POLITECNICO MILANO 1863

Tesi di Laurea Magistrale in
Nuclear Engineering - Ingegneria Nucleare
Scuola di Ingegneria Industriale e dell'Informazione

Benchmark of the TDT MOC-3D solver of APOLLO3® against Serpent during depletion

Author:
Matteo Montanari
Person Code:
10722519
Advisor:
Prof. Stefano Lorenzi
EDF Supervisor:
Dr. Marco Tiberga
CEA Supervisors:
Dr. Simone Santandrea
Dr. Andrea Gammicchia

Academic Year:
2024-25

Alla mia famiglia e a Pulce, per l'amore
che mi hanno donato in questo percorso.

Contents

List of Figures **v**

List of Tables **ix**

List of Symbols **xi**

List of Acronyms **xiii**

Abstract **xv**

Sommario **xvii**

Acknowledgments **xix**

1 Introduction: master's thesis framework and structure **1**

1.1 Master's thesis framework 1

1.2 Master's thesis structure 2

2 Theoretical Background **5**

2.1 The neutron transport equation 5

2.1.1 Derivation principles 9

2.1.2 Initial and boundary conditions 9

2.1.3 Integral form 10

2.1.4 Resolution methods 10

2.2 Fuel burnup and depletion analysis 11

2.2.1 Fuel depletion made of a single isotope 12

2.2.2 Bateman equations 13

2.2.3 CRAM (Chebyshev Rational Approximation Method) 15

2.2.4 Predictor–corrector methods . . . 16
2.3 Energy self-shielding . . . 17
2.3.1 Self-shielding in neutronics codes . . . 18
2.3.2 Principles of self-shielding modeling . . . 19
2.3.3 Resonant mixture treatment . . . 20

3 MOC - Method Of Characteristics in APOLLO3® 21
3.1 MOC in neutronics . . . 21
3.1.1 Different MOC families: short and long characteristics . . . 23
3.1.2 Transmission and balance equations . . . 24
3.1.3 Iterative strategy . . . 25
3.2 MOC in APOLLO3®: higher-order axial polynomial expansions . . . 27
3.3 Depletion coupling scheme: MENDEL code . . . 28
3.4 Self-shielding . . . 29

4 Monte Carlo methods: Serpent and TRIPOLI-4® 31
4.1 Working characteristics . . . 31
4.1.1 Pseudorandomness . . . 32
4.1.2 Sampling procedures . . . 33
4.1.3 Rejection method . . . 34
4.2 Monte Carlo methods in neutronics . . . 36
4.2.1 Tracking techniques . . . 38
4.3 Serpent, a Monte Carlo reactor physics code . . . 40
4.3.1 Serpent options . . . 42
4.4 TRIPOLI-4® . . . 46

5 UOX and UOX-Gd analyzed assemblies 49
5.1 Analyzed 3D geometry: UOX assembly . . . 49
5.2 Analyzed 3D geometry: UOX-Gd assembly . . . 51

6 Convergence analysis 53
6.1 APOLLO3® . . . 53
6.1.1 TDT parameters convergence . . . 54
6.1.2 Radial mesh convergence . . . 58
6.2 Serpent . . . 60
6.2.1 Axial mesh convergence . . . 60
6.2.2 Burnup mesh convergence . . . 64
6.2.3 Radial mesh convergence . . . 68

6.3 UOX-Gd assemblies . . . 70
6.4 APOLLO3®: 36 UOX-Gd rods assembly . . . 71
6.4.1 Radial mesh convergence . . . 71
6.5 Serpent: 36 UOX-Gd rods assembly . . . 72
6.5.1 Axial mesh convergence . . . 72
6.5.2 Burnup mesh convergence . . . 75
6.5.3 Radial mesh convergence . . . 76
6.6 8 UOX-Gd rods assembly . . . 78

7 APOLLO3® - Serpent depletion benchmark 79
7.1 APOLLO3® . . . 81
7.1.1 Self-shielding optimization . . . 81
7.1.2 Burnup mesh refinement . . . 83
7.1.3 Reflector layer subdivision . . . 84
7.1.4 Energy deposition model . . . 86
7.2 Serpent . . . 86
7.3 New burnup mesh . . . 87
7.4 Decay chain and energy deposition model . . . 89
7.5 Gadolinium assembly . . . 92

8 Conclusions and perspectives 97

Bibliography 99

A TRIPOLI-4® - Serpent zero-burnup mismatch 105
A.1 TRIPOLI-4® and Serpent options . . . 109
A.1.1 Serpent Vivian Salino's IRSN library . . . 112
A.2 Interpretation of 2D cell-by-cell graphs . . . 116
A.3 Assembly characteristics and material compositions . . . 118

B 2D fissile cell analysis 121
B.1 Decay chain . . . 121
B.2 Energy deposition model . . . 125

C Notes on Serpent boundary conditions and convergence analysis 129
C.1 Notes on Serpent boundary conditions . . . 129

C.2 Notes on convergence analysis 132

Dediche **135**

List of Figures

2.1 Fission spectrum for thermal neutron induced fission in ^{235}U [10]. 6
2.2 Comparison of cross section behaviors: fission characteristics of ^{235}U (left) and resonant self-shielding in ^{238}U (right). 18

3.1 2D trajectory-based discretization. In yellow, the angular volume associated to each trajectory and r the discretization region (from [3]). 22
3.2 Graphical representation of entering and exiting fluxes for a region along a trajectory (from [3]). 24
3.3 TDT 3D tracking strategy (from [5]). 26
3.4 Depletion-neutronics coupling with (right) and without (left) the substep method (from [18]). 28

4.1 Example of the rejection method, where $[a, b] = [3, 5]$. 35
4.2 Monte Carlo working scheme in neutronics (from [28]). 37
4.3 Delta tracking drawbacks with the presence of highly absorbing materials (from Serpent Wiki). 40

5.1 X-Y layout of the 3D assembly configuration (from [6]). 49
5.2 2D geometry of the analyzed assembly. This is a layer with grids. 50
5.3 YZ cut of the modeled fuel assembly; the green rod corresponds to the instrumentation channel. 50
5.4 Fuel rod axial dimensions: UOX assembly. 51
5.5 Fuel rod axial material mesh (UOX-Gd assembly). 52
5.6 X-Y cuts of the analyzed PRATIC UOX-Gd assemblies (from [9]). 52

6.1 Comparison between integrated fission and capture rates in 30 (reference in Equation (6.1)), 24 (CamiVVER option), and 20 horizontal angle simulations. 56
6.2 Comparison between the definitive and super-refined APOLLO3® radial meshes. 59
6.3 Fuel rod axial subdivision in 26 axial layers. 61
6.4 Fuel rod axial subdivision in 45 axial layers. 62

6.5 Fuel rod axial subdivision in 77 (stepmesh) axial layers. 62
6.6 Convergence of k_{eff} varying axial mesh. 63
6.7 Analysis of the burnup mesh influence and code-to-code comparison. 65
6.8 Burnup mesh refinement benchmark and impact of numerical algorithms on depletion accuracy. 66
6.9 Sensitivity analysis of numerical parameters: impact of predictor-corrector configurations and radial mesh refinement. 68
6.10 Comparison between integrated fission and capture rates for a 5 fuel crown convergent simulation (reference) and a 4 fuel crown one at 54 000 MWd/tHM. 69
6.11 Comparison between 2D pin-by-pin fission and capture rates for a 5 fuel crown convergent simulation (reference) and a 4 fuel crown one for the fissile layers closest to the center, at 54 000 MWd/tHM. 70
6.12 2D cross-sectional geometries of the analyzed UOX-Gd assemblies modeled in Serpent. 71
6.13 Definitive X-Y layout of the UOX-Gd 3D assembly calculation. 72
6.14 UOX-Gd assembly modeling and mesh refinement: (a) global YZ section, (b) initial material mesh, and (c-e) progressive axial discretization from 20 to 40 layers. 74
6.15 Convergence of k_{eff} varying axial mesh for the 36 UOX-Gd assembly. . . . 74
6.16 Convergence analysis for the 36 UOX-Gd assembly: (a) burnup mesh and (b) impact of radial fuel rod discretization (Listing 6.4). 76
6.17 Comparison between integrated fission and capture rates with 12 crowns for the UOX-Gd rods (reference) and with 11 UOX-Gd ones at 42 000 MWd/tHM for the 36 UOX-Gd assembly. The UOX rods have 4 crowns, as in the UOX assembly case. 77
6.18 Comparison of 2D fission and capture rates at 42 000 MWd/tHM between 12-crown (ref.) and 11-crown UOX-Gd rod discretizations. 78

7.1 First full depletion benchmark result of the APOLLO3® code against Serpent. 79
7.2 Comparison between integrated fission and capture rates of Serpent (reference) and APOLLO3® at zero burnup. 80
7.3 Partial (up to 46 000 MWd/tHM) depletion result of the APOLLO3® code without self-shielding mixtures benchmarked against Serpent. 82
7.4 Analysis of the initial burnup phase: benchmark of the APOLLO3® mesh refinement and evolution of ^{135}Xe. 83

7.5 Comparison between integrated capture rates of Serpent (reference) and APOLLO3® at zero burnup with different reflector discretizations. 85
7.6 Comparison of the burnup mesh convergence Listing 6.2 between Serpent runs with `set ures 1` (reference) and `set ures 1 0.0`. 87
7.7 Full depletion benchmark of APOLLO3® against Serpent with the optimizations described in this chapter. 88
7.8 Full 3D UOX assembly depletion benchmark of APOLLO3® against Serpent (reference) with the correct energy deposition model. 90
7.9 Comparison between integrated fission and capture rates of Serpent (reference) and APOLLO3® at 60 000 MWd/tHM with the new energy deposition model. 90
7.10 Comparison of 2D fission and capture axially integrated rates at 60 000 MWd/tHM. 91
7.11 Comparison between integrated fission and capture rates of Serpent (reference) and APOLLO3® at zero burnup for the 36 UOX-Gd fuel assembly. . 92
7.12 Depletion analysis of the 36 UOX-Gd assembly: benchmark comparison up to 9625 MWd/tHM (left) and full reference evolution performed by Serpent (right). 93
7.13 APOLLO3® vs. Serpent (up to 5875 MWd/tHM) for the 8 UOX-Gd case with the new energy deposition model. 94
7.14 Comparison between integrated fission and capture rates of Serpent (reference) and APOLLO3® at 5875 MWd/tHM for the 8 UOX-Gd fuel assembly. 94
7.15 Comparison of 2D fission and capture axially integrated rates at 5875 MWd/tHM for the 8 UOX-Gd assembly. 96

A.1 Convergence of k_{eff} with respect to the number of inactive cycles. The simulation here corresponds to the result presented in Table A.1. 106
A.2 Comparison between TRIPOLI-4® (reference) and Serpent integrated fission and capture rates at zero burnup. 107
A.3 Comparison between TRIPOLI-4® (reference) and Serpent integrated fission and capture rates at zero burnup with two anergy groups. 108
A.4 Comparison between 2D pin-by-pin fission and capture rates for the fissile layer closest to the inner reflector between Serpent and TRIPOLI-4® (reference) at zero burnup. 115
A.5 Comparison between 2D pin-by-pin capture rates for the third fissile layer (it has not the grid) between Serpent and TRIPOLI-4® (reference) at zero burnup. 115

B.1 Comparison of k_{eff} between Serpent and APOLLO3® using two different decay data libraries. 122
B.2 Comparison of isotopic concentrations between Serpent and APOLLO3® using `ch_CEAV2005_V3c` (left) and `DecayData_CEAV2005_V3` (right). 124
B.3 Pin-by-pin fission and capture rates for the first fissile layer at zero burnup correspondent to the results in Figure 7.1. 125
B.4 Comparison between integrated fission and capture rates for Serpent (reference) and APOLLO3® with 383 groups at zero burnup. 126
B.5 Full depletion 2D fissile cell simulations of APOLLO3® against Serpent (with `set U235H 193.7201`) comparing different decay chains. 128

C.1 YZ cut of a fuel assembly simulated in Serpent. The black part on top and on bottom is the "void" region. It is visible the reflective boundary condition at the z axis midplane. 131
C.2 Axial mesh refinement using the CamiVVER burnup mesh (Listing 6.1). . 134

List of Tables

4.1 Available temperatures for nuclide cross sections and $S(\alpha,\beta)_{\mathrm{H_2O}}$ thermal scattering law data in JEFF-3.1.1. 43
4.2 List of nuclides present in VTT JEFF-3.1.1 not originally in JEFF-3.1.1. . 45

6.1 TDT parameter convergence analysis using the CamiVVER radial mesh. CamiVVER default options are bold. Times are in hh:mm:ss or d-hh:mm:ss. 55
6.2 TDT parameter convergence analysis simultaneously changing different parameters. 57
6.3 Analysis with $\Delta_{xy} = 0.02$. 58
6.4 Analysis with $\Delta_{xy} = 0.03$. 58
6.5 Parameters of Serpent simulations for axial mesh convergence where n is the number of neutrons per batch and N the number of batches. 61
6.6 Explanation of Figure 6.6 legend symbols. 62
6.7 Radial mesh convergence analysis for the 36 UOX-Gd assembly. The configuration numbers refer to the previous list. 72
6.8 Parameters of Serpent simulations for the axial mesh convergence analysis (UOX-Gd assembly). 73

7.1 Refinement of the axial reflector in APOLLO3® with respect to the CamiVVER material mesh choice (zero burnup). 84

A.1 TRIPOLI-4® - Serpent zero-burnup k_{eff}. All the σ results are in pcm. . . 106
A.2 TRIPOLI-4® zero-burnup k_{eff}. All the σ results are in pcm. 109
A.3 Serpent zero-burnup k_{eff}. All the σ results are in pcm. 111
A.4 Available temperatures for nuclide cross sections and $S(\alpha,\beta)_{H_2O}$ thermal scattering law data in Vivian Salino's IRSN library. 112
A.5 Serpent zero-burnup k_{eff} with Vivian Salino's IRSN library. All the σ results are in pcm. Also the k_{eff} with its uncertainty presented in Equation (A.10) was included. 113
A.6 UOX analyzed configuration characteristics (1/8 radially). 118
A.7 36 and 8 UOX-GD configuration characteristics (1/8 radially). 118

A.8 UOX assembly material composition [10^{24}/cm^3]. 119
A.9 36 UOX-Gd assembly material composition [10^{24}/cm^3]. 120

C.1 Parameters of Serpent simulations for the axial mesh convergence analysis with the CamiVVER burnup mesh (Listing 6.1). Here, n is the number of neutrons per batch and N the number of batches. 132

List of Symbols

Variable	Description	Unit of measure
$\boldsymbol{k_{eff}}$	effective multiplication factor	—
$\boldsymbol{k_{eff}^{ref}}$	reference effective multiplication factor	—
IMP_KEFF	implicit k_{eff}	—
ANA_KEFF	analog k_{eff}	—
wt%	weight percent	—
$\boldsymbol{\sigma(X)}$	uncertainty on $\boldsymbol{X}$	$\boldsymbol{X}$ units
pcm	percent mille	10^{-5}
GWd/tHM	gigawatt-day per tonne of heavy metal	$\mathrm{GW\,day/t_{HM}}$
MWd/tHM	megawatt-day per tonne of heavy metal	$\mathrm{MW\,day/t_{HM}}$
MeV	megaelectronvolt	$10^{6}\,\mathrm{eV}$
Å	angstrom	$10^{-10}\,\mathrm{m}$
b	barn	$10^{-28}\,\mathrm{m^2}$
sr	steradian	$0–4\pi$
$\boldsymbol{\phi(\vec{r}, E, \vec{\Omega}, t)}$	angular neutron flux	$\mathrm{m^{-3}\,eV^{-1}\,sr^{-1}\,s^{-1}}$
$\boldsymbol{\Sigma}$	macroscopic cross section	$\mathrm{m^{-1}}$
$\boldsymbol{\vec{\Omega}}$	neutron angular direction	sr
$\boldsymbol{\rho}$	mass density	$\mathrm{kg\,m^{-3}}$

List of Acronyms

Acronym	Meaning
ACE	A Compact ENDF (format for cross sections)
ASNR	Autorité de Sûreté Nucléaire et de Radioprotection (Nuclear Safety and Radiation Protection Authority)
BoS	Beginning Of Step
BROND	Russian Evaluated Nuclear Data Library
CamiVVER	Codes And Methods Improvements for VVER comprehensive safety assessment
CCM	Chord Classification Method
CDF	Cumulative Density Function
CEA	Commissariat à l'énergie atomique et aux énergies alternatives (French Alternative Energies and Atomic Energy Commission)
CELI	Constant Extrapolation, Linear Interpolation (predictor-corrector method)
CENDL	Chinese Evaluated Nuclear Data Library
CFD	Computational Fluid Dynamics
CLT	Central Limit Theorem
CRAM	Chebyshev Rational Approximation Method
EASI-SMR	Ensuring Assessment of Safety Innovations for SMR
ENDF	Evaluated Nuclear Data File
EoS	End Of Step
HPC	High-Performance Computing
HSS	Hit Surface Sequence
IDT	Integro-Differential Transport (solver)
IRSN	Institut de Radioprotection et de Sûreté Nucléaire (Institute of Radioprotection and Nuclear Safety)
JENDL	Japanese Evaluated Nuclear Data Library
LELI	Linear Extrapolation, Linear Interpolation (predictor-corrector method)

Acronym	Meaning
LEQI	Linear Extrapolation, Quadratic Interpolation (predictor-corrector method)
LS	Linear Surface (approximation)
MOC	Method Of Characteristics
MOSC	Method Of Short Characteristics
MOX	Mixed OXide
NR	Narrow Resonance (slowing down model)
ODE	Ordinary Differential Equation
OpenMP	Open Multi-Processing
PDE	Partial Differential Equation
PDF	Probability Density Function
PHYSOR	International Conference on the Physics of Reactors
PRATIC	Petit REP Académique pour Tester, Innover & Concevoir (Small Academic PWR for Test, Innovation & Design)
PWR	Pressurized Water Reactor
R&D	Research & Development
SC	Step Constant (approximation)
SEM	Step Equivalent Mesh
SERMA	Service d'Études des Réacteurs et de Mathématiques Appliquées (Reactor Studies and Applied Mathematics Division)
SHEM281	Santamarina Hfaiedh Energy Mesh (281 groups)
ST	STatistical (distribution of narrow resonances within a group, slowing down model)
TDT	Two-Three Dimensional Transport (solver of APOLLO3®)
TMS	Target Motion Sampling
TR	Toute Résonance (any resonances, slowing down model)
TTA	Transmutation Trajectory Analysis
UOX	Uranium OXide
UOX-Gd	Uranium Oxide with Gadolinium
VTT	Valtion Teknillinen Tutkimuskeskus (Technical Research Center of Finland)
VVER	vodo-vodyanoi Energetichesky Reaktor (Water-Water Energetic Reactor)
WR	Wide Resonance (slowing down model)

Abstract

The Two-Three Dimensional Transport (TDT) solver of APOLLO3®, based on the Method of Characteristics (MOC), was recently extended for full 3D neutron transport depletion calculations using axial polynomial expansions for flux and macroscopic cross sections. While previous tests against TRIPOLI-4®, the French reference Monte Carlo code developed at CEA, successfully benchmarked this solver for a standard PWR UOX assembly up to 60 GWd/tHM, this thesis extends that work by comparing the solver against the Serpent Monte Carlo code. Configurations analyzed include the same UOX assembly and two other UOX-Gd assemblies, containing 36 and 8 gadolinium rods. The study performs a detailed convergence analysis to determine TDT parameters for APOLLO3® and identifying axial, radial, and burnup meshes for both codes, balancing precision and efficiency. A depletion benchmark revealed excellent agreement for the UOX assembly: k_{eff} remained within 140 pcm of the Serpent reference, and discrepancies in radially integrated fission and capture rates were below 0.5 %, dropping to less than 0.15 % for axially integrated rates. Conversely, the MOC-3D solver showed limitations for assemblies containing gadolinium. Due to high computational demand, the 36 UOX-Gd case could not be simulated up to 60 GWd/tHM within a reasonable amount of time; therefore, the analysis was limited to 10 GWd/tHM. Results revealed a mismatch exceeding 1000 pcm in absolute value for k_{eff}, while the 8-rod UOX-Gd assembly showed a decrease in the discrepancy (up to -753 pcm). These results indicate that while agreement improves as gadolinium content decreases, the current implementation remains unsatisfactory for these configurations. These findings reinforce the robustness of the TDT MOC-3D approach for UOX depletion, while confirming previous studies on the necessity of the Linear Surface (LS) method in APOLLO3® to correctly model the presence of gadolinium. This issue arises because gadolinium isotopes have high neutron capture cross sections, causing steep flux gradients near the rods, which the current version of the MOC-3D solver fails to correctly represent.

