)\mathbf{U} - \frac{1}{\rho}\nabla p + \upsilon\nabla^2\mathbf{U}, \tag{25}$$

by the conservation of energy

$$\rho c_p \frac{\partial T}{\partial t} = -\rho c_p \mathbf{U}\cdot\nabla T + \kappa_f \nabla^2 T + q''' + \dot{Q}, \tag{26}$$

with the velocity $\mathbf{U}$, with the density ρ, the pressure p, and the viscosity υ,

The MOOSE-NS module also calculates the DNP production and transport for Griffin (Eq. (22)) and for Squirrel (Eq. (17)). The relevant equations are discussed in the respective section below. The MOOSE-NS solver runs on a mesh with 100 volumes in the y direction and 235 volumes in the x direction. 100 of those are in the fuel and 135 are in the moderator. The refinement is increased closer to interface **7**.

The out-of-core is modeled as a 1D channel. The out-of-core model advects the temperature and the DNPs from the reactor outlet to the inlet. The DNPs and the temperature are advected with a constant velocity. Perfect mixing is assumed. To match the out-of-core time, the length of the out-of-core channel is adjusted. Within the outer core, the DNPs decay according to their respective decay constant λ. No DNPs are produced in the out-of-core model. In addition, Heat is removed with a heat transfer rate $\dot{Q}(r)$ to simulate the heat exchanger. Inside of the core no heat is removed.

The TH module is used in steps 1.2, 2.1, 2.2, 2.3, 2.4, and 3.1. The module is coupled with Squirrel and Griffin in these steps.

3.6. iMC

iMC is a Monte Carlo neutron transport code under development at KAIST (Kim and Kim, 2021). The code is specialized for high computing environments by utilizing parallel computing. The iMC code supports both multigroup and continuous energy nuclear data libraries. Furthermore, the code is capable of secondary photon tracking, which is required for the benchmark. Studies have been done to verify the iMC code's accuracy via code-to-code comparisons.

Several studies were conducted with the iMC code, including MSR-specialized analysis. The impact of the following fuel is considered by direct tracking of the delayed neutron precursor. After the fission neutron is produced, the emission time of the delayed neutron precursors produced from the previous cycle is sampled. The emission site is estimated based on the sampled time and velocity field from the CFD code. This results in a shifted distribution of the delayed neutrons. For the precursors decaying out of the active core, the delayed neutrons are excluded from further calculation. iMC adopts the Iterated Fission Probability (IFP) concept to produce adjoint-weighted kinetic parameters. The adjoint neutron flux stands for the importance, so it needs to be applied to evaluate effective kinetic parameters. Although the direct calculation of the adjoint flux is complicated, the IFP allows Monte Carlo codes to estimate the adjoint-weighted parameters, including the effective delayed neutron fraction. The iMC code confirmed its IFP capability by code-to-code comparison with conventional Monte Carlo and deterministic codes. Previous works showed that the fuel flow causes a noticeable reduction in reactivity and kinetic parameters. The scheme is applied in this study to measure and compare the impact of the velocity field on the delayed neutron precursor shifts and their impact on the reactor performance.

In this study, the iMC code is used in steps 1.1, 2.1, 2.2, 2.3, and 2.4. The code is coupled with an open-sourced CFD code; OpenFOAM® (Weller et al., 1998) except for step 1.1. The coupling is a file-based communication of temperature, fuel velocity, and heating distributions. All iMC calculations are performed with 500,000 particles per cycle, with 200 inactive and 500 active cycles. This results in a standard deviation of the eigenvalue of 6 pcm.

3.7. OpenFOAM

OpenFOAM® (Open-source Field Operation And Manipulation) is a powerful, free, and open-source software for continuum mechanics simulations. Widely used by engineers, scientists, and researchers, it provides a comprehensive suite of tools for modeling and analyzing fluid dynamics, turbulence, heat transfer, optimization, acoustics, chemical reactions, solid mechanics, and electromagnetics. OpenFOAM® finds applications across various industries.

Designed with modularity and flexibility, OpenFOAM® allows users to customize and extend its functionalities, enabling advanced simulations tailored to specific needs. Its open-source nature fosters collaboration, transparency, and innovation, providing a platform for shared development and knowledge exchange within the global community.

Since the benchmark model contains both fluid and solid regions, the conjugate heat transfer solvers *chtMultiRegionFoam* and *chtMultiRegionSimpleFoam* are utilized, where the latter option provides a steady-state solution. Furthermore, heating transferred from the neutronics result is suggested on 25-by-50 equivalent meshes for each region. The heating data is utilized in the OpenFOAM® with *fvOptions*.

An out-of-core model is crucial for steps 2.4 and 3.1 to estimate a fuel inlet temperature. The out-of-core model is designed with a vertical extension of the fuel region. The velocity is controlled to a constant and considered as a 1-dimensional problem. The inlet temperature is estimated by solving the cooling scheme numerically by subdividing the extended salt region.

The code is used stand-alone for step 1.2 and coupled with the iMC code for phases 2 and 3. The OpenFOAM® uses 50,000 meshes total, which gets finer near the interface. The CFD result then undergoes mesh conversion, equivalent to the heating mesh from the iMC code. The mesh conversion is performed with the postprocessing feature of the OpenFOAM®.

3.8. Nek5000

Nek5000 (Fischer et al., 2007) is a high-order spectral element code designed for the simulation of incompressible and low-Mach number flows, heat transfer, and related phenomena. Based on the spectral element method (SEM), it combines the geometric flexibility of finite elements with the spectral accuracy of global methods, enabling efficient and highly accurate solutions for complex geometries and flow dynamics. The code employs a parallel implementation using the Message Passing Interface (MPI), demonstrating excellent scalability on high-performance computing systems. Nek5000 has been extensively validated and applied to a wide range of problems, including turbulent flows, thermal convection, and magnetohydrodynamics. For this study, Nek5000 is used to solve the incompressible Navier–Stokes equations coupled with energy and passive scalar transport equations in order to generate high-fidelity reference solutions for temperature and precursor concentration distributions. Furthermore, it is complemented by a point-kinetics model to solve the transient problem for step 3.1.

Three solution variations are used:

1. *Fluid-domain simulation only*. In this simulation, 1D transversal conduction in graphite is assumed, and the conjugate heat problem is not considered. Therefore, heat transfer from the graphite is approximated by a wall heat flux. The main focus of this simulation is obtaining a converged velocity field and precursor concentration. The precursor conservation equations are solved, taking into account the recirculation through the ex-core and corresponding exponential decay. To this end, the recirculation boundary condition is implemented as a user-defined function.
2. *Conjugate heat-transfer simulation*. In this simulation, the converged velocity field from the previous step is used in a configuration explicitly including the graphite part of the domain. Velocity is not solved as temperature is treated as a passive scalar, i.e., having no effect on the flow field. The convergence of this problem is rather slow, since the conduction time scales in the graphite are very long. The Dirichlet boundary condition for temperature at the inlet is used.
3. *Full reactor modeling*. For steps 2.4 and 3.1 below, the fuel channel is extended beyond the graphite-moderated core region so as to achieve the reference ex-core residence time. The resulting out-of-core region is two-dimensional, rather than one-dimensional, but the boundary condition for both sides of the flow channel is taken as symmetry for fuel velocity and insulated for temperature and DNP concentration. Therefore, all quantities gradually converge to flat transversal distributions as they are transported through the channel. The heat-exchanger action is distributed in the 2D channel so as to match the total heat removal prescribed for the 1D reference model. Both fuel temperature and DNP concentration at the outlet of the out-of-core region are velocity-averaged and recirculated to the inlet of the core. Since the length of the out-of-core region matches the desired residence time, no further exponential decay is needed. For the modeling of reactor kinetics, a point-kinetic approach identical to the one taken by the Squirrel code above was used. The heating profile was supplied from the OpenMC reference calculation and was taken to approximate the DNP source term in the fuel. The flux profiles in fuel and graphite, as well as temperature reactivity coefficients, were also supplied from the OpenMC reference calculation. To obtain nuclear temperatures used for calculating the temperature feedback, precursor drift, and delayed-neutron source term for point kinetics, the same weighting approach as described for the Squirrel code above is used.

For spatial discretization, 2D elements were used in the fluid and solid domains with eight quadrature points per element per direction. For the simulations without explicit ex-core representation, the Nek5000-native *genbox* mesh generator was used, while for full reactor modeling, *Gmsh* was used to generate the block-structured mesh (Geuzaine and Remacle, 2008). In the graphite region, 50 uniform elements were used in the axial direction and seven in the transversal direction with mesh grading 0.77 towards the wall. Both in the core and ex-core fuel regions, 50 uniform elements were used in the axial direction and five in the transversal direction with mesh grading 0.7 towards the wall. In total, 350 solid-domain elements and 250 (no ex-core) or 500 (full reactor modeling) fluid-domain elements were used. For step 1.2 a finer mesh was used, with eight quadrature points per element per direction, 100 uniform elements were used in the axial direction and ten in the transversal direction with mesh grading 0.8 towards the wall. The difference in reactivity was estimated to be 0.3pcm. For time-dependent simulations, a variable time step based on the Courant–Friedrichs–Lewy (CFL) condition $CFL \leq 0.5$ was used.

3.9. OpenMC

OpenMC is an open-source Monte Carlo code designed for neutron and photon transport (Romano et al., 2015a), originally developed at the Massachusetts Institute of Technology (MIT) in 2011. It supports various types of simulations, including fixed source, criticality eigenvalue, and subcritical multiplication. OpenMC can effectively simulate both 2D and 3D problems with a continuous energy spectrum. Additionally, it can run in parallel using MPI and OpenMP, enabling highly accurate simulations within a short time frame. The code has undergone rigorous benchmarking to ensure its reliability and accuracy (see Labossiere-Hickman and Forget (2017), Chaudri and Mirza (2015) and Kreher et al. (2019)).

OpenMC serves as the reference for the kinetics parameters presented in Table 10 and the cross-section data provided in Tables 8 and 9. These cross-sections are utilized by SIMMER and Griffin. In step 1.1, OpenMC calculates the reactivity, temperature coefficients, flux, heating, and kappa fission. It also supplies the heating data to Squirrel, TRANSFORM, Nek5000, SIMMER, and Griffin. Furthermore, Squirrel and TRANSFORM depend on the temperature coefficients, fission source density, and flux provided by OpenMC. The OpenMC code is particularly advantageous for this benchmark, as there is a wrapper called Cardinal (Novak et al., 2022) that integrates it into the MOOSE framework. This integration ensures a seamless transfer of the calculated flux, kappa fission, and heating data.

OpenMC is modeled on a mesh with 50 cells in the y direction and 50 cells in the x direction. 25 of those are in the fuel and 25 are in the moderator. The cells are equidistantly spaced. The same mesh is reproduced in the Cardinal input. Cardinal atomically adjusts the density and the temperature. The reference Monte Carlo simulation was run with 10^6 particles, 1000 batches of which 200 were inactive.

