



**HAL**  
open science

## Automated selection of nuclides and reactions of interest in a depletion simulation. Precision loss estimation for multiple outputs

Marc Ernoult, Jiali Liang, Xavier Doligez, Nicolas Thiollière, Olivier Meplan, Sandra Bouneau, Sylvain David

### ► To cite this version:

Marc Ernoult, Jiali Liang, Xavier Doligez, Nicolas Thiollière, Olivier Meplan, et al.. Automated selection of nuclides and reactions of interest in a depletion simulation. Precision loss estimation for multiple outputs. Progress in Nuclear Energy, 2021, 135, pp.103702. 10.1016/j.pnucene.2021.103702 . hal-03171319

**HAL Id: hal-03171319**

**<https://hal.science/hal-03171319>**

Submitted on 20 Oct 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

## Highlights

### **Automated selection of nuclides and reactions of interest in a depletion simulation. Precision loss estimation for multiple outputs**

Marc Ernoult, Jiali Liang, Xavier Doligez, Nicolas Thiollière, Olivier Meplan, Sandra Bouneau, Sylvain David

- New physics-based method and explicit algorithm for automatic selection of isotopes and reactions.
- 3 simplification processes combined for better efficiency.
- Up to 40% reduction of calculation costs without significant loss in precision.

# Automated selection of nuclides and reactions of interest in a depletion simulation. Precision loss estimation for multiple outputs

Marc Ernoult<sup>a,\*</sup>, Jiali Liang<sup>a</sup>, Xavier Doligez<sup>a</sup>, Nicolas Thiollière<sup>b</sup>, Olivier Meplan<sup>c</sup>, Sandra Bouneau<sup>a</sup> and Sylvain David<sup>a</sup>

<sup>a</sup>Université Paris-Saclay, CNRS/IN2P3, IJCLab, 91405 Orsay, France

<sup>b</sup>Université de Nantes, CNRS/IN2P3, IMTA, Subatech, 44307 Nantes, France

<sup>c</sup>Université Grenoble Alpes, CNRS/IN2P3, LPSC, 38026 Grenoble, France

---

## ARTICLE INFO

**Keywords:**  
depletion  
isotopes selection  
fuel cycle  
approximation

## ABSTRACT

To accelerate depletion calculations, it is common to only consider a selection of isotopes. Simulating only a limited number of nuclides leads inevitably to some bias on the outputs of the depletion simulation. Most methods used are either implicit or dedicated to specific cases. This paper presents a new method, based on physics consideration and an explicit algorithm, that automatically selects isotopes and reactions for a depletion calculation. This method uses the creation and simplification of a nucleus tree gathering and linking together all nuclides that can possibly be created during the depletion process. This is performed by recursively adding nuclides possibly created from nuclides previously in the tree through decay or nuclear reaction. After the creation of the tree, it can be simplified by automatically cutting nuclides and reaction out through 3 processes: half-life threshold, cross-section threshold and reaction type cut. Using these simplification processes, the presented method leads to important calculation cost reduction (up to 40%) with effect on outputs lower than 3 times their stochastic standard deviation.

---

## 1. INTRODUCTION

Depletion calculations, in which the evolution of the isotopic composition of the fuel in a nuclear reactor's core is simulated, are essential to the understanding of physics and behavior of nuclear reactors as well as fuel cycles.

The most common way to do depletion calculation is to use a depletion module, that solves the Bateman equation, coupled with a neutronic transport code that calculate reactions rates or cross-sections. It can be done within the transport code when a depletion module is included like in SERPENT [14], MCNP [6] or TRIPOLI [11] or by coupling an external depletion module with a transport code like in SMURE [16].

In fuel cycle simulators, in order to decrease calculation cost of these depletion calculations, the neutronic transport calculation is often avoided and replaced by a meta-model constructed based on more standard depletion calculation. Many types of meta-models can be used in these cases and their calculation needs are very diverse. Nevertheless, there is always cost for both the creation of the meta-model and its multiple uses.

The simplest depletion meta-model used in fuel cycle simulator is nearest-neighbor interpolation, based on recipes, like it is possible in CYCLUS [8]. This kind of meta-model simply consists in taking the output of the pre-simulated depletion that is the closest to currently simulated fuel, often allowing re-normalization of these outputs. They are designed as fast as possible at run time. For these meta-models, the overall precision of the predictions is directly linked with the number of pre-simulated depletion calculations in the database. Lowering the calculation cost of each depletion calculation enable more calculations and thus a better precision.

Another common kind of depletion meta-models is based on artificial neural network predicting average cross-sections that are then used in numerical solving of Bateman equation like in CLASS [18] and FITXS [7]. These neural networks need to be trained on a set of standard depletion calculations distributed over the application domain of the model. The creation of such database implies a large number of depletion calculations and therefore a high total calculation cost. Simplifications that could reduce the calculation cost of these depletion calculations would decrease the creation cost of this kind of meta-model. Furthermore, the trained predictor does not predict directly the isotopic concentration in the depleted fuel, but predicts one group cross-sections that are then used to solve the

---

\*Corresponding author :

✉ [ernoult@ipno.in2p3.fr](mailto:ernoult@ipno.in2p3.fr) (M. Ernoult)

ORCID(s): 0000-0002-9610-3850 (M. Ernoult)

Bateman equation. If the implementation of such numerical Bateman solving is really close to the methods used in depletion modules of transport codes, the simplifications done for the main depletion module can be used in the meta-model Bateman solving method and similar acceleration can be achieved. This is the case for the depletion module of CLASS which is really close to the Runge-Kutta method of SMURE.

For meta-model using predictors directly for the isotopic composition of the spent fuel, like the sparse polynomial chaos expansion available in TR\_EVOL [24] or the neural networks available in CYCLUS [2], no decrease of running time inside the fuel cycle simulator will probably occur.

Other codes avoid direct use of meta-models specific to the fuel cycle simulator but use external codes chosen for its simplicity and small running time. For example, COSI [5] use CESAR and CYCLUS can use ORIGEN through the Cyborg module [25]. But even if this case, CESAR and ORIGEN calculations rely on cross sections library created using database of calculation done with more precise codes for which calculation cost evolution are similar to transport code coupled with depletion module. Furthermore, the Bateman solver inside the depletion tool is expected to benefit of calculation cost decrease in a similar way as the depletion module SMURE tested in this work.

Any meta-model for faster depletion module needs numerous depletion calculations, at least during their creation. Previous studies have shown that meta-models can achieve precision very close to the reference calculation and that therefore there is no need to use less approximation for database creation than the one that will be used in the meta-model [13, 2, 7]. Therefore, calculations for the databases can be simplified without negative impact on the final precision of the meta-model.

Exploratory studies trying to map possible neutronic behavior of wide range of systems [10, 26] can also lead to very large number of depletion calculations. In all these cases, each depletion calculation needs to be as fast as possible without losing precision on the results of interest.

To accelerate those depletion calculations, it is common to only consider a selection of isotopes [14, 18, 7]. Keeping only a small number of different fission products or even only the actinides are often made possible by explicit options in the code (like in SERPENT [14] or CLASS [18]) or even the only possibility for some codes that only simulate actinides (like in FITXS [7]). However, the way the selected nuclides are selected is not always clear and sometimes owes more to history than to physics-based and verified algorithm. Simulating only a limited selection among all existing nuclides leads inevitably to some bias on the outputs of the simulation, but the implicit method of selection and the absence of verification make it very difficult to know how much to trust the outputs produced, especially for indirect output like decay heat after long irradiation times.

As a possible solution to this difficulty, this paper presents a method, based on physics consideration and an explicit algorithm, that automatically selects isotopes and reactions that should be simulated in a depletion calculation.

## 2. METHODS OF SIMPLIFICATIONS

This section describes methods available in SMURE [16] for several years. They allow automatic selection of nuclides to be considered in a depletion simulation. These methods have been presented in the PhD theses that have contributed to its implementation [12, 19, 17]. However, these PhD theses in French are not easily available to the non-French-speaking community.

These methods have also, to this day, never been subject to a proper analysis to determine advantages and disadvantages regarding precision of the simulation and calculation costs.

### 2.1. Creation of a Nucleus Tree

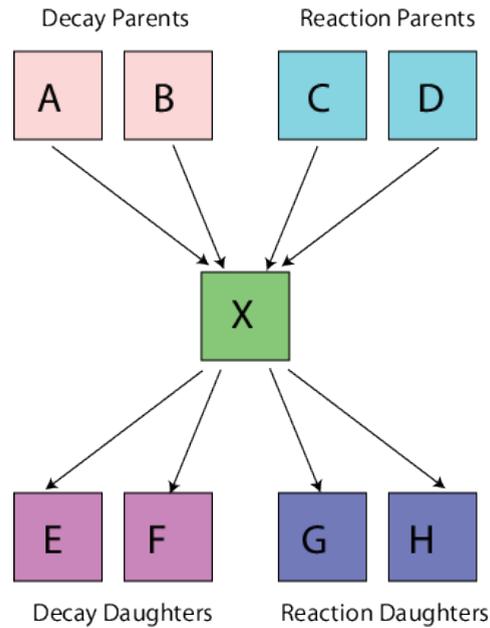
In depletion calculation, the numerical resolution of Bateman equations asks for a list of all isotopes that will be simulated before the beginning of the simulation. To construct this list of nuclides, SMURE gives the option to automatically create a Nucleus Tree gathering nuclides of interest and links between them.

The creation of the nucleus tree is a recursive process starting from the nuclides of the initial isotopic composition of the simulated material set out by the user.

From each nucleus of the initial composition, two kinds of nuclei can be created.