Keywords: APOLLO3®, Serpent, MOC-3D, benchmark, depletion, UOX, gadolinium

Sommario

Il codice Two-Three Dimensional Transport (TDT) di APOLLO3®, basato sul Metodo delle Caratteristiche (MOC), è stato recentemente esteso per consentire calcoli di trasporto neutronico 3D durante il bruciamento utilizzando espansioni polinomiali assiali per il flusso e sezioni d'urto macroscopiche. Mentre test precedenti rispetto a TRIPOLI-4®, il codice Monte Carlo di riferimento francese sviluppato al CEA, hanno validato con successo questo codice per un assemblaggio PWR UOX standard fino a 60 GWd/tHM, questa tesi estende tale lavoro confrontandolo con il codice Monte Carlo Serpent. Le configurazioni analizzate includono il medesimo assemblaggio UOX ed ulteriori due assemblaggi UOX-Gd, contenenti 36 e 8 barre di gadolinio. Lo studio conduce un'analisi di convergenza per determinare i parametri TDT per APOLLO3® e identificare le mesh assiali, radiali e di bruciamento per entrambi i codici, bilanciando precisione ed efficienza. Un confronto tra i due codici ha mostrato un ottimo accordo per l'assemblaggio UOX: il k_{eff} rimane entro 140 pcm rispetto a Serpent, e le discrepanze nei tassi di fissione e cattura integrati radialmente risultano inferiori allo 0.5 %, scendendo a meno dello 0.15 % per i tassi integrati assialmente. Al contrario, il codice MOC-3D ha mostrato dei limiti per gli assemblaggi gadoliniati. Il caso UOX-Gd a 36 barre non ha potuto essere simulato fino a 60 GWd/tHM in tempi ragionevoli; pertanto, l'analisi è stata limitata a 10 GWd/tHM. I risultati hanno rivelato un disaccordo significativo tra i due codici, con discrepanze superiori a 1000 pcm per il k_{eff} (in valore assoluto). L'analisi con 8 barre UOX-Gd ha riportato un accordo migliore, anche se la differenza relativa tra i codici abbia raggiunto i -753 pcm. Questi risultati indicano che, sebbene l'accordo migliori al diminuire del contenuto di gadolinio, l'attuale implementazione MOC-3D rimane insoddisfacente per tali configurazioni. Queste conclusioni rafforzano la robustezza dell'approccio TDT MOC-3D per il bruciamento di assemblaggi UOX e confermano studi precedenti sulla necessità del metodo Linear Surface (LS) in presenza di gadolinio. Tale criticità sorge poiché gli isotopi del gadolinio possiedono sezioni d'urto di cattura neutronica elevate che causano ripidi gradienti di flusso che la versione attuale del codice non riesce a rappresentare.

Parole chiave: APOLLO3®, Serpent, MOC-3D, bruciamento, UOX, gadolinio

Acknowledgments

The author would like to thank the APOLLO3® development team for their efforts in developing the code. APOLLO3® is a registered trademark of CEA and is developed with the support of EDF and Framatome. This work received financial support from the European Union's Horizon Europe Research and Innovation program through the EASI-SMR project (Grant Agreement No. 101 164 810).

1 | Introduction: master's thesis framework and structure

1.1. Master's thesis framework

The long-term objective of achieving one-step core calculations with high-performance computing requires continuous improvements in deterministic neutron transport solvers. Significant progress towards this goal has been marked by the TDT (Two-Three Dimensional Transport) solver of APOLLO3®, the new generation multi-purpose deterministic code for reactor analysis developed at CEA with financial support from EDF and Framatome [1]. Based on the Method of Characteristics (MOC), TDT was recently extended to three dimensions [2], enabling 3D calculations in extruded geometries.

To reduce computational cost and memory consumption, an axial polynomial expansion for the neutron flux was introduced [3, 4]. However, this initial implementation was limited to fresh fuel configurations: as burnup progresses, spatial gradients in neutron flux induce corresponding variations in nuclide concentrations and, consequently, in macroscopic cross sections. Most current transport codes treat cross section profiles as piecewise constant, an assumption that becomes increasingly inaccurate at higher burnup levels. This simplification undermines the effectiveness of higher-order flux expansions: if the flux varies significantly in space, the cross sections must also be treated with similar spatial resolution; therefore, applying higher-order approximations solely to the flux is insufficient: it is required that the same level of precision be extended to the cross section modeling as well. For this reason, the code incorporates the methodology proposed in [5], which employs a polynomial cross section expansion. This approach is described in detail in Chapter 3

This master's thesis continues the benchmark effort of the TDT MOC-3D solver implemented in APOLLO3®, building upon the work initiated in [5] and further developed within the CamiVVER (Codes and Methods Improvements for VVER Comprehensive Safety Assessment) project which was organized into seven distinct Work Packages, each addressing a specific technical domain. This master's thesis follows directly from Work Package 4, and specifically from Chapter 2 and Appendix B of Deliverable D4.6 [6]. All

open-access results and deliverables from the project are available on the dedicated website. The focus of CamiVVER was the development and benchmark of the TDT MOC-3D solver capable of producing high-fidelity reference solutions at the assembly level. This solver is integrated within a computational environment that couples it with the depletion and self-shielding capabilities of the APOLLO3® code. Such integration enables accurate modeling of the axial reflector without relying on conventional simplifications. In particular, it allows for the resolution of the pronounced thermal flux peak that typically arises near the axial boundaries which is an effect often neglected in lattice calculations or approximated using overly simplified one-dimensional models. The solver was benchmarked against TRIPOLI-4® (a Monte Carlo code developed by CEA [7]) for a standard PWR UOX assembly up to 60 GWd/tHM. In the current work, the Serpent Monte Carlo code [8] has been selected as the reference for depletion analysis. Compared to TRIPOLI-4®, Serpent offers superior depletion capabilities and higher precision, making it a more suitable tool. Hence, the aim of this thesis is both to accurately benchmark the TDT MOC-3D solver during depletion and obtain more precise results with respect to previous works [5, 6].

While the CamiVVER project focused exclusively on a UOX fuel assembly, this thesis expands the scope by also analyzing UOX-Gd assemblies whose characteristics are based on the PRATIC core ("Petit REP Académique pour Tester, Innover & Concevoir": Small Academic PWR for Test, Innovation & Design) [9], which is part of the EASI-SMR initiative (Ensuring Assessment of Safety Innovations for Small Modular Reactors). This broader analysis aims to confirm the robustness of APOLLO3® regarding gadolinium concentration. Two configurations were chosen: one with 36 UOX-Gd rods and another with 8. Hence, this work allows for a more comprehensive study of the TDT MOC-3D solver of APOLLO3® across different fuel types and reactor configurations.

1.2. Master's thesis structure

The remainder of this master's thesis is organized as follows. In Chapter 2, a theoretical introduction to fission reactor physics is presented, focusing on the neutron transport equation and major theoretical aspects encountered in this study, such as depletion analysis and energy self-shielding, particularly regarding their implementation in neutronics codes.

Chapter 3 and Chapter 4 provide the main theoretical foundations of how deterministic and probabilistic neutronics codes work, specifically regarding the MOC-3D implementation in the TDT solver of APOLLO3®, and the most important features in Serpent and TRIPOLI-4®, respectively.

Chapter 5 details the 3D geometries investigated for the UOX and UOX-Gd assemblies, while Chapter 6 outlines the preliminary tests conducted to identify the optimal set of computational parameters for APOLLO3® and Serpent. Since depletion simulations are considerably time-consuming, this accuracy-efficiency trade-off is crucial, as fine-tuning parameters can yield substantial time savings with minimal impact on results. This convergence analysis specifically addresses TDT tracking parameters alongside axial, radial, and depletion mesh refinement.

Finally, Chapter 7 presents the core results of the depletion benchmark between APOLLO3® and Serpent, followed by concluding remarks and future perspectives in Chapter 8.

Further analyses are present in the appendices. Appendix A focuses on the discrepancy at zero burnup between TRIPOLI-4® and Serpent, trying to diminish them and give an interpretation. Appendix B analyzes a 2D fissile cell of the UOX assembly in order to study the relative differences in isotopic concentrations during depletion between APOLLO3® and Serpent. The discrepancies reveal a difference in the decay chains between the two codes. Appendix C explains two interesting observations that were made during this thesis and are worth to be mentioned.

Supplementary investigations are collected in the appendices. Appendix A addresses the zero-burnup discrepancies between TRIPOLI-4® and Serpent, offering an interpretation of the results to minimize these offsets. Appendix B explores a 2D fissile cell of the UOX assembly to evaluate relative differences in isotopic concentrations, revealing variations in the decay chain modeling between the two codes. Lastly, Appendix C discusses noteworthy observations encountered during the course of this work that merit to be mentioned.

2 | Theoretical Background

This chapter aims to introduce the reader to the fundamental principles of nuclear reactor physics necessary to understand the context and objectives of this work. In particular, it provides the bases for studying neutronics and dealing with depletion problems and related techniques, which are the central topics of this thesis. A section is also dedicated to the self-shielding phenomenon and how it is treated by neutronics codes.

In a nuclear reactor, the core is the region where the nuclear fuel is located and where fission reactions takes place. This reaction occurs when neutrons collide with heavy fissile nuclei, such as ^{235}U and ^{239}Pu, causing them to split into two fission products and release additional neutrons. This is a highly exothermic reaction that releases about 200 MeV per fission event (where $1\,\text{eV} \approx 1.6 \times 10^{-19}\,\text{J}$) which is over a million times greater than the energy of a chemical reaction (e.g., coal combustion), a difference often referred to as the "mega factor" due to the transition from the eV to the MeV scale. The newly emitted neutrons can then interact with other fissile nuclei, sustaining a chain reaction that produces a stable source of heat which can be exploited, typically, to produce electricity.

Neutrons are therefore the key agents in the system, and accurately tracking their behavior is essential to predict their distribution and motion within the reactor core. To achieve this, a mathematical model is required to describe the phenomenon of the so-called neutron transport.

2.1. The neutron transport equation

The neutron transport equation, or Boltzmann equation, aims to describe the neutron distribution inside a system which depends on several variables such as the space distribution ($\vec{x}$), the angular distribution or flight direction ($\vec{\Omega}$), the energy distribution (E), and the time distribution (t).

To start with, a series of approximations are needed to establish the order of detail the physical phenomenon is decided to be studied and they shape the final equation form:

- The relativistic phenomena are not studied since the rest energy of a neutron is around $E_o^n \approx 939\,\text{MeV}$, which is way higher than the maximum energy taken into

account by neutronics codes (around $E^n_{max} \approx 20\,\text{MeV}$). This means that the neutrons' behavior is described by cross sections not relativistically adjusted;

- The magnetic nature of neutrons, so their polarization property, is neglected. This means that the cross sections in the formulas used in this work are not magnetically adjusted;

- Neutrons are considered stable even if they have a half-life (when isolated) of about $T_{\frac{1}{2}} = 10\,\text{min}$, as their average lifetime inside a thermal reactor is $\approx 10^{-5}\,\text{s}$ to $10^{-3}\,\text{s}$ and for a fast reactor is even lower;

- The neutrons wave nature is not considered. They are considered as deterministic particles without the possibility of having diffraction phenomena. This is possible because their De Broglie wavelength is lower than the smallest dimension they can see, which is the characteristic atomic dimension ($L_C \approx 1\,\text{Å}$). Indeed, their De Broglie wavelength is (knowing that $p = mv$, $E = \frac{1}{2}mv^2$, thus $\sqrt{2Em}$):

$$\lambda_{DB} = \frac{\hbar}{p} = \frac{4.55 \times 10^{-10}}{\sqrt{E(eV)}}(\text{eV})^{1/2}\,\text{cm}.$$

This means that with $E = 0.01\,\text{eV}$, $\lambda_{DB} = 4.55 \times 10^{-11}\,\text{m} = 0.455\,\text{Å}$. So, even with a lower energy than the one of interest for thermal neutrons ($E_{th} \approx 0.025\,\text{eV}$) there is not a diffraction phenomenon. Actually, according to Watt's spectrum (see Figure 2.1) there is a non-zero probability of having neutrons emerging from a fission event with a really low energy, resulting in a way higher λ_{DB}. Anyway, this probability is really low, thus the neutron wave nature can be neglected;

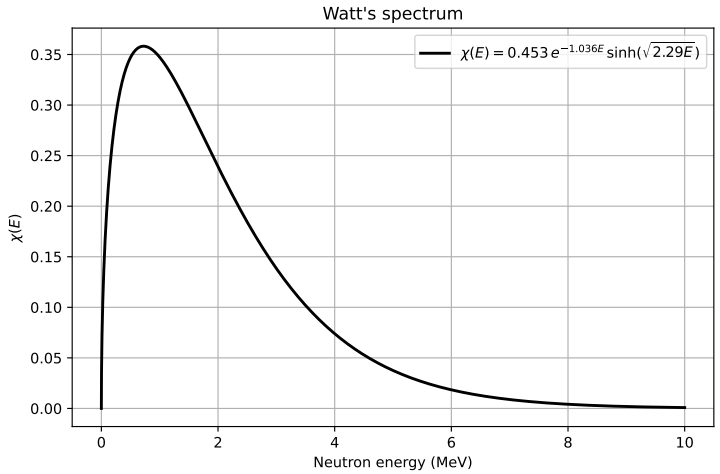


Figure 2.1: Fission spectrum for thermal neutron induced fission in ^{235}U [10].

- The Heisenberg uncertainty principle is not a concern for the dimensions of interest, since the smallest dimension in a reactor is the cladding thickness which is about $t_{clad} \approx 500\,\mu\text{m}$. Assuming to accept a maximum uncertainty of $\sigma(t_{clad}) \approx 1\,\%$, it results in $\Delta x \approx 10^{-4}\,\text{cm}$. Moreover, as $E = \frac{p^2}{2m}$, the differential of Δp is $\frac{m\Delta E}{\sqrt{2mE}}$. Since the Heisenberg uncertainty principle states $\Delta x\,\Delta p \geq \frac{\hbar}{2}$, then:

$$\frac{\Delta E}{E} \geq \frac{\hbar}{2\Delta x}\frac{\sqrt{2}}{\sqrt{E}} = 4.55 \times 10^{-6}\frac{(eV)^{1/2}}{\sqrt{E}},$$

  which means that with $E = 0.01\,\text{eV}$ (lower than thermal neutron energy) the relative uncertainty over the energy is around $\frac{\Delta E}{E} \geq 4.55 \times 10^{-5}$, that is negligible;

- The Boltzmann equation is assumed to be linear. In principle, scattering between neutrons occur, hence leading to a coupled term between neutron densities. Since the aim is to study neutron interaction with nuclei and due to the fact that neutrons are considered rarefied[1], the coupling term is neglected. By considering r the probability density ratio between neutron-nucleus interaction and neutron-neutron interaction, it can be expressed as $r = \frac{\sigma_{nN}}{\sigma_{nn}}\frac{N}{n}$ where σ_{nN} and σ_{nn} are respectively the microscopic cross sections for neutron-nucleus interaction and neutron-neutron interaction while N and n are the nuclei and neutron volumetric densities. Assuming[2] $\sigma_{nN} \approx \sigma_{nn}$ and considering a typical thermal neutron flux in a nuclear reactor core around $\phi \approx 10^{13}\,\text{cm}^{-2}\,\text{s}^{-1}$ then, considering thermal neutrons with $E = 0.025\,\text{eV}$, so $v \approx 2.2 \times 10^5\,\text{cm/s}$, and since $\phi = nv$, then $n \approx 10^8\,\text{cm}^{-3}$. Being again conservative, let's assume $\phi \approx 10^{15}\,\text{cm}^{-2}\,\text{s}^{-1}$ so $n \approx 10^{10}\,\text{cm}^{-3}$. Since a solid body (the fuel in our case) has usually $N \approx 10^{22}\,\text{cm}^{-3}$, even if underestimated and considered as the density of a gaseous medium of the order of $N \approx 10^{20}\,\text{cm}^{-3}$, the result for r is about

$$r = \frac{\sigma_{nN}}{\sigma_{nn}}\frac{N}{n} \gg \frac{N}{n} \gg \frac{10^{20}}{10^{10}} = 10^{10}.$$

  Thus, it is more than justified to neglect neutron-neutron coupling in Boltzmann equation[3].

