Applying the Serpent-DYN3D Code Sequence for the Decay Heat Analysis of Metallic Fuel Sodium Fast Reactor

Annals of Nuclear Energy 125(2019), 291-306

DOI: https://doi.org/10.1016/j.anucene.2018.11.020

Applying the Serpent-DYN3D Code Sequence for the Decay Heat Analysis of Metallic Fuel Sodium Fast Reactor

G. Pereira[1], A. E. Johnson[1], Y. Bilodid[2], E. Fridman[2], D. Kotlyar[1]

[1]*Georgia Institute of Technology, George W. Woodruff School, Nuclear and Radiological Engineering, Atlanta, USA*

[2] *Helmholtz-Zentrum Dresden-Rossendorf, Bautzner Landstraße 400, 01328 Dresden, Germany*

Keywords:
Decay heat; cross section generation; microscopic depletion; DYN3D;

Abstract

Recent developments of the reactor dynamics code DYN3D have introduced the micro-depletion model which allows for explicit calculation of radioactive decay heat. Such a unique combination of nodal diffusion, thermal hydraulic (T/H) and depletion solvers allows DYN3D to perform fuel cycle depletion and obtain detailed core isotopic concentration and decay heat distributions. The new sequence utilizes considerably less computational resources than coupled Monte Carlo-T/H-depletion systems, but with comparable accuracy. This capability was recently tested on a limited number of simple unit cell models. The main objective of this work is to further verify the decay heat calculation capabilities of DYN3D by applying it to a considerably more realistic and detailed full core model. For the purpose of the current analysis a 3D full core model of Advanced Burner Reactor (ABR) was adopted from the OECD/NEA Benchmark for Neutronic Analysis of Sodium-Cooled Fast Reactor Cores with Various Fuel Types and Core Sizes. In this work, the Monte Carlo code Serpent was used to generate macro- and microscopic parameters, and the neutron diffusion code, DYN3D, was used to perform neutronic and depletion analyses. Detailed spatial isotopic and decay heat distributions obtained with DYN3D were verified against the equivalent Serpent reference 3D full core solution. Results indicate very good agreement between the Serpent-

DYN3D code sequence and the reference Serpent solutions, with a discrepancy in total decay heat on the order of 0.5%.

1 Introduction

The ability to determine the decay heat produced by nuclear fuel after reactor shutdown is of extreme importance for analyses of accident scenarios in which heat removal systems might be compromised. It is also important for designing storage and transportation solutions for spent fuel. In Light Water Reactors (LWRs), the decay heat generated by the fuel after shutdown can be predicted with the help of the decay heat standards such as American National Standard for Decay Heat Power in Light Water Reactors issued by American Nuclear Society (ANSI/ANS, 2005). However, in the case of advanced reactors with non-conventional fuels, using such standardized calculation procedure can lead to large discrepancies in decay heat calculations. Alternative approaches must be capable of appropriately accounting for fuel composition and neutron energy spectrum (Shwageraus and Hejzlar, 2009).

DYN3D (Rohde et al., 2016) is a 3D nodal reactor dynamic code owned by Helmholtz-Zentrum Dresden-Rossendorf (HZDR) in Germany. The code was initially developed for the analysis of fuel cycles and transient scenarios in LWRs but its functionality has been extended to SFRs applications (Nikitin and Fridman, 2018a; 2018b; 2018c). Recently, DYN3D has been enhanced with the capability to perform decay heat calculations based on the explicit nuclide content which is free from approximations and limitations of the standard decay heat curves approach (Bilodid et al., 2018). A detailed isotopic composition is calculated in each spatial position (*i.e.*, node) using local neutron fluxes. Local operational conditions and nuclide content are used to obtain the interpolated few-group macroscopic and microscopic cross-sections. This newly implemented hybrid microscopic depletion approach (Bilodid et al., 2016) allows to obtain a fine resolution of microscopic parameters (*e.g.*, concentrations) with only modest computational requirements.

To the best of our knowledge, such unique capabilities were implemented only in DYN3D. However, the applicability of the method was tested only on a number of simple fuel pin and assembly models with different material compositions, operating history, and neutron spectra. Therefore, the main objective of this work is to further verify the decay heat

calculation capabilities of DYN3D by applying it to a considerably more realistic and detailed full core model.

For this purpose, we considered a 3D full core model of the Advanced Burner Reactor (ABR) adopted from the OECD/NEA Benchmark for Neutronic Analysis of Sodium-cooled Fast Reactor (SFR) Cores with Various Fuel Types and Core Sizes (NEA, 2016). The ABR core was modeled with DYN3D using a set of macroscopic and microscopic cross-sections generated by the Monte Carlo (MC) code Serpent (Leppänen et al., 2015). The performance of DYN3D was assessed by comparison to the reference full-core solution obtained with Serpent. The results indicate that detailed isotopic concentration and decay heat distributions are efficiently and accurately predicted by DYN3D.

This paper is structured as follows. A detailed description of the ABR core is presented in the next section. Section 3 provides an overview of the micro-depletion and decay heat calculation methodology implemented in DYN3D. Section 4 describes the process by which macro- and microscopic quantities were generated. This section also presents the adopted assumptions and simplifications. In Section 5, the Serpent-DYN3D code sequence is applied to decay heat analysis of 2D infinite lattice, 3D fuel assembly, and the full core model. Finally, Section 6 presents the summary of the work done and a set of conclusions.

2 Description of the reference ABR core

The reference core for this study is the sodium-cooled, metal fuel Advanced Burner Reactor (ABR), a 1000 MW_{th} fast reactor design proposed by Argonne National Laboratory (Kim et al., 2009), and included into the OECD/NEA Benchmark for Neutronic Analysis of SFR Cores with Various Fuel Types and Core Sizes (NEA, 2016). The ABR core is a compact core concept with transuranic (TRU) conversion ratio of ~0.7, with an intended operational cycle duration of 1 year at 90% capacity. This reactor design allows using TRU from LWRs as part of the fuel. Fig. 1 provides an overview of the radial distribution of the ABR core. The core consists of 2 fuel regions, an inner core region containing 78 fuel assemblies, and an outer core region with 102 fuel assemblies. Each fuel assembly, shown in Fig. 2, is subdivided into 5 axial layers with unique isotopic composition in each layer. The core also contains 19 control assemblies, 114 radial reflector assemblies, and 66 radial shield assemblies. Some of the key operating parameters and core dimensions of the ABR are

summarized in Table 1. The reader is referred to (NEA, 2016) for a full description of the reactor.

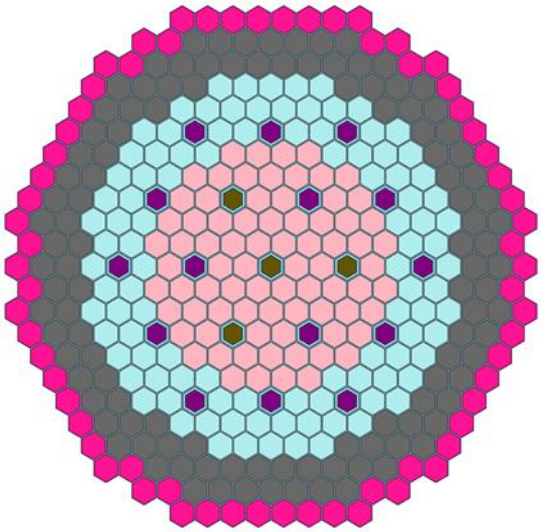


Fig. 1 Radial core layout of ABR Metallic-Fuel Core **(**Inner Core – light pink, Outer Core – light blue, Radial Reflector – dark grey, Radial Shield – dark pink, Primary Control Assemblies – purple, Secondary Control Assemblies – olive green).

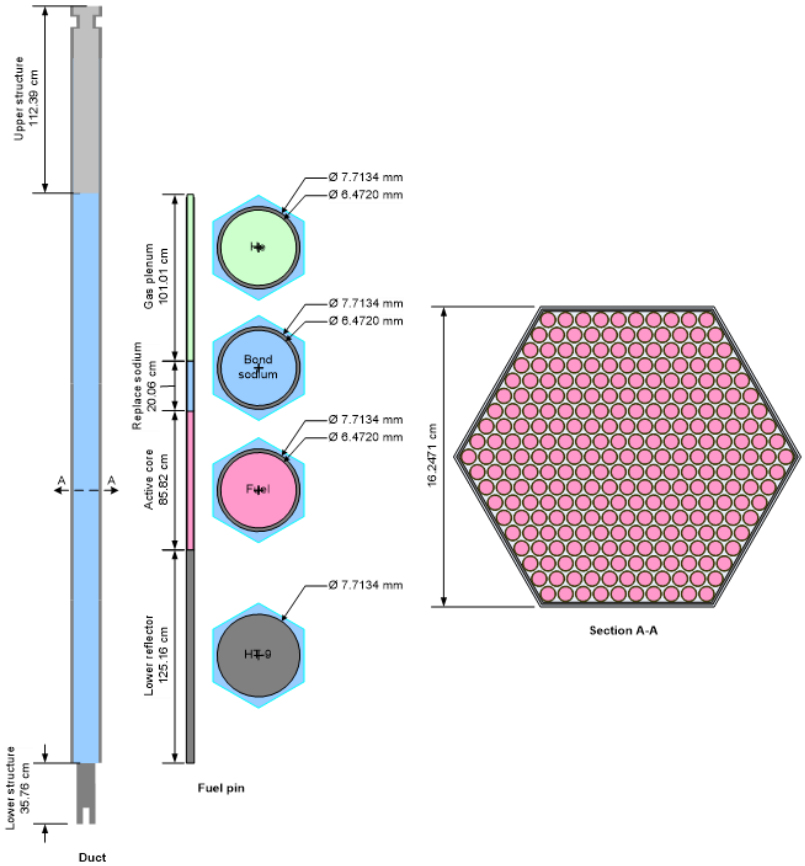


Fig. 2 Axial layout of ABR fuel assembly (NEA, 2016)**.**

Table 1: Operating parameters and key dimensions.

Parameter	**Value**
Power (MW_{th})	1000
Active core height (cm)	85.82
Total core height (cm)	480.20
Coolant temperature (^{o}C)	432.50
Average core structural temperature (^{o}C)	432.50
Average metallic fuel temperature (^{o}C)	534.00
Number of fuel pins	271
Subassembly pitch (cm)	16.2471

3 Micro-depletion and decay heat calculation methodology in DYN3D

The lattice transport - nodal diffusion sequence is a widely used approach for performing full core calculations. Typically, homogenized macroscopic cross-sections are generated by a lattice code and tabulated against the operational parameters (*e.g.*, coolant density, fuel temperature, burnup, etc.). In order to solve the depletion problem, microscopic cross-sections must also be mapped against the operational conditions. DYN3D uses both macro- and microscopic cross-sections sets to perform coupled neutron diffusion-T/H-burnup analysis on a core level.

The coupling scheme implemented in DYN3D relies on the quasi-static approach, in which multiple discrete time-steps are used to perform fuel cycle simulations. At the beginning of each time-step, DYN3D performs coupled neutron diffusion-T/H steady state calculations, with an optional criticality search. The obtained flux and temperatures distributions are considered constant during the time-step. These distributions are used to obtain local burnup and isotopic concentrations at the end-of-step (EoS). Finally, the EoS burnup distribution is used to obtain the updated macro- and microscopic cross-sections and the procedure is repeated for all the subsequent steps.