3.10. Serpent

The Monte Carlo transport code Serpent has been developed by VTT since 2004 (Leppänen et al., 2025). Initially defined as a reactor physics code, Serpent can also be used for broader applications, especially radiation shielding and fusion. The code allows for coupled neutron-photon transport and, in particular, offers the choice between different

levels of accuracy for nuclear heating simulations. Version 2.1.32 of the code was used. By default, Serpent energy deposition calculations are limited to the site where the fission reactions occur, but three other energy deposition modes have been implemented in the code to increase the accuracy of nuclear heating calculations (Tuominen et al., 2019). Mode 3, based on the Serpent existing coupled neutron-photon transport capability, is of particular interest, allowing for the most realistic energy deposition modeling, including photon transport. Using mode 3, nuclear heating due to photons is calculated using an analog photon heat deposition tally, which is scored after each interaction leading to energy loss (Compton scattering, pair production, and photoelectric effect). Accounting for photon transport, the secondary photons can be transported in the reactor using photon physics modeling. Neutrons do not deposit all their energy at the fission site but at various locations during their lifetime. This allows for the calculation of the non-local neutron heating contribution, leading to a more accurate assessment of the nuclear heating in the structural materials of the reactor. Serpent is used in step 1.1 as a reference code for calculating the reactivity, the neutron flux, and the nuclear heating in the fuel and moderator. The simulations were run using 500 batches of 10^5 particles, 200 batches being inactive. A nuclear data library containing additional data to be used for the different energy deposition modes. This library is distributed by the Serpent developers (Babcsány et al., 2023).

4. Benchmark description

The steps in this benchmark gradually increase in complexity. The benchmark starts with single physics and ends in a fully coupled multi-physics ULOF transient. This stepwise approach allows for a better understanding of the final transient and eases the implementation into code. Eventual differences in codes in the single physics or the coupled multi-physics can easily be detected, allowing for a comparison of different codes throughout the benchmark.

Phase 1: Single physics

In this phase of the benchmark, the code's ability to solve single-physics problems is tested. The goal is to evaluate the neutronics and thermal-hydraulics behavior in the reactor separately. The steps in this phase are without coupling and ensure that the codes' different approaches to single physics problems and the resulting differences are well understood. This makes it easier to evaluate discrepancies in later steps.

Step 1.1: Investigation of the criticality eigenvalue problem with no fuel flow

In this step, the criticality eigenvalue problem is evaluated. The codes should be able to evaluate the correct reactivity and the spatial heating power profile without fuel flow. The excess reactivity will be used as a reference value to calculate the change in reactivity in the following steps. This step ensures that the heating in the fuel and the moderator are evaluated consistently.

A constant material temperature of 873.15 K is imposed. The correct material density is calculated for the corresponding temperature. At 873.15 K the density of the fuel is 4126.85 kg m^{-3}, while the graphite density is constant. Fuel flow and the DNP drift are not considered in this step. The measurements are the excess reactivity and the heating profile.

For the heating, the neutron and photon energy deposition is considered. The (α, β) scattering in the moderator is evaluated using the appropriate thermal scattering library from the ENDF/B-VII.1 nuclear library. The excess reactivity is defined as

$$\rho = \frac{k_{eff} - 1}{k_{eff}}, \tag{27}$$

with the eigenvalue k_{eff}. The used parameters are:

- Fixed material temperature and density

The measured values are:

- Reactivity at 873.15 K in pcm
- Total heat deposition in the fuel and in the moderator
- Heating power in the moderator from the point (0.01 m, 0.75 m) to (0.23 m, 0.75 m)
- The temperature feedback coefficients

The temperature feedback coefficients

$$\Gamma = \frac{\partial \rho}{\partial T}, \tag{28}$$

are calculated, relative to an increase of the temperature by 500 K between 873.15 K and 1373.15 K. Within this range, the total temperature feedback is measured to be linear. The temperature feedback will be split into the total temperature feedback coefficient, the temperature feedback coefficient of the fuel, and the temperature feedback coefficient of the moderator. Furthermore, the temperature feedback coefficient caused by the Doppler effect and by a change in density are calculated. This tests the ability of the code to correctly evaluate the feedback and related change in reactivity cause by changes in temperature.

Step 1.2: Temperature distribution with fuel flow

In this step, fuel flow in the channel with a fixed inlet velocity of 0.2 ms^{-1} and the temperature profile is measured. The inlet temperature is constant at 873.15 K. The reactor is heated with the power density obtained in step 1.1.

Step 1.2 tests that the code can solve the Navier–Stokes equation with a fixed inlet boundary condition. Furthermore, the ability of the code to correctly evaluate the heat deposition and transport in the fuel, the moderator, and the conjugated heat transfer on the interface is tested. The different approaches in modeling the fuel flow and the heat transfer will become visible.

The used parameters are:

- Fixed inlet temperature 873.15 K
- Fixed heat source from 1.1
- No buoyancy

The measured values are:

- The velocity field of the fuel from point (0.0 m, 0.75 m) to point (0.01 m, 0.75 m)
- The area-averaged temperature distribution in the fuel along the y-axis
- The velocity-averaged temperature distribution in the fuel along the y-axis
- Average fuel temperature of the salt flowing out of the channel
- Radially-averaged temperature distribution along the y-axis in the moderator

Phase 2: Multi-physics

Step 2.1: Effect of the temperature field on the reactivity

In this step, the effect of the temperature field calculated in step 1.2 on the reactivity is evaluated. The step tests the ability of the codes to calculate the correct change in reactivity caused by a change in the temperature field. Additionally, this step allows us to estimate the differences introduced by the thermal hydraulics modeling approaches and in the feedback coefficients.

The measured quantity is the change in reactivity relative to step 1.1, so that

$$\Delta\rho = \rho_{2.1} - \rho_{1.1} \tag{29}$$

The used parameter is:

- Fixed temperature field from step 1.2

The measured value is:

- Reactivity change with respect to step 1.1

Step 2.2: Effect of the DNP movement on the reactivity

In this step, the impact of the moving fuel on the eigenvalue due to moving DNPs is evaluated, as well as the ability to calculate the reactivity effects stemming from the reentering of the DNPs.

Step 2.2 tests the ability of the code to correctly calculate the eigenvalue problem with moving fuel and an out-of-core model with reentering of DNPs after a specific out-of-core time. The DNPs are assumed to be perfectly mixed in x direction when reentering. The flow and temperature field remain the same as in step 1.2. The fission source density remains the same as in step 1.1.

In this step, we evaluate the change in reactivity and the delayed neutron source, equal to $\sum_{i=1} \lambda_i C_i$. The change in reactivity is calculated relative to Step 2.1, taking into account only the effects of the DNPs movement. The reactivity loss is recorded for different out-of-core times. The delayed neutron source produced is averaged horizontally over the fuel channel and for 10 s of out-of-core time.

The used parameters are:

- Fixed flow field form step 1.2
- Fixed temperature field form step 1.2
- Returning DNPs
- Out of core time $\tau \in \{0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 1, 3, 10, 30, 100, 300, 1000\}$ s

The measured values are:

- Loss of reactivity for $\Delta\rho = \rho_{2.2} - \rho_{2.1}$ considering 10 s out-of-core time
- Volume-averaged delayed neutron source in the fuel channel for 10 s out of core time, normalized to one
- DNP reactivity worth $\Delta\rho = \rho_{2.2} - \rho_{2.1}$ for different out-of-core times

Step 2.3: Effect of a change in the inlet velocity on the reactivity

In this step, the effect of a change in the velocity on the reactivity is analyzed. The flow speed is changed inside and outside of the core. The resulting contributions of the thermal feedback and the DNP movement on the change in reactivity are evaluated.

Step 2.3 analyzes the ability of the code to evaluate the relation between the flow rate, fuel, and moderator temperature, and the DNP advection with a separate out-of-core model. With these parameters, the codes have to calculate the correct change in reactivity due to DNP movement and the temperature change, relative to step 1.1. This ensures the correct implementation of the different components.

The used parameters are:

- Fixed power 350 kW
- Inlet temperature (600 °C)
- Returning DNPs
- $v_{inlet} \in \{0.02, 0.03, 0.05, 0.075, 0.1, 0.2, 0.3, 0.4, 0.5, 1, 2, 5, 10\}$ ms^{-1}

The measured values are:

- Total reactivity change $\Delta\rho = \rho_{2.3} - \rho_{2.2}$ with 10 s out-of-core time
- Temperature contribution to the reactivity change
- DNP contribution to the reactivity change

Step 2.4: Full reactor modeling

In this step, the fuel is cooled outside of the reactor core in the heat exchanger. With the heat exchanger, the steady-state solution to the reactor with a constant power of 350 kW is calculated. This tests the full steady-state coupling between neutronics and thermal hydraulics and will provide the steady-state reference reactor in the transient calculations. This is deemed necessary to ensure good comparability between the codes in the next steps.

The used parameters are:

- Fixed power of 350 kW
- Returning DNPs
- 10 s out of core with the fully implemented out-of-core model (see Section 2.2).

The measured values are:

- Steady-state inlet temperature
- Steady-state outlet temperature
- Steady-state reactivity change $\rho_{2.4} - \rho_{1.1}$
- Steady-state reactivity change DNP contribution
- Steady-state reactivity change temperature contribution

In addition to the above steps, the participants are required to confirm that the system is at a steady state by running a null transient. In this transient simulation, the power is fully coupled and develops freely. With no external change to the system, the codes must run a transient for 1000 s with no change in power or temperature.

Phase 3: Full transient coupling

Step 3.1: Unprotected loss of flow

A ULOF accident is simulated in which the fluid flow gradually decreases, e.g., due to pump failure. For simplicity, the velocity v_{inlet} at the inlet of the core changes according to

$$v_{inlet}(t) = (v_{initial} - v_{final}) \exp(-t\gamma) + v_{final} \tag{30}$$

with $v_{initial} = 0.2$ ms^{-1}, $\gamma = 0.1$ s^{-1}, and $v_{final} = 0.02$ ms^{-1}. The bulk fuel flow will converge to the final velocity $v_{final} = 0.1 v_{initial}$, by the change in the inlet velocity. The final velocity is assumed to result from natural circulation and will, in practice, depend on the details of the reactor out-of-core model.

This global change in velocity implies that fuel salt has a longer residence time in the out-of-core region, and more heat is removed from the fuel salt, cf. Eq. (1). During the transient, the flow rate is assumed not to be affected by the power level of the reactor.

In Step 3.1, the codes have to evaluate the changing power level, lower flow rate, and resulting temperature change occurring in a ULOF accident correctly. The change of power will affect the temperature and the DNP production in the reactor. The changed temperature will impact the current power level, testing the heat transfer between the graphite and the fuel, since the heat transfer depends on the velocity. The DNP movement will be affected by the change in power and the changing flow field. The periodic reentering of the DNPs will, in turn, change the power level of the reactor. Differences that occurred in previous steps will occur in the power evolution of the reactor.