- The neutrons' quantum mechanics properties are neglected so the fact that are fermions ($s = \frac{1}{2}\hbar$) obey to Pauli exclusion principle is not considered as their density is way lower than the one of solid materials, as just explained;

[1] Indeed Boltzmann equation is originally derived for rarefied gases.
[2] It is a strong conservative assumption since it is well known that microscopic cross sections, evaluated in cm^2 or barns refer to the *geometrical* area occupied by solid bodies; therefore , $\sigma_{nN} \gg \sigma_{nn}$.
[3] Assumption not justified in the weapon engineering field since the neutron fluxes are way higher.

- Availability of fissile nuclide is considered constant during the transport process. In fact, the fissile nuclide density depends on the burnup (the more energy is produced, the more fuel is consumed) leading to a further source of non-linearity. Since burnup time scales are way longer than transport ones, it is justified to apply this approximation in order to consider macroscopic cross sections ($\Sigma = \sigma N$) constant in time. Usually, in depletion calculations, this approximation is performed by subdividing the time domain in intervals in which Σ is considered constant, as explained in more details in Section 2.2;

- The cross sections do not depend on temperature. This is a further removal of non-linearity source since in principle temperature variations cause variations of system properties (feedbacks). Indeed, it has been historically chosen to decouple thermal feedbacks and neutronics: first the study of neutronics is carried out, then, if a multi-physics approach is needed, temperature dependencies are introduced. It is essential to exploit this decoupling technique to diminish the computational burden otherwise most likely unbearable;

- The Boltzmann equation describes the *average* neutron behavior in the studied system, indeed the neutron population is a statistical one along with its uncertainty. Since the neutron transport is a Markovian process[4] described by Poisson probabilities, the uncertainty on a variable X is $\sigma = \sqrt{X}$. This means that in a thermal system, and making conservative assumptions as before $n \approx 10^{10}\,\mathrm{cm}^{-3}$, at most $\sigma \approx 10^{5}\,\mathrm{cm}^{-3}$, so the variation is very small compared to the mean value.

Now that the framework is set, a series of definitions must be introduced:

- $N(\vec{r}, E, \vec{\Omega}, t)$ is the neutron angular density which is measured in $\mathrm{cm}^{-3}\,\mathrm{MeV}^{-1}\,\mathrm{sterad}^{-1}$ and $N(\vec{r}, E, \vec{\Omega}, t)\, d^3\vec{r}\, dE\, d\vec{\Omega}$ expresses the number of neutrons in the infinitesimal volume of space centered in $\vec{r}$ having energy E within dE and flight direction $\vec{\Omega}$ within $d\vec{\Omega}$. The *sterad* is the unit of measure of the solid angle that goes from 0 to 4π;

- $\phi(\vec{r}, E, \vec{\Omega}, t) = v(E)\, N(\vec{r}, E, \vec{\Omega}, t)$ is the neutron angular flux measured in $\mathrm{cm}^{-2}\,\mathrm{MeV}^{-1}\,\mathrm{sterad}^{-1}\,\mathrm{s}^{-1}$ where $v(E)$ is the neutron velocity and $\phi(\vec{r}, E, \vec{\Omega}, t) d^3\vec{r}\, dE\, d\vec{\Omega}$ expresses the number of neutrons that cross the infinitesimal area perpendicular to the fly direction $\vec{\Omega}$ having fly direction $\vec{\Omega}$ within $d\vec{\Omega}$ and energy E within dE;

[4] A Markovian process refers to a system where transitions between states occur independently of what happened during the particle history, making it memoryless.

- $\phi(\vec{r}, E, t) = v(E)N(\vec{r}, E, t) = \int_{4\pi} \phi(\vec{r}, E, \vec{\Omega}, t)d\vec{\Omega}$ is the neutron flux measured in $\mathrm{cm}^{-2}\,\mathrm{MeV}^{-1}\,\mathrm{s}^{-1}$. it is the integral over all the solid angle of the neutron angular flux;
- $\Sigma_i(\vec{r}, E, \vec{\Omega}, t)$ is the macroscopic cross section of reaction i measured in cm^{-1}.

2.1.1. Derivation principles

Since the aim of the equation is to follow motion, let's consider an arbitrarily chosen infinitesimal volume $d^3\vec{r}$ located in $\vec{r}$ at time t and assume to wait a time Δt and then check the new neutron angular density at $N(\vec{r} + v\vec{\Omega}\Delta t, E, \vec{\Omega}, t + \Delta t)$. Let's furthermore assume that the Δt is small enough to guarantee that the neutrons have at most 1 collision (so they experience zero or one collision) then, the physical phenomena that have to be taken into account for each neutron are the following:

- If the neutron does not interact, so its energy and fly direction do not change;
- If the neutron interacts, so at least one between energy and fly direction change;
- There is the production of one or more neutrons in that infinitesimal volume thanks to fission event or thanks to an external source.

It can be proved [10] that, assuming $\Delta t \to 0$ and since the infinitesimal volume is arbitrarily chosen, it is obtained the so-called Boltzmann equation or neutron transport equation:

$$\begin{aligned}\frac{1}{v(E)}\frac{\partial\phi(\vec{r}, E, \vec{\Omega}, t)}{\partial t} + \vec{\Omega}\cdot\nabla\phi(\vec{r}, E, \vec{\Omega}, t) + \Sigma_t(\vec{r}, E)\,\phi(\vec{r}, E, \vec{\Omega}, t) = \\ \iint_{\vec{\Omega}', E'} \Sigma_s(\vec{r}, \vec{\Omega}', E' \to \vec{\Omega}, E)\,\phi(\vec{r}, E', \vec{\Omega}', t)\,d\vec{\Omega}'dE' + S(\vec{r},\vec{\Omega}, E, t).\end{aligned} \tag{2.1}$$

2.1.2. Initial and boundary conditions

Being a (integro-)differential equation, boundary conditions are needed to guarantee the uniqueness of the solution. In particular, two conditions are usually chosen:

1. *Initial condition:* the flux at time t_0 is a known value, where t_0 is the time instant in which the system of interest starts to be monitored. Mathematically: $\phi(\vec{r}, E, \vec{\Omega}, t_0) = \phi_0(\vec{r}, E, \vec{\Omega})$;

2. *Vacuum Boundary condition:* there is no neutron entering the system, so neutrons can only exit the system[5]. This translates mathematically in:

$$\phi(\vec{r}_s, E, \vec{\Omega}, t) = 0 \quad \forall \vec{r}_s \in \partial V \quad \text{such that} \quad \vec{r}_s \cdot \vec{n}_{\vec{\Omega}} \leq 0, \tag{2.2}$$

where V is the volume of the system of interest and ∂V the surface that surrounds it, while $\vec{n}_{\vec{\Omega}}$ is the unit vector normal to the surface correspondent to the fly direction $\vec{\Omega}$. This just presented is only one possible boundary condition, different ones can be adopted according to the system analyzed.

2.1.3. Integral form

The integral form of the Boltzmann equation is the one used by deterministic (numerical or mathematical) and probabilistic (Monte Carlo) resolution methods which are respectively addressed in Chapter 3 and Chapter 4. This form allows a better exploitation of resolution methods rather than the integro-differential one (Equation (2.1)).

Its derivation is quite easy starting from the integro-differential form of Boltzmann equation (Equation (2.1)) and noting that it can be written in the form

$$\frac{1}{v}\frac{\partial \phi}{\partial t} = \hat{L}_{\vec{\Omega}}\phi + \hat{L}_{\Sigma}\phi + \hat{L}_t\phi + S = \hat{L}_{\vec{\Omega}}\phi + \hat{L}_{\Sigma}\phi + q, \tag{2.3}$$

where $\hat{L}_{\vec{\Omega}} = (-\vec{\Omega} \cdot \nabla)$ is the differential operator, $\hat{L}_{\Sigma} = (-\Sigma)$ the algebraic operator, $\hat{L}_t = \iint_{\vec{\Omega}', E'} \Sigma_s(\vec{r}, \vec{\Omega}', E' \to \vec{\Omega}, E)\phi(\vec{r}, E', \vec{\Omega}', t)d\vec{\Omega}'dE'$ the transfer operator and $q = \hat{L}_t\phi + S$ the generalized source term. It can be demonstrated [11] that, following a series of mathematical derivations, the final result is

$$\phi(\vec{r}, E, \vec{\Omega}, t) = \int_0^{+\infty} q(\vec{r} - s'\vec{\Omega}, \vec{\Omega}, E, t - \frac{s'}{v})\, e^{-\tau(\vec{r} - s'\vec{\Omega} \to \vec{r}, E)} ds', \tag{2.4}$$

where $\tau(\vec{r} - s'\vec{\Omega} \to \vec{r}, E) = \int_0^{s'} \Sigma(\vec{r} - \bar{s}\vec{\Omega}, E)d\bar{s}$ is called the optical path or kernel of free flight.

2.1.4. Resolution methods

Once the equation is derived and its form (differential or integro-differential) is chosen, a resolution method must be selected to analyze it and study neutron behavior in the system of interest.

[5] This is an arbitrary decision to simplify the mathematical model. Indeed, nothing avoids having a concave (re-entrant) *physical* system but it is wiser to study a *mathematical* convex system that contains the physical one and is surrounded by void. This choice simplifies a lot the calculations.

In particular, one key parameter in nuclear reactor physics is the k-effective (effective neutron multiplication factor or k_{eff}). It expresses the ratio between the rate of neutron production and the rate of neutron loss or, analogously, the ratio between the number of neutrons of the current generation and the number of neutrons in the previous generation. Where a generation refers to the complete life cycle of a group of neutrons, from the moment they are released during a fission event (or from an external source) to the moment they are lost (leakage or capture). It is straightforward to understand that if $k_{eff} = 1$ the number of neutrons inside the system is constant, so there is a stable condition, called criticality. The discrepancy between the criticality condition and the k_{eff} calculation must be accurately calculated to avoid supercriticality accidents or unwanted reactor shutdowns.).

The range of possible strategies to retrieve physically meaningful parameters from Equation (2.1) and Equation (2.4), such as k_{eff}, is quite broad and exhaustively discussed in [10, 11]. The numerical or mathematical ones often rely on simplifications of the various independent variables on which the neutron flux depends, such as:

- *Angle*: Legendre polynomials or the S_N approximation may be employed;
- *Space*: spatial mesh discretization can be applied;
- *Energy*: it may be treated using a multigroup approximation (or even a single-group one), rather than treating it as a continuous variable.

Beyond these mathematical treatments, several solution approaches exist to solve the neutron transport problem. They can be broadly categorized into deterministic methods, which provide an exact solution of an approximated version of the transport equation, and probabilistic (Monte Carlo) methods, which rely on statistical sampling to estimate the neutron behavior without applying major simplification.

This thesis focuses on the Method Of Characteristics (MOC) as deterministic method and Monte Carlo methods for probabilistic ones. Their basic principles are respectively explained in Chapter 3 and Chapter 4.

2.2. Fuel burnup and depletion analysis

As already explained, the macroscopic cross section Σ is considered to be constant in time when the Boltzmann equation is solved; however, in order to perform an analysis in time that considers the material depletion of a multiplying medium, the macroscopic cross sections vary in time since the medium is consumed and its density N changes.

This occurs because, during the depletion simulation, material compositions evolve due to fission, neutron capture, or both. These changes affect the macroscopic cross sections, which in turn alter the reaction rates. In each independent region with a certain material composition, these values are assumed to be constant to simplify calculations; therefore, if the material refinement is too coarse, the simulation cannot accurately model these local variations, leading to unrealistic or misleading results.

To properly follow the Σ variation along time, the period of interest must be divided into several intervals in which N and Σ may be considered constant. These time intervals must communicate with each other, where interval t_i provides information to t_{i+1}. This is the goal of a depletion solver which can implement a variety of techniques to perform this task based on a simple idea:

- Solve the Boltzmann equation within a time interval t_i to get the neutron flux and considering it constant[6] while solving the depletion equations to obtain values for the cross sections at the end of the time interval;
- These results are then used as input data for the time interval t_{i+1} to solve again the neutron transport equation and get the neutron flux for the t_{i+1} time interval and so on.

2.2.1. Fuel depletion made of a single isotope

To understand what calculations are performed by a depletion solver, let's imagine studying just the consumption of an isotope where $\sigma(\vec{r})$ expresses the microscopic cross section of consumption. The evolution of its density can be expressed as follows:

$$\frac{\partial N(\vec{r},t)}{\partial t} = -N(\vec{r},t)\,\sigma(\vec{r})\,\phi(\vec{r},t), \tag{2.5}$$

where the E and $\vec{\Omega}$ dependencies are eliminated to get a lighter notation. The result is

$$N(\vec{r},t) = N(\vec{r},\bar{t})\,e^{-\sigma(\vec{r})\int_{\bar{t}}^{t}\phi(\vec{r},t')dt'}. \tag{2.6}$$

By considering the flux constant in time, the expression becomes

$$N(\vec{r},t) = N(\vec{r},\bar{t})\,e^{-\sigma(\vec{r})\,\phi(\vec{r})\,(t-\bar{t})}. \tag{2.7}$$

[6] Actually, there is also the possibility to consider a constant power released within the time interval. This approach leads to different results but, for sufficiently small time intervals, the two approaches are, at first approximation, analogous [10].

2.2.2. Bateman equations

The simple example just explained is only a hint on how the mathematical problem is tackled since it is way more complex due to the presence of different isotopes (thus, an equation is needed for each isotope) and each isotope can transform into another one (capture, fission, decay, etc.) making the equations coupled to each other; therefore, the result is a system of coupled partial differential equations in time as many as the number of isotopes that are chosen to be studied: the so-called Bateman equations [12].

The Bateman equations can be expressed as:

$$\frac{dN_i}{dt} = \sum_j (\gamma_{j,i}\sigma_{f,j})_g \, N_j \, \phi^g + \sigma^g_{\gamma,i-1} \, N_{i-1} \, \phi^g + \lambda_{i'} \, N_{i'} - \sigma^g_{f,i} \, N_i \, \phi^g - \sigma^g_{\gamma,i} \, N_i \, \phi^g - \lambda_i \, N_i \quad (2.8)$$

The last term refers to the loss due to radioactive decay of the isotope of interest i, while $+\lambda_{i'} N_{i'}$ refers to the decay on another isotope i' to the isotope i. The term $-\sigma^g_{\gamma,i} N_i \phi^g$ expresses the neutron capture possibility of the isotope i and $-\sigma^g_{f,i} N_i \phi^g$ its fission occurrence, while $+\sigma^g_{\gamma,i-1} N_{i-1} \phi^g$ indicates the neutron capture of isotope $_{i-1}$. The first term is the sum all over the isotopes that can undergo fission and can produce ($\gamma_{j,i} \neq 0$) isotope i as fission product. The g letter indicates that the quantities are group constants, so energy independent. For example:

$$(\gamma_{j,i}\sigma_{f,j})_g = \frac{\int_{E_g}^{E_{g-1}} \gamma_{j,i}(E) \, \sigma_{f,j}(E) \, \phi(\mathbf{r}, E, t) \, dE}{\phi_g(\vec{r}, t)}, \quad (2.9)$$

where

$$\phi_g(\vec{r}, t) = \int_0^\infty \phi(\mathbf{r}, E, t) \, dE. \quad (2.10)$$

Equation (2.8) can also be written in matrix form:

$$\frac{d\underline{N}}{dt} = \underline{\underline{A}} \, \underline{N}(t) + \underline{F}(t), \quad (2.11)$$

where, by adding $\underline{F}(t)$, also external sources are considered.

In principle, these equations are non-linear because the fluxes also depend on the isotopic densities; however, by choosing a sufficiently small time interval Δt, the flux and the matrix $\underline{F}(t)$ can be approximated as constant. This linearization allows the resulting ordinary differential equation to be solved as follows [10]:

$$\underline{N}(t + \Delta t) = e^{\underline{\underline{A}} \, \Delta t} \underline{N}(t) + \underline{\underline{A}}^{-1} (e^{\underline{\underline{A}} \, \Delta t} - 1) \underline{F}(t). \quad (2.12)$$

Then, having determined the densities at Δt, a new Boltzmann equation solution is performed to get the new fluxes and so one for each time step. If the time step is small, a Taylor expansion can be performed:

$$e^{\underline{\underline{A}}\,\Delta t} = 1 + \underline{\underline{A}}\,\Delta t + \frac{1}{2}(\underline{\underline{A}}\,\Delta t)^2 + \cdots = \sum_{k=0}^{\infty} \frac{(\underline{\underline{A}}\,\Delta t)^k}{k!}. \tag{2.13}$$

Then, by stopping the series at a certain order (the one that ensures convergence of the results within a chosen precision), the solution can be obtained.

For example, as shown in [13], a linear chain with n distinct decay constants λ_i, $i = 1, \ldots, n$, and branching ratios $b_{i,i+1}$ can be solved analytically. Assuming that only the first nuclide has an initial atomic density, $x_1(0)$, the atomic density of the n-th nuclide after time t is

$$N_n(t) = N_1(0)\, B_n \sum_{i=1}^{n} \alpha_i^n\, e^{-\lambda_i t}, \tag{2.14}$$

where

$$B_n = \prod_{j=1}^{n-1} b_{j,j+1}, \qquad \alpha_i^n = \frac{\prod_{j=1}^{n-1} \lambda_j}{\prod_{\substack{j=1 \\ j\neq i}}^{n} (\lambda_j - \lambda_i)}. \tag{2.15}$$

It has to be pointed out that there is a further complication for the resolution of this system of equations that is the presence of thousands of different nuclides (thus thousands of equations) with widely varying half-lives, resulting in the need of a significant computational time to solve it. This means that the burnup matrix has a wide eigenvalue spectrum making its approximation difficult. Moreover, all the eigenvalues of the $\underline{\underline{A}}$ matrix representing a general time problem, like the one in Equation (2.11), must have their real part lower than zero to ensure the stability in time of the solutions. Due to the complexity and high computational time cost of this problem, different approaches have been investigated.

One could think to lump several nuclides with similar half-lives into an *effective* one, diminishing the number of equations; however, this approach results in a non-physical representation of the problem. Other techniques could consider the use of simplified burnup chains or treat the most short-lived nuclides separately when computing a matrix exponential solution. Studies carried out in the past years pointed out the better performance of CRAM (Chebyshev Rational Approximation Method) [13, 14].