The microscopic data generation stage is described in Section 4.2, whereas the remainder of this section focuses on describing the general depletion calculations to obtain the nodal nuclide field, $\mathbf{N}(t)$, as a function of time. DYN3D obtains the nodal nuclide vector

$\mathbf{N}(\mathrm{t})$, composed of elements $N_{\mathrm{j}}(\mathrm{t})$, representing the atom density of nuclide j, by solving the depletion problem formulated in Eq. (1):

$$\frac{dN_j(t)}{dt} = -\lambda_j N_j(t) - \bar{\sigma}_{a,j}\phi N_j(t) + \sum_{\substack{k=1 \\ k\neq j}} \left(\lambda_{k\to j} + \bar{\sigma}_{k\to j}\phi\right) N_k(t) \tag{1}$$

where, λ_j is the decay constant of nuclide$j$, $\bar{\sigma}_{a,j}$ is the energy average absorption cross-section, ϕ is the total flux, $\lambda_{k\to j}$ is the decay constant from nuclide k to nuclide j, and $\bar{\sigma}_{k\to j}$ is the average transmutation cross-section of the nuclide k that leads to the production of nuclide j. The set of multiple first order differential equations, Eq. (1), can be represented in a matrix form:

$$\frac{d\mathbf{N}(t)}{dt} = \mathbf{A}\,\mathbf{N}(t) \tag{2}$$

where $\mathbf{A}$ is the transmutation matrix, which consists of the radioactive decay and neutron induced reactions rates terms. The solution of Eq. (2) is obtained using the matrix exponential, as described in Eq. (3), assuming constant reaction rate during time-step Δt.

$$\mathbf{N}(t) = \exp[\mathbf{A}\Delta t]\,\mathbf{N}(t_0) \tag{3}$$

where, $\mathbf{N}(t_0)$ describes the known nuclide vector at time t_0, and Δt is the length of the depletion time-step. Detailed discussion on the matrix components and their generation is given in Section 4.2.

The method implemented in DYN3D to solve the matrix exponential (*i.e.*, $\exp[\mathbf{A}\Delta t]$) is Chebyshev rational approximation (Gonchar and Rakhmanov, 1989; Pusa, 2011). This method was proven to be particularly suitable (Pusa and Leppänen, 2010) for fuel depletion and radioactive decay problems.

The macroscopic cross-sections for nodal diffusion calculation, as well as microscopic cross-sections for depletion calculation, are interpolated from a pre-generated library using local burnup and operational parameters. The operational history is taken into account by correcting the macroscopic cross-sections using local nuclide concentrations (Bilodid et al.,

2016). Detailed spatial nuclide content distribution also allows to calculate the radioactive decay heat via the isotopic summation method, without relying on semi-empirical correlations, as described in Eq. (4):

$$P^n(t) = \sum_j N_j^n(t)\lambda_j q_j \quad (4)$$

where, $P^n(t)$ is the decay heat rate in node n at time t, $N_j^n(t)$ is the time dependent concentration of nuclide j in node n, and q_j is the energy release per decay of nuclide j. To capture the contribution of all the heat producing nuclides, DYN3D explicitly tracks over 1,200 isotopes (by default) via their decay or transmutation reactions. However, the transmutation chains can be easily extended, provided that the homogenized microscopic cross-sections are supplied to DYN3D by the lattice code.

This method was implemented in DYN3D and tested only on simple 2D infinite lattices (Bilodid et al., 2018). The main goal of this work is to demonstrate the applicability of this method to predict spatial isotopic concentrations and decay heat distributions in 3D full core case.

4 Data generation

Generation of homogenized macro- and micro- group constants is a key step to enable accurate core modeling. Section 4.1 describes the procedure used to generate the few-group macroscopic parameters. Section 4.2 describes the data required to perform micro-depletion calculations with DYN3D. This section presents the preparation of the decay matrix, few-group microscopic cross-sections, and fission yields. Finally, issues associated with data management and processing are described in Section 4.3

4.1 Generation of homogenized few-group macroscopic cross-sections

Generation of the homogenized few-group macroscopic cross-sections is an important step in utilizing the Serpent-DYN3D code sequence. The few-group cross section methodology adopted in this work relies on the previous studies performed at HZDR (Fridman and Shwageraus, 2013; Nikitin, Fridman, and Mikityuk, 2014).

- Cross-sections for fuel regions were generated using a single 3D model of a fuel assembly with reflective boundary conditions radially and black boundary conditions axially. The model generates group constants for each of the axial fuel layers simultaneously, as depicted in **Fig. 3**.
- All group constants for non-multiplying regions are generated using 2D super-cell models (Nikitin, Fridman, and Mikityuk, 2014), as depicted in **Fig. 4**. The regions of interest are placed in the center of the model and surrounded by fuel assemblies to best approximate the flux the assembly would experience in a full core. The group constants are only homogenized over the central region of interest.

Throughout this work, cross sections are generated with a 24-group structure. The choice of energy structure is based on the study presented by Fridman and Shwageraus (Fridman and Shwageraus, 2013).

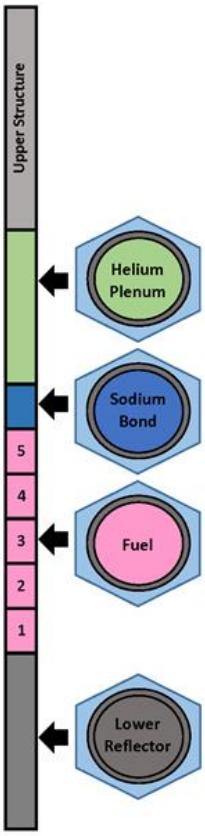


Fig. 3 Axial representation of fuel regions for cross section generation

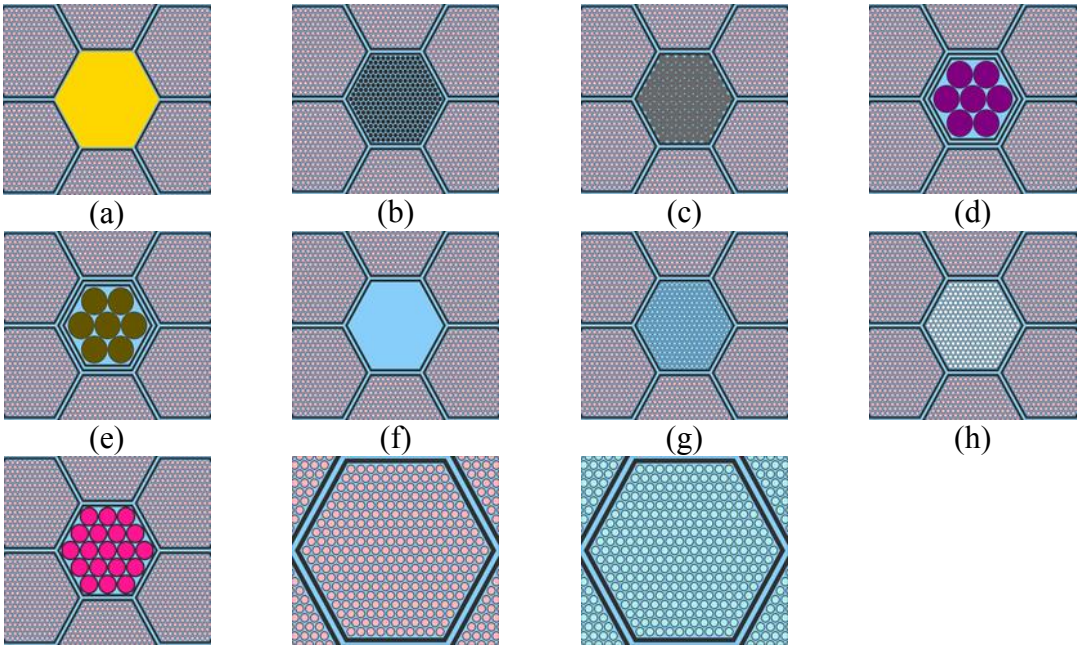


Fig. 4 Super-cell and lattice models: (a) lower structure, (b) lower reflector, (c) radial reflector, (d) primary control subassembly, (e) secondary control subassembly, (f) empty duct, (g) sodium plenum, (h) helium gas plenum, (i) radial shield, (j) inner core fuel, (k) outer core fuel.

4.2 Generation of the transmutation matrix

This section describes the data preparation associated with solving the depletion problem (*i.e.*, Bateman equations). More specifically, this section describes the procedure adopted to assemble the transmutation matrix and the corresponding simplifications made to ease the data management. In our study, a total of ~1,400 nuclides are included in the transmutation matrix, which is a sparse matrix (denoted as **A**) as shown in **Fig. 5**.

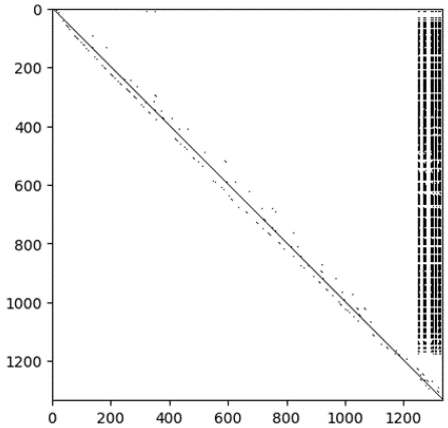


Fig. 5 Sparsity of the transmutation matrix.

Fig. 5 shows a dense diagonal ($A_{j,j}$ elements) that represents the decay constants (λ_j) for each isotope *j* and the group absorption reaction rates as shown in Eq. (5).

$$A_{j,j} = -\left(\lambda_j + \sum_g \sigma_{a,j}^g \phi_g \right) \tag{5}$$

where, ϕ_g is the neutron flux in group g and the summation is performed over all the energy groups (one-group absorption reaction rates). The off-diagonal, $A_{k,j}$, elements are composed by including the decay from isotope *k* to *j* ($\lambda_{k\rightarrow j}$) and the transmutation rate ($\sigma_{k\rightarrow j}^g \phi_g$) from isotope *k* to *j*.

$$A_{j,k} = \sum_k (\lambda_{k\rightarrow j} + \sum_g \sigma_{k\rightarrow j}^g \phi_g) \quad , \quad k \neq j \tag{6}$$

Lastly, the dense array of columns in **Fig. 5** correspond to the production of fission fragments from fissions events originated from different fissionable elements as described by Eq. (7).

$$A_{j,k} = \sum_g \gamma_g^{k\rightarrow j} \sigma_{f,k}^g \phi_g \quad , \quad k \neq j \tag{7}$$

where, $\gamma_g^{k\rightarrow j}$ is the energy dependent fission yield that leads to the production of isotope *j* (*e.g.*, Xe-135), given that a fission event occurred in isotope k (*e.g.*, Pu-239).

Several key points related to the compilation of the transmutation matrix are given below:

Transmutation matrix

The transmutation matrix, which includes around 1,400 isotopes, was obtained directly from Serpent. The latter does not use a fixed number of nuclides or decay and depletion chains to construct a transmutation matrix. Instead, Serpent builds the transmutation chains

automatically starting from the initial fuel composition and using the reaction modes described in radioactive decay and cross section data libraries (Serpent WIKI). Under the current implementation, the decay matrix in DYN3D is fixed based on the initial isotopic composition of the ABR fuel. There is no error associated with the mentioned simplification. However, if a completely different fuel composition is taken (*e.g.*, thorium-based fuel), the transmutation matrix should be re-generated.