The used parameters are:

- Reduction of inlet velocity according to Eq. (30)
- Free power evolution over time
- Out-of-core model from Section 2.2 with changing $U(t)$
- 2 m out-of-core length
- 1000 s transient time

The measured values are:

- Total reactor power as a function of time
- Maximum temperature change in the moderator as a function of time
- Salt temperature change at the outlet as a function of time

Table 14
Contribution to the steps.

	Step 1.1	Step 1.2	Multi-physics steps	Transient
SIMMER	X	X	X	X
TRANSFORM		X	X	X
Squirrel			X	X
Griffin	X		X	X
MOOSE-NS		X	X	X
iMC	X	X	X	
OpenFOAM		X	X	
Nek5000		X	X	X
OpenMC	X			
Serpent	X			

Table 15
ρ at 873.15 K evaluated by the participants.

	SIMMER	Griffin	iMC	OpenMC	Serpent
ρ (pcm)	3772	3768	2014 ± 4	2016 ± 3	2014 ± 16

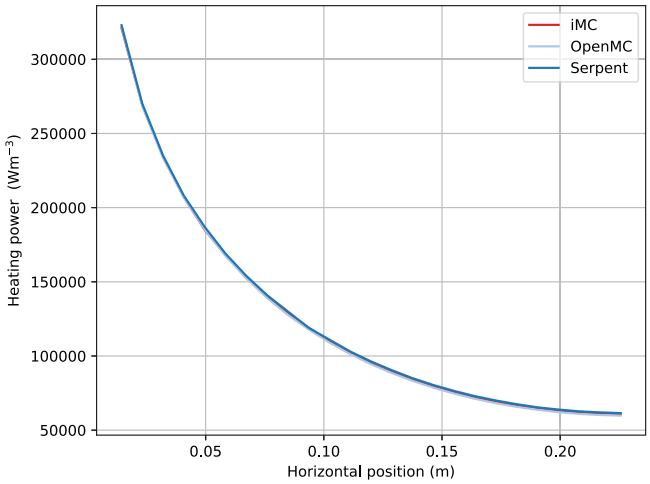

Fig. 4. Moderator heating power (Wm^{-3}) along line center.

5. Results

In this section, the results for each of the steps presented in Section 4, obtained by each participant, are presented. Depending on the code's ability, some of the steps are not evaluated. SIMMER is able to solve both the neutronics and thermal-hydraulics steps. Squirrel, TRANSFORM, and Nek5000 are using the neutronics results generated by OpenMC. Griffin and Squirrel rely on the MOOSE-NS solver to account for the thermal-hydraulics, and iMC is coupled with OpenFOAM. Which code is involved in which single physics step is shown in Table 14.

The coupled steps are simulated by SIMMER, TRANSFORM, Squirrel coupled with MOOSE-NS, Griffin coupled with MOOSE-NS, iMC coupled with OpenFOAM, and Nek5000.

For some measurements, the comparison is redundant, e.g., Squirrel, TRANSFORM, and Griffin use the heating profile obtained with OpenMC. Therefore, their heating power profile is not reported. Whenever a code produces a unique solution, the results are reported. The resulting data for the plots can be downloaded from Philip Pfahl (2025). For all stochastic results obtained by Monte Carlo codes, uncertainties are given as one standard deviation.

Since there is no analytical solution to take as a reference, the approach to estimate a discrepancy presented by Tiberga et al. (2020) is taken here. At a single point, the average value

$$\bar{V} = \frac{1}{N}\sum_{i}^{N} V_i \tag{31}$$

obtained from all N codes is taken. The relative discrepancy between code i and the average solution is calculated as

$$D_i = \left| \frac{V_i - \bar{V}}{\bar{V}} \right|. \tag{32}$$

Phase 1: Single physics

In this phase of the benchmark, the results of the single physics evaluations are shown, first, the investigation of the critical eigenvalue problem, and then the solution to the pure thermal hydraulics problem.

Step 1.1: Investigation of the critical eigenvalue problem with no fuel flow

In Table 15, the reactivity at 873.15 K for each participant is presented. The Monte Carlo codes agree within two pcm. The excess reactivity is slightly above 2000 pcm. SIMMER and Griffin overestimate the excess reactivity with ≈1750 pcm, compared to the Monte Carlo results. The difference between SIMMER and the Monte Carlo solver is a known issue.

In Table 16, the reactivity feedback coefficients evaluated by each participant are shown. The coefficients are split into the total temperature feedback, the fuel temperature feedback, the moderator temperature feedback, the Doppler effect, and the density contribution. OpenMC and iMC agree on the values for the moderator feedback. However, the Monte Carlo codes differ slightly in terms of temperature feedback in the fuel. The difference is more significant when splitting between the Doppler and density feedback. The difference in the temperature treatment might explain this. While OpenMC linearly interpolates the temperature data, iMC uses a single temperature point and performs an on-the-fly Doppler broadening treatment. Serpent uses a Doppler-broadening preprocessor routine enabling the temperature of a free-atom cross-section to be increased below the energy region of unresolved resonances. The Doppler-broadening is performed analytically using the Solbrig's kernel.

The OpenMC heating results are used as a reference for SIMMER, TRANSFORM, Squirrel, Griffin, and Nek5000. Therefore, the values are not reported. SIMMER deviates slightly from the OpenMC results, with 0.3 pcm/K for the fuel temperature feedback; these differences are mainly from the density feedback. The good agreement obtained with SIMMER of the reactivity coefficients shows that the difference in the initial reactivity is a systematic deviation. The total feedback coefficient obtained by Griffin is lower than the SIMMER result. While the single results are in agreement with the Monte Carlo results. Due to the provided cross-section data, it is not possible to obtain the Doppler and density feedback.

In Table 17, the heat deposition in the moderator and the fuel is shown, as well as the ratio of power deposited in the moderator to total power. iMC and OpenMC agree on heating the moderator with a difference of 0.03% (94 W). OpenMC and Serpent also agree with a difference of 0.02% (84 W). The reference for SIMMER, Griffin, and Squirrel is again OpenMC. These codes obtained the heating profile from OpenMC and are not reported.

In Fig. 4, the heat deposition within the moderator along the center line is shown, testing the ability of the code to calculate a correct heating profile. The results are only presented for the Monte Carlo codes since the diffusion codes are not used to calculate the photon transport. All codes, iMC, OpenMC, and Serpent, agree on the exponential decay of the heat deposition in the moderator.

Step 1.2: Temperature distribution with fuel flow

At the inlet of the fuel channel (boundary **1** in Fig. 1), the fuel enters with a constant velocity of 0.2 ms^{-1}. Due to the no-slip boundary condition on interface **7**, the flow slows down near the graphite. Due to the conservation of mass, the slowing down in one region results

Table 16
Temperature reactivity coefficients Γ.

	SIMMER	Griffin	iMC	OpenMC	Serpent
Γ (pcm/K)	−7.495	−7.347	−7.680 ± 0.009	−7.832 ± 0.009	−7.839 ± 0.05
Γ_{fuel} (pcm/K)	−3.449	−3.835	−3.699 ± 0.009	−3.732 ± 0.009	−3.802 ± 0.05
$\Gamma_{moderator}$ (pcm/K)	−3.544	−3.591	−3.588 ± 0.009	−3.588 ± 0.009	−3.559 ± 0.05
$\Gamma_{Doppler}$ (pcm/K)	−1.402		−1.485 ± 0.009	−1.437 ± 0.009	−1.463 ± 0.05
$\Gamma_{density}$ (pcm/K)	−2.046		−2.202 ± 0.009	−2.302 ± 0.009	−2.383 ± 0.05

Table 17
Heating in the fuel and in the moderator.

	iMC	OpenMC	Serpent
P_{fuel} (W)	323 790	323 234	323 314
$P_{moderator}$ (W)	26 859	26 765	26 686
$P_{moderator}/P_{total}$ (%)	7.67	7.64	7.62

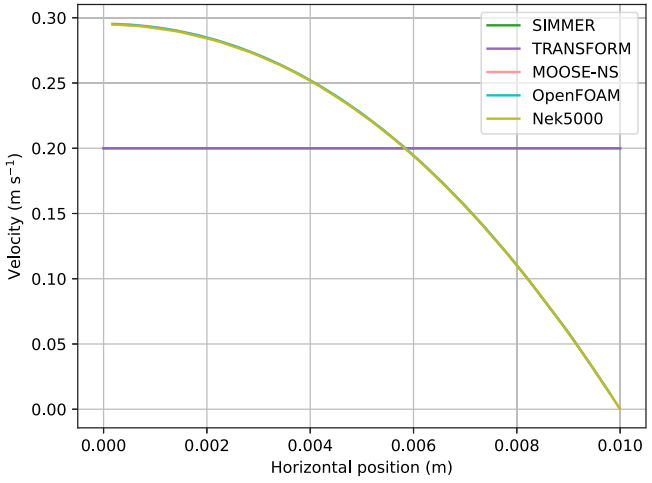

Fig. 5. Flow field (ms^{-1}) along the center line.

Table 18
Velocity (ms^{-1}) at different points along the center line.

Point in x and y (m)	MOOSE-NS	OpenFOAM	Nek5000	Average
(0.0002, 0.75)	0.29537	0.29514	0.29476	0.29509
(0.005, 0.75)	0.22642	0.22659	0.2261	0.22637
(0.0098, 0.75)	0.01219	0.01217	0.01218	0.01218
Mean discrepancy	0.07024%	0.06664%	0.0789%	0.07192%

in an increased velocity in the middle region of the channel. The resulting converged velocity profile solution has a parabolic shape and corresponds to the analytical solution for laminar flow between parallel plates.

In Fig. 5, the flow field along the center line of the fuel channel is shown for each participant. The codes SIMMER and TRANSFORM have only one radial volume in the fuel channel and, therefore, have a constant velocity across the fuel channel. The other codes evaluate the momentum in the Navier–Stokes equation to calculate the flow pattern. For MOOSE-NS, OpenFOAM, and Nek5000, the parabolic velocity profile is not yet fully developed. The maximum velocity of 0.3 ms^{-1}, predicted by the Poiseuille equation for a channel with these dimensions, has not been reached (White and Majdalani, 2006).

In Table 18 the flow field for the fully resolved solvers is shown for the points (0.75, 0.0002), (0.75, 0.005), and (0.75, 0.0098) m. Additionally, the average value and the mean discrepancy (Eq. (32)) are presented.

Inside the core, the fuel is heated in two ways: directly by fission within the salt and from the heated moderator. The heating by fission is directly proportional to the fission rate density in the reactor. It follows a sinusoidal function with an offset along the fuel channel.

Table 19
Average outlet fuel temperature (K).