The first kind are nuclei created after radioactive decay, named **decay daughters**. These decay daughters are determined from decay data read in nuclear databases like JEFF [9] or ENDF [4]. All decay daughters regardless of probability of decay and half-life are added to the nucleus tree. Multiple decay daughters are managed by averaging the branching ratios over standard reactor spectra (thermal and fast are available) as described in [12]. The initial nuclide is then a decay parent of its decay daughters.

The second kind are nuclei created after nuclear reactions, named **reaction daughters**. These reaction daughters are by looking all available reaction cross-sections on the considered nuclide in nuclear cross-section databases like JEFF [9] or ENDF [4] then the nuclide created when each available reaction occurs are added to nucleus tree, the decay daughter of these newly added nuclide will need another run of decay daughter listing to be added to the nucleus tree. For each of these cross-sections, the nuclide created when the reaction occurs is added to the nucleus tree regardless of cross-section value. Actually, these values cannot be known because the creation of the nucleus tree is performed before any neutronic simulation calculation and thus without knowing the spectrum of the reactor. Multiple reaction daughters for a single reaction are managed by branching ratio averaged over standard reactor spectra as described in [12] The initial nuclide is then a reaction parent of its reaction daughters.



**Figure 1:** Building block of a nucleus tree : a nuclide and its neighbors

Metastable nuclides, i.e. excited state of the corresponding nuclide, are treated as independent nuclide for the nucleus tree creation and simplification. Their decay to the ground state is treated as a radioactive decay. This is indicated with a star (\*) at the right of the element name in the rest of the paper.

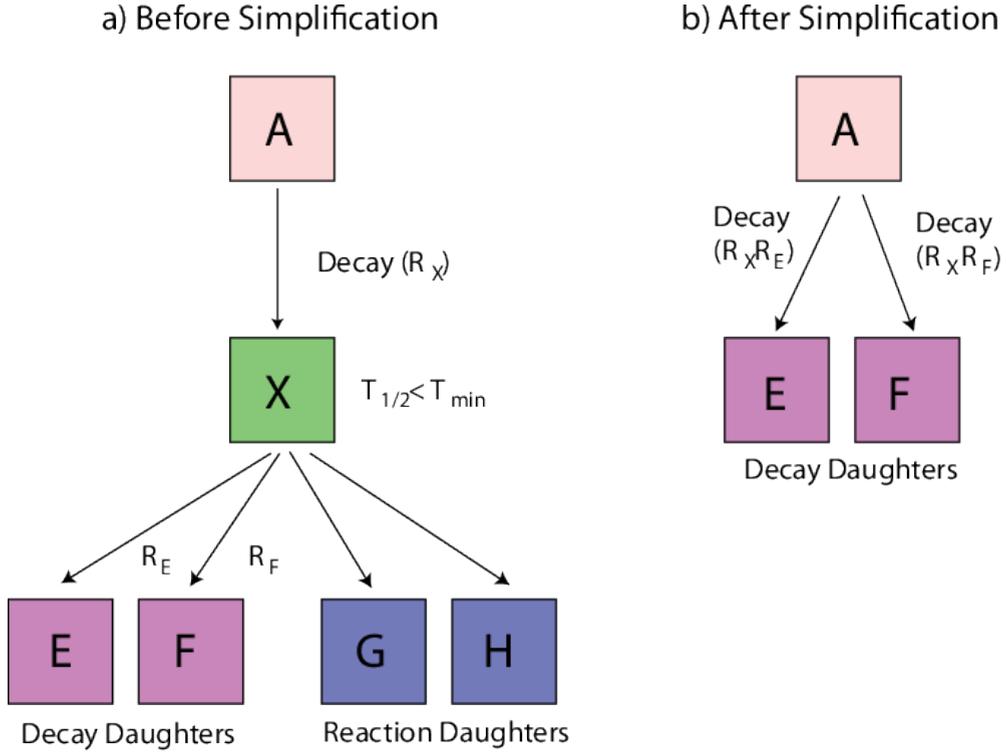
The recursive process converges when the Nucleus Tree graph become cyclic and all newly daughters found are already in the tree. The nucleus tree then regroups numerous nuclides linked together by decay and reaction connections. Each nuclide in the tree is linked with four kinds of neighbors as presented in figure 1 : reaction parents, decay parents, reaction daughters and decay daughters.

Created from material in a standard PWR UOx fuel, the generated nucleus tree usually regroups more than 2000 nuclides. A number of nuclides so large is not practical in a simulation. To reduce the number of nuclides several automated simplification methods are available.

## 2.2. Half Life threshold

The first simplification method uses a threshold on half-life. This half-life threshold has a default value of 1 hour but can be put arbitrarily lower or higher by the user. The simplification process is described in figure 2.

In the example used to illustrate the method, the nucleus A decays into the nucleus X with a branching ratio of  $R_X$ . This nucleus X decays into the nuclei E and F with respective branching ratios  $R_E$  and  $R_F$  or produces through nuclear reactions its reactions daughters G and H. If the nucleus X has a half-life  $T_{1/2}$  shorter than the threshold, the simplification process make the approximation that its decay is instantaneous. Its instantaneous decay lead to an inventory of zero for X that does no longer need to be registered or simulated. This simplification process could lead to significant deviation for short-lived nuclides with a high production rate that still have non-trivial saturated inventories. However, to check this kind of situation, we would need to intertwine the half-life threshold simplification process with



**Figure 2:** Method to decrease number of nuclides by half-life threshold

the cross-section one and this would lead to convergence problem in our algorithm. To avoid it, we choose to rely of the expertise of the user guiding a choice of an adapted threshold.

Because the decay is instantaneous, no nuclear reaction can happen to the intermediate nucleus X, the reaction daughters of X, G and H are therefore not produced. If X was their only parents they are removed from the nucleus tree. The decay daughters of X, E and F become direct decay daughters of A, with respective branching ratios of  $R_X * R_E$  and  $R_X * R_F$ .

If the cut nuclide is reaction daughter of A instead of a decay daughter the process behaves similarly but the decay daughters of X, E and F become direct reaction daughters of A.

Nuclides present in the initial materials are never cut even if their half-life is below the threshold.

### 2.3. Cross-section Threshold

The second method of nucleus tree simplification aim to disregard non-significant nuclear reactions, namely reactions with very low average occurrence. Because the simplification of the nucleus tree is done before any depletion calculation and the simplified tree is kept for the whole depletion, it is impossible to know if one nuclide will be present in high quantity during the depletion and the quantity of nuclide cannot be used. The significance of nuclear reactions is therefore evaluated using one group cross-section. However, because the nucleus tree is built and simplified before the first neutronic calculation, real one group cross-sections are unknown.

To overcome this limitation, a representative average one group cross-section is calculated for each reaction in the nucleus tree with the formula:

$$\langle \sigma \rangle = \frac{\int_a^b \sigma(E) d(\log(E))}{\int_a^b d(\log(E))} = \frac{\int_a^b \sigma(E) \frac{dE}{E}}{\int_a^b \frac{dE}{E}} \quad (1)$$

This integration of the cross-section along the logarithm of the energy is equivalent to use a  $1/E$  spectrum, which corresponds to the approximation that the spectrum is flat in lethargy like in a case of perfect non-absorbent moderation.

(n, f)	(n, g)	(n,2n)	(n,3n)	(n,4n)	(n, p)
0	1	2	3	4	5
(n, d)	(n, t)	(n,He3)	(n, $\alpha$ )	(n, n+p)	(n, n+d)
6	7	8	9	10	11
(n, n+t)	(n, n+He3)	(n, n+ $\alpha$ )	(n,2n+p)	(n,3n+p)	(n,2n+d)
12	13	14	15	16	17
(n,2n+ $\alpha$ )	(n,3n+ $\alpha$ )	(n,2p)	(n,2 $\alpha$ )	(n,3 $\alpha$ )	(n, n+2p)
18	19	20	21	22	23
(n, n+2 $\alpha$ )	(n, n+3 $\alpha$ )	(n, p+d)	(n, p+t)	(n, p+ $\alpha$ )	(n, d+ $\alpha$ )
24	25	26	27	28	29
(n, d+2 $\alpha$ )	(n, t+2 $\alpha$ )	(n, n+p+ $\alpha$ )	(n,2n+2 $\alpha$ )	(n, n+d+2 $\alpha$ )	(n, n+t+2 $\alpha$ )
30	31	32	33	34	35

**Table 1**

Reactions types usable in SMURE

This approximation keeps reactions with high cross-sections at low energy and is not so far to an idealized thermal reactor. A master thesis ([15]) checked that this kind of cut is efficient even for fast reactors.

This quantity  $\langle\sigma\rangle$  is then compared to a threshold. This cross-section threshold has a default value of  $10^{-4}$  barn but can be put arbitrarily low or high by the user. The simplification process is close to the one used with half-life threshold.

To illustrate the method, consider a nucleus X that decays into the nuclei E and F with respective branching ratios  $R_E$  and  $R_F$  or produces through nuclear reactions its reactions daughters G and H with respective average cross-sections  $\langle\sigma_G\rangle$  and  $\langle\sigma_H\rangle$ . If  $\langle\sigma_G\rangle$  is lower than the threshold, the simplification process makes the approximation the cross-section is exactly zero. This zero cross-section leads to a complete suppression of the reaction and removes the need of even trying to calculate the reaction rate or reading data about it in the databases. The reaction daughter of X, G is therefore not produced. If X was its only parent G is removed from the nucleus tree. If  $\langle\sigma_H\rangle$  is higher than the threshold, the reaction daughter H continues to be considered and produced in the simulation.

Neutron capture on  $^{238}\text{U}$  is never cut even if its average cross-section is below the threshold.