2.2.3. CRAM (Chebyshev Rational Approximation Method)

The method employed in this thesis is a matrix exponential approach that differs from the Taylor expansion method in how it handles the $e^{\underline{\underline{A}}\,\Delta t}$ term. This approach relies on the observation that the eigenvalues of the burnup matrix generally remain bounded near the negative real axis in practical calculations [14]. Consequently, the exponential function can be accurately represented using a Chebyshev rational approximation over the interval $(-\infty, 0]$.

Recalling the Cauchy integral formula, the solution of Equation (2.11) (assuming $\underline{F}(t) = 0$, i.e., no external sources) is expressed as a contour integral:

$$\underline{N}(t) = \underline{N}_0\, e^{\underline{\underline{A}}\,t} = \frac{1}{2\pi i} \int_\Gamma e^z\, (z\underline{\underline{I}} - \underline{\underline{A}}\,t)^{-1} \underline{N}_0\, dz, \tag{2.16}$$

where Γ is a closed contour of the spectrum of $\underline{\underline{A}}\,t$. Since the elements of the resolvent operator $(z\underline{\underline{I}} - \underline{\underline{A}}\,t)^{-1}$ are rational functions [14], calculating $\underline{N}(t)$ reduces to evaluate integrals where the singularities coincide with the eigenvalues of $\underline{\underline{A}}\,t$, as stated by the theory of analytic functions, where

$$I = \frac{1}{2\pi i} \int_\Gamma e^z f(z)\, dz. \tag{2.17}$$

When these singularities are confined near the negative real axis, Γ can be widened into a parabolic or hyperbolic shape. In this region, the integrand decreases exponentially, allowing for efficient approximation via numerical quadrature. As demonstrated in [14], such quadrature formulas are mathematically equivalent to rational approximations. Specifically, the quadrature nodes and weights correspond to the poles and residues of a rational function $r(z) \approx e^z$.

The Chebyshev rational approximation $\hat{r}_{k,k}$ is specifically constructed to minimize the absolute deviation from the exponential function on the negative real axis. Because its poles $\{\theta_1, \ldots, \theta_k\}$ are distinct, the approximation can be computed using a partial fraction expansion:

$$e^z \approx \alpha_0 + \sum_{i=1}^{k} \frac{\alpha_i}{z - \theta_i} = \alpha_0 + \mathrm{Re}\left(\sum_{i=1}^{k/2} \frac{\alpha_i}{z - \theta_i} \right). \tag{2.18}$$

The final mathematical formulation for the density vector becomes

$$\underline{N}(t) \approx \alpha_0\, \underline{N}_0 - \mathrm{Re}\left(\sum_{i=1}^{k/2} (\theta_i \underline{\underline{I}} + \underline{\underline{A}}\,t)^{-1} \alpha_i\, \underline{N}_0 \right). \tag{2.19}$$

This allows the system to be solved by computing $k/2$ sparse linear systems. Efficiency is further improved by indexing nuclides in ascending order of mass number, which concentrates non-zero elements around the diagonal. This structure enables the use of symbolic lower–upper (LU) factorization and Gaussian elimination to solve the burnup matrix rapidly [15, 16].

2.2.4. Predictor–corrector methods

As already explained, the most straightforward approach for burnup calculations is the stepwise depletion method which proceeds as follows [17]:

1. Solve neutronics using the beginning-of-step (BOS) (t_0) material compositions to obtain cross sections $\sigma(t_0)$ and flux;
2. Predict $\sigma(t)$ for $t_0 \leq t \leq t_1$ using $\sigma(t_{-1})$ and $\sigma(t_0)$. Then, the step average is computed by

$$\bar{\sigma} = \frac{1}{t_1 - t_0} \int_{t_0}^{t_1} \sigma(t)\, dt. \tag{2.20}$$

   So, materials are depleted from t_0 to t_1 using constant step average reaction rates;
3. Solve neutronics for $\sigma_{pred}(t_1)$ using EOS compositions obtained in the previous steps;
4. Re-predict $\sigma(t)$ using $\sigma(t_{-1})$, $\sigma(t_0)$, and $\sigma_{pred}(t_1)$;
5. Recompute $\bar{\sigma}$ and deplete materials;
6. The resulting end-of-step (EOS) compositions then become the starting compositions for the next step.

While simple, this method is inaccurate because reaction rates generally vary within each step, necessitating very short step lengths. To improve accuracy, predictor-corrector methods are commonly used. These methods require two neutronics solutions per step: one at BOS and one at EOS. But the trade-off allows longer steps. A standard predictor-corrector scheme proceeds by modifying steps 2 and 5 as follows [17]:

- Divide the step into S substeps: $T_0 = t_0 < t_1 < \cdots < t_S = T_1$;

- For each substep $s = 1, \ldots, S$, compute

$$\bar{\sigma} = \frac{1}{t_s - t_{s-1}} \int_{t_{s-1}}^{t_s} \sigma(t) dt, \tag{2.21}$$

  and deplete materials over that sub-interval.

The simplest choice is to use a constant BOS value, equivalent to constant extrapolation (CE), while the corrector step uses a linear interpolation (LI) between BOS and EOS values.

Accuracy can be further improved by including information from the previous step. Using cross sections and flux from the previous step (σ_{PS}), BOS (σ_{BOS}) and EOS (σ_{EOS}) of the current step, one can construct higher-order fits: linear for the predictor step (LE) and quadratic for the corrector step (QI). No additional neutronics solutions are needed because previous step values are available.

Since the Bateman solutions require constant reaction rates during each step, higher-order predictions are approximated by step averages:

$$\bar{\sigma} = W_{\mathrm{PS}}\sigma_{\mathrm{PS}} + W_{\mathrm{BOS}}\sigma_{\mathrm{BOS}} + W_{\mathrm{EOS}}\sigma_{\mathrm{EOS}}, \tag{2.22}$$

where the weights $W_{\mathrm{PS}}, W_{\mathrm{BOS}}, W_{\mathrm{EOS}}$ correspond to the contribution of each point. The same approach applies to fluxes and any derived quantities, such as power per unit flux or total power.

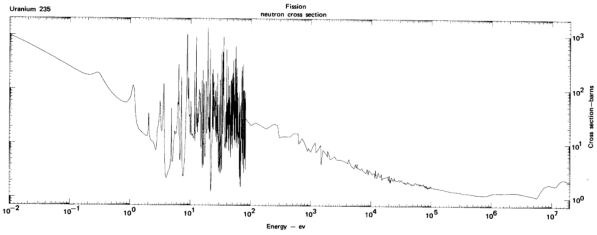
Predictor steps can use constant (CE) or linear extrapolation (LE), while corrector steps can use linear (LI) or quadratic interpolation (QI). It is also possible to omit the corrector step and halve the step length instead. This leads to six variants: CE, LE, CELI, CEQI, LELI, LEQI. It has to be noted that linear extrapolation and quadratic interpolation cannot be used at the first step due to the absence of previous step data, thus being downgraded to CE and LI. Studies [17–19] proved that the implementation of LELI with 3 predictor and 5 corrector substeps offers the best choice in terms of accuracy and computational cost. Actually, several further studies [20–23] show that the previous sentence is actually really restricted to the particular cases studied in [17–19] and should not be extended without checking thoroughly to the particular system of interest one is studying.

2.3. Energy self-shielding

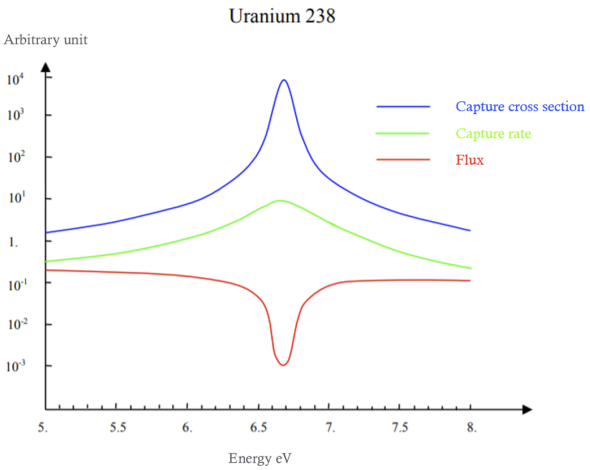
The physical energy self-shielding phenomenon occurs when the neutron flux drops as a function of energy in a medium where the macroscopic capture cross section is high.

This results in reaction rates that exhibit variations much slower than those of the cross sections and flux, as reaction rates are the product between the flux and cross sections. In deterministic multigroup codes, modeling this compensation is indispensable; otherwise, for example, the absorption rate of ^{238}U is overestimated and the effective multiplication factor is underestimated.

A multigroup approach consists of integrating (thus averaging) the neutron transport Boltzmann equation over a certain energy range. Hence, the energy self-shielding is particularly relevant at nuclide resonance energies, where the cross section has steep peaks rather than a smooth decreasing behavior, as seen for the fission cross section of ^{235}U in Figure 2.2a. This behavior is also evident in capture cross sections, such as the 6.67 eV peak for ^{238}U (see Figure 2.2b), where the cross section varies by four decades while the reaction rate varies by about one decade.



(a) Fission cross section of ^{235}U (from [10]).



(b) Energy self-shielding for ^{238}U (from [5]).

Figure 2.2: Comparison of cross section behaviors: fission characteristics of ^{235}U (left) and resonant self-shielding in ^{238}U (right).

Taking into account the self-shielding phenomenon is vital for multigroup transport codes and can be seen as the final step of cross section treatment. In thermal neutron reactors, the heavy nuclei absorption resonances are the major ones to be treated; however, for fast neutron reactors or specific reflector calculations, scattering resonances of lighter nuclei such as sodium or iron must also be taken into account.

2.3.1. Self-shielding in neutronics codes

The data for the multigroup equation are multigroup cross sections defined to ensure that, for each group g, the multigroup reaction rate equals the integrated pointwise reaction

rate. The goal is to preserve the number of reactions per unit time and volume in each group:

$$\sigma^g(\vec{r})\phi^g(\vec{r}) = \int_g \sigma_g(E)\phi(\vec{r},E)dE. \tag{2.23}$$

Since reaction rates depend on space through the flux, it is impossible to build universal, geometry-independent multigroup libraries. Strictly speaking, the pointwise flux in Equation (2.23) is unknown before solving the transport problem. To simplify this, libraries often use a known weighting spectrum $\phi_w(E)$ independent of space, leading to *infinite-diluted* cross sections:

$$\sigma^g_{g,\text{inf}} = \frac{\int_g \sigma(E)\phi_w(E)du}{\int_g \phi_w(E)du}. \tag{2.24}$$

While this simplification works for slowly varying cross sections, it is unacceptable in the resonance range where cross sections can vary by decades within a single group. In such cases, self-shielded cross sections must be calculated for the specific geometry.

2.3.2. Principles of self-shielding modeling

Self-shielding calculation requires assessing the actual flux $\phi(\vec{r},E)$, typically in the slowing-down range. The medium is discretized into self-shielding regions α where the flux $\phi_\alpha(E)$ is assumed space-constant but energy-pointwise:

$$\sigma^g_\alpha = \frac{\int_g \sigma(E)\phi_\alpha(E)du}{\phi^g_\alpha}. \tag{2.25}$$

Two main families of methods exist for this task [24]:

- *Equivalence methods*: these establish an equivalence with an infinite homogeneous medium using a *fine structure equation*. Pointwise reaction rates are derived from existing tabulations, followed by a *multigroup equivalence* to preserve these rates in the Boltzmann equation;
- *Subgroup methods*: these assume that resonant cross sections and slowing-down sources are uncorrelated. This allows for a direct solution of the slowing-down equation in each region α without specific modeling of the resonant operator.

2.3.3. Resonant mixture treatment

A major difficulty arises when resonant isotopes overlap, as in MOX fuel due to plutonium and uranium capture cross sections. The treatment often considers the mixture as a unique resonant entity. In the *Livolant-Jeanpierre* method, the fine structure equation for M resonant nuclei is written as [24]:

$$\begin{cases} \phi_\alpha(E) = \sum_\beta c_{\alpha\beta}(E) R_{0\beta}\, \phi_\beta(E) + SS_\alpha(E) \\ R_{0\beta}\, \phi_\beta(E) = \sum_{m=1}^{M} a_{0m\beta}\, r_{0m\beta}\, \phi_\beta(E), \end{cases} \tag{2.26}$$

where $\phi_\alpha(E)$ is the fine structure factor in region α, $r_{0m\beta}\, \phi_\beta(E)$: slowing-down operator of the m component of the resonant mixture in region β and $a_{0m\beta}$ is the proportion in the resonant mixture of component m in region β. $c_{\alpha\beta}(E)$ matrix is deduced from the probability collision matrix [24] that gives the space coupling between the different self-shielding regions and the vector $S_\alpha(E)$, the sources resulting from non-resonant nuclei. Because multigroup calculations lose fine energy transfer details, operators r_{0m} are simplified using slowing-down models like Narrow Resonance (NR), Wide Resonance (WR), or *Toute Résonance* (TR). The TR model, the recommended one, approximates the scattering probability and its accuracy improves with finer mesh refinement.

To eliminate modeling errors, an *equivalent dilution* is sought for each component m so that the TR-computed absorption rate matches that of an infinite homogeneous medium. The final corrected reaction rates are then obtained by solving the fine structure equation on an extremely fine mesh of over ten thousand groups, ensuring a nearly exact result.

3 | MOC - Method Of Characteristics in APOLLO3®

Mathematically, the Method of Characteristics (MOC) is a well-established technique for solving first-order linear Partial Differential Equations (PDEs). This method is particularly suitable for addressing the Boltzmann transport equation (Equation (2.1), Equation (2.4)) due to its structure. The idea behind MOC is to transform the original PDE into a more manageable form by switching to a curvilinear coordinate system. In this new framework, one identifies specific paths in the domain, known as characteristic curves, along which the PDE simplifies significantly. By integrating the equation along these curves, the original PDE is effectively reduced to an Ordinary Differential Equation (ODE). This transformation is advantageous as ODEs are generally easier to solve, both analytically and numerically. The simplification occurs because, along the characteristic lines, the gradient that typically involves multiple partial derivatives can be expressed as a single-variable derivative. This means that the multidimensional nature of the problem collapses into a one-dimensional form along each characteristic, allowing for a more straightforward analysis of the system's behavior.

3.1. MOC in neutronics

In neutronics, MOC is a primary method for solving the monogroup neutron transport equation in reactor lattice calculations. It focuses on the integral transport equation to compute the angular flux along characteristic lines across the computational domain. The following derivation is based on [24], where MOC is presented in 2D within an energy group g at constant time. Equation (2.1) can be written as

$$\vec{\Omega} \cdot \nabla \phi(\vec{r}, \vec{\Omega}) + \Sigma(\vec{r}) \phi(\vec{r}, \vec{\Omega}) = q(\vec{r}, \vec{\Omega}), \tag{3.1}$$

where $q(\vec{r}, \vec{\Omega})$ represents the source terms. The method relies on specific discretizations:

- *Space variable*: The domain $\mathcal{D}$ is partitioned into I homogeneous regions $\mathcal{D}_i$. In the so-called Step Characteristics (SC) approximation, cross sections and sources are

assumed to be constant per region:

$$\Sigma(\vec{r}) = \Sigma_i, \quad q(\vec{r}, \vec{\Omega}) = q_i(\vec{\Omega}), \quad \forall \vec{r} \in \mathcal{D}_i; \tag{3.2}$$

- *Angle variable*: The S_N formulation selects a finite set of discrete directions $\{\vec{\Omega}_n, \omega_n\}$. Angular integration is performed via a quadrature formula:

$$\int_{4\pi} f(\vec{\Omega})\, d\vec{\Omega} = \sum_{n=1}^{N} \omega_n\, f(\vec{\Omega}_n). \tag{3.3}$$

For each direction, characteristic lines intercepting region boundaries provide points for the homogeneous approximation. Each trajectory represents a portion of the domain, and parallel trajectories cover the full domain for each direction. The approximation improves as trajectories become closer, as shown in Figure 3.1.

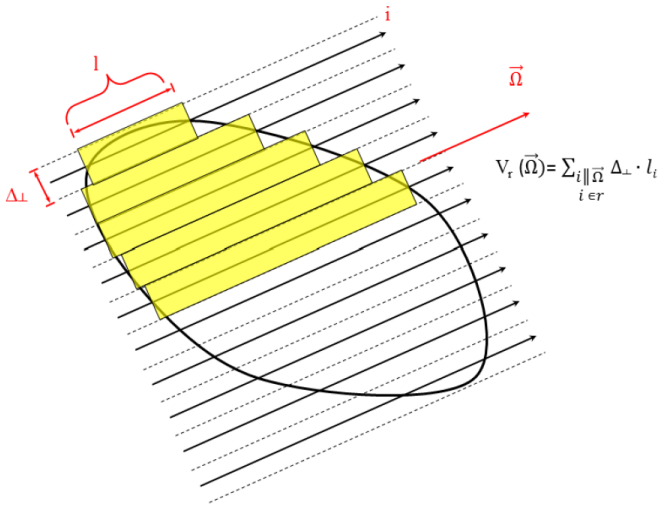


Figure 3.1: 2D trajectory-based discretization. In yellow, the angular volume associated to each trajectory and r the discretization region (from [3]).

On a characteristic line i with direction $\vec{\Omega}$, the transport equation becomes a 1D problem along the curvilinear abscissa s:

$$\frac{d\phi(\vec{r}_{0i} + s\vec{\Omega}, \vec{\Omega})}{ds} + \Sigma_t(\vec{r}_{0i} + s\vec{\Omega})\phi(\vec{r}_{0i} + s\vec{\Omega}, \vec{\Omega}) = q(\vec{r}_{0i} + s\vec{\Omega}, \vec{\Omega}). \tag{3.4}$$

It can be noted that trajectory-based quantities inherit an angular dependence. The the angular volume $V_r(\vec{\Omega})$ and the numerical volume V_r are defined as

$$V_r(\vec{\Omega}) = \sum_{i||\vec{\Omega}, i\in r} \Delta_\perp(\vec{\Omega})\, l_i, \quad V_r = \oint \frac{d\vec{\Omega}}{4\pi} V_r(\vec{\Omega}), \tag{3.5}$$

where l_i is the chord length and $\Delta_\perp$ is the distance between parallel characteristics. While numerical values tend to analytical ones as $\Delta_\perp \to 0$, they often differ in real applications. Unlike some codes that normalize chords to match analytical volumes, the solver analyzed in this work, TDT (Two-Three Dimensional Transport), does not normalize chords in order to preserve the physical representation of particle transport. This results in the need of a sufficiently fine mesh. Moreover, one noteworthy feature of MOC is its high versatility, thanks to its capability of treating complex Cartesian geometries without requiring specific regularity.