	SIMMER	TRANSFORM	MOOSE-NS	Nek5000	OpenFOAM
Outlet temperature (K)	915.7	915.6	915.6	915.6	915.6

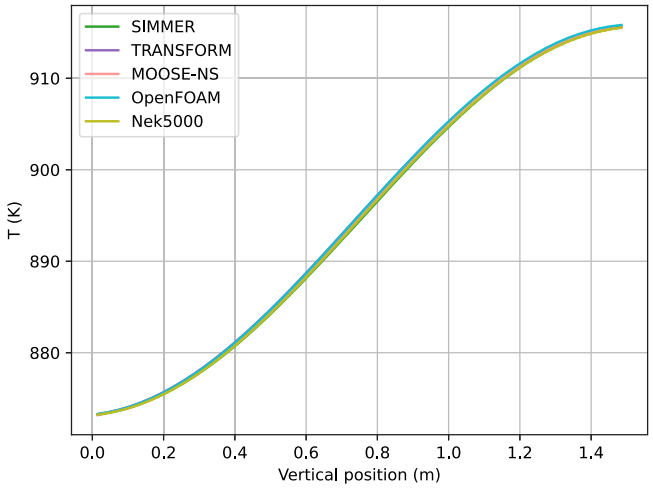

Fig. 6. Velocity-averaged fuel temperature, plotted along the y direction.

The heating from the moderator wall depends on the temperature gradient across the boundary. At a steady state, the total amount of heat energy that the fuel is moving out of the core is equal to the reactor power of 350 kW. The analytical solution for the outflowing fuel temperature is 915.6 K, and the simulated temperature is shown in Table 19. In Fig. 6, the average salt temperature in y direction is shown for each participant. These values correspond to the velocity-averaged fuel temperature, showing the ability of the code to solve the enthalpy flow correctly. In Table 20 the velocity-averaged fuel temperature along the y-axis at 0.015, 0.75, and 1.485 m is shown for all codes. Additionally, the average value and the mean discrepancy are presented. The good agreement is expected, since all codes are required to be able to conserve energy.

In Fig. 7, the volume-averaged fuel temperature in y direction for each participant is shown. This temperature is the temperature that the codes use to calculate the neutron material interaction in the upcoming sections. MOOSE-NS, OpenFOAM, and Nek5000 fully describe the developed flow profile and the conjugate heat transfer. The codes agree on the rise in fuel temperature. SIMMER and TRANSFORM have only one radial cell in the salt; the velocity profile is, therefore, not resolved in the radial direction (see Fig. 5). Their temperature, therefore, is the velocity-averaged temperature reported in Fig. 6. The following steps of the benchmark must take into account the difference in the fuel temperature.

Neutrons and photons emitted by fission heat the graphite. The heating profile is shown previously in Fig. 4. Since the moderator is insulated on all boundaries, the heat can only flow out of the moderator at the fuel-moderator interface **7** into the flowing fuel. Therefore, the

Table 20
Velocity-averaged fuel temperature (K) at different points along the y-axis.

Point in y direction (m)	SIMMER	TRANSFORM	Moose	OpenFOAM	Nek5000	Average
0.015	873.22	873.29	873.17	873.31	873.26	873.25
0.75	894.42	895.09	894.88	895.07	894.67	894.80
1.485	915.69	915.57	915.55	915.82	915.60	915.64
Mean discrepancy	0.01781%	0.01425%	0.00853%	0.01762%	0.00792%	0.07192%

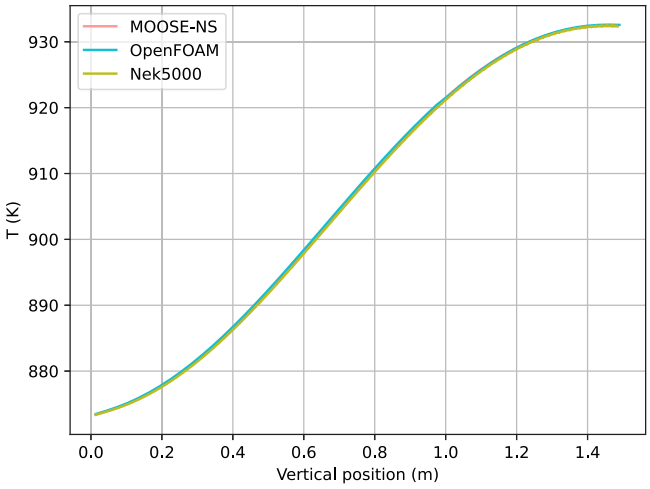

Fig. 7. Horizontally volume-averaged fuel temperature, plotted in y direction.

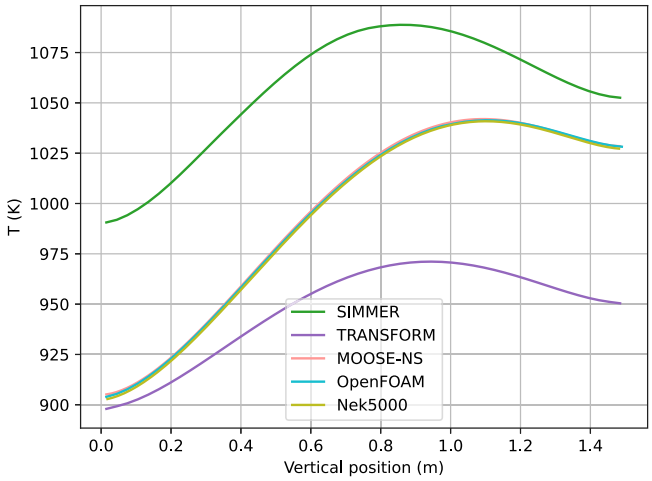

Fig. 8. Horizontal average moderator temperature (K), plotted in y direction.

moderator temperature is significantly higher than the fuel temperature. In this benchmark, it is assumed that the heat deposited in the moderator is symmetric with respect to the center line.

The heat is conducted through the graphite moderator, resulting in an axial heat flow from the hot center region into the less heated regions at the top and bottom of the reactor. Despite the heating symmetry, the maximum temperature will be shifted upwards since the fuel is hotter at the outlet of the reactor, reducing the cooling of the graphite.

In Fig. 8 the average moderator temperature along the y-axis is shown for each participant. There is good overall agreement, and all codes capture the shift of the maximum peak to the top of the reactor. The temperature gradient at the top and the bottom of the reactor flattens out, due to the no-flux boundary condition. MOOSE, OpenFOAM, and Nek5000 fully implement conjugate heat transfer, radial and axial heat conduction, and a refined mesh. These solutions should be taken as a reference. TRANSFORM and SIMMER model axial and radial heat conduction on a significantly coarser mesh. As 1D thermal-hydraulic coarse mesh codes, they do not incorporate subchannel modeling. Consequently, the spatially developing temperature field within the channel cannot be simulated. Instead, the model inherently assumes a fully developed thermal and velocity profile. While this assumption is generally acceptable for turbulent flows, where entrance effects are negligible due to the short entrance length, it presents a limitation in this particular case, as thermal boundary layers are still developing along the channel length. This phenomenon is accounted for by locally varying the Nusselt number in Eq. (8), which assumes constant wall heat flux, thereby introducing an error. Furthermore, the presence of internal heat generation within the fluid translates to an increase in wall temperature, ultimately reducing heat transfer (Poppendiek and Palmer, 1954). This phenomenon is fully neglected in the present study due to the lack of a suitable correlation applicable to transient simulations. For TRANSFORM, this omission is the cause of the systematic underestimation of graphite temperature throughout the entire domain. These discrepancies are known limitations of 1D thermal-hydraulic codes and are consistent with their simplified representation of fluid flow. SIMMER additionally assumes only two values for radial heating distribution: one for the salt and one for the graphite. The overestimation of the graphite temperature cannot be explained by the difference in the heating distribution alone. An intensive study of this phenomenon has been conducted without providing any conclusive explanation.

Table 21
Average graphite temperature along the y-axis.

Point in y direction (m)	MOOSE-NS	OpenFOAM	Nek5000	Average
0.015	905.15	903.93	902.75	903.95
0.75	1019.07	1017.98	1017.24	1018.09
1.485	1028.47	1028.31	1027.22	1028.00
Mean discrepancy	0.09166%	0.0143%	0.09758%	0.07192%

In Table 21 the averaged graphite temperature for the fully resolved solvers is shown along the y-axis at 0.015, 0.75, and 1.485 m. SIMMER and TRANSFORM are excluded, since the solutions obtained with the parameterized heat transfer are very different. Additionally, the average value and the mean discrepancy are presented.

Phase 2: Multi-physics

Step 2.1 Effect of the temperature field on the reactivity

The change in temperature reported in step 1.2 affects the system's neutronics behavior. In Table 22, the reactivity change with the temperature field obtained in step 1.2 relative to step 1.1 is shown for each participant. The neutronics codes coupled to a high fidelity thermal hydraulics solver, Squirrel, Griffin, iMC, and Nek5000, agree within 7 pcm on the mean of −586.3 pcm. This agreement is expected since these codes use similar temperature fields obtained in Phase 1. SIMMER overestimates the temperature feedback due to the higher graphite temperature (See Fig. 8). The opposite is true for TRANSFORM. The lower graphite temperature field discussed previously leads to an underestimation of the reactivity feedback. The difference in reactivity between SIMMER and TRANSFORM is 300 pcm. Griffin estimates a lower temperature feedback as shown in Table 16, which results in lower reactivity in this step.

Table 22
Reactivity change $\Delta\rho = \rho_{2.1} - \rho_{1.1}$.

	SIMMER	TRANSFORM	Squirrel	Griffin	iMC	Nek5000	Average
$\Delta\rho$ (pcm)	−670	−370	−593	−558	−580 ± 6	−586	−559.5
Discrepancy	19.75%	33.87%	5.99%	0.27%	3.66%	4.74%	11.37%

Table 23
Reactivity change $\Delta\rho = \rho_{2.2} - \rho_{2.1}$.

	SIMMER	TRANSFORM	Squirrel	Griffin	iMC	Nek5000	Average
$\Delta\rho$ (pcm)	−275	−276	−260	−261	−253 ± 7	−262	−264.5
Discrepancy	3.97%	4.35%	1.70%	1.32%	4.35%	0.95%	2.77%

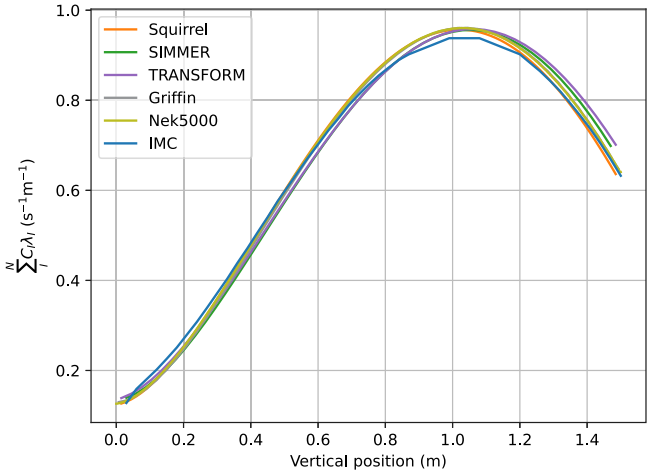

Fig. 9. Average delayed neutron source along the y-axis.