## 2.4. Reaction type

The third simplification method takes advantage of grouping the reactions by their reaction types. In SMURE, up to 36 reactions can be taken into account, they are characterized within the code by a number between 0 and 35. The link between numbers and reactions types is presented in table 1.

Users have the option to force cut all reactions of a certain type out of the nucleus tree. For example if the user forces the cut of reaction type 4, all  $(n, 4n)$  reactions on all nuclides of the nucleus tree are considered to have zero cross-section. Such reactions are then cut out of the nucleus tree by a process identical to the one used to remove reaction below the threshold in the cross-section threshold method.

Before closing this section on simplification method, we must note a fourth simplification of the nucleus tree that occurs without user action: the cut of nuclides and reactions that lack data. If a reaction available in SMURE for a nuclide present in the nucleus tree has no available data, its cross-sections is assumed to be zero and is therefore cut accordingly. If a nuclide present in the nucleus tree is neither stated as stable in the available data nor has any decay half-life given, it is assumed to have an arbitrarily low half-life and is therefore cut accordingly.

## 2.5. Indicator of calculation cost and output size

In the following depletion calculations will be performed to evaluate the performance of these 3 simplification methods. To evaluate their impact on the calculation cost of the simulation, several indicators are considered, linked to the different types of depletion calculation.

The codes used for the test of the methods are SMURE and CLASS. Their depletion modules are identical except for the method used to calculate the one group cross-section used in Bateman equation. SMURE uses MCNP to calculate them while CLASS uses artificial neural networks.

For depletion calculation done with a neutronic physics code, calculation times of different parts of a SMURE simulation are considered:

- **Total time** : the total time of the simulation.
- **Nucleus Tree creation time** : the time needed for the creation and the simplification of the nucleus tree.
- **Runge-Kutta time** : the time spent in the numerical resolution of Bateman equation using Runge-Kutta algorithm.
- **Monte-Carlo time** : the time spent in the Monte-Carlo code used in SMURE to solve the transport equation and calculate the reactions rates.

CRAM method has been recently introduced in SMURE [3], decreasing significantly the time needed for the numerical resolution of Bateman equation. However, this implementation is still very recent and less tested than the Runge-Kutta implementation. Therefore Runge-Kutta method has been kept in this study. CRAM method is expected to be much faster than the Runge-Kutta algorithm, but an extensive decrease in Runge-Kutta time would suggest in potential gain with CRAM methods, especially when a meta-model is used for the neutronic part.

For CLASS depletion calculation done using a meta-model to predict average cross-section, we considered:

- **Nuclides** : total number of nuclides in the tree which gives an indication on the size of the result to be stored between runs of the predictors and sent to them.
- **Actinides** : total number of actinides in the tree ( $Z \geq 90$ ) gives an indication on the size of the result to be stored if only inventories of actinides are needed.
- **Reactions** : total number of reactions in the tree which gives an indication of the number of cross-sections that have to be predicted.
- **Reaction types** : number of different reaction type, indication of the complexity of the Bateman matrix and thus of the code development needed.

These indicators that are directly linked with the number of predictors to be trained and stored give a clear view of the cost of creating and storing the meta-models designed for fuel cycle simulators. For nearest-neighbor meta-model, number of nuclides and actinides gives almost directly the size of the data stored for each recipe and a lower value enable larger database for more precise meta-model.

### 3. REFERENCE SIMULATION

To analyze the effect of the use of a nucleus tree with application of the simplification methods presented in the last section on a depletion calculation, we compared the results of depletion calculations made with SMURE using different level of simplification. The calculation with the fewest cuts in the nucleus tree is then used as a reference.

#### 3.1. Characteristics of the simulated assembly

To serve as a reference simulation on which the effects of the aforementioned simplification, a well-defined PWR UOx assembly already used for many tests has been chosen: the assembly described in the benchmark published by Nuclear Energy Agency (NEA) in January 2013 [23] and summarized in [22].

Because the objective of this study is not to compare ourselves with the benchmark but only to compare the effect of simplification on a representative assembly, some simplifications have been done, specifically on the power history.

To minimize Monte-Carlo source convergence problems, only a slice of assembly of 40 cm has been simulated. It corresponds to around 51 kg of Heavy Metal. The depletion simulated represents an irradiation campaign of 1000 consecutive days at full power, bringing the fuel to a burn-up of  $39465 MWd/t_{HM}$ . For the reference, a half-life threshold of 100 seconds and a cross-section threshold of  $10^{-4}$  barns has been used without cutting any reaction types in SMURE (35 reaction types authorized). The number of particles in the Monte-Carlo transport calculation of each time step of the SMURE calculation has been adjusted to have around 100 pcm of statistical uncertainties on  $k$  at the beginning of the irradiation.

In order to decrease the statistical error, 50 depletion calculations have been made and results have been averaged to create the references values.

	DH at 5y	Total Pu	Total M.A.	K_BOC	K_EOC
average	147 kW	818.7 g	56.6 g	1.214	0.883
std. dev.	0.07%	0.12%	0.30%	112 pcm	176 pcm

**Table 2**

Reference values and Statistical errors on the outputs of interest

<sup>235</sup> U	<sup>237</sup> Np	<sup>238</sup> Pu	<sup>239</sup> Pu	<sup>240</sup> Pu	<sup>241</sup> Pu	<sup>242</sup> Pu	<sup>241</sup> Am	<sup>242</sup> Am*	<sup>243</sup> Am	<sup>242</sup> Cm	<sup>244</sup> Cm
0.07%	0.48%	0.33%	0.17%	0.40%	0.18%	0.28%	0.22%	0.26%	0.74%	0.20%	0.79%

**Table 3**

Statistical error on common actinides 5 years after the end of irradiation

### 3.2. Reference outputs used to evaluate effects of simplifications

To evaluate the impact of simplifications on the precision of the simulation, several outputs will be followed in this paper. These outputs have been selected because they are representative of outputs often of interest in depletion studies.

- **K\_EOC**: the multiplication factor at the end of the irradiation
- **K\_BOC**: the multiplication factor at the beginning of the irradiation. Because this value is not dependent of the nucleus tree, it is only presented for the reference calculation.
- **Total Pu**: the total mass of plutonium (all isotopes with  $Z = 94$ ) in the simulated slice of assembly at the end of the irradiation.
- **Total M.A.**: the total mass of minor actinides (all isotopes with  $Z = 93$  or  $Z \geq 95$ ) in the simulated slice of assembly at the end of the irradiation.
- **DH at 5y**: the total decay heat produced by the simulated slice of assembly, 5 years after the end of the irradiation.

For the reference calculation the decay heat has been calculated with the graphical user interface provided with SMURE [16]. This method uses matrix exponential method for the calculation of inventories during decay and then multiplication of the quantity of each of isotopes present after decay with a Heat factor coming from databases.

The results are presented in table 2.

Using the standard deviation of the result between the 50 reference depletion calculations done, an evaluation of the statistical uncertainty of these results has been done. The statistic used for each of these depletion (which is kept the same for all other calculations done afterwards) guarantees a very low uncertainty, far below 1% as presented in table 2. Statistical uncertainties of major actinides have been calculated for information and are presented in table 3.

For the rest of this study, a script has been developed to automatically calculate the decay heat 5 years after the end of irradiation as linear function of isotopic composition of the spent fuel. To achieve that, a list of isotopes from the spent fuel at discharge and their contribution to the decay heat at 5 years (added with the one from all of their decay daughters and descendant created by decay during the 5-year waiting time) has been created. The first 40 nuclides of this list ordered by contribution to the total decay heat is presented in table 4. The table present the name of the nuclides, their individual contribution to the decay heat in the column "Part of Total DH" and the relative residual that is not explained when considering only this nuclide and the nuclides with larger contribution to the decay heat. The part of decay heat contributed by each nuclide and the part not explained by these nuclides and all other nuclides with bigger contribution are given as a ratio of the total decay heat. For example, the <sup>106</sup>Ru present at the end of irradiation is responsible of 8.3% of the decay heat 5 years after the end of irradiation and 25.5% of the decay heat is not explained by the contribution of <sup>106</sup>Ru and nuclides with higher contribution to decay heat. Using this list of 40 nuclides, decay heat 5 years after the end of irradiation is then calculated using the linear function of these 40 elements, which multiply the corresponding inventories by equivalent heat factors including all heat produced by the corresponding element and all its descendants 5 years after the end of the irradiation.