3.1.1. Different MOC families: short and long characteristics

Two main families of MOC-based methods are used for neutron transport along straight lines: *Long Characteristics* and *Short Characteristics* (also known as MOSC). The TDT solver implements the long characteristics approach, where trajectories cover the entire domain from boundary to boundary, following particles along the full length of each line.

In contrast, the short characteristics approach, adopted by solvers like the Integro-Differential Transport (IDT) one[25], solves the integral transport equation across individual regions. In this method, trajectories are used only during an initial phase to compute coupling coefficients between volume emission densities and surface fluxes. These coefficients allow for the propagation of surface fluxes across the domain without treating each chord individually. Because surface flux transmission is computationally cheaper than trajectory-based tracking, MOSC generally offers lower execution costs, though it requires significantly more memory to store the coefficients which is already an issue for long characteristics MOC.

Furthermore, the short characteristics method involves certain geometric limitations. Since surface fluxes are typically expanded using low-order polynomials on planar surfaces, modeling irregular boundaries often requires fine triangular or tetrahedral meshes, leading to a high number of regions. While IDT introduces optimizations to identify identical cells and can handle heterogeneous Cartesian cells with concentric circles, it remains less geometrically general than the long characteristics approach.

The strength of the long characteristics method in TDT lies in its ability to treat arbitrary region shapes. By recognizing line and arc segments as boundaries, it provides an exact representation of standard reactor geometries, such as extruded grids of rectangular cells containing fuel rods and guide tubes. Its limitation is the need to identify each

intersection point between a trajectory and a region boundary.

Despite its flexibility, TDT is demanding in terms of floating-point operations and memory needed to store chord positions, hence these requirements have historically slowed the extension of MOC to 3D; nevertheless, the recent introduction of high-order spatial methods, such as axial polynomial expansions for the flux and macroscopic cross sections, has mitigated storage issues, allowing for efficient 3D calculations on modern high-performance computers.

3.1.2. Transmission and balance equations

In actual calculations, the integral transport equation is adapted to the domain's discretization into homogeneous regions. Within these regions, cross sections are constant between integration boundaries, simplifying the formulation of Equation (2.4) into

$$\phi^+(\vec{\Omega}) = \phi^-(\vec{\Omega})e^{-\Sigma_r l} + \int_0^l q(\vec{r}(t), \vec{\Omega})e^{-\Sigma_r(l-t)}dt, \tag{3.6}$$

where ϕ^+ and ϕ^- represent the exiting and entering fluxes along a trajectory of length l. The notation is detailed in Figure 3.2, where φ is used instead of ϕ.

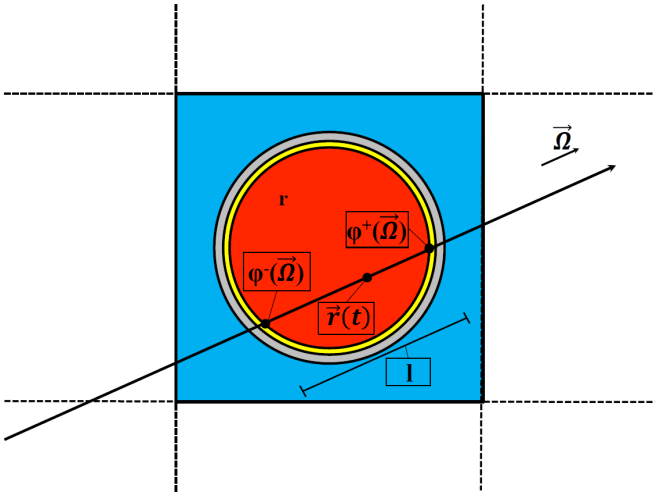


Figure 3.2: Graphical representation of entering and exiting fluxes for a region along a trajectory (from [3]).

The MOC solution is defined by two fundamental equations:

1. *Balance Equation*: this equation imposes neutron conservation within each region V_r. Integrating Equation (3.6) and applying the divergence theorem yields [3]

$$\Sigma_r \bar{\phi}_r(\vec{\Omega}) = q_r(\vec{\Omega}) - \Delta J_r(\vec{\Omega}), \tag{3.7}$$

where the current term $\Delta J_r(\vec{\Omega})$ represents the net flux difference across the region boundaries, computed by summing the contributions of all trajectories crossing the region.

2. *Transmission Equation*: this evaluates the flux variation along characteristics. Under a constant source approximation, it is formulated in the following way to minimize floating-point operations and handle small cross sections efficiently [3]:

$$\phi^+(\vec{\Omega}) - \phi^-(\vec{\Omega}) = \left(\frac{q_r(\vec{\Omega})}{\Sigma_r} - \phi^-(\vec{\Omega}) \right) (1 - e^{-\Sigma_r l}). \tag{3.8}$$

Indeed, the term $(1 - e^{-\Sigma_r l})$ is typically pre-tabulated or evaluated via Taylor expansion for small optical paths.

3.1.3. Iterative strategy

The solution of the transport equation via the method of characteristics is obtained through several steps which consist in nested iteration loops. At the beginning, the flux is initialized in each fissile region and the outermost iteration loop starts initializing the fission source term for each energy group, using the initial angular flux moments guess. Then, starting from the highest energy group, a transfer contribution coming from other groups is added to the source term. This defines the second iteration loop while the third level is constituted by the transfer within the same group. This final contribution allows one to define the source term of the balance equation (Equation (3.7)).

The last term of this equation is computed using the trajectory sweep by solving the transmission equation (Equation (3.8)) for each chord of each 3D trajectory. The balance equation is used to compute the average angular flux in each region, which is to estimate a new value for the moments of the flux for the considered energy group. This new evaluation is used to update the transfer term within the same group, until a converged value is obtained. This inner-most level of iteration is generally referred to as inner iterations. At this point, it is possible to update the transfer term coming from the group just computed, and passing to the solution of the successive energy level. The procedure is repeated until the fluxes have converged in each group. Consequently, the new set of fluxes is used to compute again the fission source for all the energy groups, and a new outer iteration starts. The new eigenvalue estimation is also computed using the fission integrals of two consecutive outer iterations. The procedure is repeated until convergence of the angular flux moments in each group and for each region is attained. In this work, convergence criteria were set at 10^{-5} for the eigenvalue and inner iterations,

and 10^{-4} for the fission integral.

To manage the significant time and memory requirements, several techniques are employed:

- *Chord Classification Method (CCM)*: MOC-3D is memory-intensive, with chord counts reaching 10^6 to 10^9 [3] in a typical reactor geometry. For extruded geometries, CCM exploits the regularity of 3D tracking based on a 2D footprint, as illustrated in Figure 3.3. Chords are classified into three types:
  - *V-chords*: chords crossing two vertical surfaces: $l^V_{i,3D} = \frac{l_{i,2D}}{\sin\theta}$.
  - *H-chords*: chords crossing two successive horizontal surfaces: $l^H_{i,3D} = \frac{\Delta z}{\cos\theta}$.
  - *Mixed chords (M-chords)*: chords entering through a horizontal surface and exiting a vertical one (or vice-versa). These are stored in a non-recognized chord structure.

  This reduces storage and limits floating-point operations by pre-computing transmission coefficients ($\beta = 1 - e^{-\Sigma_r l}$) for classified chords;

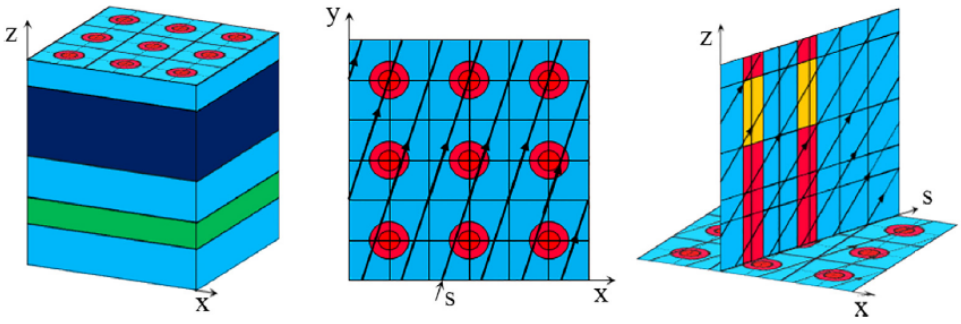


Figure 3.3: TDT 3D tracking strategy (from [5]).

- *Hit Surfaces Sequence (HSS)*: This method compacts tracking information by storing sequences of surface impacts as integers. A number > 1 indicates consecutive impacts of the same type (Vertical or Horizontal), with the sign distinguishing the type. Mixed chords are represented by a "1". This allows the solver to sweep trajectories while efficiently identifying chord types across the 2D footprint and axial layers;

- *Parallel Strategy*: An OpenMP-based algorithm parallelizes the trajectory sweep. Since trajectories are independent, they are distributed among threads. To avoid *race conditions* (simultaneous attempt to modify the same variable) when updating

the current term $\Delta J_r(\vec{\Omega})$, each thread maintains a private copy of the structure. A load-balancing algorithm divides trajectories into packages sorted by computational cost and assigns them dynamically to maximize efficiency;

- *Acceleration Strategies (DP_n)*: To reduce the number of iterations, a synthetic acceleration approach is used for internal and external iterations. It solves a transport problem where the source term depends on the difference between successive iterations, providing a better estimation of the converged flux.

3.2. MOC in APOLLO3®: higher-order axial polynomial expansions

APOLLO3® is the deterministic neutron transport simulation code developed by the French Alternative Energies and Atomic Energy Commission (CEA) in collaboration with EDF and Framatome. Within its TDT MOC-3D solver, a new higher-order axial expansion method for the neutron flux was introduced in [3], which is based on the following space-angle moments expansion for the angular neutron flux ϕ:

$$\phi(\vec{r},\vec{\Omega}) \approx \sum_{n=1}^{N_m} A_n(\vec{\Omega})\,\Phi_n(\vec{r}), \quad \text{with} \quad \Phi_n(\vec{r}) = \int \frac{d\vec{\Omega}}{4\pi} A_n(\vec{\Omega})\,\phi(\vec{r},\vec{\Omega}), \tag{3.9}$$

which is the classical flux expansion with respect to the set of real spherical harmonics $\vec{A}(\vec{\Omega})$ truncated at order $N_m = (K+1)^2$, with K being the maximum anisotropy order. Each angular moment is further expanded according to

$$\Phi_n(\vec{r}) \approx \sum_{p=0}^{N_p} P_p(\vec{r})\,\Phi_{n,p,\mathcal{D}}, \quad \text{with} \quad P_p(\vec{r}) = \left(\frac{z-\bar{z}_{\mathcal{D}}}{\Delta z_{\mathcal{D}}/2}\right)^p, \tag{3.10}$$

where N_p is the chosen maximum polynomial order, and the axial coordinate z is normalized in such a way that it ranges between -1 and 1 within each 3D computational region $\mathcal{D}$ ($\Delta z_{\mathcal{D}}$ being the region's height and $\bar{z}_{\mathcal{D}}$ the midheight z value within $\mathcal{D}$, respectively). Then, a further study [5], expanded also the macroscopic cross sections using the same polynomial basis as the flux, given the interdependence between the two physical quantities:

$$\Sigma(\vec{r}) \approx \sum_{p=0}^{N_p} \Sigma_{p,\mathcal{D}}\,P_p(\vec{r}). \tag{3.11}$$

This polynomial expansion is limited to the axial direction, as typical nuclear reactor geometries are extruded. On the radial plane, therefore, the classical step-constant approximation (i.e., a different constant value in each region) is still adopted for both flux and cross sections.

Trajectories in 3D are traced according to the strategy illustrated in Figure 3.3. For each 2D characteristic s on the x-y plane, the corresponding 3D characteristics are traced on the s-z plane, obtained by extruding s along the axial direction.

The interested reader can find in [3, 5] all details regarding the reformulation of the MOC transmission and balance equations, as well as the DP_N equations to incorporate the flux and cross sections polynomial approximations.

3.3. Depletion coupling scheme: MENDEL code

An exact solution to the depletion problem would require a fully coupled system involving the time-dependent Boltzmann equation, the Bateman equations, and thermal-hydraulic feedbacks. Due to the computational cost, a simplified strategy decouples the neutron transport and depletion calculations. This separation remains physically consistent through the quasi-static approximation: nuclide density evolution is slow enough that their time derivatives can be neglected during flux calculations. In this framework, the MENDEL code [26] solves the Bateman equations at each time step Δt. To optimize the interaction with APOLLO3®, predictor-corrector schemes are employed, utilizing a dual time mesh: a coarse mesh for transport calculations and a finer mesh for depletion substeps, as reported in Figure 3.4.

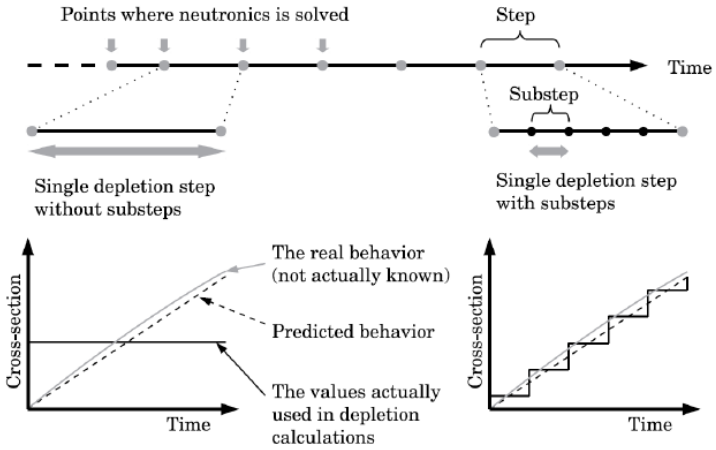


Figure 3.4: Depletion-neutronics coupling with (right) and without (left) the substep method (from [18]).

Each time step in APOLLO3® consists of:

- *Predictor phase*: estimation of nuclide concentrations using extrapolated fluxes from previous solutions;
- *Corrector phase*: refinement of concentrations using fluxes interpolated between predicted and previous values;
- *Step halving*: if corrected concentrations exceed a set tolerance compared to predicted ones, the time step is halved and the process repeats.

It should be noted that this approach differs significantly from the predictor-corrector methods adopted by some stochastic codes (discussed in the next chapter). In those cases, the correction acts as a refinement of the predicted values rather than a convergence check against a tolerance. These tools typically avoid repeating calculations with halved time steps to save computational costs, as probabilistic methods are inherently more time-consuming than deterministic ones. Consequently, they require a rigorous depletion mesh convergence analysis, as will be detailed in later chapters. Indeed, if the chosen step is too coarse, the correction fails to properly adjust the results, leading to inaccurate outcomes.

It is really important to know that the higher-order spatial method developed in [5] has not yet been extended to the depletion solver, which currently handles only spatially constant quantities; therefore, an equivalence between the two spatial representations: polynomial and stepwise constant has to be set. Consequently, the coupling strategy adopted employs the use of both a step-constant and a polynomial mesh, named after the axial expansion employed for the flux and the cross sections: starting from macroscopic cross sections values which are constant in each region of the former mesh, the polynomial coefficients for the corresponding region of the coarser mesh are retrieved imposing the conservation of polynomial moments. After performing the flux calculation on the polynomial mesh, the flux is converted into its step-constant mesh by the inverse procedure of the cross sections conversion and received as input by APOLLO3® depletion solver, MENDEL.

3.4. Self-shielding

As explained in Section 2.3, two major techniques can be adopted to model cross section behavior in proximity of resonances: the Livolant-Jeanpierre method (which is an equivalence method) and the Subgroups method. The first consists basically in a more accurate representation of the in-group flux which is solution of a slowing-down problem under the hypotheses of pure elastic and isotropic scattering based on a homogeneous-heterogeneous

equivalence. The latter is applicable when the group discretization is fine enough (refined energy meshes), thus one can approximate the average of the product of two quantities over a group as the product of their averages. Hence, it is reasonable to assume that the resonances of different isotopes do not overlap and thus it is acceptable to treat one resonant nuclide at a time.

Overall, the Livolant-Jeanpierre technique introduces greater approximations, particularly regarding the fine-structure treatment of the slowing-down term. On the other hand, the subgroups method requires a finer energy discretization to be consistent. This is the main reason why the Livolant-Jeanpierre approach is usually adopted for the case studies considered in this work, as the SHEM281 [27] energy group subdivision was adopted (instead of a finer one).

Finally, the higher-order spatial description that this thesis extends to macroscopic cross sections does not involve the self-shielding procedure. Thus, as for the depletion code, the fine-structure flux and the infinite dilution multigroup cross sections, as well as the resulting self-shielded ones, remain constant within each computational region. Hence, an ad hoc procedure is required each time the flux and the self-shielding solvers have to interact. More specifically, the step-constant self-shielded cross sections need to be converted into polynomial at the beginning of each flux calculation and, if more than one depletion step is to be performed, the computed flux is then reconverted into step-constant format and used as input for the depletion solver, before the next application of the self-shielding method. This strategy implies the need to have two spatial meshes. It is usually chosen to have the same SC mesh for the self-shielding and depletion to simplify mesh generation.

4 | Monte Carlo methods: Serpent and TRIPOLI-4®

A Monte Carlo method is a probabilistic (or stochastic) approach that has been extensively applied in radiation and particle transport simulations since the late 1940s. This technique operates in three spatial dimensions, is adaptable to arbitrarily fine spatial resolutions, and provides highly accurate modeling of transport physics without relying on major approximations or constraints tailored to specific applications, making it ideal for complex geometries and heterogeneous media; however, the main drawback of Monte Carlo methods has always been their high computational cost, which can lead to extremely long simulation times. For instance, a 3D assembly depletion simulation using TRIPOLI-4® can require over a month of calculation time. This computational challenge prompted CEA, EDF, Framatome and the French Nuclear Safety and Radiation Protection Authority (ASNR) to invest in the development of a new proprietary code, TRIPOLI-5®, specifically designed to alleviate this burden. The new code is expected to become fully operational in the early 2030s.

Historically, the excessive runtime has been the primary limiting factor of Monte Carlo methods, particularly in scenarios involving iterative simulations or sensitivity analyses, both of which are fundamental in nuclear reactor design and safety assessments. Consequently, until the late 20th century, deterministic transport methods were the only viable option for such applications; however, the rise of High Performance Computing (HPC) and parallel processing technologies towards the end of the millennium allowed the use of Monte Carlo techniques. The first Monte Carlo burnup calculation codes emerged in the late 1990s and their use became a vibrant and promising field of research.