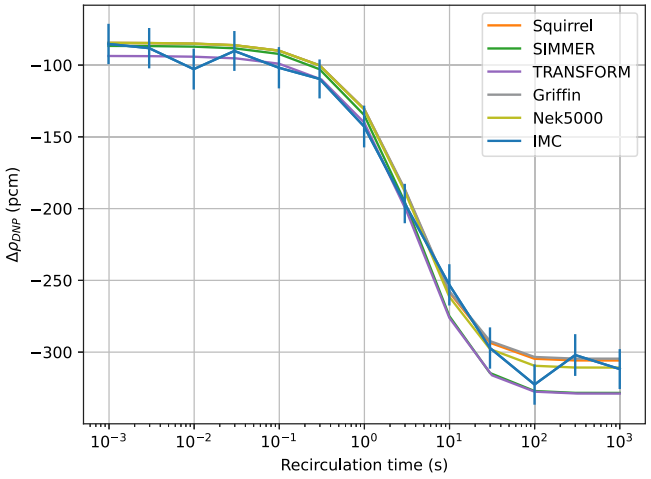

Fig. 10. Reactivity change $\Delta\rho = \rho_{2.2} - \rho_{2.1}$ for different recirculating times. For the iMC code the 2 σ error is shown.

Step 2.2 Effect of the DNP movement on the reactivity

The DNPs are produced by fission and move upwards with the salt in the reactor. In Fig. 9, the resulting volume-averaged and normalized delayed neutron emission along the y-axis is shown. The delayed emissions are clearly skewed towards the upper part of the reactor. The maximum value of the delayed neutron source occurs above the center line of the reactor. Due to a lower fission rate in the outer regions and some fast-decaying DNP groups, the delayed neutron source reduces in the uppermost region. The DNPs reenter the channel at the bottom.

The difference observed between on the one hand SIMMER and TRANSFORM and on the other hand iMC, Griffin, Nek5000, and Squirrel, is due to the different velocity profiles shown in Fig. 5. With the fully developed flow, the DNPs reside longer close to the fuel moderator interface compared to a uniform flow. Therefore, fewer DNPs are advected outside of the core. The effect of flow patterns in fuel channels is discussed in Tovar et al. (2024).

As a result of DNPs moving, neutrons are emitted in regions away from their origin. The shift of DNPs away from the center line of the core leads to the production of delayed neutrons in regions of lower importance. Outside of the core, the emitted neutrons do not contribute to the chain reaction. Both of these effects lead to a reduction in reactivity. In Table 23 the reactivity change due to the advection of DNPs relative to step 2.1 is shown for each participant. The codes largely agree on the loss in reactivity. SIMMER and TRANSFORM estimate the loss to be slightly higher than the other codes. Again, this difference is mainly a result of the different flow patterns. The results obtained by Squirrel, Griffin, iMC, and Nek5000 agree within 9 pcm. The most significant difference occurs for long out-of-core times, where the behavior of the short-lived DNPs dominates. For those DNPs, the flow near the fuel graphite interface has the largest impact, and small deviations in the fission rate density and the flow pattern have the largest impact.

In Fig. 10, the reactivity change due to the advection of DNPs relative to step 2.1 is shown for each participant and different out-of-core times. For iMC the error bars with one standard deviation are also shown. For longer out-of-core times, the reactivity loss increases since the DNPs stay out of the core for longer and have a higher likelihood to decay outside of the core. For short out-of-core times, the DNPs will reenter the core sooner. Therefore, fewer DNPs decay outside of the core, and the loss in reactivity is only due to the advection of the DNPs into core regions of lower importance.

All codes agree on the increased loss of reactivity for longer out-of-core times. The disagreement is maximal for long out-of-core times with 24 pcm. Longer out-of-core times additionally increase the difference between the codes since the flow pattern has a significant impact on the number of DNPs advected into the out-of-core region. With a fully developed flow pattern, the DNPs close to the wall stay in the core for longer, as a result, fewer DNPs leave the core. With a constant flow, the opposite is true, and more DNPs leave the core, resulting in a more significant difference for longer out-of-core times.

Step 2.3: Effect of a change in the inlet velocity on the reactivity

Changing the bulk velocity affects the energy balance and the advection of the DNPs. The reactivity change can be split into two components: the change in the temperature field and the DNPs flushing from the core and their return. With slower-flowing fuel, less heat is moved out of the core, and the salt temperature increases, causing an overall hotter reactor and a negative temperature feedback. Additionally, the slower fuel flow advects fewer DNPs, leading to a positive feedback. For faster flow rates, the loss of reactivity due to temperature feedback will be lower, while due to a stronger DNP advection, the loss of reactivity will be higher.

Table 24
Reactivity change at bulk velocity = 0.02 (ms^{-1}).

	SIMMER	TRANSFORM	Squirrel	Griffin	Nek5000	IMC	Average
$\Delta\rho$ (Only temperature)	−1551.1	−1571.92	−2091.64	−1882.88	−2065.5	−2268.87	−1905.32
Discrepancy	18.59%	17.5%	9.78%	1.18%	8.41%	19.08%	12.42%
$\Delta\rho$ (Only DNP)	204.1	194.71	189.68	186.5	190.86	184.01	191.64
Discrepancy	6.5%	1.6%	1.03%	2.68%	0.41%	3.98%	2.7%
$\Delta\rho$ (Total feedback)	−1347.0	−1377.38	−1901.96	−1696.37	−1874.64	−2084.86	−1713.7
Discrepancy	21.4%	19.63%	10.99%	1.01%	9.39%	21.66%	14.01%

Table 25
Reactivity change at bulk velocity = 0.02 (ms^{-1}) (without system codes).

	Squirrel	Griffin	Nek5000	IMC	Average
$\Delta\rho$ (Only temperature)	−2091.64	−1882.88	−2065.5	−2268.87	−2077.22
Discrepancy	0.69%	9.36%	0.56%	9.23%	4.96%
$\Delta\rho$ (Only DNP)	189.68	186.5	190.86	184.01	187.76
Discrepancy	1.02%	0.67%	1.65%	2.0%	1.33%
$\Delta\rho$ (Total feedback)	−1901.96	−1696.37	−1874.64	−2084.86	−1889.46
Discrepancy	0.66%	10.22%	0.78%	10.34%	5.5%

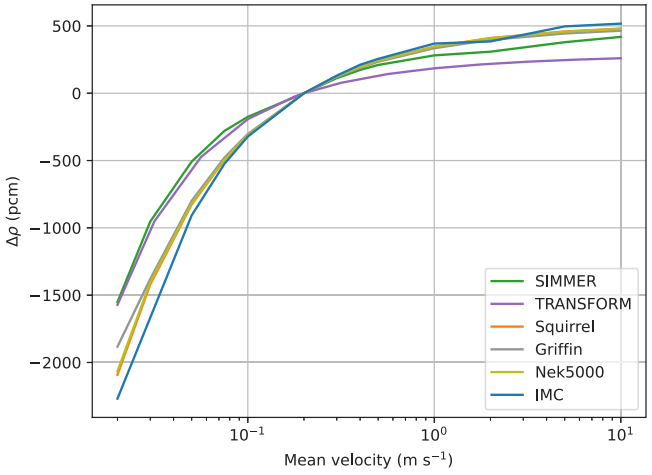

Fig. 11. Reactivity change due to temperature change $\Delta\rho = \rho_{2.3} - \rho_{2.2}$ for different fuel velocities.

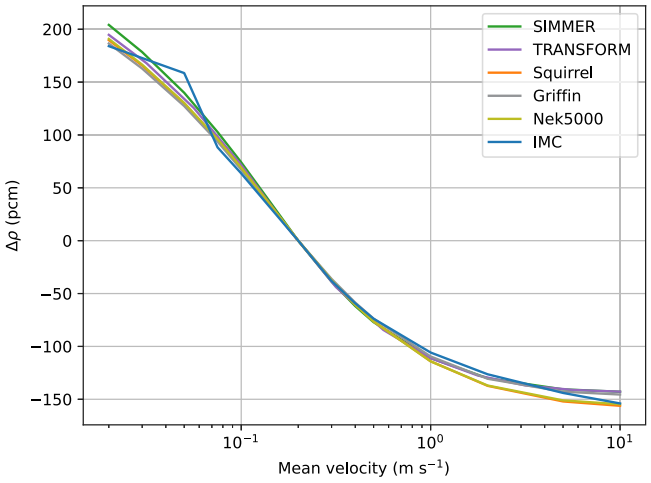

Fig. 12. Reactivity change due to DNP movement $\Delta\rho = \rho_{2.3} - \rho_{2.2}$ for different fuel velocities.

In Fig. 11, the temperature component of the reactivity change relative to step 2.2 is shown for each participant and different velocities. The codes agree that the loss of reactivity caused by the temperature feedback is lower for higher velocities. The coarse-mesh thermal-hydraulics codes SIMMER and TRANSFORM exhibit a weaker negative reactivity insertion at low flow velocities. This behavior arises from the implemented heat transfer correlation (Eq. (8)), which tends to overestimate heat transfer from the graphite to the fuel. As a result, the graphite remains at a lower temperature compared to the predictions of the CFD solvers under low-flow conditions. Conversely, at high flow velocities, TRANSFORM, in particular, deviates from the other codes as the graphite temperature approaches the reference temperature. Consequently, the graphite temperature feedback is relatively weak in this regime. Since Fig. 11 shows deviations from step 2.2, the observed magnitude is smaller due to the lower baseline case values, as presented in Table 19.

In Table 24 the reactivity change due to temperature effects, the DNP drift and the total effect are shown at a bulk velocity of 0.02 ms^{-1}. The reactivity change due to the change in temperature shows a larger discrepancy due to the differences in the heat transfer. In Table 25 the same data is shown for the fully resolved CFD codes. The average discrepancy between the fully resolved CFD codes in the reactivity due to the DNP drift went down to 1.33%, while the average discrepancy in the temperature feedback went down from 12.42% to 4.96%.

An increase in the fuel velocity increases the DNP advection out of the core and to regions of lower importance. At the same time, a faster flow reduces out-of-core time. These two phenomena have opposite effects. More DNPs are moved to regions of lower importance, but fewer DNPs decay outside of the core and reenter. Since the half-life of most DNPs is short relative to the out-of-core time, the reentering effect is considered to be small. In Fig. 12 the DNP component of the reactivity change relative to step 2.2 is shown for each participant at different velocities. The codes agree that the loss of reactivity caused by the DNP movement is higher for higher velocities. This effect is well captured by all codes and the average discrepancy between the codes is 2.7%.