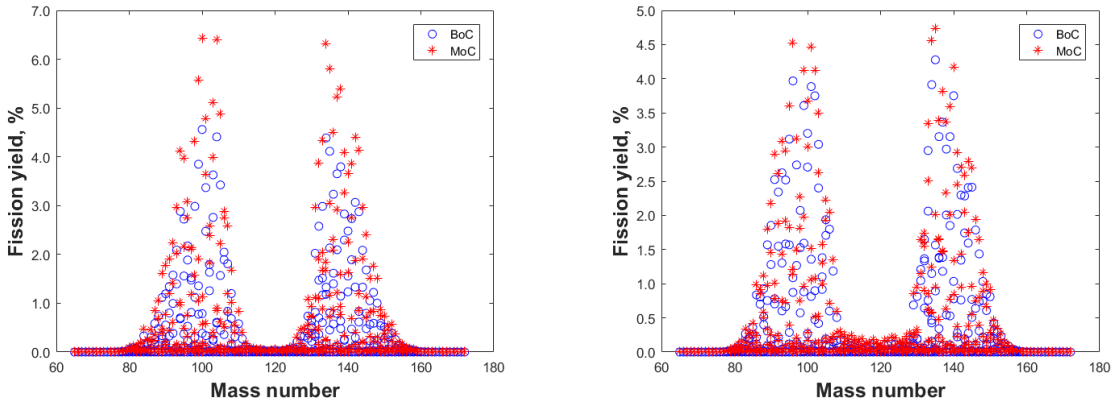
Microscopic cross-sections

Serpent allows to generate homogenized microscopic cross-sections for any isotope and reaction ("mdep" card) without the need to define dedicated detectors/tallies. This option was exploited to generate tabulated few-group microscopic cross-sections with the same energy structure used for the macroscopic quantities. Only the (n,γ), (n,fission), (n,2n), (n,3n), and (n,α) reactions were considered in the transmutation chains. Microscopic cross-sections were generated for precisely 290 of the 1,400 isotopes corresponding to the isotopes included within the ENDF/B-VII cross section library. It must be emphasized that Serpent includes many more reactions, such as (n,t), which were not considered in DYN3D. However, the error associated with this simplification has a negligible effect on isotopic concentration of most nuclides, with higher impact on the lighter nuclides which are more likely to undergo some of these rare reactions.

Fission yields

During calculations, Serpent obtains energy-dependent fission yields directly from the ENDF formatted fission yield data files. Typically, fission yield data is specified for several particular incident neutron energies. To obtain the effective fission yields for depletion calculations, Serpent interpolates the energy-dependent fission yield for every fissionable isotope that undergoes a fission event during the transport tracking routine. As a result, the fission yields in Serpent vary with burnup and spectrum and are composition dependent. This continuous energy approach cannot be reproduced exactly in DYN3D nor in any diffusion code because the code relies on energy discretization. Consequently, it was decided to generate energy-independent sets of fission yields for every fissionable isotope based on the Serpent beginning-of-life data. Finally, Serpent produces fission yield values at every burnup point, leading to prohibitively large sets. Therefore, it was also decided to keep the fission yields constant throughout the depletion cycle.

This simplification in fission yields treatment leads to discrepancies in the fission products concentrations. Fig. 6 presents the fission yields distribution for Pu-239 and U-238 at different burnup points. It is clear that this difference is propagated through the depletion analysis and has non-negligible impact on some specific fission products. There are various potential approaches to avoid storing the fission yields as a function of operational conditions, but none were studied here.



(a) Pu-239 (b) U-238

Fig. 6 Weighted fission yields at beginning-of-cycle (BoC) and middle-of-cycle (MoC).

4.3 Data processing requirements

The data storage and processing requirements associated with tabulating macroscopic quantities are fairly negligible. However, this is not the case if microscopic quantities are added to the tabulation process. Energy dependent microscopic cross-sections of different type (*e.g.* σ_f) must be stored for hundreds of nuclides and mapped against the operational conditions. For each fuel type, a unique library with microscopic parameters must be generated. In addition, fission yields for each possible parent-daughter pair must also be included in the pre-generated libraries.

In the current analyses, 10 unique fuel types and hence libraries were considered. The cross-sections in each library were generated for 290 isotopes, five reaction types, and 24 energy groups, resulting in a large amount of data.

It must be pointed out that Serpent generates output files in MATLAB format. The few-group macroscopic cross-sections are printed in the main result file (*i.e.*, "_res.m"), which occupied more than 70 MB in the current study. The microscopic cross-sections are printed in a separate "_mdx.m" file (6 MB) for each depletion step.

The results were processed using the serpent-tools package, developed by the CORE group at Georgia Tech (Johnson et al., 2018). The serpent-tools Python package is a collection of parsing tools and containers aimed at expediting analysis of Serpent outputs. Files that would make MATLAB extremely slow or unresponsive can be processed within fractions of seconds, with no loss of data. The data is stored in an object-oriented framework that mimics the physical nature of the quantities represented, *e.g.*, Detector objects have tallies and grid structures, Depleted Material objects have names and associated atomic density, toxicity, and burnup matrices. Many of the readers and containers have routines for expediting common analyses, with heavy emphasis on plotting. Plots of Cartesian and hexagonal detector meshes, flux spectra, homogenized group constants, and depletion parameters are made accessible to the user, without requiring a high learning curve. The project is hosted on GitHub with a permissive MIT license and is undergoing constant development and improvement. A thorough overview of the supported file types with examples can be found through the repository (Johnson et al., 2018).

5 Results

The results presented in the following sub-sections were produced for three cases, with increasing levels of complexity. First, an infinite fuel assembly (Section 5.1) was investigated due to its simplicity and the ability to accurately predict the few-group fluxes. The next test case (Section 5.2) focused on an infinite heterogeneous 3D fuel assembly, in which the spatial flux variation affects the axial decay heat predictions. Finally, the core level results are summarized in Section 5.3, in which a variable power scheme was applied. It must be emphasized that macro and microscopic cross-sections were tabulated only as a function of burnup.

5.1 Infinite 2D fuel assembly analysis

As a first step, an infinite 2D assembly is depleted in Serpent, with cross-sections being generated throughout the depletion cycle. The macro and microscopic cross-sections are then used in DYN3D to repeat the depletion calculations. For this scenario, the power of the simulated system was varied (**Fig. 7**) throughout the depletion to investigate the effect this would cause on using the code sequence to calculate decay heat. Criticality reported by

Serpent and DYN3D is compared and plotted in **Fig. 8** along with the difference in reactivity.

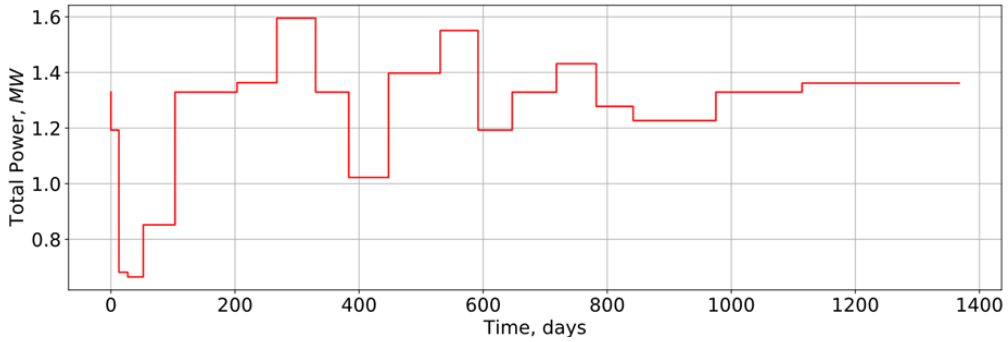


Fig. 7 Assembly power profile throughout the cycle

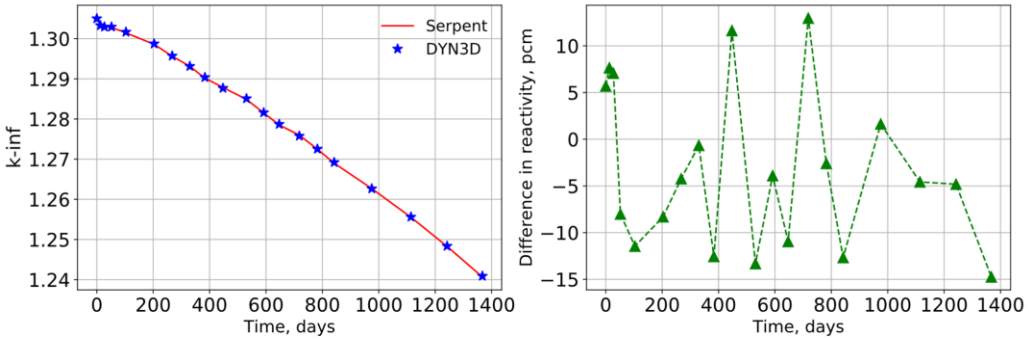
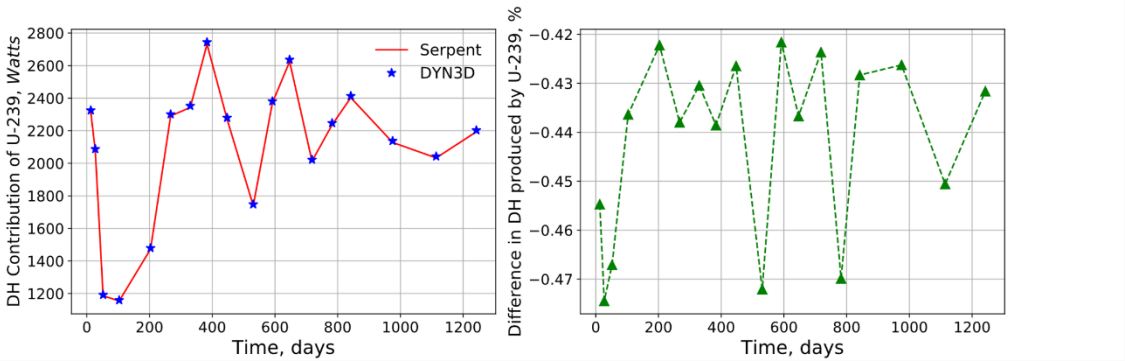
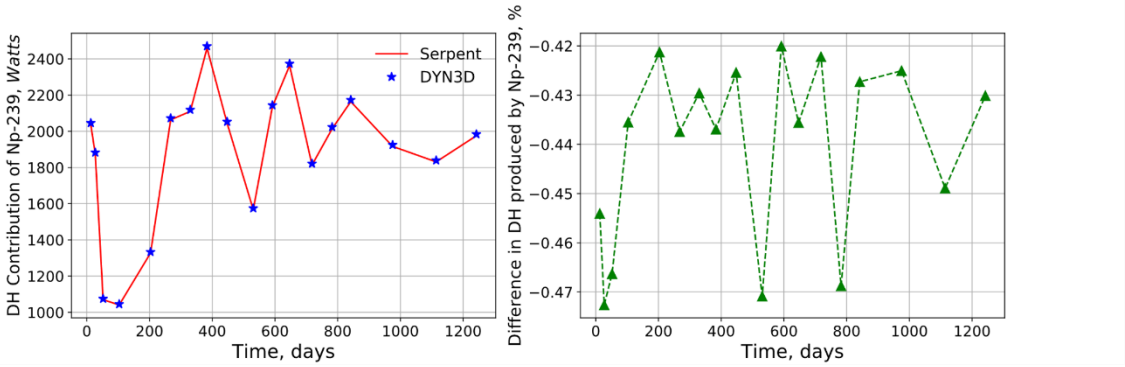


Fig. 8 Criticality throughout depletion cycle for Serpent vs. DYN3D infinite lattice

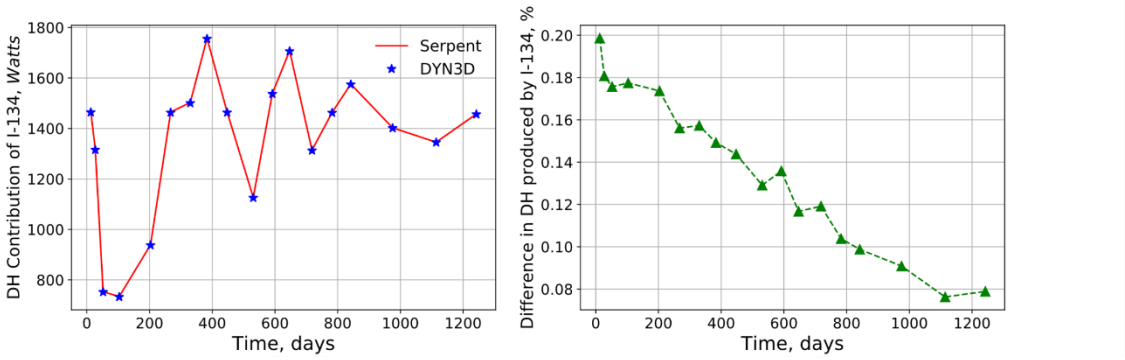
The ability of each code to keep track of isotopic concentration throughout the depletion cycle is crucial for calculation of decay heat. The individual contribution of the top five isotopes to decay heat is compared between Serpent and DYN3D in **Fig. 9**. To determine which isotopes constitute the largest decay heat (DH) share, the contribution of each isotope is averaged over the depletion cycle. It can be seen that these isotopes are in very good agreement, with a difference in decay heat produced on the order of 0.3%.