The method available in SMURE Graphical User Interface is undoubtedly closer to the physics than the used linear function but it is not more precise. The linear function using the selected nuclides can reproduce the decay heat five

Isotope	Part of Total DH	Not yet explained	Isotope	Part of Total DH	Not yet explained
<sup>137</sup> Cs	0,33045	0,66955	<sup>110</sup> Ag*	0,00026	0,00090
<sup>90</sup> Sr	0,16874	0,50082	<sup>242</sup> Am*	0,00021	0,00065
<sup>134</sup> Cs	0,16238	0,33844	<sup>244</sup> Am	0,00018	0,00043
<sup>106</sup> Ru	0,08266	0,25578	<sup>239</sup> Np	0,00008	0,00025
<sup>244</sup> Cm	0,06455	0,19123	<sup>125</sup> Sn	0,00004	0,00017
<sup>238</sup> Pu	0,05378	0,13745	<sup>242</sup> Pu	0,00003	0,00013
<sup>144</sup> Ce	0,04329	0,09415	<sup>151</sup> Sm	0,00002	0,00011
<sup>154</sup> Eu	0,02566	0,06849	<sup>245</sup> Cm	0,00002	0,00009
<sup>241</sup> Pu	0,02396	0,04453	<sup>152</sup> Eu	0,00001	0,00007
<sup>85</sup> Kr	0,01216	0,03237	<sup>119</sup> Sn*	0,00001	0,00005
<sup>239</sup> Pu	0,00659	0,02578	<sup>113</sup> Cd*	0,00001	0,00005
<sup>240</sup> Pu	0,00597	0,01980	<sup>121</sup> Sn*	0,00001	0,00004
<sup>147</sup> Pm	0,00596	0,01384	<sup>237</sup> Np	0,00001	0,00003
<sup>242</sup> Cm	0,00485	0,00899	<sup>236</sup> Pu	0,00000	0,00003
<sup>125</sup> Sb	0,00358	0,00541	<sup>238</sup> U	0,00000	0,00002
<sup>241</sup> Am	0,00278	0,00263	<sup>236</sup> U	0,00000	0,00002
<sup>155</sup> Eu	0,00066	0,00197	<sup>246</sup> Cm	0,00000	0,00002
<sup>238</sup> Np	0,00041	0,00156	<sup>235</sup> U	0,00000	0,00002
<sup>243</sup> Am	0,00039	0,00117	<sup>234</sup> U	0,00000	0,00002
<sup>147</sup> Nd	0,00027	0,00090	<sup>243</sup> Cm	0,00000	0,00002

**Table 4**

Nuclides of importance for Decay Heat 5 years after the end of irradiation

	Ref (in s)	std. dev.	From Total time
<b>NT creation Time</b>	82	3	4%
<b>RK time</b>	804	16	38%
<b>MC Time</b>	1214	1214	58%
<b>Total Time</b>	2100		100%
<b>Nuclides</b>	843	-	
<b>Actinides</b>	87	-	
<b>Reactions</b>	1437	-	
<b>Reaction types</b>	18	-	

**Table 5**

Calculation costs indicators for the reference calculation

years after the end of irradiation with a loss of precision lower than  $1.10^{-5}$ . This loss of precision due to the use of the linear function is order of magnitude lower than the stochastic uncertainty due to our use of Monte-Carlo transport code for reaction rate estimation: ( $1.10^{-5} \ll 7.10^{-4}$ ). This simple fitted linear function have allowed us to streamline the decay-heat calculation and speed its calculation replacing matrix exponential by one matrix product.

### 3.3. Indicators of calculation cost and output size

These 50 reference depletion calculations have been used to evaluate the calculation cost of the simulation without using the nucleus tree and the simplifications.

The results are presented in table 5.

Size of SMURE output files is almost directly proportional to the sum of the number of nuclides and reactions. For each cell, at each time step a name and a value is stored for each of them (inventory for each nuclide, microscopic cross-section for each reaction). In addition, the flux within each cell is recorded as well as the multiplication factor of the whole simulated system. In all cases in this study we have  $Nuclides + Reactions \gg 2$ , therefore, in order to avoid adding another indicator, the file size will be estimated by looking at the number of nuclides and reactions simulated.

Without simplification the calculation time is almost equally divided between Runge-Kutta and Monte-Carlo and the creation and simplification of the nucleus tree takes only 4% of the total calculation time. Even if none of the 36

possible reactions in SMURE have been cut, only 18 are used in the simulation, principally due to lack of data and very low cross-sections.

## 4. EFFECTS OF EACH CUT ON NUCLEUS TREE

To evaluate impact of the three simplification processes presented above on calculation cost and outputs precision, we first tested them independently. For each of the processes, the level of cuts has been increased while keeping the 2 other processes at the level used in the reference calculations. For each of the tested thresholds, comparison has been made with the average of the 50 reference calculations to evaluate the impact of the simplification process on calculation cost and outputs precision.

### 4.1. Half Life threshold

The first simplification process whose impact has been evaluated is the use of half-life threshold. Several depletion calculations have been made for increasingly high half-life threshold and their result have been compared with the average of the 50 reference calculations. The result of the simulation for the reference outputs defined in section 3.2 are presented in figure 3 in which the error introduced by the simplification is normalized to the statistical standard deviation on these outputs within the 50 reference calculations that is presented in table 2.

We can see that below  $5 \cdot 10^4$  seconds (14 h) all outputs stay within 3 stochastic standard deviations and therefore that no effect of the simplification is visible. With thresholds higher than  $5 \cdot 10^4$  seconds, significant errors appear on the total plutonium mass, the total mass of minor actinides and multiplication factor at the end of irradiation. The number of nuclides newly cut is not specifically high, but the cut nuclides are closer to nuclides of interest. The effects at  $5 \cdot 10^4$  seconds comes from the cut of  $^{244}\text{Am}$  which leads to a faster increase of  $^{240}\text{Pu}$  during irradiation.

The decay heat has also a big jump in error at this threshold with errors getting close to 10 standard deviations. It presents an even bigger jump at  $5 \cdot 10^5$  seconds (almost 6 days) with errors approaching 100 statistical standard deviations. This second jump is linked with the cut of  $^{238}\text{Np}$

When these limits on the half-life threshold are exceeded, the errors get large really fast due to the cut of major Am and Np or even Pu isotopes.

In addition to comparison of the results, a comparison of the change of the different calculation cost indicators with the increase of the threshold has been done. The results are presented in figure 4.

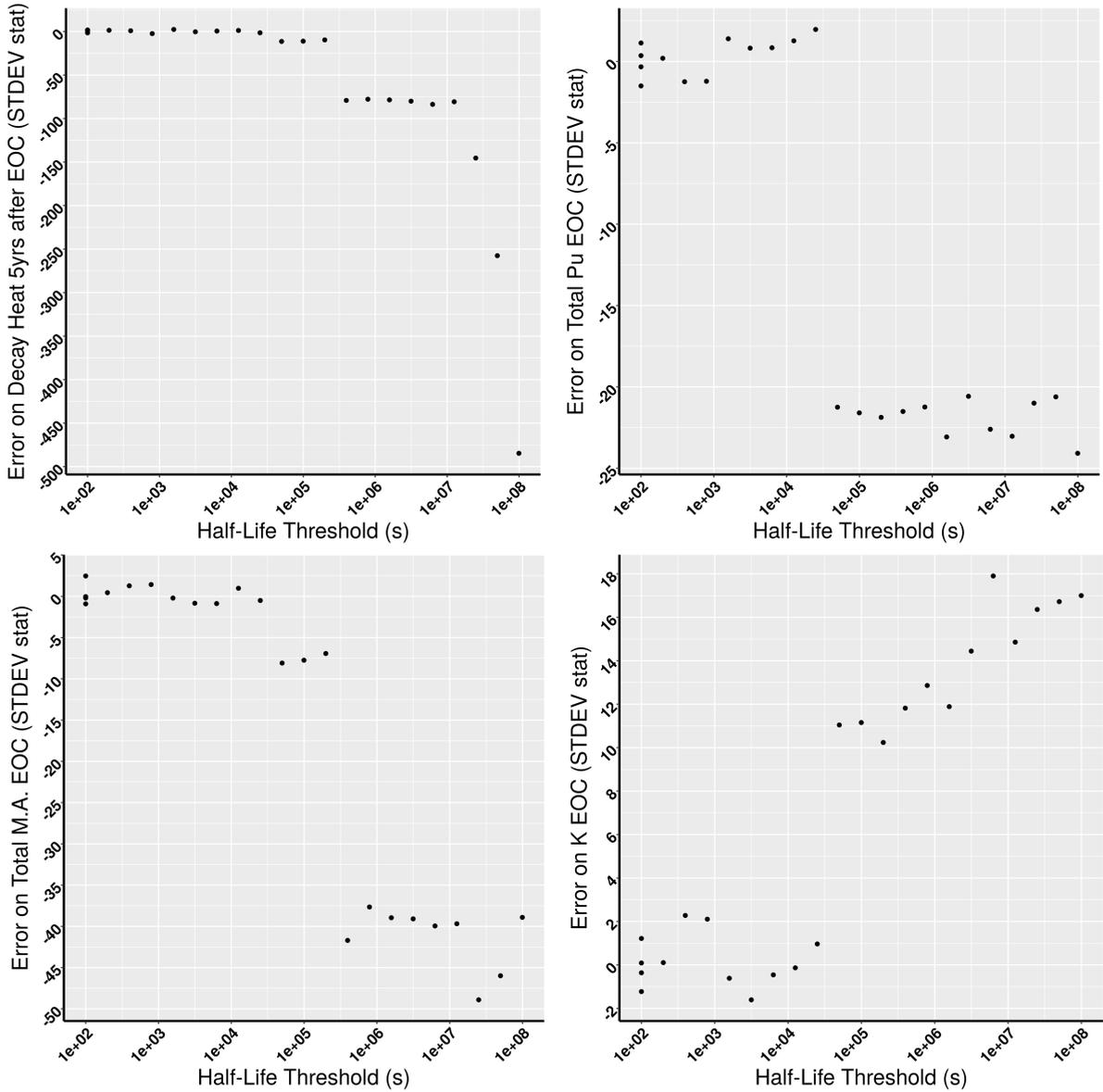
A big decrease of the number of isotopes tracked in the simulation is visible in the figure 4. This is most due to cut of short-lived fission product. This decrease of the number of isotopes tracked leads to a big gain on memory usage, both for storing the results and for their analysis. We see also a fast and exponential decrease of the time needed in the Runge-Kutta solving of the Bateman equation. Part of this decrease is due to a decrease of the size of the Bateman matrix from the decrease of number of isotopes simulated. Furthermore, Runge-Kutta method needs inner time step within the numerical resolution to be smaller than the smallest eigenvalue of the Bateman matrix. Cutting short-lived elements removes the smallest eigenvalue and allows for larger inner time steps in Runge-Kutta, decreasing their number. This big gain on Runge-Kutta time is significant for coupled depletion simulation but has even bigger relative impact for meta-models where the Runge-Kutta resolution takes most of the execution time. There is also a decrease of time needed for transport calculation, but it appears mainly for thresholds higher than  $10^5$  seconds when the precision has already decreased a lot. Before the precision limit, the gain is limited, only some percent.