In Fig. 13, the reactivity change due to the advection of DNPs and the change in temperature at different velocities relative to step 1.1 is shown for each participant and different velocities.

The overall agreement of the relative difference to a reference reactor is one of the main findings in this benchmark. Even though the codes disagree on the steady state results, the response to a perturbation seems to be similar. These results explain the good agreement of the codes in the ULOF transient (See step 3.1).

Step 2.4: Full reactor modeling

In the heat exchanger, the salt cools and then recirculates back into the core. The cooling of the salt depends on its temperature difference to a reference value (See. Eq. (1)). With a fixed power, a fixed velocity,

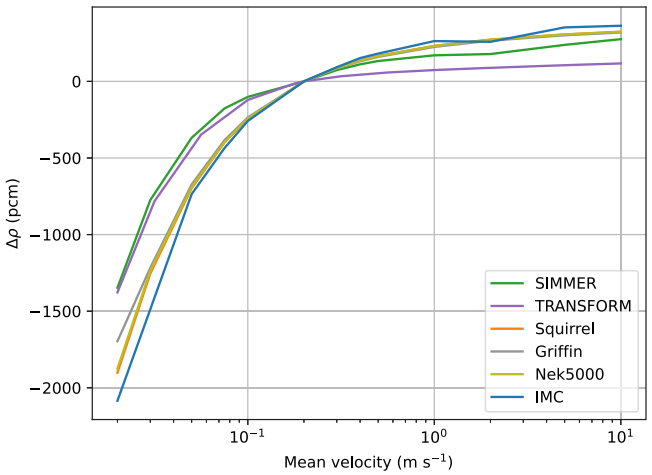

Fig. 13. Reactivity change $\Delta\rho = \rho_{2.3} - \rho_{1.1}$ for different fuel velocities.

Table 26
Steady state temperatures and reactivity change $\Delta\rho = \rho_{2.4} - \rho_{1.1}$.

	T inlet (K)	T outlet (K)	$\Delta\rho$	$\Delta\rho$ DNP	$\Delta\rho$ temperature
SIMMER	1064.2	1106.4	−2182	−275	−1907
TRANSFORM	1064.7	1107.1	−2048	−276	−1772
Squirrel	1064.7	1107.1	−2255	−260	−1995
Griffin	1064.3	1106.5	−2432	−273	−2158
iMC	1065.9	1108.6	−2362 ± 7	−266 ± 7	−2096 ± 7
Nek5000	1063.0	1105.4	−2237	−262	−1975

and a heat exchanger, the temperature of the inlet and the outlet are well defined through the energy balance. This step ensures that the different codes have been correctly implemented in the heat exchanger.

In Table 26, the inlet and outlet temperature and the change in reactivity for a constant power are shown. In addition, the contribution of the temperature feedback and the DNP contribution is shown. The reason for the disagreement of the Nek5000 results compared to the other results is due to the ex-core modeling. Since the ex-core model is an extension of the core fuel channel so as to achieve the reference ex-core residence time, the transition between Pouiseuille flow and plug flow and mixing of the temperature in the ex-core region are not immediate. Thus, hotter fuel, originally located near the fuel/moderator boundary, experiences longer residence time than colder fuel in the channel center, leading to more effective cooling action of the heat exchanger overall. This explains why the core inlet temperature (i.e., ex-core outlet temperature) is lower using the Nek5000 model than models with 2D core and 1D ex-core thermal-hydraulic modeling, such as Squirrel or TRANSFORM.

All codes ran a null transient and confirmed that an actual steady state with full power coupling could be simulated. This null transient is the initial state for the next step.

Phase 3: Full transient coupling

Step 3.1: Unprotected loss of flow

During a ULOF event, the reactor power evolves freely as the flow rate changes, without any active measures taken to control the reactor. The feedback mechanisms within the reactor should maintain stability during the transient period until natural circulation is established, allowing the reactor to stabilize at a lower power output.

In an MSR, a reduction in fuel flow results in two primary effects: reduced reactor cooling from the fuel coolant salt and the DNPs being advected at a lower velocity. Both of these factors influence the reactor power level. During the ULOF, the velocity in the out-of-core region reduces at the same rate as inside the core. For the present, no other changes are assumed, such as an intervention by the operator or a change in the secondary temperature T_H. This simplified model of the out-of-core heat exchanger neglects two effects that are present in more realistic settings during a ULOF event. By assuming a constant heat transfer coefficient in the heat exchanger (Eq. (1)) and the secondary side of the heat exchanger being modeled as an infinite heat sink, with a constant temperature, the cooling of the primary salt is overestimated. This leads to an underestimation of the recirculating fuel temperature. With colder fuel entering the core, a positive reactivity feedback can be observed. This effect might be less dominant with a fully modeled secondary loop or actions taken by an operator.

The reduced fuel flow will lead to a longer residence time of the fuel inside the core, increasing fuel and graphite temperature, and reducing the reactivity. While the fuel temperature changes fast, the large thermal mass and the relatively low amount of heating in the graphite make the temperature change of the moderator slow. The reduced advection of the DNPs increases the reactivity, since fewer DNPs are advected out of the core. This leads to a positive reactivity insertion since more DNPs are present in the core. All these effects happen on different time scales and need to be correctly accounted for.

The power evolution during the ULOF is shown in Fig. 14. Initially, as the flow rate decreases, fewer DNPs are advected with the fuel, leading to an increase in the power level. However, this rise in power, coupled with a slowdown in fuel flow, causes the reactor temperature to rise. This leads to a peak in power output, which then rapidly declines to a minimum approximately 50 s after the transient begins. The temperature of the fuel entering the core drops due to a longer out-of-core time and reduced power. As a result, the reactor power increases until around 120 s, although it does not exceed the initial power level, except for SIMMER. Following this peak, power output continuously decreases due to the increasing graphite temperature.

All codes agree on the initial increase in power due to the return of the DNPs. The difference in the maximum power is 11 kW. The differences can be explained by the different return patterns of the DNPs caused by the differences in the modeling of the fuel flow in the channel and the outer core. The velocity profile collapse is faster than the bulk velocity decrease, thus, more DNPs are retained in the active core. Therefore, the codes simulating the bulk flow predict a lower power peak. Also, the reduction in power due to the heating of the fuel is captured by all codes. The codes differ on the amplitude of the reduction by 25 kW. This is mainly due to differences in the temperature feedback and the fuel temperature. All codes agree on the rise at 50 s after the initiation of the transient caused by the reentry of colder fuel. The long-term behavior is dominated by the slow heating of the moderator. Step 1.2 demonstrates the differences in the heat conduction and the steady-state temperature distribution. These differences and the difference in the temperature feedback can explain the behavior here. The underestimate of the power evolution of Griffin can be explained by the differences in the temperature feedback observed in step 1.1.

The increase in outlet temperature is depicted in Fig. 15. The outlet temperature reaches its maximum after 100 s and subsequently decreases due to the reduction in reactor power and colder fuel reentering the core. All codes agree on the rate the temperature is increasing and the time at which the maximum is reached. SIMMER and TRANSFORM estimate a much higher outlet temperature. The other codes agree much more on the time development. Griffin estimates a lower power, compared to Squirrel and Nek5000 and subsequently has a lower outlet temperature.

From a safety perspective, the material temperature is a relevant aspect of this transient. At certain temperatures, the structural integrity of the reactor cannot be ensured. Monitoring the maximum temperature of the graphite is necessary to ensure that the reactor is able to withstand this transient. In Fig. 16, the maximum moderator temperature is shown. The moderator temperature increases near the fuel channel due to heat transfer from the fuel and increased direct

Table 27
Transient results after 1000 s.

	SIMMER	TRANSFORM	Squirrel	Griffin	Nek5000	Average
Power (W)	313 088.19	302 228.53	265 426.63	260 274.41	266 756.77	281 554.91
Discrepancy	11.2%	7.34%	5.73%	7.56%	5.26%	7.42%
Δ max moderator temperature (K)	9.98	44.66	76.2	70.4	78.47	55.94
Discrepancy	82.17%	20.16%	36.21%	25.85%	40.27%	40.93%
Δ outlet temperature (K)	191.19	180.77	118.98	112.83	121.18	144.99
Discrepancy	31.86%	24.68%	17.94%	22.18%	16.42%	22.62%

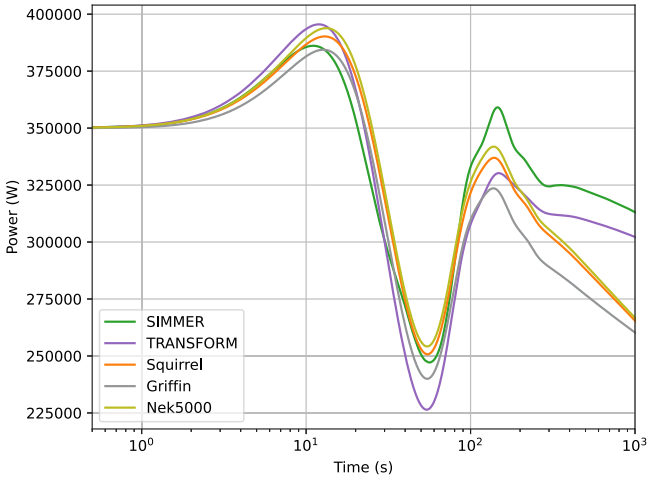

Fig. 14. Power evolution during the ULOF.

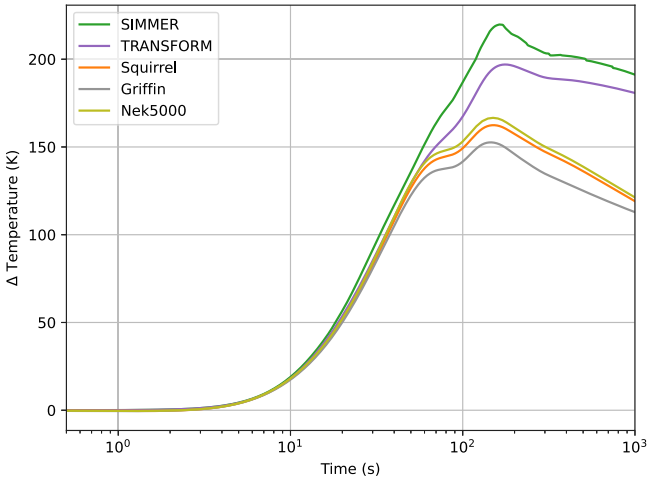

Fig. 15. Change of the outlet temperature evolution during the ULOF.

heating. The location of the maximum temperature at steady state is furthest from the fuel channel. This results in an initial plateau in the maximum graphite temperature. After 30 s, the graphite temperature starts to rise for the fine thermal-hydraulics solvers. The coarser nodal models estimate the rise to occur later.

In Table 27 the difference in power, outlet temperature, and maximal moderator temperature is shown. The average discrepancy in power is 7.42%. This difference is mainly caused by SIMMER and TRANSFORM. Without these codes, the average discrepancy in power is 0.98% shown in Table 28.