4.1. Working characteristics

A Monte Carlo method is a computational algorithm that relies on the generation of *random*[1] numerical quantities to simulate particle behavior within a given environment. By employing statistical sampling and probabilistic tracking, this method enables the

[1] In Section 4.1.1, the true *pseudorandom* nature of a Monte Carlo method will be addressed.

estimation of physical quantities of interest with high accuracy, especially in complex systems.

The term "Monte Carlo" originates from the Monte Carlo Casino in Monaco, once renowned as the world's most famous gambling venue.[2] The name reflects the inherent randomness of the method, which resembles games of chance[3].

Monte Carlo simulations can be interpreted as controlled random experiments, and they are particularly valuable when the outcomes of a system cannot be determined analytically (except by making several approximations) due to the presence of numerous uncertain parameters. In such cases, analytical solutions may be either infeasible or extremely time-consuming. This is precisely why Monte Carlo methods are often preferred over deterministic approaches such as the Method of Characteristics (MOC) implemented in APOLLO3®, especially when it is essential to capture the full complexity and variability of the physical system under study.

4.1.1. Pseudorandomness

Monte Carlo simulation methods do not necessarily require truly random numbers to be effective. In fact, many of the most widely used techniques rely on deterministic pseudorandom sequences. This intrinsic characteristic is closely tied to the pseudorandom nature of computers, which are the calculators Monte Carlo codes work on. When referring to a pseudorandom sequence of numbers, a specific set of operations is performed by the computer:

1. Sampling a random integer value between 0 and $2^n - 1$, where n is the number of bits used by the computer to represent numbers (typically 32 or 64);
2. Normalization by dividing the sampled value by 2^n to obtain a number $N_1 \in [0, 1)$;
3. Generating a sequence of N_{tot} numbers within the interval $[0, 1)$ by applying the recurrence relation:

$$N_i = N_{i-1} \times K_1 + K_2, \tag{4.1}$$

[2]The term was popularized by scientists working on the Manhattan Project, notably Stanislaw Ulam and John von Neumann.

[3]Interestingly, the name "Monte Carlo" comes from the Monte Carlo Casino, which was the most famous gambling venue at the time, as Las Vegas had not yet become the established global capital of gambling. Had the situation been different, we might today be referring to it as the "Las Vegas method" instead. Other rumors say that an uncle of Stanislaw Ulam was a gambling addict who frequented the casino. Because Ulams's team was working the Manhattan project, the method needed a secret code name and Monte Carlo seemed perfectly fitting.

where $K_1, K_2 \in [0, 2^n - 1]$ are constants. These constants are chosen to ensure that the resulting values remain within the valid range $[0, 2^n - 1]$.

With this strategy, the entire sequence of numbers is determined once the initial value is selected. This initial value is referred to as the *seed*. This property ensures that a Monte Carlo simulation produces the same results if started from the same seed; therefore, to obtain different outcomes, the seed must be changed.

4.1.2. Sampling procedures

Since a Monte Carlo code relies on a stochastic approach, a method to extract meaningful information from random variables must be developed. These variables are first described by probability distributions, typically expressed as $dP = f(x)\,dx$, where $f(x)$ is a probability density function (PDF) that is always non-negative. This implies that the probability of the variable x falling within a specific interval, say $x \in (a, b)$, is given by

$$P(a < x < b) = \int_a^b f(x)\,dx. \tag{4.2}$$

Moreover, due to the normalization condition of a PDF, the total probability over the entire domain must be equal to one:

$$\int_{-\infty}^{+\infty} f(x)\,dx = 1. \tag{4.3}$$

To facilitate sampling, a cumulative distribution function (CDF) is defined as

$$F(x) = P(x' < x) = \int_{-\infty}^{x} f(x')\,dx'. \tag{4.4}$$

This function represents the cumulative probability of obtaining a value less than or equal to x. If a uniform probability distribution is defined over an interval (a, b),

$$f(x) = \begin{cases} \frac{1}{b-a} & \text{if } a < x < b \\ 0 & \text{elsewhere,} \end{cases} \tag{4.5}$$

the corresponding CDF becomes

$$F(x) = \begin{cases} 0 & x < a \\ \frac{x-a}{b-a} & \text{if } a \leq x \leq b \\ 1 & \text{elsewhere.} \end{cases} \tag{4.6}$$

Using the so-called *inverse method*, a pseudorandom number $\xi \in [0, 1)$ is generated. Since the CDF always ranges between 0 and 1 ($0 \leq F(x) \leq 1$), if the CDF is invertible, the corresponding value of x can be obtained by

$$x = F^{-1}(\xi). \tag{4.7}$$

Basically, sampling a uniform CDF generates values of x that follow the desired probability distribution, starting from uniformly distributed pseudorandom numbers ξ.

If a PDF relevant to the neutronics domain is considered,

$$f(x) = \begin{cases} \Sigma\, e^{-\Sigma x} & \text{if } a < x < b \\ 0 & \text{elsewhere,} \end{cases} \tag{4.8}$$

the corresponding CDF can be shown to be

$$F(x) = 1 - e^{-\Sigma x}. \tag{4.9}$$

This leads to the inverse relation

$$x = -\frac{1}{\Sigma} \ln \xi. \tag{4.10}$$

This means that by sampling a uniform random variable ξ, one can reconstruct values of x that follow the exponential distribution described by the PDF. This technique is particularly useful for sampling the mean free path $\bar{l}$ of a neutron. Knowing that $e^{-\Sigma_t x}$ represents the probability of not having an interaction occurring up to distance x, the mean free path is given by

$$\bar{l} = \int_0^{+\infty} x\, e^{-\Sigma_t x}\, dx = \frac{1}{\Sigma_t}. \tag{4.11}$$

Therefore, if the values of x are sampled a *sufficient* number of times, their average converges to $\frac{1}{\Sigma_t}$. Where *sufficient* stands for the fact that the number of samplings is chosen in order to achieve a certain precision on the variable sampled.

4.1.3. Rejection method

In order to sample random numbers from an arbitrary probability density function $f(x)$ whose support lies in the interval $[a, b]$, an ad hoc method must be developed. The one reported in the following is called *rejection method*.

First, it is necessary to define

$$H \geq \max_{x \in \mathbb{R}} f(x), \tag{4.12}$$

and a function $g(x)$ such that

$$g(x) = \begin{cases} H & x \in [a, b] \\ 0 & \text{elsewhere,} \end{cases} \tag{4.13}$$

as illustrated in Figure 4.1.

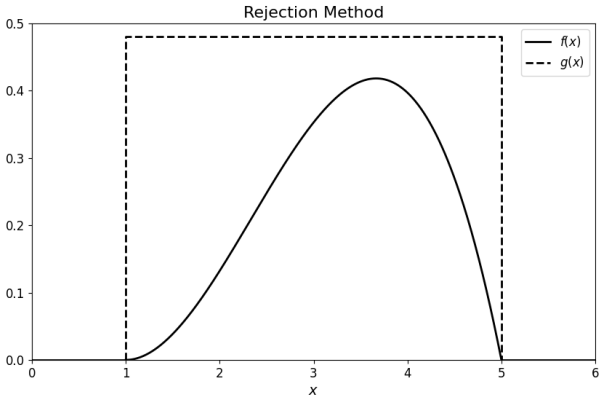


Figure 4.1: Example of the rejection method, where $[a, b] = [3, 5]$.

Then, by sampling from the uniform distribution over $x \in [a, b]$, a candidate sample can be generated as

$$x' = R \times (b - a) + a. \tag{4.14}$$

To determine whether this candidate should be accepted, an acceptance probability is defined as

$$P = \frac{f(x')}{H}, \tag{4.15}$$

where $f(x')$ is the target PDF. If the sampled value is greater than P, the candidate is rejected, otherwise, it is accepted.

To maximize efficiency, the optimal choice for H is

$$H = \max_{x \in [a,b]} f(x). \tag{4.16}$$

However, if $f(x)$ is highly peaked or asymptotic, the rejection area becomes significantly larger than the acceptance one, which reduces the efficiency of the method.

The efficiency η of the rejection sampling method can be defined as the ratio between the area under the target PDF and the area under the envelope function $g(x)$:

$$\eta = \frac{\int_a^b f(x)\,dx}{\int_a^b g(x)\,dx} = \frac{1}{(b-a)\cdot H}. \tag{4.17}$$

Clearly, if $(b-a) \to \infty$ or $H \to \infty$, then $\eta \to 0$, making the method increasingly inefficient.

4.2. Monte Carlo methods in neutronics

Neutron life is simulated from their birth (fission or external source emission) to death (absorption or leakage) using probability density functions and getting statistical estimations of quantities of interest, always together with their uncertainty due to the fact the method is stochastic.

To make the simplest example, let's focus on the k_{eff} analysis of a system. The life of a certain quantity of neutrons, let's call it n, is simulated multiple times, let's assume N (called batches or generations). For each $j = 1, ..., N$ batch, the k_{eff} can be evaluated in different ways, the most logical one is to apply the physical definition of k_{eff}. Thus,

$$k_{eff} = \frac{\text{Number of neutrons in the next generation}}{\text{Number of neutrons in the current generation}}.$$

This is called the *analog* technique and it can be referred to as ANA_KEFF.

At the end of n neutron life simulation, the k_{eff} is stored and, once the N simulations end, the final result is the average k_{eff}. The scheme of the working principle of a Monte Carlo code applied to neutronics is represented in Figure 4.2.

Since the k_{eff} of each simulations inherits statistical uncertainty due to the stochastic nature of Monte Carlo codes and due to the intrinsic physical uncertainties of experimental quantities such as cross sections, the final average k_{eff} has an uncertainty according with the well-known relation:

$$\sigma(\overline{k_{eff}}) = \frac{\sigma_{batch}}{\sqrt{N}} \approx \frac{1}{\sqrt{N_{tot}}}. \tag{4.18}$$

Where $N_{tot} = n \times N$. This relation holds because the uncertainty (standard deviation of the mean) follows the Central Limit Theorem (CLT). Since each individual k_{eff} is based on n histories, the variance of a single batch is proportional to $1/n$. When averaging over N batches, the total uncertainty scales as Equation (4.18).

It has to be said that Equation (4.18) holds provided that the N k_{eff} results are not correlated. Unfortunately, this is not exactly true because each batch is influenced by the

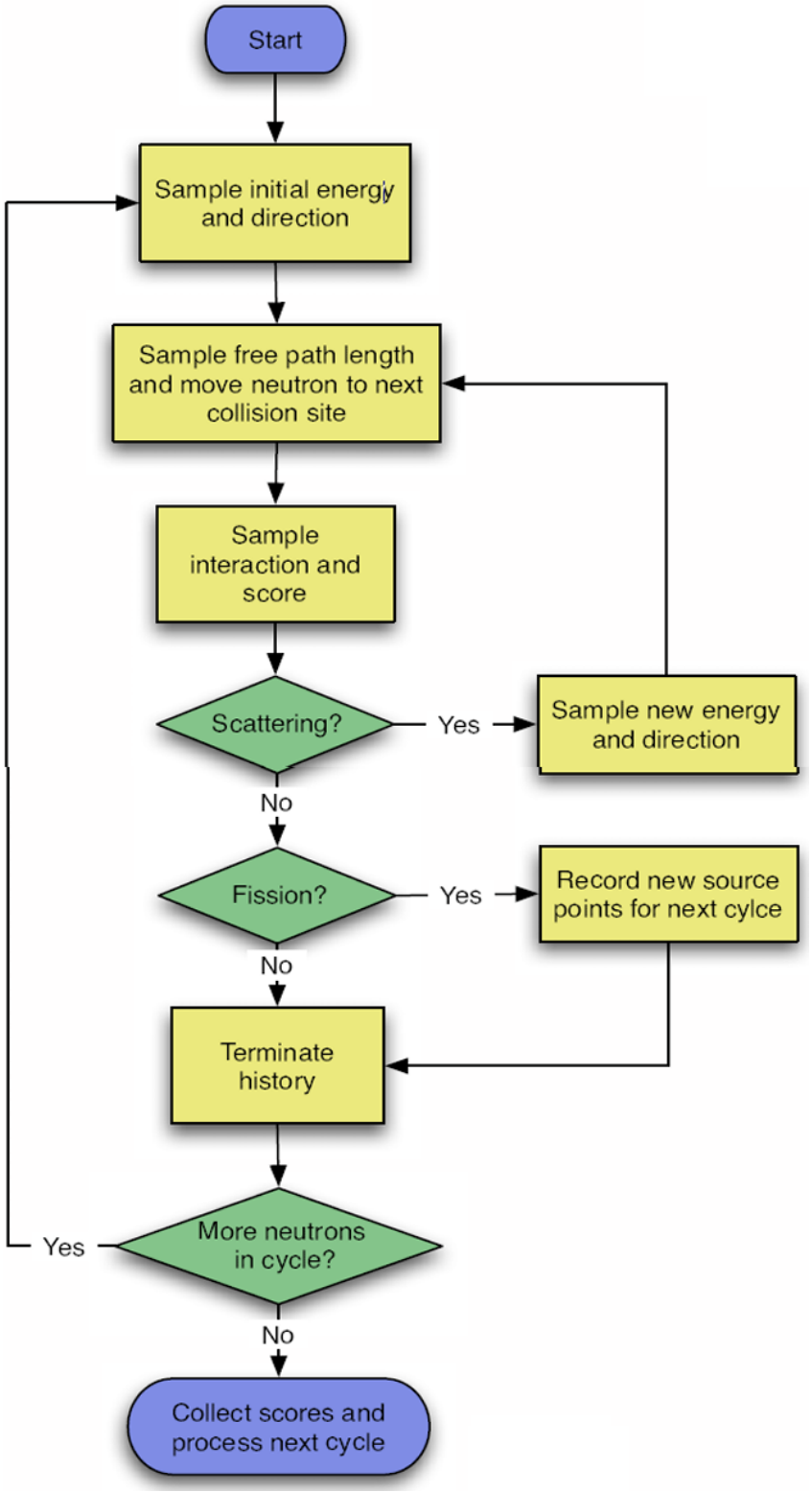


Figure 4.2: Monte Carlo working scheme in neutronics (from [28]).

configuration of the previous one (since the next batch uses as neutron source the fission sites retrieved in the current one) and the resultant uncertainty is usually higher than the one reported in Equation (4.18).

According to Equation (4.18), the inherent scaling of Monte Carlo methods implies that to reduce the uncertainty, for instance, by a factor of 2, the total number of histories (N) must be increased four times.

Basically, this means that in Monte Carlo codes:

$$\text{Precision} \times N \Rightarrow \text{Computational cost} \times N^2.$$

For example, if a simulation runs with 50 000 neutrons for 1000 batches, this means:

$$\sigma(\text{ANA\_KEFF}) \propto \frac{1}{\sqrt{5 \times 10^7}} \approx 14.14\,\text{pcm}. \tag{4.19}$$

The overline on $\sigma(\overline{\text{ANA\_KEFF}})$ that stands for its mean value is omitted to have a lighter notation and this choice will be applied to the remainder of this work.

In the next chapters several uncertainties are reported with a lower value than the one obtained according to Equation (4.18) and Equation (4.19). This occurs because Equation (4.18) and Equation (4.19) are just an estimate: the proportionality constant is unknown. The uncertainties found are coherent as they do not differ from their estimations by order of magnitudes.

4.2.1. Tracking techniques

By definition, the probability of interaction per unit path length is governed by the macroscopic total cross section Σ_t, typically expressed in units of inverse centimeters. Assuming the particle moves through an infinite, homogeneous medium with a constant total cross section, the free path length between interactions follows an exponential distribution. This distribution can be sampled using the inverse method, yielding the distance to the next collision site as

$$l = -\frac{1}{\Sigma_t} \log \xi, \tag{4.20}$$

which is expressed by Equation (4.10); however, Equation (4.20) is valid only under the assumption of a homogeneous medium. If the particle crosses different materials, this assumption no longer holds, and the sampling procedure must be adjusted accordingly. There are two major techniques commonly adopted to address this issue.

The first technique is known as *surface-tracking*. In traditional algorithms, the particle's trajectory is interrupted at each material boundary. Since the probability of interaction is independent of the particle's history, any point along the trajectory can be treated as the starting point of a new path. When a sampled path intersects a boundary, the track is truncated at the point of intersection, and a new path length is sampled using the cross section of the new material. To implement this method, the algorithm must calculate the distance to the nearest surface boundary which is typically done by looping through all candidate surfaces and selecting the shortest distance. While surface-tracking is widely adopted and considered the standard in Monte Carlo transport codes, it has limitations in terms of computational efficiency, especially in geometries with a large number of surfaces or when the particle's mean free path is long relative to the dimensions of the geometry.

The other technique is called *delta-tracking* which was originally introduced in [29] and avoids the need to calculate surface distances by using rejection sampling (explained in Section 4.1.3) thanks to the key concept of virtual collision. Basically, by defining the *majorant* cross section as the maximum of all material total cross sections at a given energy:

$$\Sigma_m(E) = \max[\Sigma(\vec{r}, E)], \tag{4.21}$$

the path length is retrieved by

$$l = -\frac{1}{\Sigma_m} \log \xi, \tag{4.22}$$

and at the end of each sampled path, the algorithm performs a rejection sampling where the probability of accepting the collision is given by

$$P = \frac{\Sigma(\vec{r}, E)}{\Sigma_m(E)}. \tag{4.23}$$

If the collision is rejected, thus resulting in the so-called virtual collision, a new path length is sampled using Σ_m, and the particle is moved to the next tentative collision site. Since Σ_m is always greater than or equal to the current cross section Σ, the average path lengths in delta-tracking are shorter than those in surface-tracking; however, the physical mean free path is preserved because some paths span multiple virtual collisions, and interaction probabilities are conserved.

The main advantage of this technique is that it eliminates the need to compute surface distances from each collision site and stop particles at material boundaries. This significantly improves computational performance in geometries where the mean free path is long compared to the size of the geometry. Moreover, because the majorant cross section

is independent on position, delta-tracking can be effectively applied to inhomogeneous materials.

However, delta-tracking also has limitations. If the geometry contains localized regions with very high absorption cross sections such as control rods or burnable absorber pins, these regions dominate the majorant cross section. Since they occupy only a small volume, the rejection sampling becomes inefficient, especially in regions where particles spend most of their time, such as the moderator or coolant. In Figure 4.3, the left panel shows how the majorant cross section is dominated by the high capture cross sections of ^{155}Gd and ^{157}Gd. The right panel illustrates the resulting high rejection probability in regions like the coolant and moderator, where neutrons spend most of their lifetime. This leads to a significant drop in sampling efficiency with a high rate of rejection.