A reasonable half-life threshold, like  $10^4$  seconds, would allow a big decrease in memory usage and in calculation time, especially in meta-models. However, the creation cost of the predictors would not be significantly impacted because the number of cross-sections does not decrease a lot.

### 4.2. Cross-section Threshold

The second simplification process whose impact has been evaluated is the use of half-life threshold. Several depletion calculations have been made for increasingly high cross-section threshold and their results have been compared with the reference. The result of the simulation for the reference outputs presented in section 3.2 are presented in figure 5 in which the error created by the simplification is normalized to the statistical standard deviation on these outputs within the 50 reference calculations that been presented in table 2.

We can see that below 0.05 barns, all outputs stay within 3 stochastic standard deviations and therefore that no effect of the simplification is visible. After this limit large errors appear on total mass of plutonium and minor actinides as well as decay heat. No effect is visible on the multiplication factor. This big and fast change of outputs is due to cuts

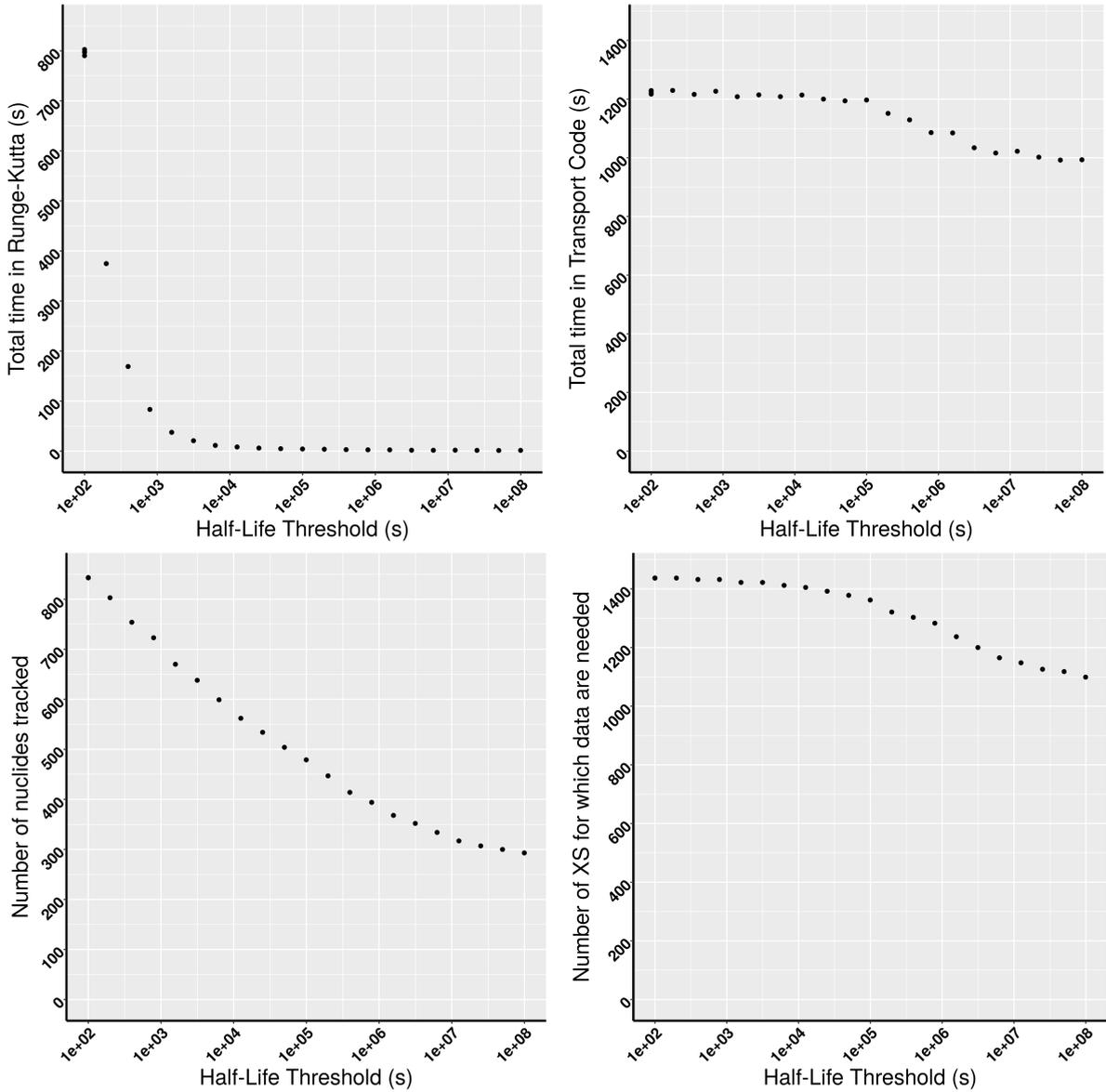


**Figure 3:** Error after simplification by half-life threshold

of major reactions. At 0.05 barns, the reaction (n,2n) on  $^{238}\text{U}$  is cut. At 0.1 barns, the reaction (n, fission) on  $^{238}\text{U}$  is cut. These two cuts create major change on which isotopes are undergoing fission and thus change drastically the fission products created.

The multiplication factor is surprisingly not impacted by this simplification, even after the cut of the reaction (n, fission) on  $^{238}\text{U}$ , which constitute a non-negligible part of the fission reactions. This is because the cuts are only effective in the Bateman resolution and on which reaction rates are recorded in the transport code, but do not change the inner workings of the transport code: all reactions are still simulated within the transport code.

Only a small decrease of the number of isotopes tracked in the simulation is visible in the figure 6. This is most due to cut of actinides far from the uranium that are not created anymore. The linear decrease in the time needed for Runge-Kutta calculation with the lower thresholds is due to the simplification of the matrix caused by the cuts. Almost no effect is seen on the transport calculation time because the cut reactions are still present in the transport calculation and the number of isotopes present in the simulation does not change a lot.



**Figure 4:** Impact of half-life threshold simplification on calculation cost

The only big effect is on the total number of cross-section evaluated where a big decrease is clear. Even with thresholds assuring no precision loss, the number of cross-section simulated can be divided by 3.

A reasonable cross-section threshold, like 0.01 barns, would allow a big decrease in time needed for the creations of meta-models' predictors. However, almost no gain can be achieved in memory usage and in calculation time because the number of isotopes and the Runge-Kutta time do not decrease significantly.

### 4.3. Reaction type

For the third and last simplification process was the cuts done by reaction types. The evaluation process used is exactly the same. The result of the simulation for the reference outputs defined in section 3.2 are presented in figure 7 in which the error introduced by the simplification is normalized to the statistical standard deviation on these outputs within the 50 reference calculations that has been presented in table 2. The reaction types are always cut cumulatively in the decreasing order of their numbers in the table 1.