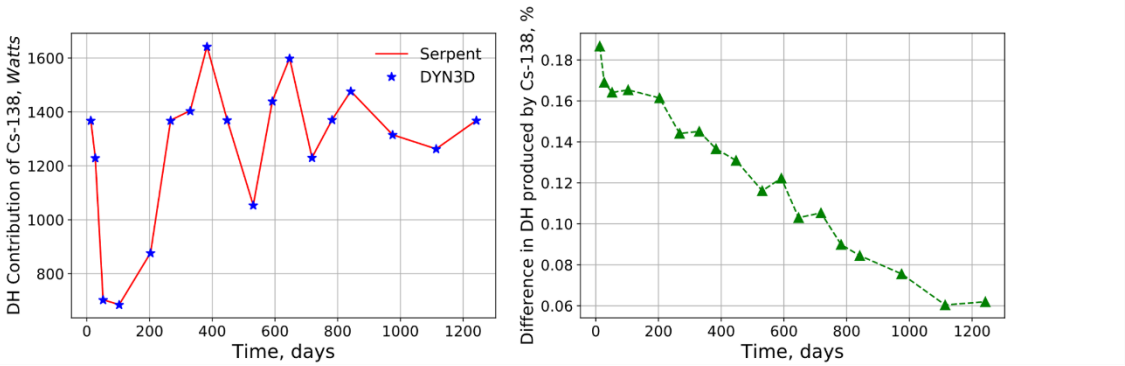
(a) U-239



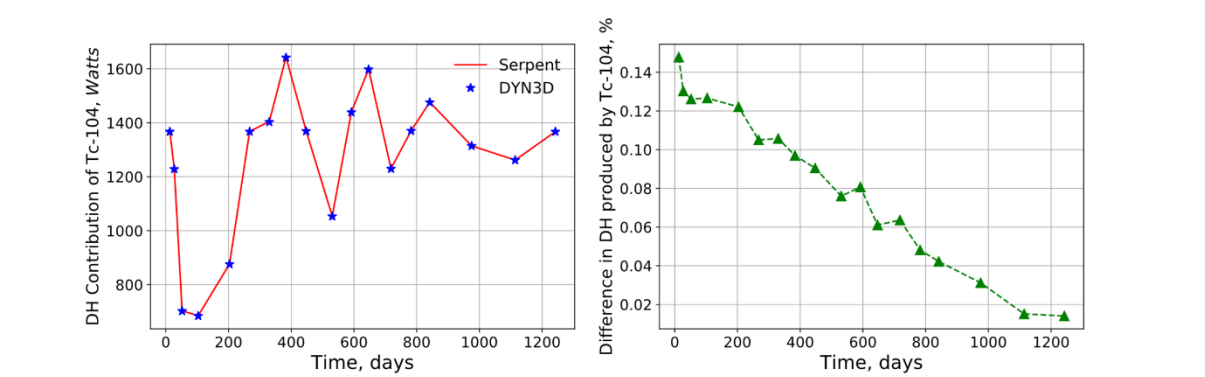
(b) Np-239



(c) I-134



(d) Cs-138



(e) Tc-104

Fig. 9 Decay heat contribution and difference in decay heat calculated by Serpent and DYN3D for the five most contributing isotopes.

Finally, the total decay heat in the system calculated by Serpent and DYN3D is compared. Very good agreement is achieved, with the difference between codes remaining close to constant throughout the cycle, even though the power is varied greatly in the cycle.

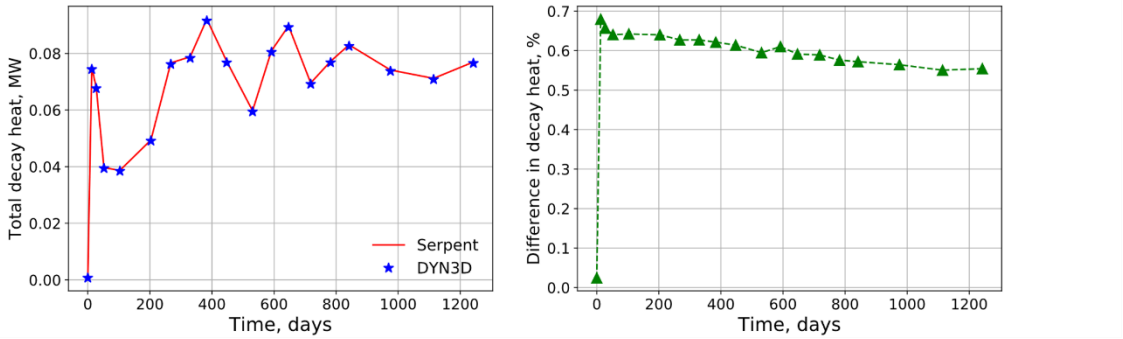


Fig. 10 Total decay heat in the assembly throughout cycle and difference between Serpent and DYN3D.

5.2 Infinite 3D fuel assembly analysis

The assembly active fuel region is axially divided into five heterogeneous layers with unique isotopic concentrations, as shown in **Fig. 3**. The cross-sections for the fuel are generated throughout the depletion cycle, while a single zero-burnup cross-sections set is used for non-multiplying regions. The criticality calculated by Serpent and DYN3D are compared in **Fig. 11**.

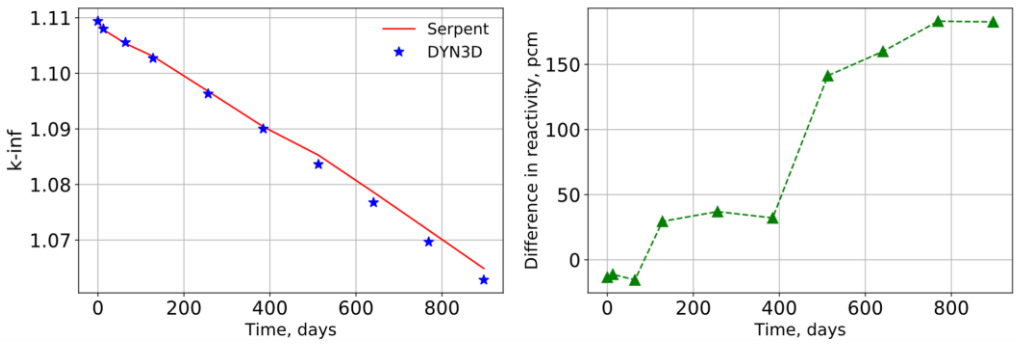


Fig. 11 Criticality throughout depletion cycle for Serpent vs DYN3D 3D fuel assembly.

In general, the agreement between the codes is very good, though there appears to be a consistent error build-up as the assembly continues to be depleted. The origin of the discrepancy arises from the fuel-reflector heterogeneity, but this is still considered to be a relatively small difference. For sake of comparison, the average standard deviation reported by benchmark participants at the End of Cycle (EoC) is in the order of 700 pcm (NEA, 2016), therefore certain variation is to be expected when different methods are used. In order to identify the axial layer (fuel region) with the largest contribution to the error, the burnup calculated by each code is compared for each fuel region and is depicted in **Fig. 12**. There is good agreement for the burnup calculated by Serpent and by DYN3D in all fuel regions, with a maximum difference of less than 1% throughout the depletion cycle.

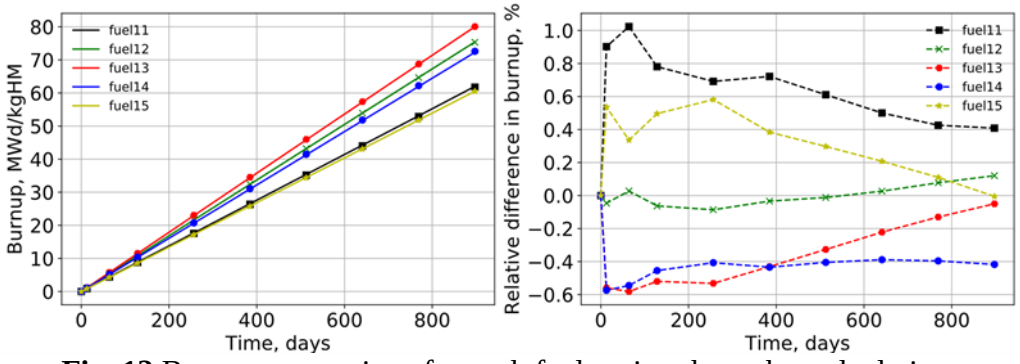


Fig. 12 Burnup comparison for each fuel region throughout depletion.

The isotopic composition calculated by each code is compared by taking the root mean square (RMS) error for each isotope i, over the five layers ($J = 5$) and the nine time-steps ($K = 9$) according to Eq. (8). The RMS differences for fission products and actinides are shown in **Fig. 13**. It should be noted that only isotopes with concentration values above

$10^{13} at./cm^3$ were included in the comparison. Very good agreement is found between the isotopic concentrations reported by Serpent and DYN3D, both for fission products and actinides, with a maximum RMS on the order of 0.5% in a few outlier cases.

$$RMS_i = \sum_k \sum_j \sqrt{\frac{100\%}{K \times J} \left[\frac{N_j^{Serpent}(t_k) - N_j^{DYN3D}(t_k)}{N_j^{Serpent}(t_k)} \right]^2} \quad (8)$$

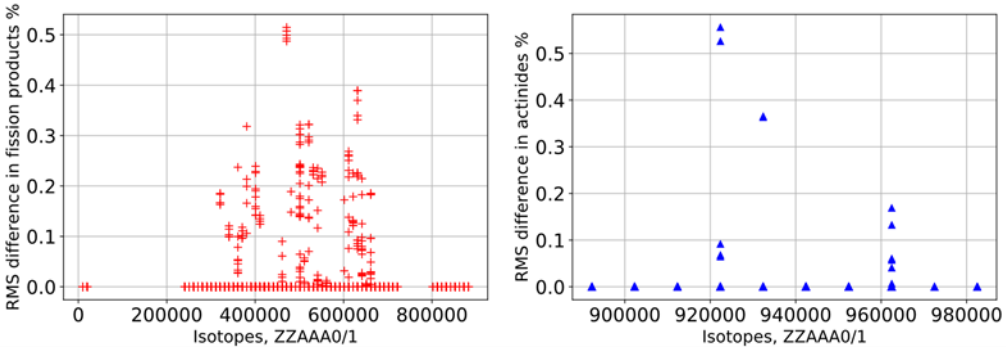


Fig. 13 RMS difference in atomic density calculated by Serpent and DYN3D for all isotopes.