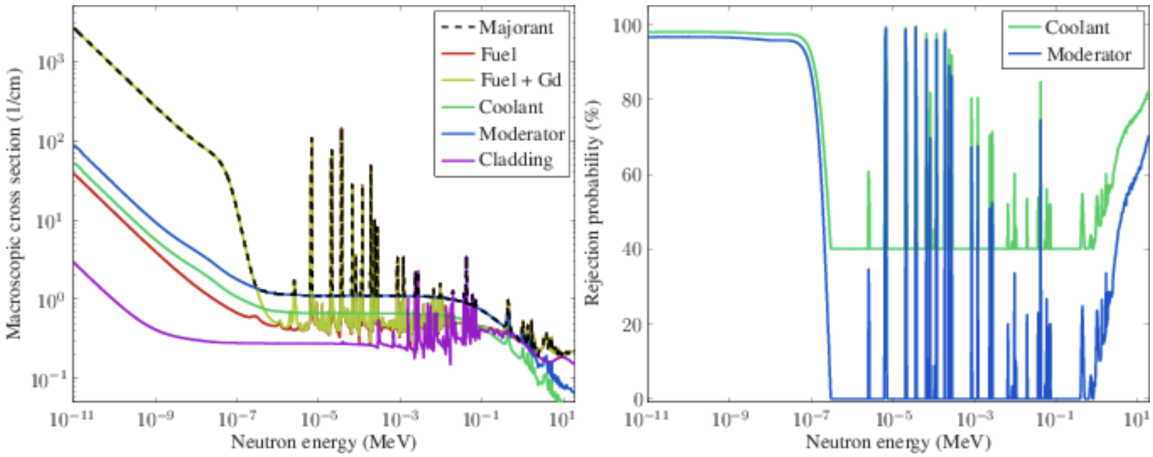


Figure 4.3: Delta tracking drawbacks with the presence of highly absorbing materials (from Serpent Wiki).

4.3. Serpent, a Monte Carlo reactor physics code

Serpent [8] is a continuous-energy Monte Carlo reactor physics code developed since 2004 by Jaakko Leppänen and collaborators at the VTT Technical Research Center of Finland. It is freely available for academic and non-commercial use, while commercial applications require a dedicated license. Unlike TRIPOLI-4®, which is proprietary and restricted to EDF and its partners, Serpent is openly distributed and widely adopted. It supports a broad range of applications, including group constant generation, coupled multi-physics simulations, fusion neutronics, and radiation shielding. In addition to neutron transport and depletion calculations, Serpent also supports photon transport.

In the context of this master's thesis, Serpent has been employed to further benchmark APOLLO3® results, rather than relying solely on TRIPOLI-4®. This choice is motivated

by Serpent's high computational efficiency in depletion simulations. In terms of execution time, for the present work, a depletion history up to 60 GWd/tHM is completed by Serpent in less than three days, whereas TRIPOLI-4® would require months to reach an equivalent level of statistical accuracy.

Furthermore, Serpent eliminates the need for TriAGe that is required when using TRIPOLI-4®. TriAGe is a function of APOLLO3® that translates all APOLLO3® data into TRIPOLI-4® ones (e.g., geometry, material composition and temperatures, reaction rates to be scored). Moreover, in order to correctly compare the two codes during depletion one must take into consideration that Monte Carlo methods adopt a stepwise approximation for cross sections, while APOLLO3® not. This means that the axial mesh adopted for the deterministic calculation to exchange with depletion and self-shielding is certainly too coarse. Supplementary axial layers are needed by TRIPOLI-4® in order to be more accurate and have a fair comparison with APOLLO3®. This mesh is referred as the Step-Equivalent Mesh (SEM), or stepmesh, as in the remainder of this work. The procedure to generate this mesh is all but straightforward and explained in detail in [6].

Beyond practical feasibility, the reliance on the TriAGe workflow is methodologically problematic regarding the principle of code benchmark. The TriAGe function operates by translating the isotopic compositions calculated by APOLLO3® directly into TRIPOLI-4® inputs. This coupling introduces a potential flaw in the process: the data generated by the code under test (APOLLO3®) is used to feed the reference code (TRIPOLI-4®) intended to benchmark it.

Such a procedure is inherently susceptible to *false positives*. If the depletion physics or isotopic evolution in APOLLO3® were wrong, TRIPOLI-4® would merely confirm the consistency of a wrong result rather than identifying the error. This technique would exclusively benchmark the TDT MOC-3D flux solver (not the depletion one) because the Monte Carlo counterpart would start from the same inputs in terms of isotopic compositions. Since a rigorous benchmark framework requires that the controller remains entirely independent from the controlled system, the results obtained via standalone Serpent simulations provide a more robust, thorough, and scientifically reliable benchmark, providing a full benchmark (flux solver and depletion solver). Therefore, Serpent depletion results are considered more reliable than those obtained with TRIPOLI-4®.

4.3.1. Serpent options

Serpent offers the possibility to adjust certain options to get a trade-off between computational cost and precision. The input syntax manual of the open-source Serpent Wiki[4] deeply describes each of the options. Here are reported the main choices adopted in this master's thesis:

Neutron population: for what concerns the neutron population and its distribution in batches, literature [30] shows that performing a simulation with few or many neutrons per batch does not affect convergence. This means that if convergence is reached after 100 batches with 100 000 neutrons per batch, simulating the same calculation with 100 neutrons per batch sees that same convergence trend at 100 batches. So, choosing whether to increase or diminish the total number of batches is just a matter of how precise is the final results in terms of uncertainty. What is important is not to use too few neutrons per batch, [30] shows that with less than 10 000, a systematic underestimation of the k_{eff} occurs, while using whether 10 000 or 100 000 neutrons per batch leads to the same result. It should be noted that [30] refers specifically to TRIPOLI-4® and to simulations aimed at achieving criticality, but this reasoning can be extended also to other codes, such as Serpent. In contrast, the depletion calculations presented in this master's thesis involve also subcritical systems. For this reason, using a sufficiently *large* number of neutrons per batch is recommended to ensure adequate sampling of the system's geometry throughout the entire simulation, especially as neutron population decreases with increasing burnup. For this reason, each neutrons per batch and number of batches choices will be adequately justified.

Burnup calculation mode: `set bumode 2`, which activates the Chebyshev Rational Approximation Method (CRAM) (see Section 2.2.3). This method has been demonstrated to outperform the Transmutation Trajectory Analysis (TTA), the alternative option available in Serpent, as shown in [13]. By default, selecting option 2 applies CRAM with an order of 14, which is precisely the configuration tested in [13] and found to yield superior results compared to TTA. For further details, refer to the set bumode *card*[5] section in the Serpent Wiki.

Unresolved resonance probability table sampling: `set ures`. In nuclear simulations, unresolved resonance probability table sampling is a statistical method used to model neutron interactions in the unresolved resonance region which is an energy range where

[4] At the time of writing, the Serpent Wiki is undergoing a transition to a new online documentation platform. Consequently, should any links in this work fail to redirect to the intended webpage, the reader is encouraged to consult the updated Serpent documentation directly at serpent.vtt.fi/docs.

[5] An input card defines a single parameter, option, or specific data entry related to, for example, geometry or material definition in Serpent.

individual nuclear resonances are too closely spaced to be experimentally distinguished. Instead of using averaged cross sections, it is chosen to rely on probability tables that represent the statistical distribution of possible cross section values for reactions such as capture, fission, and scattering. During a Monte Carlo simulation, these tables are used to randomly sample cross sections based on the neutron energy and material properties, allowing for a more realistic representation of self-shielding and resonance interference effects.

This option can be set on or off. Although the Serpent Wiki notes that: *"The effect is usually not significant in thermal systems, but can lead to noticeable discrepancies in fast-spectrum systems,"* it was decided to always enable this feature to ensure the most physically accurate results; however, enabling this option was observed to have a slight negative impact computational performance. This is likely due to the fact that unresolved resonance probability table sampling is performed *on-the-fly*, preventing Serpent from precomputing and storing the relevant data. Anyway, the computational cost increase showed to be manageable.

Temperature treatment: the `tmp` card was used instead of the `tms` card. The system temperature, including fuel, coolant, and cladding, was set isothermally to 624 K, which corresponds to one of the temperatures for which the thermal scattering law $S(\alpha, \beta)$ of light water is available in the JEFF-3.1.1 [31] nuclear data library. This library is also used by TRIPOLI-4®, while APOLLO3® relies on the CEAV-514 library, an improved version of CEAV-512 (that is derived from the JEFF-3.1.1), with enhanced self-shielding treatment capabilities. The choice of 624 K was made to prioritize interpolation of nuclide cross sections over interpolation of $S(\alpha, \beta)_{H_2O}$ data. JEFF-3.1.1 provides both nuclide cross sections and $S(\alpha, \beta)_{H_2O}$ data at the temperatures listed in Table 4.1.

Table 4.1: Available temperatures for nuclide cross sections and $S(\alpha, \beta)_{\mathrm{H_2O}}$ thermal scattering law data in JEFF-3.1.1.

Data Type	**Temperatures [K]**
Cross Sections	300, 600, 900, 1200, 1500, 1800
$S(\alpha, \beta)_{\mathrm{H_2O}}$	293.6, 323.6, 373.6, 423.6, 473.6, 523.6, 573.6, 623.6, 647.2, 800, 1000

To evaluate cross sections at the desired temperature, Serpent offers two approaches: the `tmp` and `tms` cards. The `tmp` card utilizes a Doppler-broadening pre-processing routine, while the `tms` card applies the Target Motion Sampling (TMS) method during runtime. The Doppler-broadening routine [32] analytically increases the temperature of free-atom

cross sections below the unresolved resonance energy range, making it well-suited for thermal systems like the ones that are analyzed in this thesis. This method is based on Solbrig's kernel [33] and is computationally efficient. Since it is a pre-processing step, cross sections are calculated once at the beginning and stored, avoiding repeated evaluations.

In contrast, the TMS method [34] samples the energy and motion direction of the target nuclide from a Maxwellian distribution at each neutron collision, thus being more physically compliant. A coordinate transformation to the target-at-rest frame is then used to apply the appropriate temperature-dependent cross section. For thermal scattering data, temperature treatment relies on interpolation between fixed temperature points [35], similar to the interpolation used in unresolved resonance sampling.

Due to its on-the-fly nature, TMS significantly increases computational cost. Simulations were conducted to assess the impact of using the `tms` card compared to `tmp`, with results presented in Appendix A. The findings confirmed that `tms` leads to a substantial increase in runtime with a slight change in k_{eff}, justifying the decision to use `tmp`.

All simulations in this thesis were initialized with nuclides at 600 K, the closest available temperature to the target 624 K in JEFF-3.1.1, according to the Serpent User's Manual: *"Experience has shown that the errors remain small, at least if the nearest library temperature is used as the basis for broadening."*

Predictor-corrector method: `set pcc 3 3 5`. This command configures the predictor-corrector scheme used at each burnup mesh point. The selected method is linear extrapolation and linear interpolation (LELI), which has been shown to outperform both constant extrapolation with linear interpolation (CELI) and linear extrapolation with quadratic interpolation (LEQI), as demonstrated in [19] and explained in Section 2.2.4. It is important to note that when linear extrapolation (LE) is selected, Serpent uses constant extrapolation for the first time step.

The final two numbers, 3 5, specify the number of substeps for the predictor and corrector phases, respectively. This configuration offers an optimal balance between accuracy and computational cost, as each additional substep introduces more operations, as discussed in [17] and explained in Section 2.2.4

Optimization mode: `set opti 4`. This command sets the optimization mode, which directly affects both computational performance and memory usage. Mode 4 is the default setting in Serpent and is designed to maximize performance, albeit at the cost of increased memory consumption [36]. According to the Serpent Wiki, this mode is: *"Suitable for group constant generation and 2D assembly burnup calculations with a limited number of depletion zones;"* therefore, before opting for this maximum optimization level, it is important to carefully evaluate the number of independent burnup regions defined in the simulation. This is essential because mode 4 activates advanced features such as the

delta-tracking variance reduction technique, which can lead to more accurate results.

Group constant generation: `set gcu −1`. This setting disables the generation of group constants, as they were not among the quantities evaluated in the benchmark between APOLLO3® and Sepent. By turning off this feature, a reduction in computational cost was observed.

Library choice: since the goal of this work is to benchmark APOLLO3® results, it is essential to use a Serpent library that closely matches the one used by APOLLO3® (CEAV-514). The default Serpent VTT JEFF-3.1.1 library includes JEFF-3.1.1 data [31] supplemented with 52 additional nuclides sourced from other libraries, as detailed in Table 4.2.

To ensure maximum compatibility with the CEAV-514 library, all nuclides listed in Table 4.2 were excluded from the Serpent simulations, except for ^{253}Cf from the ENDF/B-VII library, which is also present in CEAV-514; nevertheless, simulations were initially tested using the default VTT JEFF-3.1.1 library, and no significant discrepancies were observed in the results. The absence of notable differences can likely be attributed to the nature of the excluded nuclides: most of the additional ("foreign") nuclides in the VTT library are either in natural isotopic composition or have limited relevance; therefore, their impact on the simulation outcomes is minimal.

Table 4.2: List of nuclides present in VTT JEFF-3.1.1 not originally in JEFF-3.1.1.

ENDF/B-VII.0 (USA, 2006)	^{7}Be, ^{71}Ga, ^{74}As, ^{113}Sn, ^{123}Xe, ^{133}Ba, ^{136}Ce, ^{138}Ce, ^{139}Ce, ^{153}Gd, ^{156}Dy, ^{158}Dy, ^{166m}Ho, ^{239}U, ^{240}U, ^{241}U, ^{253}Cf
ENDF/B-VI.8 (USA, 2001)	Mg-nat, S-nat, Cl-nat, K-nat, Ca-nat, Ti-nat, Zr-nat, Mo-nat, Cd-nat, In-nat, Hf-nat, W-nat
JENDL-3.2 (Japan, 1994)	^{12}C, Si-nat, ^{51}V, Cr-nat, Fe-nat, Ni-nat, Cu-nat, Ge-nat, Ag-nat, Sb-nat, Eu-nat, Pb-nat
JENDL-3.3 (Japan, 2002)	^{69}Ga
JEFF-2.2 (Europe, 1992)	^{64}Zn
JEFF-3.2 (Europe, 2014)	^{36}Cl
BROND-2.2 (Russia, 1992)	He-nat, Sm-nat, Gd-nat, Re-nat, Ir-nat
CENDL-2.1 (China, 1995)	Sn-nat, Lu-nat, Hg-nat

4.4. TRIPOLI-4®

TRIPOLI-4® is the fourth generation of the continuous-energy radiation transport Monte Carlo code developed by the Service d'Études des Réacteurs et de Mathématiques Appliquées (SERMA) at CEA Saclay [7]. It is designed for a wide range of applications including shielding analysis, reactor physics with depletion, criticality safety, and nuclear instrumentation, for both fission and fusion systems.

As mentioned at the beginning of this chapter, TRIPOLI-4® is not optimized for depletion simulations: a full 3D fuel assembly depletion calculation may require several months of computational time. For this reason, Serpent was chosen in this master's thesis to perform depletion calculations; however, TRIPOLI-4® was still used in this work for a zero-burnup analysis, exploring different configurations to benchmark and cross-check Serpent results at zero burnup. During these comparative simulations, a notable mismatch at burnup zero was observed between the two codes. Several tests were conducted to identify which TRIPOLI-4® parameters could yield results closer to Serpent. The analysis is discussed in Appendix A.

Two geometry representations were used in TRIPOLI-4® for these simulations: the *surfacique* geometry and the ROOT geometry developed at CERN. The goal was to verify whether the results remained consistent across different input formats. If so, the mismatch with Serpent could not be attributed to possible input file errors.

The *surfacique geometry* is TRIPOLI-4®'s native format. It describes volumes using bounding surfaces, such as planes, cylinders, and spheres, combined logically to define regions. Particle tracking is then based on surface intersection calculations. This method is efficient for simple or moderately complex geometries and supports combinatorial modeling using predefined shapes and Boolean operators. The *ROOT geometry*, in contrast, is a hierarchical, object-oriented framework developed at CERN and widely used in high-energy physics [37]. It allows the construction of complex geometries through nested volumes and placements, managed via the `TGeoManager` class. TRIPOLI-4® can directly import ROOT files, enabling reuse of geometries from other frameworks like Geant4. Internally, the code translates the ROOT structure into its own tracking system, allowing simulations without rewriting the geometry in surfacique format.

The other TRIPOLI-4® options addressed were:

- *Sampling of the Velocity of the Target nucleus (SVT):* In Monte Carlo neutron transport simulations, this method is used to account for the thermal motion of nuclei during neutron scattering events. This effect becomes particularly significant in the thermal and epithermal energy ranges, where the relative motion between

the neutron and the target nucleus strongly influences scattering kinematics and cross sections. At low neutron energies, the common assumption that nuclei are stationary (as they were at 0 K) is no longer valid. SVT addresses this by sampling the target nucleus's velocity from a Maxwell-Boltzmann distribution, allowing for a more accurate calculation of the relative velocity between the neutron and the nucleus. This approach is essential for correctly modeling Doppler broadening, thermalization, and resonance scattering phenomena. It has to be said that Serpent does not have this feature;

- *Unresolved probability table sampling:* It is the analogous to `set ures` option explained for Serpent.

5 | UOX and UOX-Gd analyzed assemblies

5.1. Analyzed 3D geometry: UOX assembly

The first analyzed assembly is the CamiVVER one [6]: the analyzed geometry is the 3D UOX fuel assembly whose X-Y layout is reported in Figure 5.1.

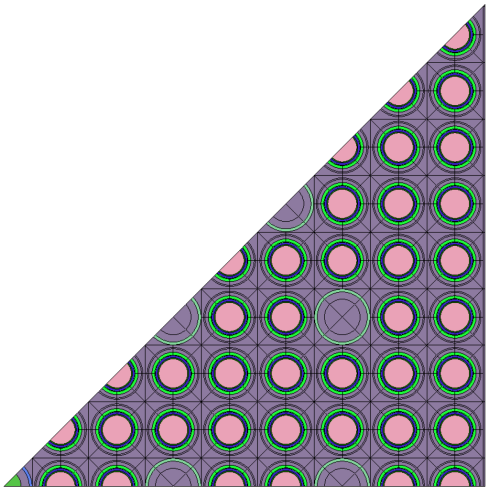


Figure 5.1: X-Y layout of the 3D assembly configuration (from [6]).