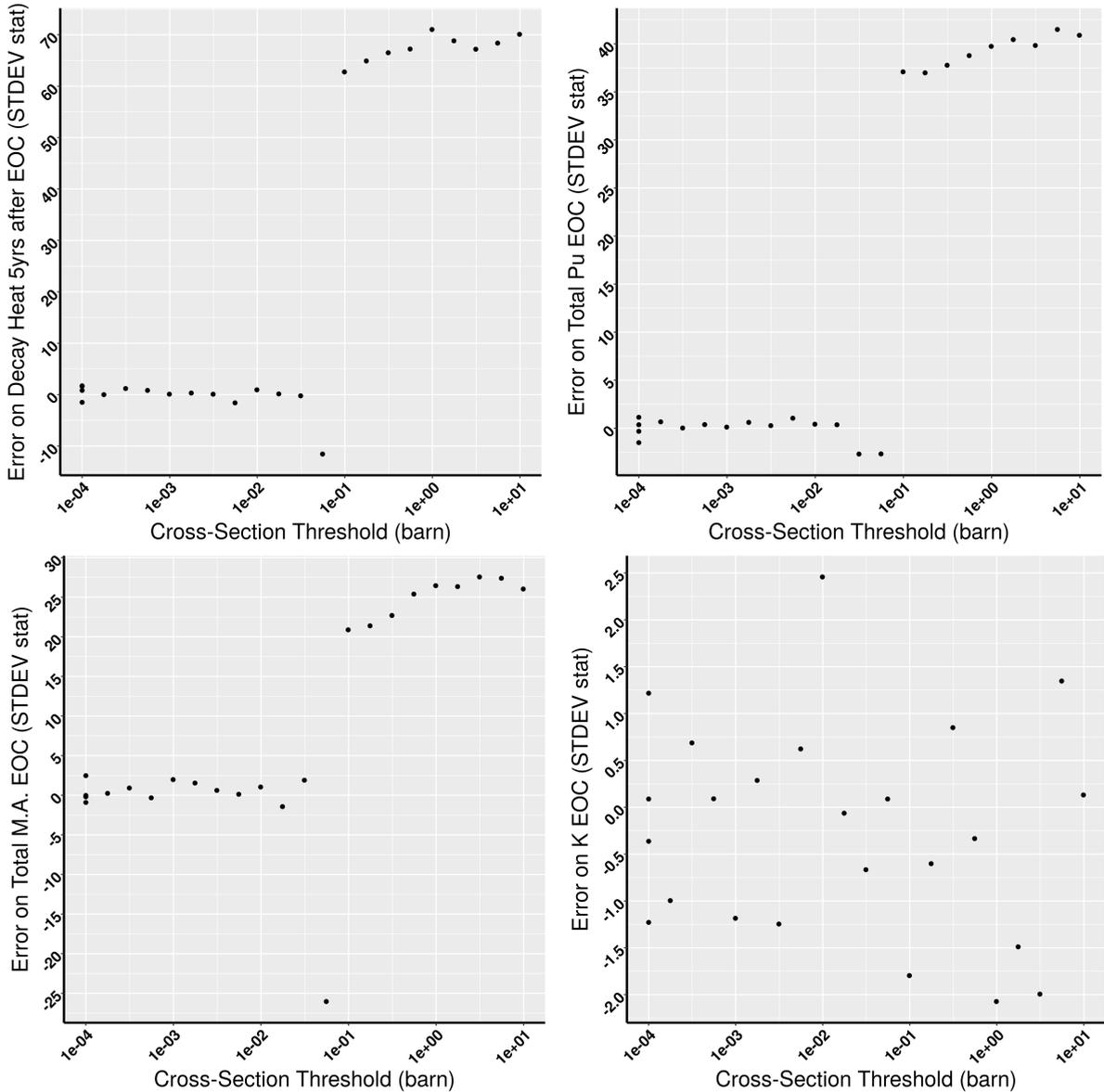


Figure 5: Error after simplification by cross-section threshold

No effect of this simplification process is visible on reference outputs except for the cut of the last reaction type ( $n,2n$ ). This means that as long as capture, fission and ( $n,2n$ ) are simulated, a good precision on the reference outputs can be achieved. For plutonium mass and reactivity, good precision is even possible without ( $n,2n$ ) reactions.

For calculation costs, we can see in figure 8 that only the last 5 reaction types have effects on the time needed for Runge-Kutta and the number of isotopes simulated. A little more reaction types have effect on the total number of cross-section to be predicted while no effect is visible on the time needed for transport calculation even when cutting ( $n,2n$ ). If we limit ourselves to cuts of reactions types causing small errors, it means that the only gain possible is on the number of cross-section to be predicted. And even this gain is small compared to what can be achieved with cross-section threshold.

However, knowing that good precision can be achieved with only 3 reaction types can facilitate the development process of new depletion modules during which each reaction type taken into account needs specific code to be written.

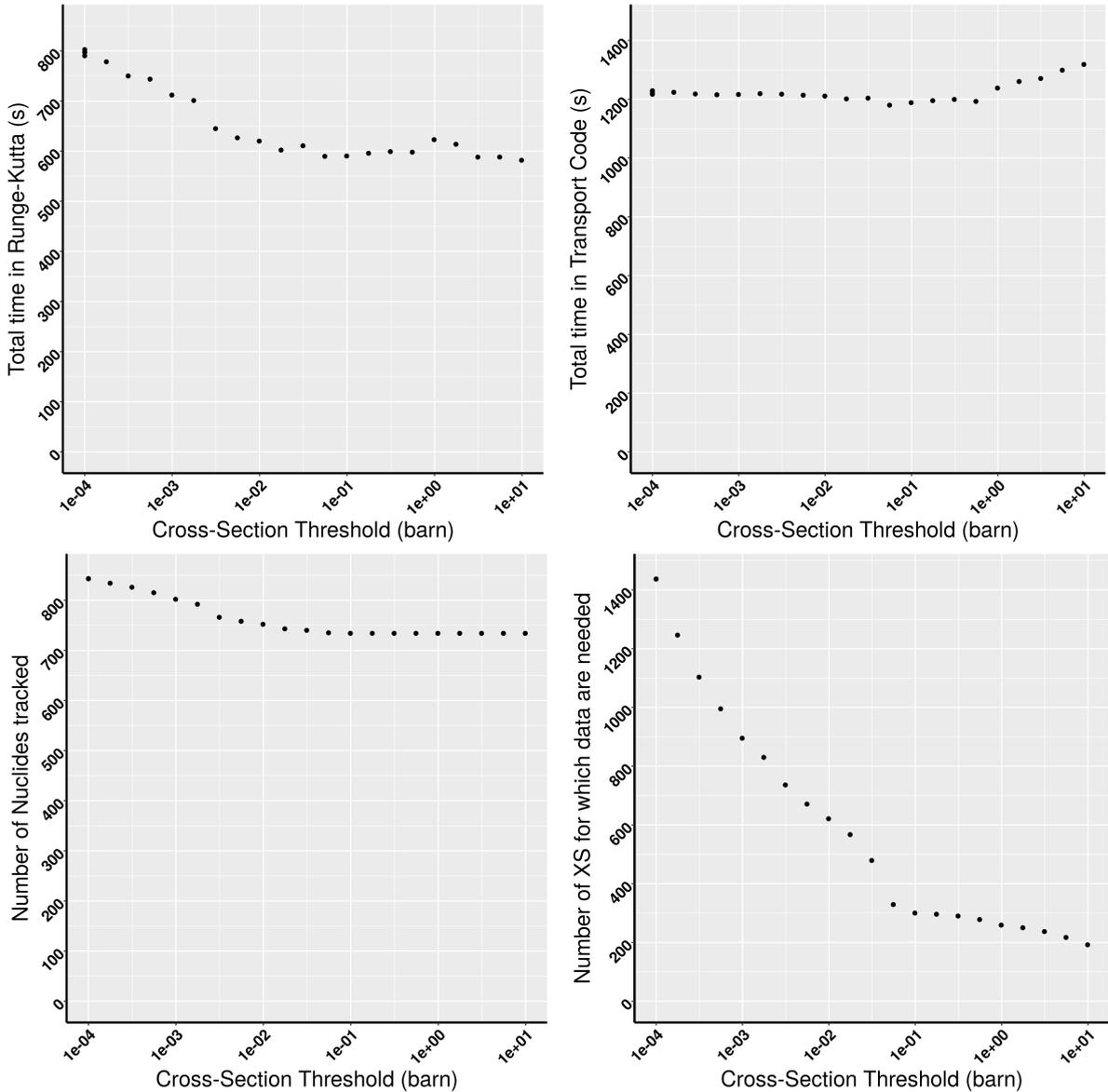


Figure 6: Impact of cross-section threshold simplification on calculation cost

## 5. SENSIBILITY OF OUTPUT TO NUCLEUS TREE CUTS

### 5.1. Calculation methods

After the evaluation of the impact of each of these simplification processes independently, a specific study has been carried out to evaluate their effect in interaction. For this purpose a sensitivity analysis has been done using the sensitivity package [20, 1] available within R [21].

Following the recommended method in the publications, several calculations have been made with half-life threshold, cross-section threshold and number of reaction types cut sampled through an Latin hypercube sampling [27]. Because half-life threshold, cross-section threshold varies in a large range covering several orders of magnitudes, the log of these values has been sampled in the Latin hyper-cube sampling and not their value. Half-life threshold ( $HL$ ) has been sampled in this way between  $10^2$  and  $10^9$  seconds and cross-section threshold ( $XS$ ) between  $10^{-4}$  and 10 barns. Because correlations can be obtained through the package only with uniform Latin hyper-cube sampling, correlations with the log of the threshold and not the threshold themselves have been obtained.