Finally, the decay heat generated in each fuel region is calculated by Serpent and by DYN3D, and results are compared in **Fig. 14**. A good agreement is observed between Serpent and DYN3D, with the highest difference in decay heat happening at the lower fuel region, at a maximum of 1.6%, but the difference decreases to 0.6% at the end of the depletion cycle. This relatively high discrepancy in decay heat occurs due to the slightly different spatial flux values predicted by DYN3D, which leads to slightly different nuclide densities.

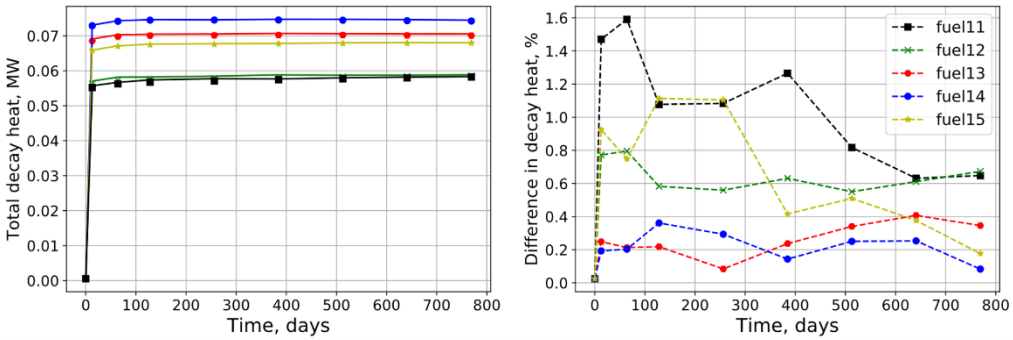


Fig. 14 Total decay heat generated in each fuel region throughout the depletion cycle.

5.3 Full core analysis

The reference solution for the full core depletion case is generated with the Monte Carlo code Serpent. Since coupled Monte Carlo–T/H calculations, in conjunction with depletion, are prohibitively expensive, all calculations were performed with fixed temperature distribution. For the cross-sections library generation, the fuel assemblies of the inner and outer core (shown in **Fig. 4**) were independently depleted for a full cycle, with the cross-sections for each axial fuel region being generated for each burnup step. The cross-sections for non-multiplying regions were assumed to remain constant throughout the cycle. Here, the B1 leakage corrected cross-sections (Fridman and Leppänen, 2011) were applied, only for the fuel assemblies, to account for the neutron leakage and nearly critical core configuration. The arbitrarily chosen operational power history is shown in **Fig. 15**.

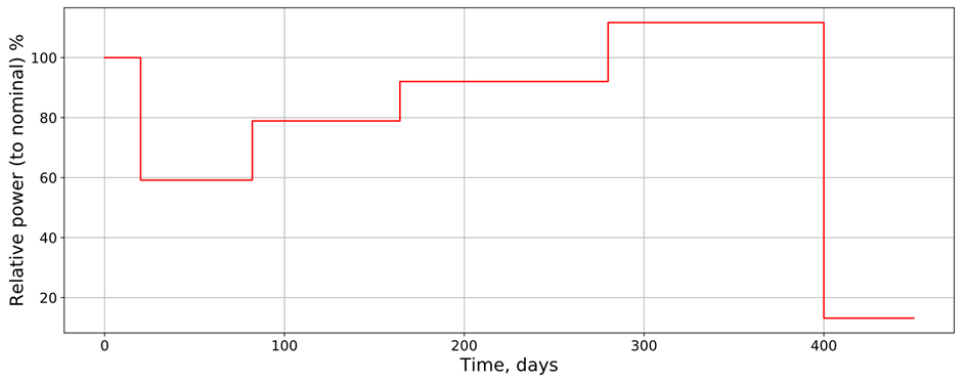


Fig. 15 Power level throughout depletion cycle.

A good agreement is observed for the criticality values (**Fig. 16**) calculated by both codes. The difference in radial power distribution reported by both codes is compared in **Fig. 17** and **Fig. 18** for one quarter of the core. These figures show good agreement in the radial power results produced by Serpent and DYN3D, with a maximum difference of 1.38% at the Beginning of Cycle (BoC) and 2.11% at the End of Cycle (EoC). The calculation of difference in power distribution was repeated for selected intermediate steps throughout the cycle. The difference in criticality and the radial power RMS error are summarized in Table 2.

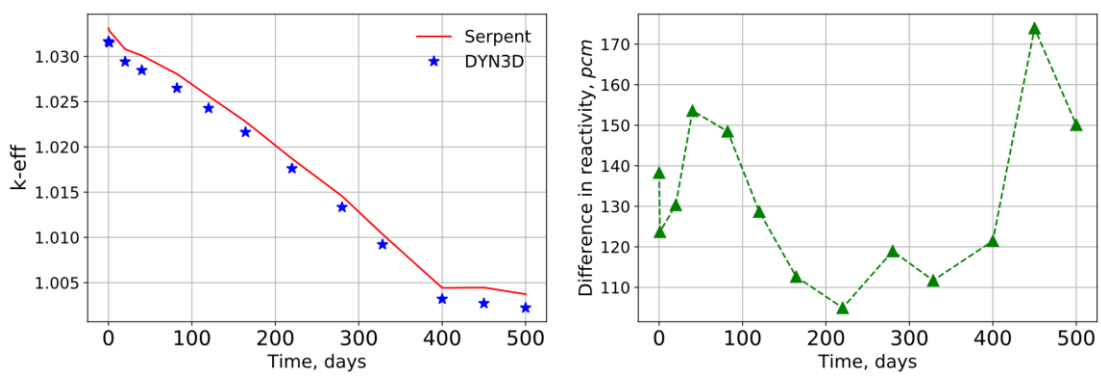
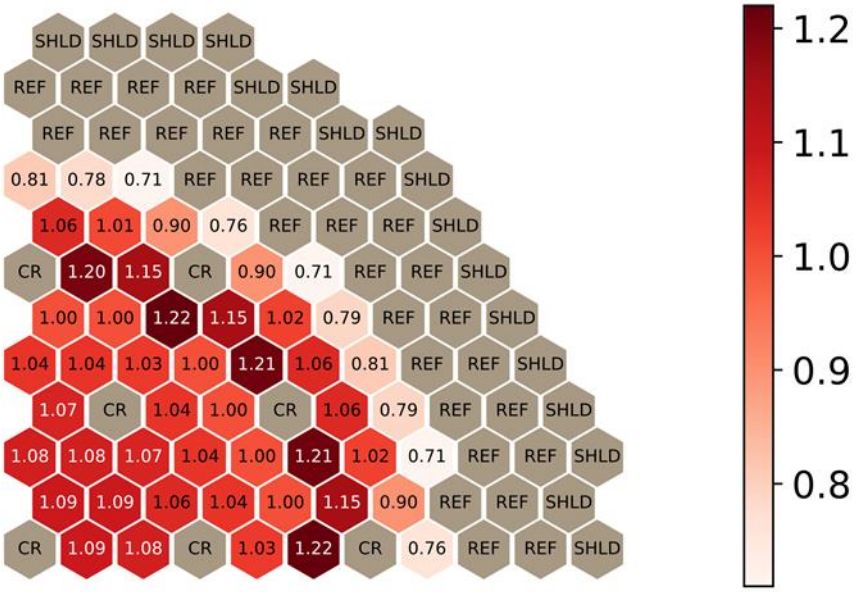
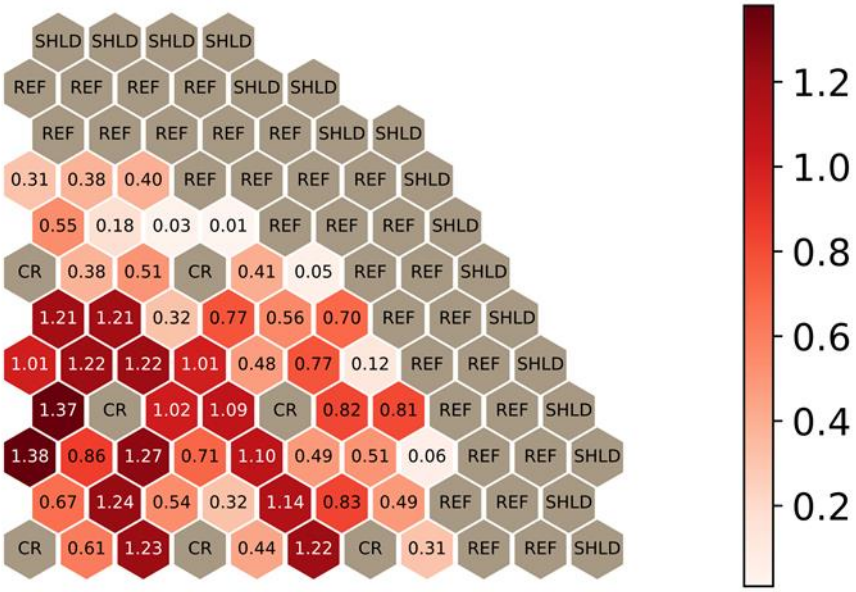


Fig. 16 Criticality throughout depletion cycle for Serpent vs DYN3D full core model.

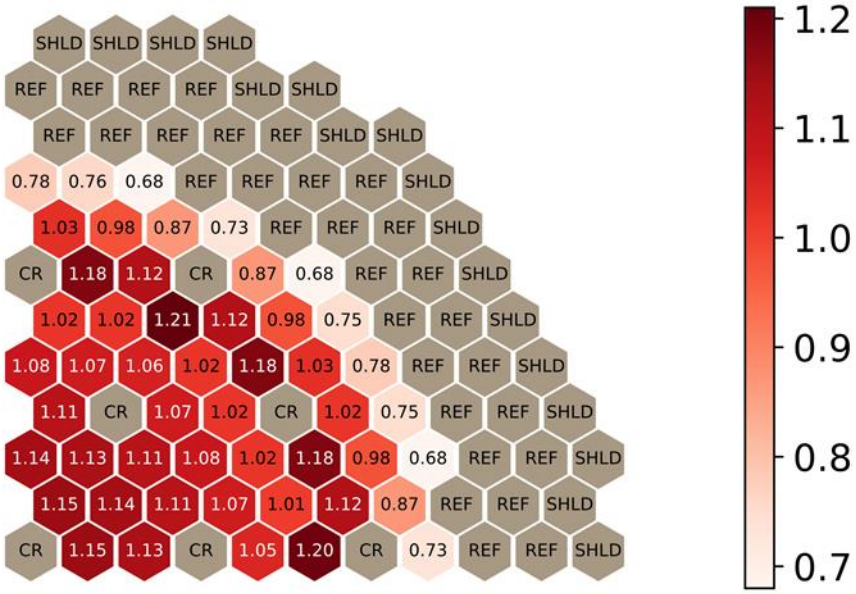


(a) Power peaking values

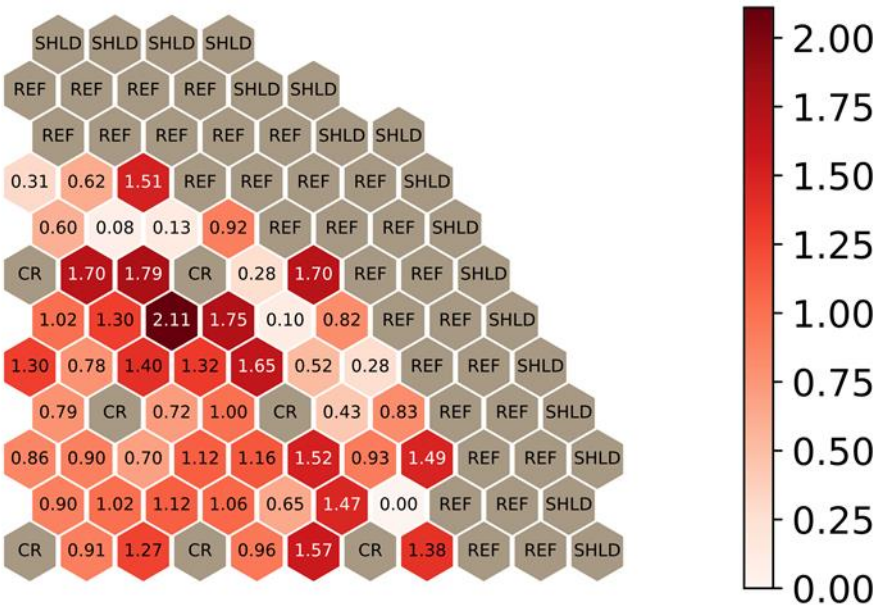


(b) Relative difference in power (%)

Fig. 17 Difference in radial power distribution at beginning of cycle.