6. Conclusions and future work

This paper presents a benchmark for thermal MSRs intended to test the performance of the multi-physics codes. The chosen reactor

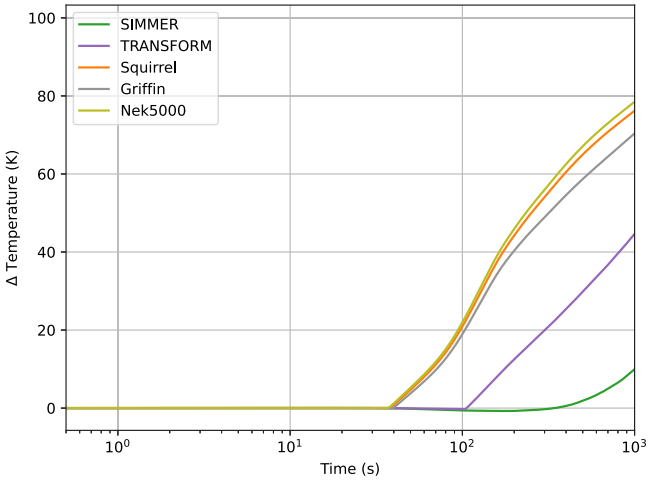

Fig. 16. Change of the maximum temperature of the moderator during the ULOF.

Table 28
Transient results after 1000 s (without system codes).

	Squirrel	Griffin	Nek5000	Average
Power (W)	265 426.63	260 274.41	266 756.77	264 152.6
Discrepancy	0.48%	1.47%	0.99%	0.98%
Δ max moderator temperature (K)	76.2	70.4	78.47	75.02
Discrepancy	1.57%	6.16%	4.59%	4.11%
Δ outlet temperature (K)	118.98	112.83	121.18	117.67
Discrepancy	1.12%	4.11%	2.99%	2.74%

model is a graphite-moderated slab reactor with an out-of-core model, allowing for the analysis of the strong coupling of thermal hydraulics and neutronics with moving DNPs inside and outside of the core.

Four different groups of codes were benchmarked:

- The coarse-mesh thermal-hydraulics codes, SIMMER and TRANSFORM, utilizing two-group neutron transport and modified point kinetics for the neutronics, respectively.
- The MOOSE-based modules Squirrel and Griffin, coupled with the thermal-hydraulics module, evaluate the neutronics with modified point kinetics and two-group diffusion.
- The high fidelity Monte Carlo code iMC coupled with OpenFOAM.
- The spectral element CFD code Nek5000 with a modified point kinetics code.

The stepwise increase in complexity of the benchmark allows for a comparison of the codes and to assess model limitations. In step 1.1, the agreement was generally found to be good, and differences were only reported for the reactivity coefficients. The agreement on the temperature field calculated in step 1.2 was good for the high-fidelity codes that used conjugate heat transfer. The codes (TRANSFORM and SIMMER), which use the correlation formulated in terms of the Graetz

number, did not fully resolve the same temperature field as the high-fidelity codes. This discrepancy also affects the calculated change in reactivity in step 2.1. The loss of reactivity caused by DNP transport, calculated in step 2.2, was mainly affected by the velocity field in the reactor core. In step 2.3, the change in reactivity as a function of the velocity field was evaluated. The results agreed on the trend of the change in reactivity, but not on the magnitude. All participating codes found steady-state solutions in step 2.4, with a good agreement on the loss of reactivity. The ULOF transient, step 3.1, started from an initially steady-state reactor. The resulting power, outlet temperature, and maximum graphite temperature were measured.

The solvers utilizing a modified point kinetics show good agreement with higher-fidelity neutronics models. Disagreements, however, are observed for different thermal-hydraulics solvers, independent of the neutronics solver applied. The simulated flow pattern has a strong impact on the loss of reactivity due to DNP advection. The differences in the initial temperature feedback in step 2.1 can be explained by the different modeling of the conjugate heat transfer, which causes differences in the temperature feedback.

One of the major findings was that even though the participating codes had different modeling assumptions and, therefore, varying single physics results, the agreement on the transient simulation is good. This implies that a good agreement of transient results does not guarantee good agreement on the single physics results. The observations made here leave an open question on how much confidence in the ability of the code to describe such a reactor can be drawn from a full-power transient. This is relevant when considering the verification of codes on the experimental results of the Molten Salt Reactor Experiment. It was also shown that small differences in the modeling assumptions have a large impact on the results of steady-state physics, especially on the temperature and the corresponding change in reactivity.

In summary, this benchmark allows the testing of different models ranging from a coarse thermal-hydraulics mesh coupled to a point kinetics solver, to a high-fidelity thermal-hydraulics coupled with a Monte Carlo solver. This enables the testing of high- and low-fidelity codes at steady state and during transients. The benchmark is accessible for both legacy and newly developed codes, improving confidence in the modeling of thermal MSR.

Future work will include the modeling of a secondary loop, showing the impact the secondary loop has on long-term transients. The model can be expanded to account for more transient phenomena. One would be xenon poisoning and its removal, including possible accidents in the removal system. Another relevant transient would be an overcooling or a refueling accident, increasing the reactivity due to colder or more fissile salt entering the reactor from the bottom. Additional investigation on the effect of the modeling assumptions on the transients can be done. This includes the effect of buoyancy and of turbulent flow. The latter is particularly interesting, since the heat transfer with the graphite is coupled with the velocity of the salt.

CRediT authorship contribution statement

P. Pfahl: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **B. Kędzierska:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **I. Kim:** Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation. **A. Chambon:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Formal analysis, Data curation, Conceptualization. **L. Fischer:** Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation. **L. Bureš:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Methodology, Investigation, Formal analysis, Data curation. **J. Groth-Jensen:** Writing – review & editing, Validation, Supervision, Software, Investigation, Formal analysis. **Y. Kim:** Writing – review & editing, Validation, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization. **A. Rineiski:** Writing – review & editing, Validation, Supervision, Software, Resources, Project administration, Formal analysis. **B. Lauritzen:** Writing –review & editing, Supervision, Resources, Project administration, Conceptualization.

Declaration of competing interest

Lubomír Bureš, Jacob Groth-Jensen, and Lorenz Fischer declare a competing interest due to affiliation in the form of employment with Saltfoss Energy (former Seaborg Technologies), which is a company actively involved in the development of molten salt reactor technology.

Acknowledgments

Research in this publication made use of the following Idaho National Laboratory (INL) software: Griffin. This software is licensed by Battelle Energy Alliance, LLC (BEA), Managing & Operating Contractor of INL. A software license may be available by contacting INL at agradmin@inl.gov. This software is experimental in nature. BEA and the U.S. Government make no representations; extend no warranties; and disclaim any liability for the use of this software or software outputs in this publication. Use of this software is not an endorsement by BEA or the U.S. Government.

Philip Pfahl would also like to acknowledge Saltfoss Energy for their financial support and the granted cluster access. He would also like to thank the Korea Advanced Institute of Science and Technology, especially Yonghee Kim, for hosting him during the time the benchmark was developed. Barbara Kędzierska would like to express gratitude to the JAEA for the provision of SIMMER under a JAEA-KIT/CEA international cooperative agreement.

Lubomír Bureš would like to acknowledge the Nek5000 User Group and especially Prof. Paul Fischer for their support regarding Nek5000-related queries.

Data availability

The date presented in figures is shared on: https://github.com/philipJFpfahl/Thermal_Molten_Salt_Reactor_Benchmark.

References

Abou-Jaoude, A., et al., 2021. A workflow leveraging MOOSE transient multiphysics simulations to evaluate the impact of thermophysical property uncertainties on molten-salt reactors. Ann. Nucl. Energy 163, 108546, https://linkinghub.elsevier.com/retrieve/pii/S0306454921004229.

Akcasuh, Z., 2012. Mathematical Methods in Nuclear Reactor Dynamics. Elsevier.

Babcsány, B., et al., 2023. Results and lessons learned from the Generation IV SCWR-FQT comprehensive Monte Carlo computational benchmark. Ann. Nucl. Energy 191, 109903.

Balay, S., et al., 2001. PETSc. See http://www.mcs.anl.gov/petsc.

Bennett, T.D., 2019. Correlations for the Graetz problem in convection – Part 1: For round pipes and parallel plates. Int. J. Heat Mass Transfer 136, 832–841, https://www.sciencedirect.com/science/article/pii/S0017931018351925.

Bohl, W.R., et al., 1990. AFDM: An Advanced Fluid-Dynamics Model. Tech. rep. LA-11692-MS-Vol.1, Los Alamos National Lab., NM (USA).

Brovchenko, M., et al., 2013. Design-related studies for the preliminary safety assessment of the molten salt fast reactor. Nucl. Sci. Eng. 175 (3), 329–339.

Buckel, G., Hesselschwerdt, E., Kiefhaber, E., Kleinheins, S., Maschek, W., 1999. A New SIMMER-III Version with Improved Neutronics Solution Algorithms. tech. rep. Report FZKA 6290, Forschungszentrum Karlsruhe, INIS Reference Number: 30052277.

Cetiner, S.M., et al., 2016. Nuclear Hybrid Energy Systems FY16 Modeling Efforts at ORNL. tech. rep., Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States).

Chadwick, M., Herman, M., Obložinský, P., et al., 2011. ENDF/B-VII.1 nuclear data for science and technology: Cross sections, covariances, fission product yields and decay data. Nucl. Data Sheets 112 (12), 2887–2996, Special Issue on ENDF/B-VII.1 Library. https://www.sciencedirect.com/science/article/pii/S009037521100113X.

Chaudri, K.S., Mirza, S.M., 2015. Burnup dependent Monte Carlo neutron physics calculations of IAEA MTR benchmark. Prog. Nucl. Energy 81, 43–52.

Clarno, K.T., et al., 2023. Evaluation of SCALE, Serpent, and MCNP for Molten Salt Reactor applications using the MSRE Benchmark. Ann. Nucl. Energy 194, 110092.

Delpech, M., et al., 2003a. Benchmark of dynamic simulation tools for molten salt reactors. In: GLOBAL 2003 - Nuclear Science and Technology : Meeting the Global Industrial and R&D Challenges of the 21st Century. American Nuclear Society, p. 2182, https://in2p3.hal.science/in2p3-00020303.

Delpech, M., et al., 2003b. Benchmark of dynamic simulation tools for molten salt reactors. In: Proceedings of the International Conference GLOBAL. pp. 2182–2187.

Diamond, D.J., Brown, N.R., Denning, R., Bajorek, S., 2018. Phenomena Important in Molten Salt Reactor Simulations. tech. rep., Brookhaven National Lab.(BNL), Upton, NY (United States).

Diniz, R.C., da Rosa, F.S.d.S., da Cruz Gonçalves, A., 2022. Calculation of delayed neutron precursors' transit time in the external loop during a flow velocity transient in a Molten Salt Reactors. Ann. Nucl. Energy 165, 108640.