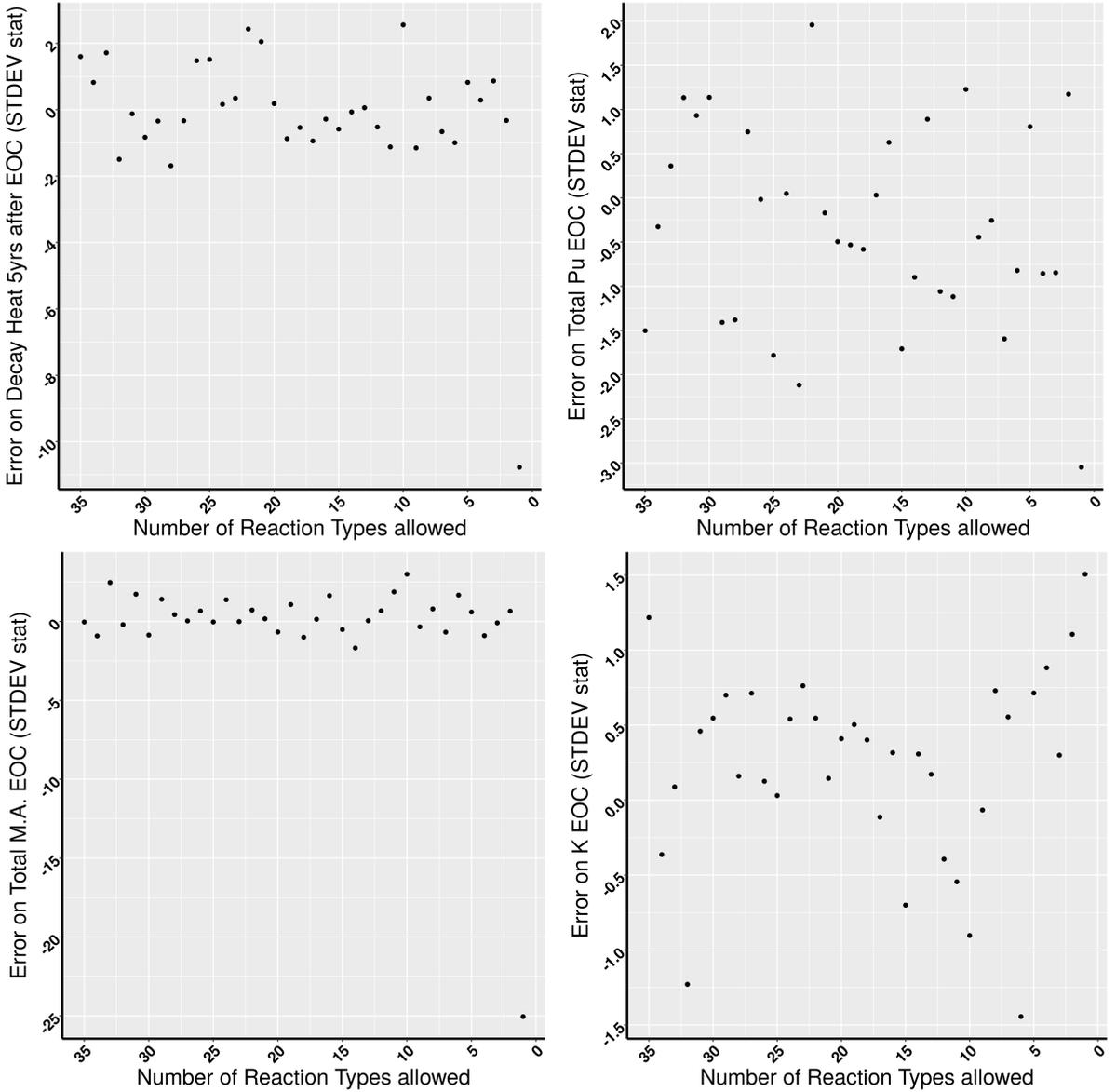
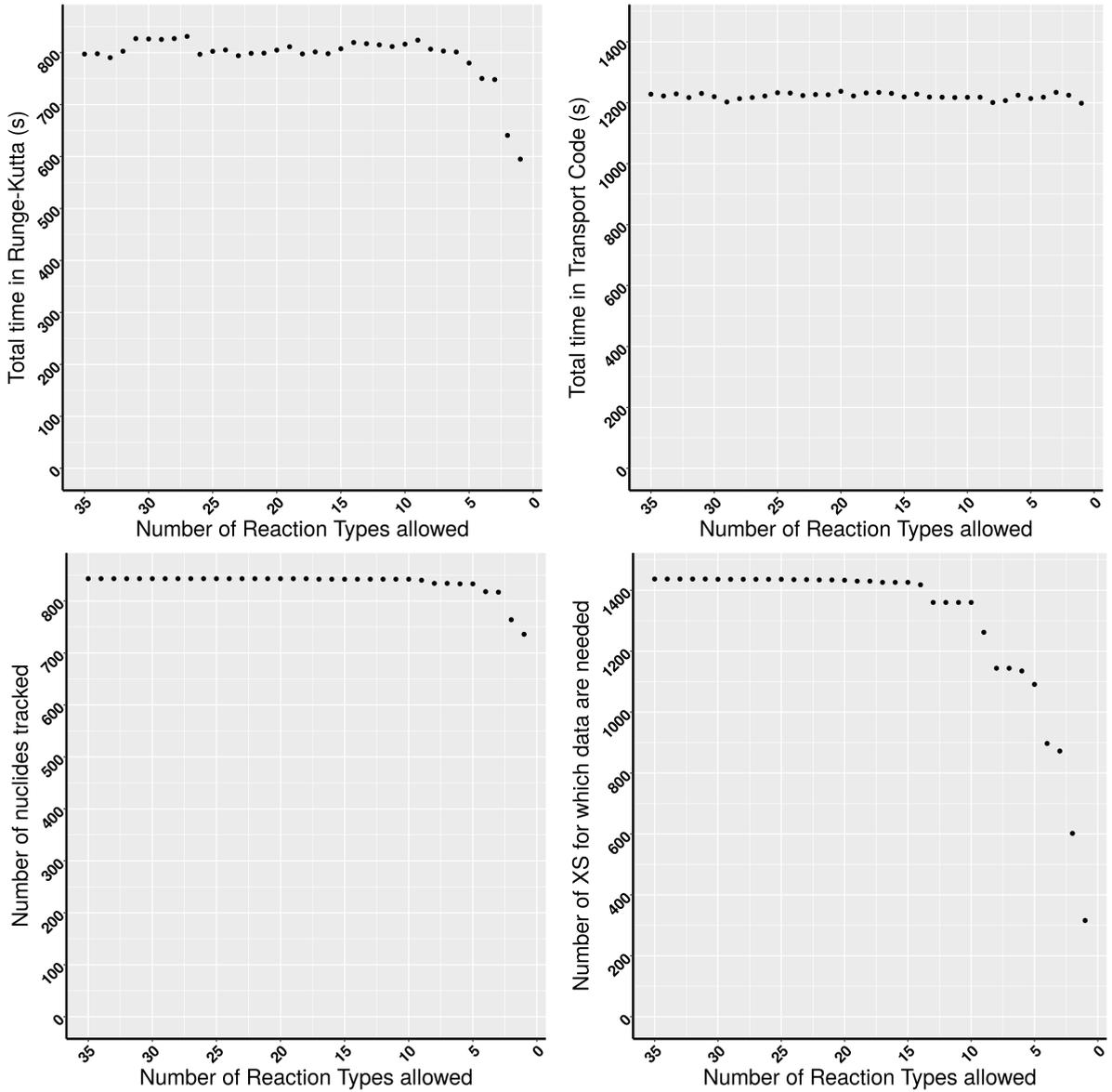


Figure 7: Error after simplification by reaction type

150 sets of thresholds have been sampled this way to assure a good precision on the correlation coefficients.

## 5.2. Correlation between thresholds and outputs

The analysis of the 150 depletion calculation made using the sensitivity package [20, 1] gave the correlation coefficients shown in table 6. The half-life and cross-section thresholds considered span multiple orders of magnitude and that, as shown in the previous section, effects on reference outputs and indicator are not so large, the direct correlation coefficients between these thresholds and the outputs would be really close to 0. To better follow the effect of each of the simplification process the correlation coefficients with the log of the half-life thresholds and cross-section thresholds have been calculated. Because  $\log(HL)$  and  $\log(XS)$  are strictly monotonous functions in the respective thresholds the sign of the coefficient is kept. A correlation coefficient of  $-1$  between  $\log(HL)$  and a calculation time would mean that when the half-life threshold is multiplied by 10, the calculation time decrease of 1%. The small size of the sample leads to fairly high errors that forbid us to go into a detailed analysis, but they are not an obstacle for our



**Figure 8:** Impact of reaction type simplification on calculation cost

goal which is to find the major contributions of each simplification process.

The values of these correlation coefficients confirm what was observed with the one at a time analysis. Increase of half-life threshold allows a good decrease of calculation cost, both in the Runge-Kutta part of the code and the transport one. It also allows a big decrease in the number of nuclides in general and actinides in particular that are followed in the simulation, but the effect on the number of reactions and reaction types is relatively small. We should keep in mind that these effects are calculated over the full range of the possible half-life threshold and the non-negligible correlation coefficients with the outputs indicate that some of these decreases in calculation costs indicators are linked with a loss of precision on the outputs. Increase of cross-section threshold has a small influence on calculation time of full depletion calculation however leads to a big decrease in the number of reactions and reactions types to be predicted within meta-models. In a similar way that with half-life, effects on the outputs are observed so attention should be paid to not increase the threshold excessively for it could cause large losses in precision on decay heat and inventories. However, the multiplication factor evaluation seems safe even at high cross-section threshold because the correlation is

	log(HL)	error	log(XS)	error	Allowed RT	error
Nucleus Tree Creation Time	-0.11	0.08	-0.02	0.08	-0.05	0.08
Runge-Kutta time	-0.55	0.05	-0.13	0.08	0.12	0.08
Monte-Carlo Time	-0.61	0.05	0.06	0.08	0.05	0.08
Total Time	-0.74	0.03	-0.03	0.08	0.10	0.08
Nuclides	-0.94	0.01	-0.25	0.07	0.13	0.08
Actinides	-0.91	0.01	-0.33	0.07	0.16	0.08
Reactions	-0.29	0.07	-0.90	0.01	0.15	0.08
Reaction types	-0.11	0.08	-0.86	0.02	0.31	0.07
DH 5y after EOC	-0.42	0.11	0.06	0.14	0.27	0.13
Total Pu EOC	-0.44	0.10	0.73	0.07	0.00	0.14
Total MA EOC	0.33	0.13	-0.18	0.13	0.18	0.14
K_EOC	-0.02	0.08	0.15	0.08	-0.06	0.08

**Table 6**

Correlation between outputs of interest and the threshold of the cuts

very small. Cutting more reactions types has generally a very low correlation with all measured indicators and outputs, even with the number of types of reactions in the simulation confirming the lack of effect of most cuts observed in the one at a time study.

## 6. CONCLUSIONS

In this paper, we presented a method, based on physics consideration and an explicit algorithm, that automatically selects nuclides and reactions for a depletion calculation. The method is based around the creation and simplification of a nucleus tree gathering and linking together all nuclides that can possibly be created during the depletion process. The nucleus tree is created by recursively adding nuclides possibly created from nuclides previously in the tree through decay or nuclear reaction. After the creation of the tree, it can be simplified by automatically cutting nuclides and reactions out through 3 processes: half-life threshold, cross-section threshold and reaction type cut.

The half-life threshold process removes, from the nucleus tree, nuclides with half-life below the threshold and reconnects the other nuclei of the tree as if the cut nucleus was an unstable intermediate state with instant decay. It allows an important decrease in calculation cost of Runge-Kutta resolution of Bateman equation and on the number of nuclei to keep track of, principally among the fission products, without losing precision. Some decrease of calculation cost of Monte-Carlo transport code is possible but only at the expense of big loss in precision.