(a) Power peaking values



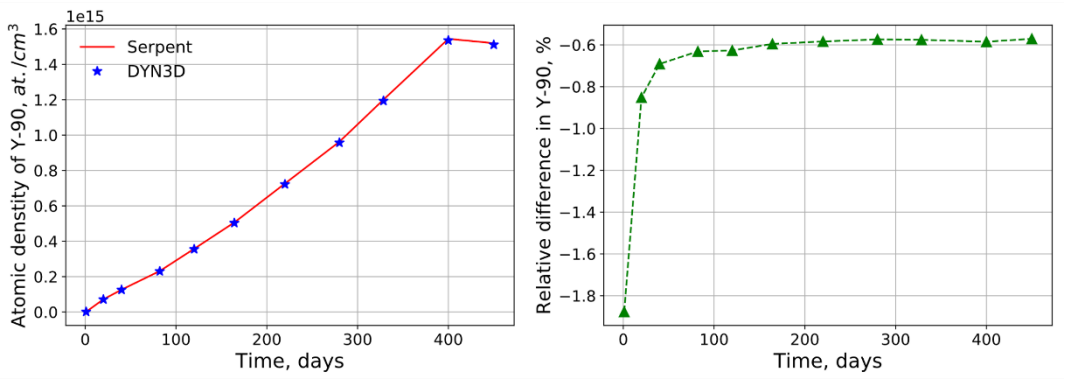
(b) Relative difference in power (%)

Fig. 18 Difference in radial power distribution towards the end of cycle (328 days).

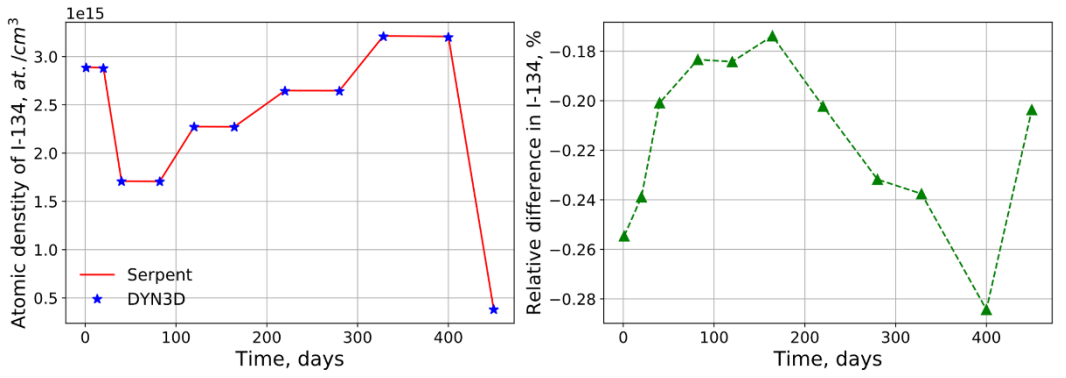
Table 2. Serpent vs. DYN3D comparison for selected intermediate depletion points.

BU, MWd/kg	Time, Days	k_{eff} DYN3D	k_{eff} Serpent	Uncertainty, pcm	Difference, pcm	Radial Power RMS error, %
0	0	1.031646	1.03312	5	138	0.86
3.042	40	1.028463	1.03009	5	154	0.83
6.236	82	1.026493	1.02806	5	148	0.88
12.168	160	1.021623	1.02280	4	113	0.97
24.945	328	1.009211	1.01035	5	112	1.17

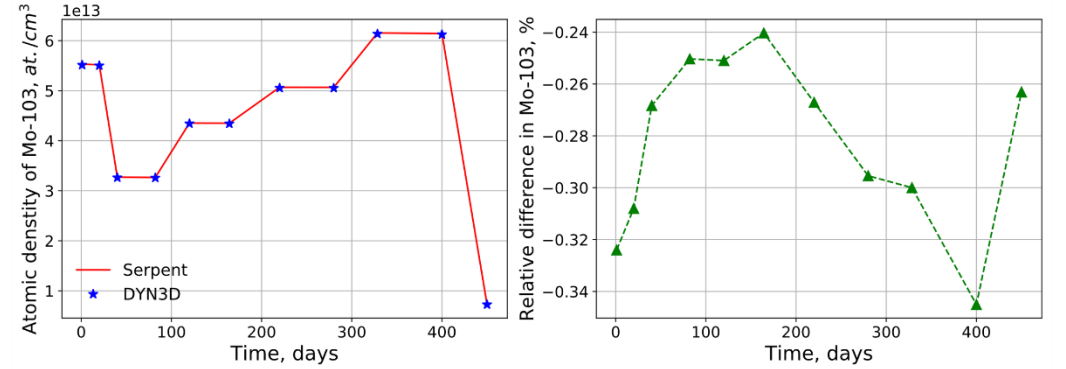
The core average isotopic composition of specific important fission products and actinides are presented in **Fig. 19** and Fig. 20 respectively.



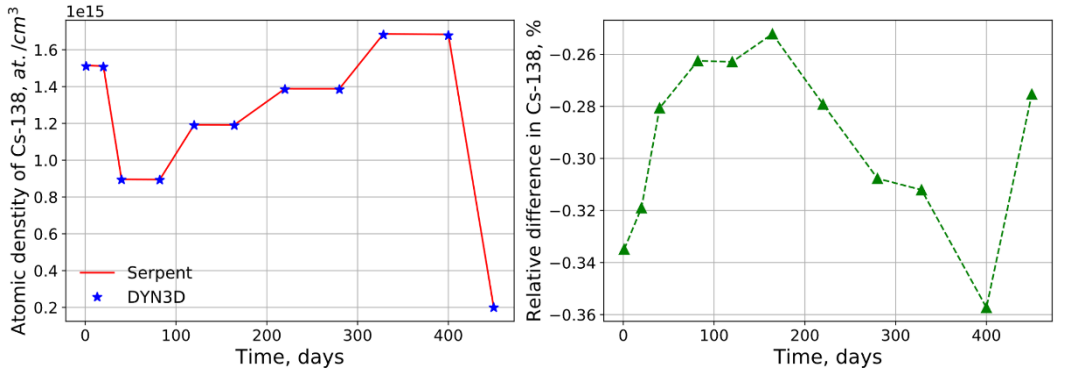
(a) Y-90



(b) I-134

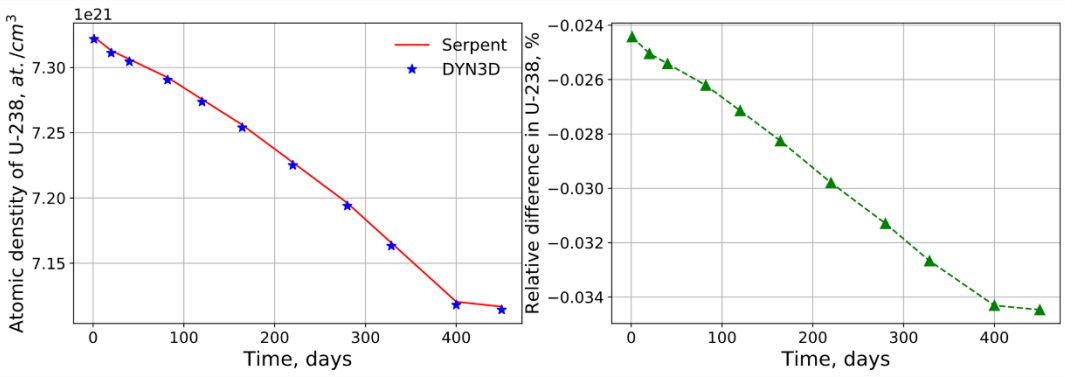


(c) Mo-103

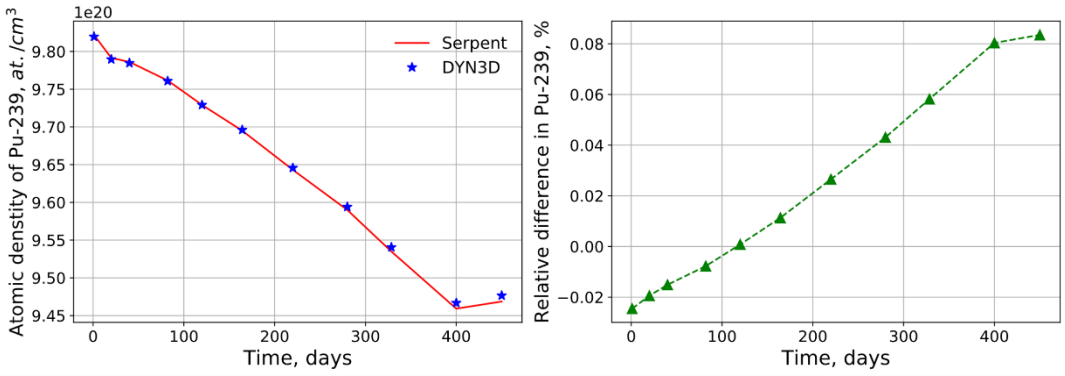


(d) Cs-138

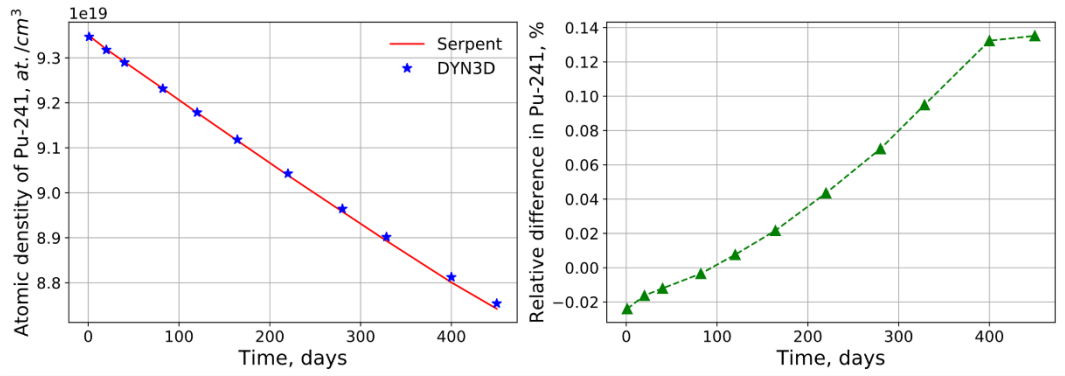
Fig. 19 Atomic concentrations and differences calculated by Serpent and DYN3D for various fission products.