Fei, T., et al., 2024. MSR transient simulation and MSRE transient benchmark with SAM and SPECTRA. Nucl. Technol. 1–17.

Fiorina, C., et al., 2012. Analysis of the MSFR core neutronics adopting different neutron transport models. In: International Conference on Nuclear Engineering. Vol. 44991, American Society of Mechanical Engineers, pp. 219–228.

Fiorina, C., et al., 2014. Modelling and analysis of the MSFR transient behaviour. Ann. Nucl. Energy 64, 485–498.

Fischer, L., Bureš, L., 2024. Application of Modelica/TRANSFORM to system modeling of the molten salt reactor experiment. Nucl. Eng. Des. 416, 112768, https://www.sciencedirect.com/science/article/pii/S0029549323006179.

Fischer, P., Lottes, J., Tufo, H., 2007. Nek5000. https://www.osti.gov/biblio/1231166.

Gaston, D.R., et al., 2015. Physics-based multiscale coupling for full core nuclear reactor simulation. Ann. Nucl. Energy 84, 45–54.

Geuzaine, C., Remacle, J.-F., 2008. Gmsh: a three-dimensional finite element mesh generator with built-in pre-and post-processing facilities (2008). URL: http://www.geuz.org/gmsh.

Giudicelli, G., et al., 2024. 3.0 - MOOSE: Enabling massively parallel multiphysics simulations. SoftwareX 26, 101690, https://www.sciencedirect.com/science/article/pii/S235271102400061X.

Graetz, L., 1882. Ueber die wärmeleitungsfähigkeit von flüssigkeiten. Ann. Phys., Lpz. 254 (1), 79–94.

Greenwood, M.S., 2017. TRANSFORM - TRANsient simulation framework of reconfigurable models. Web. https://github.com/ORNL-Modelica/TRANSFORM-Library.

Greenwood, M.S., Betzler, B.R., Qualls, A.L., Yoo, J., Rabiti, C., 2020. Demonstration of the advanced dynamic system modeling tool TRANSFORM in a molten salt reactor application via a model of the molten salt demonstration reactor. Nucl. Technol. 206 (3), 478–504.

Groth-Jensen, J., et al., 2021. Verification of multiphysics coupling techniques for modeling of molten salt reactors. Ann. Nucl. Energy 164, 108578, https://www.sciencedirect.com/science/article/pii/S0306454921004540.

Hansel, J., Andrs, D., Charlot, L., Giudicelli, G., 2024. The MOOSE thermal hydraulics module. J. Open Source Softw. 9 (94), 6146, https://doi.org/10.21105/joss.06146.

Haubenreich, P.N., Engel, J., 1970. Experience with the molten-salt reactor experiment. Nucl. Appl. Technol. 8 (2), 118–136.

Haubenreich, P., Engel, J., Prince, B., Claiborne, H., 1964. MSRE Design and Operations Report. Part III. Nuclear Analysis. tech. rep., Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States).

Jaradat, M.K., Choi, N., Abou-Jaoude, A., 2024. Verification of Griffin-Pronghorn-coupled multiphysics code system against CNRS molten salt reactor benchmark. Nucl. Sci. Eng. 1–34.

Kędzierska, B., Rineiski, A., Tobita, Y., 2025. Modified algorithm for taking into account delayed neutron precursor movement in the improved quasi-static scheme of the SIMMER-III code. In: International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering. M&C 2025, Denver, CO.

Kim, H., Kim, Y., 2021. Unstructured mesh-based neutronics and thermomechanics coupled steady-state analysis on advanced three-dimensional fuel elements with Monte Carlo code iMC. Nucl. Sci. Eng. 195.

Kirk, B.S., Peterson, J.W., Stogner, R.H., Carey, G.F., 2006. libMesh: a C++ library for parallel adaptive mesh refinement/coarsening simulations. Eng. Comput. 22, 237–254.

Kreher, M., Forget, B., Smith, K., 2019. Single-batch Monte Carlo multiphysics coupling. In: Proc. M&C.

Labossiere-Hickman, T.J., Forget, B., 2017. Selected VERA core physics benchmarks in OpenMC. Trans. Am. Nucl. Soc. 117, 1520–1523.

Laureau, A., et al., 2017. Transient coupled calculations of the Molten Salt Fast Reactor using the transient fission matrix approach. Nucl. Eng. Des. 316, 112–124.

Laureau, A., et al., 2022. Unmoderated molten salt reactors design optimisation for power stability. Ann. Nucl. Energy 177, 109265.

LeBlanc, D., 2010. Molten salt reactors: A new beginning for an old idea. Nucl. Eng. Des. 240 (6), 1644–1656.

Leppänen, J., Valtavirta, V., Rintala, A., Tuominen, R., 2025. Status of Serpent Monte Carlo code in 2024. EPJ Nucl. Sci. Technol. 11, 3.

Lindsay, A., et al., 2023. MOOSE Navier–Stokes module. SoftwareX 23, 101503.

Mattioli, A.S., Fiorina, C., Lorenzi, S., Cammi, A., et al., 2021. Derivation and implementation in OpenFOAM of a point-kinetics model for Molten Salt Reactors. Trans. Am. Nucl. Soc. 125 (1), 448–451.

Morita, K., Fischer, E.A., 1996. Simplified Analytical Equation-of-State Model for SIMMER-III ver. 2. A. tech. rep., Oarai Engineering Center, Ibaraki, Japan, p. 22.

Novak, A., et al., 2022. Coupled Monte Carlo and thermal-fluid modeling of high temperature gas reactors using cardinal. Ann. Nucl. Energy 177, 109310, https://www.sciencedirect.com/science/article/pii/S0306454922003450.

O'Dell, R.D., 1977. Standard Interface Files and Procedures for Reactor Physics Codes. Version IV. LA-6941-MS, Los Alamos National Lab. (LANL), Los Alamos, NM (United States), https://www.osti.gov/biblio/5369298.

Pfahl, P., Chambon, A., Groth-Jensen, J., Lauritzen, B., 2025. Squirrel; a MOOSE based APP for solving point kinetics in molten salt reactors. Nucl. Sci. Eng..

Philip Pfahl, 2025. Thermal Molten Salt Reactor benchmark results: https://github.com/philipJPfahl/Thermal_Molten_Salt_Reactor_Benchmark.

Ponce Tovar, M.A., 2021. Multiphysics Simulation and Verification of Molten Salt Reactors using GeN-Foam (MA thesis). Universitat Politècnica de Catalunya.

Poppendiek, H.F., Palmer, L., 1954. Forced Convetion Heat Transfer between Parallel Plates and in Annuli with Volume Heat Sources within the Fluids. tech. rep. ORNL-1701, Oak Ridge National Laboratory.

Rader, J., Smith, M., Greenwood, S., Harrison, T., 2019. Nuclear Thermal Propulsion Dynamic Modeling with Modelica. Oak Ridge National Laboratory (ORNL), Oak Ridge, TN (United States), https://www.osti.gov/biblio/1543223.

Rineiski, A., Sinitsa, V., Maschek, W., Wang, S., 2005. Kinetics and cross-section developments for analyses of reactor transmutation concepts with SIMMER. In: M&C 2005: International Topical Meeting on Mathematics and Computation, Supercomputing, Reactor Physics and Nuclear and Biological Applications. SFEN, France, INIS Reference Number: 40084930.

Romano, P.K., et al., 2015a. OpenMC: A state-of-the-art Monte Carlo code for research and development. Ann. Nucl. Energy 82, 90–97, Joint International Conference on Supercomputing in Nuclear Applications and Monte Carlo 2013, SNA + MC 2013. Pluri- and Trans-disciplinarity, Towards New Modeling and Numerical Simulation Paradigms. https://www.sciencedirect.com/science/article/pii/S030645491400379X.

Romano, P.K., et al., 2015b. OpenMC: A state-of-the-art Monte Carlo code for research and development. Ann. Nucl. Energy 82, 90–97.

SAMOFAR, 2025. SAMOFAR: http://samofar.eu/.

SAMOSAFER, SAMOSAFER: https://samosafer.eu/.

Shen, D., Ilas, G., Powers, J.J., Fratoni, M., 2021. Reactor physics benchmark of the first criticality in the molten salt reactor experiment. Nucl. Sci. Eng. 195 (8), 825–837.

Shen, D., et al., 2018. Zero-power criticality benchmark evaluation of the molten salt reactor experiment. In: Proceedings of the International Conference on Physics of Reactors. PHYSOR, Vol. 4012.

Singh, V., Wheeler, A.M., Lish, M.R., Chvála, O., Upadhyaya, B.R., 2018. Nonlinear dynamic model of Molten-Salt Reactor Experiment–Validation and operational analysis. Ann. Nucl. Energy 113, 177–193.

Steffy, Jr., R., 1970. Experimental Dynamic Analysis of MSRE with ^{233}U Fuel. tech. rep., Oak Ridge National Lab..

Tiberga, M., et al., 2020. Results from a multi-physics numerical benchmark for codes dedicated to molten salt fast reactors. Ann. Nucl. Energy 142, 107428, https://www.sciencedirect.com/science/article/pii/S0306454920301262.

Tovar, M.P., Fischer, L., Bureš, L., 2024. On the impact of fuel velocity profile on loss of delayed neutron precursors in the molten salt reactor experiment. In: Proceedings of the International Conference on Physics of Reactors. PHYSOR 2024, Seaborg Technologies, San Francisco, CA, pp. 1050–1059.

Tuominen, R., Valtavirta, V., Leppänen, J., 2019. New energy deposition treatment in the Serpent2 Monte-Carlo transport code. Ann. Nucl. Energy 129, 224–232.

Vieira, T., Carvalho, Y., Silva, G., et al., 2024. Steady state coupled calculations (Serpent-GeN-FOAM) applied to Molten Salt Fast Reactor (MSFR). In: Proceedings of the INAC 2024: International Nuclear Atlantic Conference. Nuclear Energy: Assuring Energy, Health and Food.

Wang, Y., et al., 2025. Griffin: A MOOSE-based reactor physics application for multiphysics simulation of advanced nuclear reactors. Ann. Nucl. Energy 211, 110917.

Weller, H.G., Tabor, G., Jasak, H., Fureby, C., 1998. A tensorial approach to computational continuum mechanics using object-oriented techniques. Comput. Phys. 12 (6), 620–631.

White, F.M., Majdalani, J., 2006. Viscous Fluid Flow, vol. 3, McGraw-Hill New York.

Yamaji, B., Aszódi, A., Kovács, M., Csom, G., 2014. Thermal–hydraulic analyses and experimental modelling of MSFR. Ann. Nucl. Energy 64, 457–471.

Yamano, H., et al., 2003. SIMMER-III: A Computer Program for LMFR Core Disruptive Accident Analysis. JNC TN9400 2003-071, O-arai Engineering Center, Japan Nuclear Cycle Development Institute.