The cross-section threshold process removes, from the nucleus tree, reactions with average cross-sections below the threshold and removes the nuclides that are not any more linked to the rest of the nucleus tree. It allows a decrease in calculation cost of Runge-Kutta resolution of Bateman equation but smaller than the one from half-life threshold. There is also a significant decrease in the number of nuclides and reactions to be followed leading to a decrease of output file size and memory use which, as described in section 2.5, is practically proportional to the sum of these numbers.

The reaction type cut removes all reactions of the certain type from the tree. Only the cut of the last types of reaction have effects (reaction type <10). Even for these lasts reactions types, the decrease in calculation costs of any sort is small. However, these cut of last reactions types allows a significant decrease of number of cross-section to predict while keeping reasonable precision on all outputs.

Using these simplification processes for a depletion calculation performed with a coupled transport neutronic code that has already build-in capabilities to manage numerous reaction types, the most efficient use of the presented method is to set a half-life threshold around  $5 \cdot 10^4$  s ( around 14 hours) and a cross-section threshold of 0.05 barns which leads to important calculation cost reduction as presented in table 7 with effect on outputs lower than 3 times their stochastic standard deviation.

The analysis of these simplification processes also shows that simulating only (n, g), (n, fission) and (n, 2n) reactions in a depletion module is enough to calculate actinides inventories, reactivity evolution and to predict decay heat after several years of cooling. When creating new depletion modules where each reaction type has to be re-implemented

	Without NT	With NT	Rel. Gain
<b>NT creation Time (in s)</b>	0	88	+100%
<b>RK time (in s)</b>	804	7	-99%
<b>MC Time (in s)</b>	1214	1098	-10%
<b>Total Time (in s)</b>	2018	1193	-41%
<b>Nuclides</b>	843	508	-40%
<b>Actinides</b>	87	55	-37%
<b>Reactions</b>	1437	594	-59%
<b>Reaction types</b>	18	5	-72%

**Table 7**

Calculation costs (in seconds) and their differences between naive and efficient nucleus tree use

specifically, limit oneself to implement only these 3 reactions type can be enough to calculate most fuel cycle observable while decreasing the complexity of code.

The paper focuses of typical problems seen in fuel cycle studies, with a prime importance on inventories of wastes and fissile content as well as transportation, usually limited by decay heat. All of this in a context where most reactors are PWRs.

However, the results can be extrapolated to other problems. In other reactor technologies, fuel composition tends to have a lower impact on the neutron spectrum. Therefore, we expect less correlation between the simulation of the different nuclides and such less effect of the simplification on the major nuclides (which are not cut until very drastic simplifications). The study done in sodium cooled fast reactor in [15] goes in this direction and seems to show that our result would be conservative for other simulation of other systems but such kind of extrapolation of guideline should be done extremely carefully. If only criticality evolution is needed from the depletion calculation even bigger cuts are possible and calculation can be achieved even faster. The extension of our results on different outputs of the code is more difficult. Loss of precision due to simplification on decay heat after more than 5 years of cooling will probably be even lower than on the decay heat studied in this paper. However other output, or even decay heat with less than 5 years of cooling, could suffer large loss of precision with simplification levels that appear mild in this study. Any decay heat evaluated for time lower than 3 times the half-life threshold will for example most probably be completely wrong due to the cut of major contributors. For radio-protection studies extrapolation is even more difficult because limitations are often driven by a very few nuclides with small inventories.

## References

- [1] Asserin, O., et al., 2001. Global sensitivity analysis in welding simulations -what are the material data you really need? *Finite Elements in Analysis and Design* 47, 1004–1016.
- [2] Bae, J.W., Rykhlevskii, A., Chee, G., Huff, K.D., 2020. Deep learning approach to nuclear fuel transmutation in a fuel cycle simulator. *Annals of Nuclear Energy* 139, 107230.
- [3] Becker, M., 2020. Code and data enhancements of the mure c++ environment for monte-carlo simulation and depletion, in: *Kerntechnik*, Leipzig, Germany.
- [4] Brown, D.A., Chadwick, M., Capote, R., Kahler, A., Trkov, A., Herman, M., Sonzogni, A., Danon, Y., Carlson, A., Dunn, M., et al., 2018. Endf/b-viii.0: the 8th major release of the nuclear reaction data library with cielo-project cross sections, new standards and thermal scattering data. *Nuclear Data Sheets* 148, 1–142.
- [5] Coquelet-Pascal, C., Tiphine, M., Krivtchik, G., Freynet, D., Girieud, R., Eschbach, R., 2015. Cosi6: a tool for nuclear transition scenarios studies, in: *Proc. of Global*, pp. 21–24.
- [6] Fensin, M.L., Galloway, J., James, M., 2015. Performance upgrades to the mcnp6 burnup capability for large scale depletion calculations. *Progress in Nuclear Energy* 83, 186–190.
- [7] Halász, M., Szieberth, M., Fehér, S., 2017. Fitxs: A fast and flexible burn-up scheme based on the fitting of one-group cross-sections. *Annals of Nuclear Energy* 104, 267–281.
- [8] Huff, K.D., Gidden, M.J., Carlsen, R.W., Flanagan, R.R., McGarry, M.B., Opatowsky, A.C., Schneider, E.A., Scopatz, A.M., Wilson, P.P., 2016. Fundamental concepts in the cyclus nuclear fuel cycle simulation framework. *Advances in Engineering Software* 94, 46 – 59.
- [9] Koning, A., Bauge, E., Dean, C., Dupont, E., Fischer, U., Forrest, R., Jacqmin, R., Leeb, H., Kellett, M., Mills, R., et al., 2011. Status of the jeff nuclear data library. *J. Korean Phys. Soc* 59, 1057–1062.
- [10] Krivtchik, G., et al., 2017. Scenarios with cosi6: Optimization, uncertainty and beyond, in: *Technical Workshop On Fuel Cycle Simulation*, Columbia, SC, USA.
- [11] Lee, Y., Roesslinger, B., Tsilanizara, A., 2005. Tripoli-pepin depletion code and its first numerical benchmarks for pwr high-burnup uo {sub 2} and mox spent fuel .

- [12] Leniau, B., 2013. Caractérisation des sources radioactives du cycle du combustible. Applications au cycle thorium: synthèse de l'uranium 232 en combustibles solides. Ph.D. thesis. Ph. D, Université Paris Sud, Orsay.
- [13] Leniau, B., Mouginot, B., Thiolliere, N., Doligez, X., Bidaud, A., Courtin, F., Ernoult, M., David, S., 2015. A neural network approach for burn-up calculation and its application to the dynamic fuel cycle code class. *Annals of Nuclear Energy* 81, 125–133.
- [14] Leppanen, J., et al., 2014. The serpent monte carlo code: Status, development and applications in 2013. *Annals of Nuclear Energy* 82, 142–150.
- [15] Liang, J., 2018. Effects of cut-offs on the precision of calculation of SFR spent fuel decay heat with SMURE. Ph.D. thesis. Ms, Université Paris Sud, Orsay.
- [16] Meplan, O., et al., 2005. Mure : Mcnp utility for reactor evolution - description of the methods, first applications and results, in: ENC, Paris, France.
- [17] Michel-Sendis, F., 2006. Production d'<sup>233</sup>U en Combustible MOX- T hP u en Réacteur à Eau légère. Développement du Code MURE d'Evolution du Combustible. Scénarios de transition Français vers un Cycle du Thorium. Ph.D. thesis. Ph. D, Université Paris Sud, Orsay.
- [18] Mouginot, B., Leniau, B., Thiolliere, N., Ernoult, M., David, S., Doligez, X., Bidaud, A., Meplan, O., Montesanto, R., Bellot, G., et al., 2014. Core library for advanced scenario simulation, class: principle & application, in: International Conference "The Role of Reactor Physics toward a Sustainable Future"(PHYSOR 2014), pp. 12–p.
- [19] Onillon, A., 2014. Prédiction des taux de fission des coeurs de Chooz et estimation des incertitudes associées dans le cadre de l'expérience Double Chooz. Ph.D. thesis. Nantes, Ecole des Mines.
- [20] Pujol, G., Iooss, B., Iooss, M.B., DiceDesign, S., 2015. Package 'sensitivity'.
- [21] R Development Core Team, 2008. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing. Vienna, Austria. URL: <http://www.R-project.org>.
- [22] Rahnama, F., et al., 2009. 2-dimensional pwr and bwr whole core benchmark problems, in: International Conference on Mathematics, Computational Methods and Reactor Physics, Saratoga Springs, NY, USA.
- [23] Roque, B., Erlund, M., Galloway, J., James, M., 2013. International comparison of a depletion calculation benchmark on fuel cycle issues. Nuclear Science Committee .
- [24] Skarbeli, A., Alvarez-Velarde, F., 2020. Sparse polynomial chaos expansion for advanced nuclear fuel cycle sensitivity analysis. *Annals of Nuclear Energy* 142.
- [25] Skutnik, S.E., Sly, N.C., Littell, J.L., 2016. Cyborg: An origen-based reactor analysis capability for cyclus. *Transactions* 115, 299–301.
- [26] Thiolliere, N., et al., 2018. A methodology for performing sensitivity analysis in dynamic fuel cycle simulation studies applied to a pwr fleet simulated with the class tool. *EPJ Nuclear Sci. Technol.* 4.
- [27] Ye, K., 1998. Orthogonal column Latin hypercubes and their application in computer experiments. *Journal of the American Statistical Association* 93, 1430–1439.