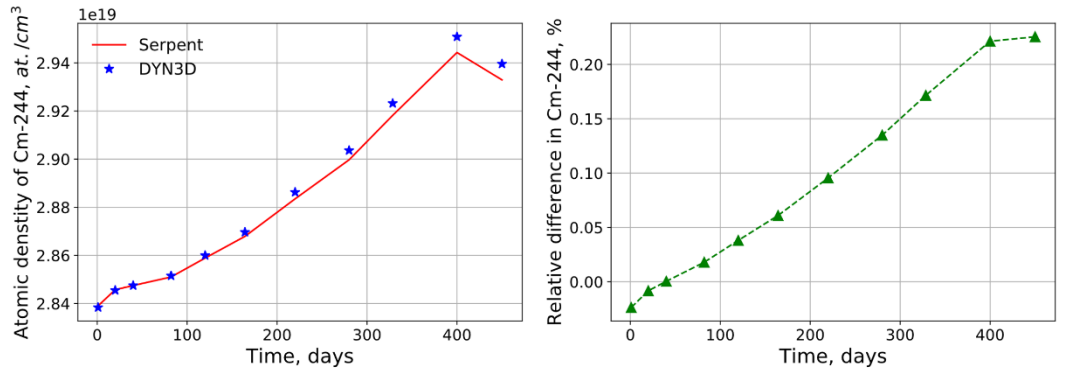
(a) U-238



(b) Pu-239



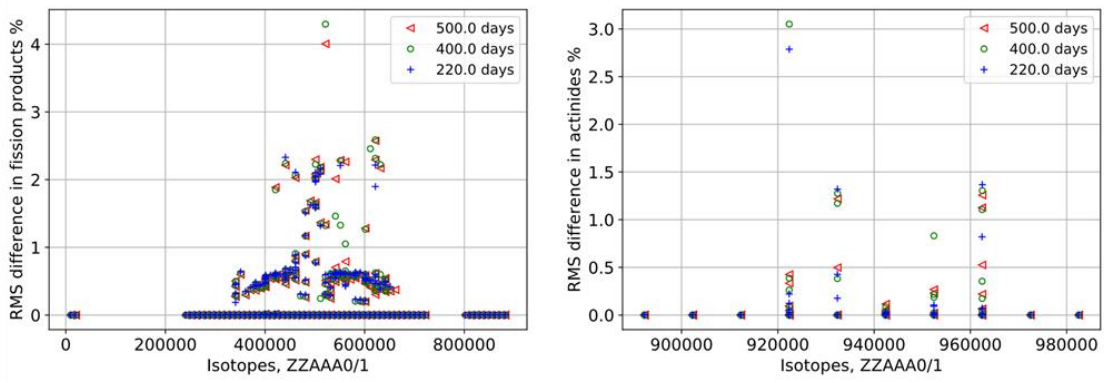
(c) Pu-241



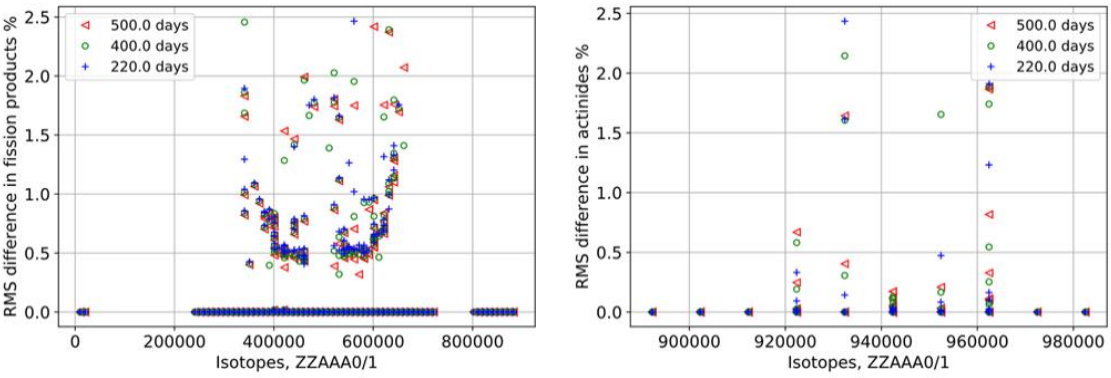
(d) Cm-244

Fig. 20 Atomic concentration and difference calculated by Serpent and DYN3D for various actinides.

In order to understand the overall isotopic concentration difference, the RMS error for all actinides and fission products tracked are plotted in Fig. 21 for three different points in the depletion cycle. In general, the results are in good agreement, with most of the isotopic difference below 1%. However, there are some fission products with differences in the 1-2% range and a small fraction of nuclides experience differences above 2%. The primary contributor to the depicted errors can be attributed to the mismatch in the spatial flux distribution between the codes. However, a fraction of the error in the concentrations of some fission products is due to the fixed fission yields used in our study.



(a) Inner core region

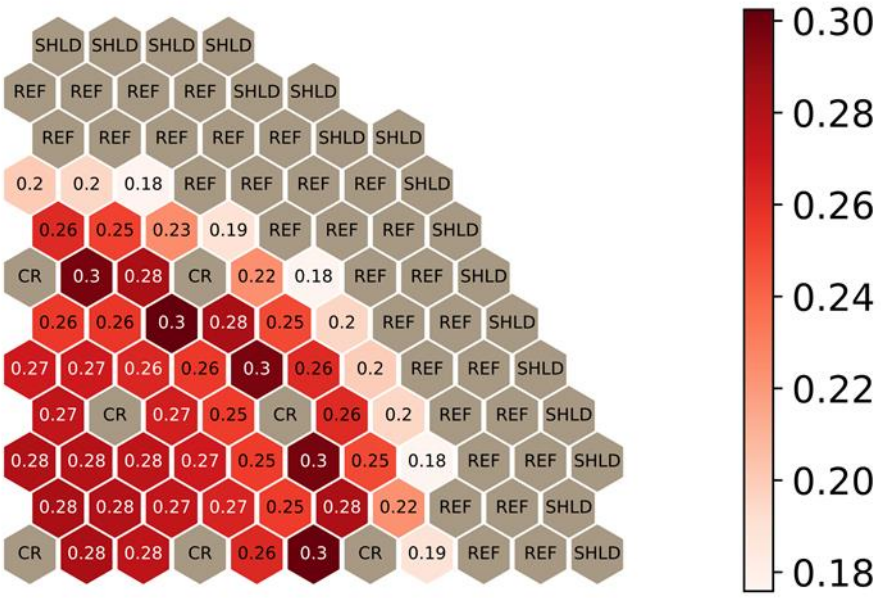


(b) Outer core region

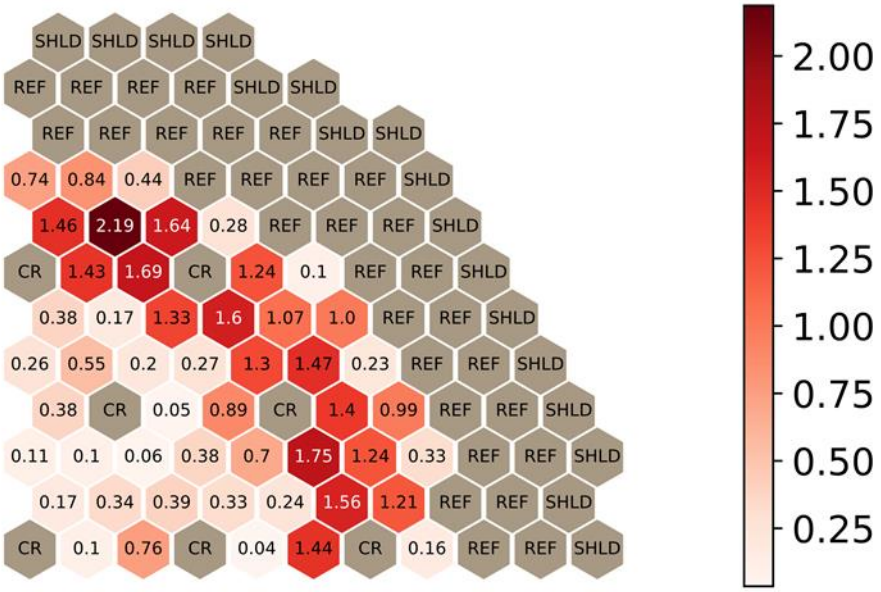
Fig. 21 RMS difference in atomic density calculated by Serpent and DYN3D for all isotopes at different time-points of the cycle.

Finally, the decay heat calculated by Serpent and DYN3D is compared in **Fig. 22** and **Fig. 23**. First, the decay heat in each radial assembly channel is calculated. Results are

presented for a time step at the middle of the cycle, when the relative power is at 90% of the nominal, and also at the end of cycle, where the relative power is at about 11% of the nominal.

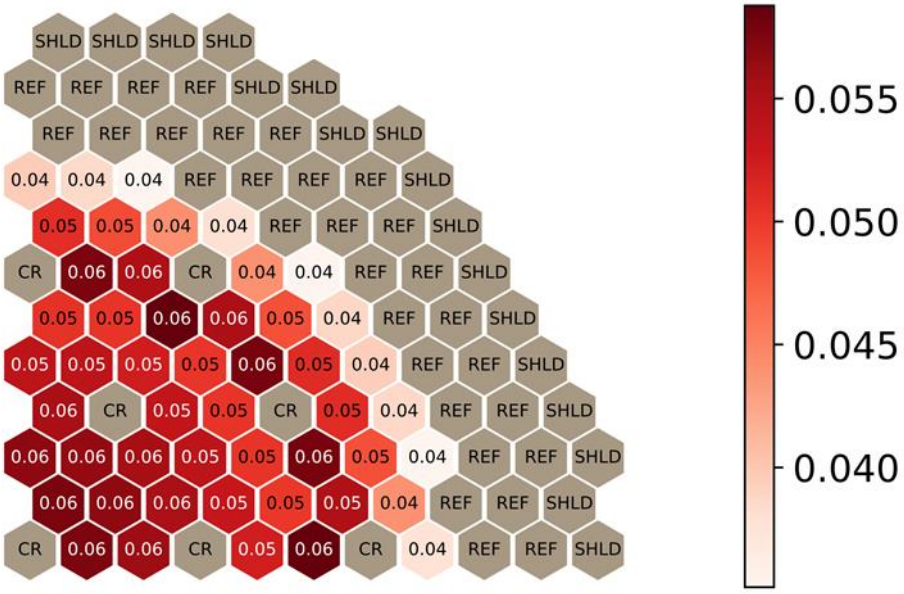


(a) Decay heat values (MW)

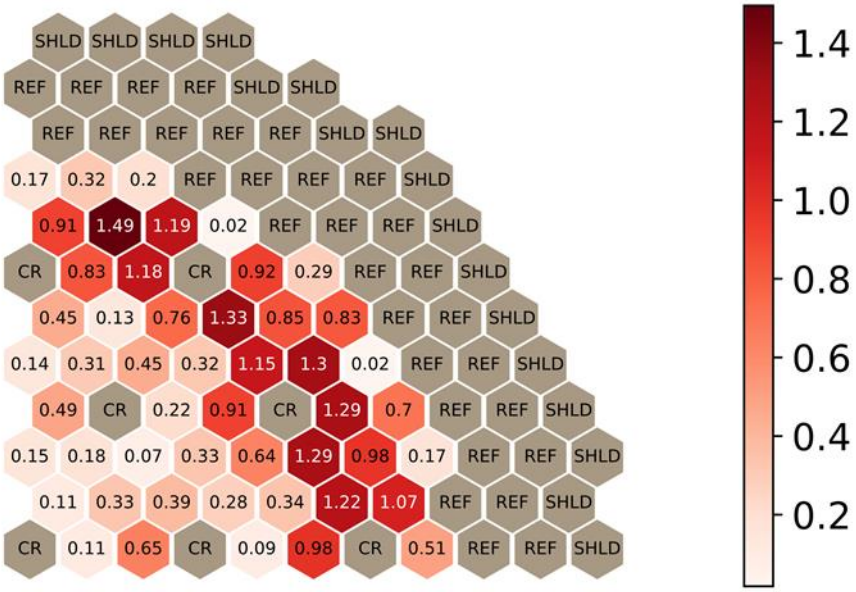


(b) Relative difference in decay heat (%)

Fig. 22 Decay heat calculated by Serpent and DYN3D and the difference between codes in the middle of the cycle, 90% of the nominal power level.



(a) Decay heat values (MW)



(b) Relative difference in decay heat (%)

Fig. 23 Decay heat calculated by Serpent and DYN3D and the difference between codes at the EoC, 11% of the nominal power level.

The radial comparison between codes demonstrates a very good agreement in both cases presented. A maximum difference in decay of around 2% was observed in the case in which the reactor power is at 90% of the nominal power, with most assemblies presenting a difference of around 1.5% and below. As for the case in which reactor power is low, close to 11% of the nominal, the observed difference in decay heat between Serpent and DYN3D is even smaller, with a maximum difference of 1.49%. It can be noted that, although great agreement is achieved for both cases, the difference in decay heat does tend to increase in the outer core region as compared to the inner core. This can be attributed to the higher fuel-reflector heterogeneity.

Lastly, the overall decay heat for the system calculated by each code is also compared and presented in **Fig. 24**. As expected based on the previous agreement achieved, the overall decay heat calculated by Serpent and DYN3D is in a very good agreement, with a maximum difference of only 0.6%. This analysis demonstrates that even for a full 3D core, with a varying power profile, the Serpent-DYN3D code sequence is able to accurately and efficiently perform decay heat calculations.

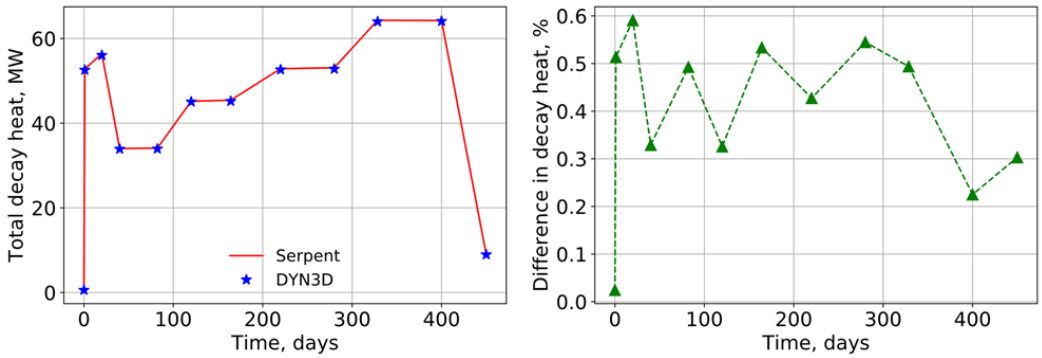


Fig. 24 Total decay heat in the core throughout cycle and difference between Serpent and DYN3D.

5.4 DYN3D performance considerations

For this study, the core was divided into 900 burnable regions (180 fuel assemblies, each with 5 axial layers) both in DYN3D and Serpent. The Serpent reference solution was obtained using 500,000 neutron histories, with 200 inactive and 700 active cycles per depletion step. As a result, the maximum statistical uncertainty in power was on the order of 0.1%. It took 91 hours to complete the MC burnup calculations on 8 CPUs using the Open Multi-Processing (OpenMP).

DYN3D is a serial code typically executed on a single CPU. To improve the efficiency of DYN3D for fuel cycle analysis, a parallel version was created. The parallelization was implemented only for the depletion routine using the OpenMP. In order to perform a fair comparison, the same system and number of processors were utilized by DYN3D. The overall computational time of DYN3D was 28 minutes, in which the multi-group diffusion solution occupied 30% and the depletion calculations share was 70%.

In summary, we find that performing the decay heat analysis using DYN3D leads to results that are in good agreement with the reference Serpent solution, while requiring significantly less computational resources.

6 Conclusions

The Serpent-DYN3D code sequence was used to perform decay heat analysis of metallic fuel fast reactors using the hybrid microscopic depletion approach. The ability to calculate the decay heat generated in the reactor is crucial for analyzing safety parameters and accident scenarios, a key step in the development of advance reactors. However, the nature of the analyses presented herein allows for the methodology to be applied to any type of system and fuel.

The Serpent MC code was used for cross-sections generation and to generate reference solutions for the cases analyzed. The neutronic and decay heat calculations were performed using the nodal diffusion code DYN3D. The Python package serpent-tools was used to process results and comparison between Serpent reference solutions and solutions obtained using the Serpent-DYN3D code sequence.

Nodal diffusion codes typically require macroscopic cross-sections libraries, which are tabulated against various operational conditions. The new Serpent-DYN3D sequence utilized in the current study introduces a somewhat challenging data management scheme: the need to tabulate microscopic cross-sections for each isotope, reaction type, and burnable material. In principal, the fission yields are energy dependent, but tabulating these for each father-daughter pair is computationally prohibitive. Therefore, this research relied on some simplifications in the data generation stage, namely by assuming constant fission yield values. More sophisticated data management methods must be envisioned to enhance the efficiency of the Serpent-DYN3D sequence.

In order to verify and demonstrate the new capabilities implemented in DYN3D, decay heat analyses were performed for a 2D infinite lattice, a 3D fuel assembly, and finally for a 3D full core model. In all the analyzed cases, very good agreement in criticalities, power distributions, isotopic concentrations, and spatial (and total) decay heat distributions were observed. The maximum difference in radial heat distribution was found to be 2.19%.

These results demonstrate that the Serpent-DYN3D code sequence is a useful tool in decay heat analysis, especially for advanced systems in which experimental data and correlations are not readily available. Applying this methodology saves time and computational resources, allowing for the analyses of various accident scenarios to be performed efficiently and with high fidelity.

Acknowledgements

This work was partially funded through the Nuclear Regulatory Commission project number NRC-HQ-84-14-G-0058.

References

"ANSI/ANS-5.1-2005: Decay Heat Power in Light Water Reactors." 2005.

Bilodid, Y., E. Fridman, D. Kotlyar, and E. Shwageraus. 2018. "Explicit Decay Heat Calculation in the Nodal Diffusion Code DYN3D." *Annals of Nuclear Energy* 121 (November): 374–81. https://doi.org/10.1016/J.ANUCENE.2018.07.045.

Bilodid, Y., D. Kotlyar, E. Shwageraus, E. Fridman, and S. Kliem. 2016. "Hybrid Microscopic Depletion Model in Nodal Code DYN3D." *Annals of Nuclear Energy* 92: 397–406. https://doi.org/10.1016/j.anucene.2016.02.012.

Fridman, E., and J. Leppänen. 2011. "On the Use of the Serpent Monte Carlo Code for Few-Group Cross Section Generation." *Annals of Nuclear Energy* 38 (6): 1399–1405. https://doi.org/10.1016/j.anucene.2011.01.032.

Fridman, E., and E. Shwageraus. 2013. "Modeling of SFR Cores with Serpent-DYN3D Codes Sequence." *Annals of Nuclear Energy* 53: 354–63. https://doi.org/10.1016/j.anucene.2012.08.006.

Gonchar, Andrei Aleksandrovich, and Evguenii Andreevich Rakhmanov. 1989. "Equilibrium Distributions and Degree of Rational Approximation of Analytic Functions." *Mathematics of the USSR-Sbornik* 62 (2): 305. https://doi.org/10.1070/SM1989v062n02ABEH003242.

Johnson, Andrew, Dan Kotlyar, Gavin Ridley, Stefano Terlizzi, and Paul Romano. 2018.

"CORE-GATECH-GROUP/Serpent-Tools: 0.5.2a1 - Pre-Release," June. https://doi.org/10.5281/ZENODO.1301036.

Kim, T K, W S Yang, C Grandy, and R N Hill. 2009. "Core Design Studies for a 1000 MW Th Advanced Burner Reactor." http://www.elsevier.com/copyright.

Leppänen, Jaakko, Maria Pusa, Tuomas Viitanen, Ville Valtavirta, and Toni Kaltiaisenaho. 2015. "The Serpent Monte Carlo Code: Status, Development and Applications in 2013." *Annals of Nuclear Energy* 82: 142–50. https://doi.org/10.1016/j.anucene.2014.08.024.

NEA. 2016. "Benchmark for Neutronic Analysis of Sodium-Cooled Fast Reactor Cores with Various Fuel Types and Core Sizes, NEA/NSC/R(2015)9," no. February: 85. https://www.oecd-nea.org/science/docs/2015/nsc-r2015-9.pdf.

Nikitin, E., and E. Fridman. 2018a. "Extension of the Reactor Dynamics Code DYN3D to SFR Applications – Part I: Thermal Expansion Models." *Annals of Nuclear Energy* 119 (September): 382–89. https://doi.org/10.1016/j.anucene.2018.05.015.

Nikitin, E., and E. Fridman. 2018b. "Extension of the Reactor Dynamics Code DYN3D to SFR Applications – Part II: Validation against the Phenix EOL Control Rod Withdrawal Tests." *Annals of Nuclear Energy* 119 (September): 411–18. https://doi.org/10.1016/j.anucene.2018.05.016.

Nikitin, E., and E. Fridman. 2018c. "Extension of the Reactor Dynamics Code DYN3D to SFR Applications – Part III: Validation against the Initial Phase of the Phenix EOL Natural Convection Test." *Annals of Nuclear Energy* 119 (September): 390–95. https://doi.org/10.1016/j.anucene.2018.05.017.

Nikitin, E., E. Fridman, and K. Mikityuk. 2014. "Solution of the OECD/NEA Neutronic SFR Benchmark with Serpent-DYN3D and Serpent-PARCS Code Systems." *Annals of Nuclear Energy* 75: 492–97. https://doi.org/10.1016/j.anucene.2014.08.054.

Pusa, Maria. 2011. "Rational Approximations to the Matrix Exponential in Burnup Calculations." *Nuclear Science and Engineering* 169 (2): 155–167. https://doi.org/10.1016/j.anucene.2010.10.019.

Pusa, Maria, and Jaakko Leppänen. 2010. "Computing the Matrix Exponential in Burnup Calculations." *Nuclear Science and Engineering* 164 (2): 140–50. https://doi.org/10.1016/j.anucene.2010.10.019.

Rohde, Ulrich, Soeren Kliem, Ulrich Grundmann, Silvio Baier, Yuri Bilodid, Susan Duerigen, Emil Fridman, et al. 2016. "The Reactor Dynamics Code DYN3D - Models, Validation and Applications." *Progress in Nuclear Energy* 89: 170–90. https://doi.org/10.1016/j.pnucene.2016.02.013.

Shwageraus, Eugene, and Pavel Hejzlar. 2009. "Decay Heat in Fast Reactors with Transuranic Fuels." *Nuclear Engineering and Design* 239 (12): 2646–53. https://doi.org/10.1016/j.nucengdes.2009.07.010.