In the bottom left corner there is the instrumentation channel with the fission chamber (green) and in blue the guide tube that surrounds it. Since the fission chamber channel is usually at the center of a fuel assembly in a PWR, it is clear that the figure represents one-eighth of the X-Y cross section of the assembly. The simulated geometry is overall one-sixteenth of the original since an axial cut at the midplane ($z = 0$) is performed. The cells with only the green material are the guide tubes while the others are the fuel rod cells. It has to be noted that in CamiVVER simulation, the fuel cells were subdivided in four radial subregions[1] and in eight equispaced angle regions from the cladding and gap crown in light green: the cladding and the gap are modeled together since the gap is about

[1] The colors are not completely visible but in order from the inner to the outer ones are: pink, green, dark blue and dark green, which is almost visible.

500 µm, thus a negligible physical dimension with respect to the fuel assembly (1.26 cm just for the cell pitch), while the guide tube cells are subdivided in four equispaced angle regions. Furthermore, on the right of the Figure 5.1, there is the water blade that divides the fuel assemblies from each other inside a nuclear reactor core. On the top of that water blade there is a little triangle; it represents the lower extremity of the water blade belonging to the vertically adjacent fuel assembly. From now on, it will be addressed as water corner.

Figure 5.2 represents the configuration that has been analyzed with Serpent, where 1/4 radially is shown. Thanks to the `set usym` Serpent card, the reflection with respect to the bisector of the first and third quadrants (clearly visible as a light black line in figure) does not duplicate the burnup zones. Instead, it treats the reflected zones as part of the original ones, thus resulting in better statistics. This strategy is adopted because, to the author's knowledge, Serpent does not allow to successfully model a triangular geometry such as the one simulated with APOLLO3®. Furthermore, the fuel assembly grid is modeled with the orange circles that can be observed around each fuel rod.

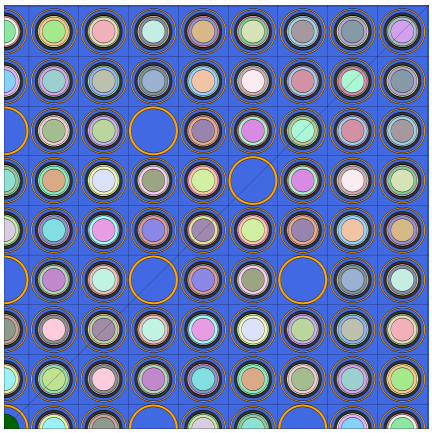


Figure 5.2: 2D geometry of the analyzed assembly. This is a layer with grids.

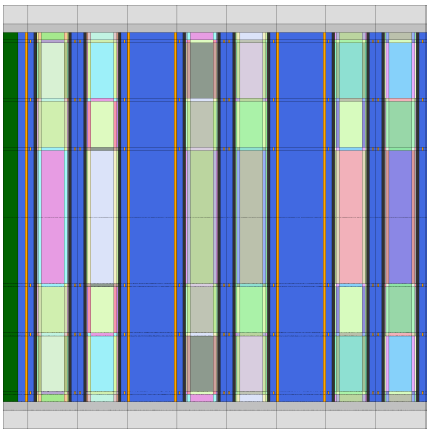


Figure 5.3: YZ cut of the modeled fuel assembly; the green rod corresponds to the instrumentation channel.

In Figure 5.3, a YZ cut of the material geometry is shown. One can observe the clear symmetry line at $z = 0$ (which is at the midplane of the image), thanks to the fact that Serpent represents the same depletion region with the same color. The two gray layers are the reflector ones and have a different isotopic composition.

An axial representation of a fuel rod is in Figure 5.4, where within each layer there is written its thickness and each grid layer, in dark gray, is 3.3 cm thick. In total, considering also the two reflector layers with different material composition, the assembly half-height

is 225 cm.

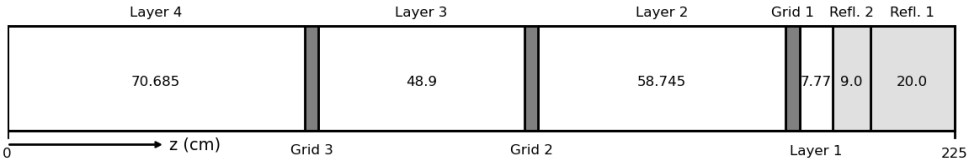


Figure 5.4: Fuel rod axial dimensions: UOX assembly.

To actually simulate a full assembly, reflective boundary conditions were applied radially and at the axial midplane, while black ones were imposed at the top of the outer reflector. This task is relatively easy with APOLLO3® while it led to the development of an ad hoc method with Serpent, as addressed in Appendix C.

The UOX assembly analyzed in this master's thesis its characteristic properties reported in Table A.6 and the material ones in Section A.3. It has to be noted that the sum of the atomic densities of each isotope results in the material one itself. When H_2O is present, the atomic density to take into account is 3 times that value: once for the oxygen and twice for the hydrogen.

5.2. Analyzed 3D geometry: UOX-Gd assembly

As already explained in Chapter 1, other than just continuing the CamiVVER project work, it was chosen to analyze also an assembly with rods with gadolinium which is a strong neutron absorber. A positive benchmark result would lead to a further confirmation of the robustness of APOLLO3® in its presence. In particular, it was chosen to study two configurations of the PRATIC core, which is a soluble-boron-free PWR, SMR developed at CEA: one assembly with 36 UOX-GD rods and the other one with 8. Respectively, with a 1/8 radial cut, the rods are 7 and 1[2]. Figure 5.6 shows their X-Y cuts where the black squares are guide tube cells, green ones are UOX-GD rods, while the blue one is the instrumentation channel.

The objective of this study is to evaluate the performance of APOLLO3® in relation to the gadolinium concentration within the core. It is well-established that the 3D Method of Characteristics (MOC), in the absence of the Linear Surface (LS) approximation, struggles to accurately model the presence of gadolinium [38, 39]. This limitation arises from the exceptionally high capture cross section of gadolinium isotopes, which induces significant localized gradients in the neutron flux. Indeed, in the proximity of UOX-Gd rods

[2]It has to be noted that the result is not simply a division by 8 of the total number since the rods can lie in the cutting plane, so just a half of them is modeled and counted anyway as +1.

it cannot be adequately captured by a standard SC approximation. In contrast, the LS approximation allows the neutron source term to vary linearly along each characteristic segment, providing the mathematical fidelity necessary to resolve these sharp spatial variations. For these reasons, the LS approximation is also essential for the accurate modeling of control rod insertion; like gadolinium, control rods act as strong local absorbers that generate high flux gradients, which the simpler SC approach fails to resolve correctly. That is why the all-rods-out configuration was analyzed. Nonetheless, it was decided to assess the current capability of APOLLO3® to model gadolinium presence without the LS method, as this feature remains unavailable at the time of writing.

The 36 UOX-Gd fuel assembly material compositions is reported in Table A.9. Both asembly configuration characteristics are reported in Table A.7. It has to be noted that the instrumentation channel is empty, without fission chamber, and in these configurations the grids are not modeled, thus the fuel rods have the axial configuration reported in Figure 5.5. Each fuel axial layer as a thickness of 10.0281 cm and the each reflector one of 5 cm, for a total assembly height of 120.2807 cm[3].

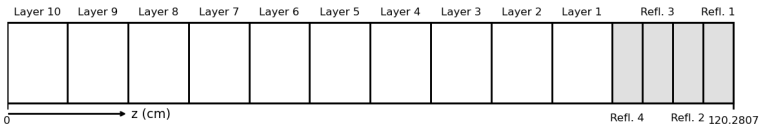
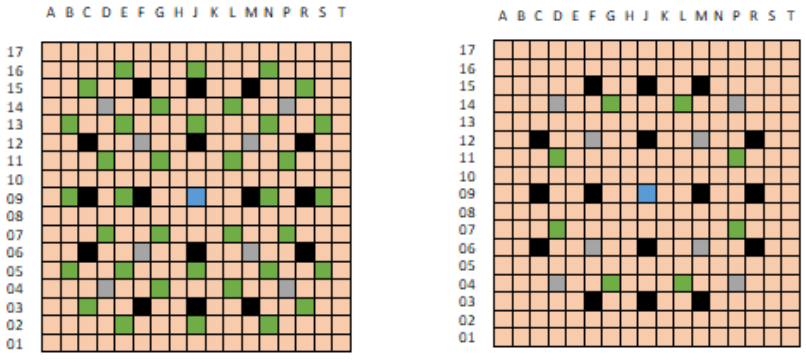


Figure 5.5: Fuel rod axial material mesh (UOX-Gd assembly).

Furthermore, with respect to Table A.9, the only modifications for the 8 UOX-GD assembly involve the uranium enrichment in the UOX rods, specifically (in units of $10^2 4/\mathrm{cm}^3$): ^{16}O (4.5377×10^{-2}), ^{234}U (4.8143×10^{-6}), ^{235}U (5.7429×10^{-4}), ^{236}U (2.6305×10^{-6}), and ^{238}U (2.2107×10^{-2}).



(a) 36 UOX-GD rods. (b) 8 UOX-GD rods.

Figure 5.6: X-Y cuts of the analyzed PRATIC UOX-Gd assemblies (from [9]).

[3]It has to be noted that 10 fuel axial layers times 10.0281 cm plus the reflector layers does not give the mentioned height. This is due to the fact that the idea was to divide the axial active region in 10 layers with a precision of 10^{-4}, so the third, sixth and ninth layers (from the midplane one) are 10.028 cm thick.

6 | Convergence analysis

As already explained in Chapter 4, a preliminary benchmark of the Serpent output was performed against TRIPOLI-4® at zero burnup to check if the results were compatible with CamiVVER ones. The outcome of this study is reported in Appendix A.

To continue with the goal of benchmarking APOLLO3® against Serpent, it is necessary to establish a reliable simulation setup: a preliminary series of tests must be conducted to understand which parameter influences convergence and how. Once a stable configuration is identified, parameters may be further refined or relaxed in precision, allowing for faster computations at the expense of reduced precision or vice versa. This trade-off between accuracy and computational cost is a fundamental aspect of numerical simulations across all disciplines, and it is particularly critical in neutronics, where depletion calculations can require several days to complete. Even a minor adjustment to a single parameter can result in substantial time savings while only marginally affecting the precision of the results.

All calculations were performed using a development version (based on the v3.1) of APOLLO3® identified by the SHA1 hash 2a6b1a70aa7388a96848c0e8b5c547c8fa3c9f01 on the official Tuleap Git repository of the code[1]. While for Serpent, the simulations were performed using version 2.2.1 (March 24, 2023). It should be noted that the Serpent version used is not the most recent one; the latest release (at the time of writing) is version 2.2.2 (June 19, 2025); however, since this master's thesis project began in March 2025, the most up-to-date version available at that time was adopted.

6.1. APOLLO3®

All calculations were performed with the SHEM281 [27] energy groups subdivision. Furthermore, the self-shielding correction was performed for each burnup step, thus the self-shielding time mesh is the same as the depletion one.

Flux calculations were performed imposing a convergence criterion of 10^{-5} on the eigenvalue and the thermal and inner iterations precision, and of 10^{-4} on the fission

[1]This version is available on Cronos at:
/software/restricted/apollo3/MAD/AP3_v3.1_20250618_2a6b1a70.

integral precision, as already pointed out in Section 3.1.3.

The strategy adopted to achieve a sufficiently convergent simulation with APOLLO3® was structured as follows:

1. Start from the radial mesh shown in Figure 5.1 and the axial material mesh defined in CamiVVER [6];
2. Verify the convergence of the TDT MOC-3D flux solver parameters in APOLLO3® at zero burnup;
3. Refine the radial mesh using the solver parameters identified in Step 2 to ensure convergence at zero burnup;
4. Test the converged mesh in depletion and compare it with Serpent.

It should be noted that the choices made in Steps 2 and 4 were also influenced by the computational cost. Excessive refinement of simulation parameters can lead to a significant increase in runtime, which may outweigh the benefits in terms of precision.

6.1.1. TDT parameters convergence

The following TDT parameters and their respective values were tested:

- `Horizontal_angles` (N_θ): [18, 20, 24, 30, 36]. It is the number of angles over π (for the 2D horizontal plane) used for the angular quadrature formulas in the tracking;
- `Vertical_angles` (N_φ): [4, 5]. It corresponds to the number of angles over the 3D vertical interval $\left(0, \frac{\pi}{2}\right)$ used for the angular quadrature formulas in the tracking. The values are not lower than 4, since calculations are performed with P_3 anisotropy;
- `MaxNberMeshesPerPeriod` (N_{mesh}): [100, 200, 300, 400, 500, 600]. It refers to the maximum number of mesh cells crossed by a trajectory over one complete tracking cycle;
- `Transversal_integration_step` (Δ_{xy}): [0.01, 0.015, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.1, 0.2]. It controls the step size used during the integration of the neutron flux along each characteristic in the transversal plane (x-y plane);
- `Transversal_Vertical_step` (Δ_z): [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7]. It specifies the transverse integration step in 3D along the z-axis.

The parameter ranges were chosen to be reasonable and centered around the CamiVVER values [6].

To determine the best configuration, the convergence of both k_{eff} and integrated reaction rates over each material axial layer were analyzed. This is because k_{eff} is an integrated quantity and may be affected by error counterbalance; therefore, convergence in k_{eff} alone is a necessary but not sufficient condition for overall convergence. When the results differ *significantly* from more refined simulations, it indicates a lack of convergence. Here, "*significantly*" means in a way that is distinguishable from the numerical uncertainty defined in Monte Carlo methods by the standard deviation and in deterministic methods by the tolerances of the various algorithms.

Computational time was also recorded, although it should be considered only indicative, as the HPC cluster nodes used did not have uniform performance and simulations were not run on nodes with the same characteristics. In particular, the cluster exploited was the EDF R&D's proprietary one, Cronos.

Table 6.1 presents all the results. Relative differences are computed using Equation (6.1), where the reference is the k_{eff} obtained with the most precise simulation in each row.

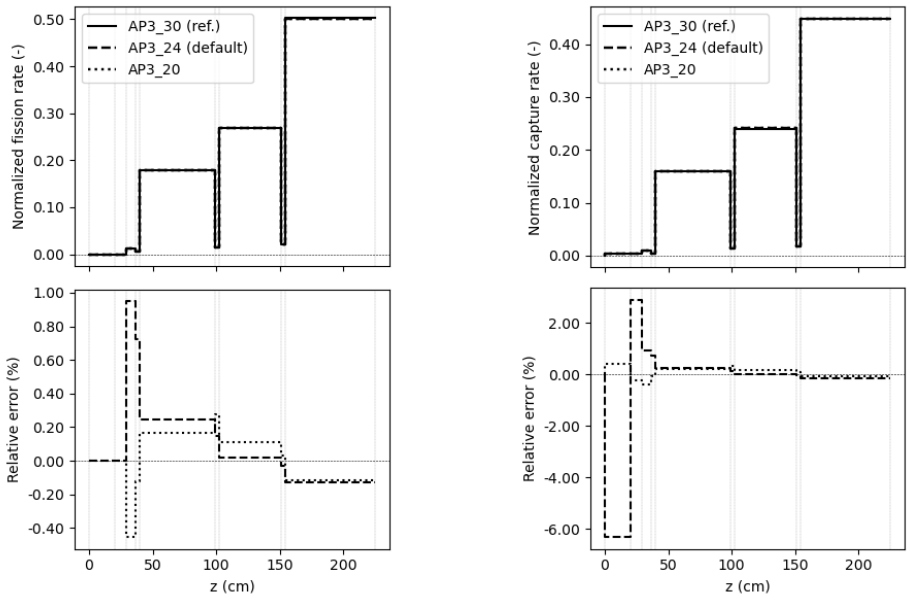
$$\varepsilon_k = \frac{k_{eff} - k_{eff}^{ref}}{k_{eff}^{ref}} = \frac{k_{eff}}{k_{eff}^{ref}} - 1, \tag{6.1}$$

Table 6.1: TDT parameter convergence analysis using the CamiVVER radial mesh. CamiVVER default options are bold. Times are in hh:mm:ss or d-hh:mm:ss.

N_θ	18	20	**24**	30	36		
k_{eff}	1.29573 (-41)	1.29570 (-43)	1.29603 (-18)	1.29605 (-16)	1.29626		
time	6:24:47	6:49:22	7:13:52	7:21:05	8:49:15		
N_φ	**4**	5					
k_{eff}	1.29603 (-3)	1.29607					
time	7:13:52	8:05:24					
N_{mesh}	100	200	**300**	400	500	600	
k_{eff}	1.29609 (+5)	1.29600 (-2)	1.29603 (+1)	1.29602 (0)	1.29602 (0)	1.29602	
time	6:32:46	7:04:51	7:13:52	8:47:11	10:00:19	11:36:06	
Δ_{xy}	0.01	0.015	0.02	0.03	0.04		
k_{eff}	1.29610	1.29606 (-3)	1.29603 (-5)	1.29611 (+1)	1.29603 (-5)		
time	19:42:44	14:01:45	12:07:05	9:40:20	7:21:24		
Δ_{xy} **(cont.)**	**0.05**	0.06	0.07	0.1	0.2		
k_{eff}	1.29607 (-2)	1.29606 (-3)	1.29587 (-18)	1.29452 (-122)	1.29391 (-169)		
time	7:13:52	7:39:49	6:52:00	5:31:37	5:14:39		
Δ_z	0.1	0.2	**0.3**	0.4	0.5	0.6	0.7
k_{eff}	1.29602	1.29602 (0)	1.29603 (+1)	1.29602 (0)	1.29603 (+1)	1.29603 (+1)	1.29601 (-1)
time	13:36:51	8:43:32	7:13:52	6:13:27	6:14:44	5:42:38	5:33:47

It can be observed that:

- N_φ, N_{mesh}, and Δ_z appear stable even when deviating significantly from the CamiVVER default values;
- Non-convergent behavior was observed for the integrated rates for Δ_z values of 0.5, 0.6, and 0.7;
- Small changes in Δ_{xy} have a strong impact on computational time. Values of 0.1 and 0.2 are not convergent, and analysis of integrated rates also shows non-convergence for 0.07;
- For N_θ, the default value could be further tested with 30 angles, as the increase in computational time seems negligible[2]. Moreover, relative differences for this parameter are the largest in magnitude compared to others, which further justifies deeper investigation. Non-convergent behavior in integrated rates was also observed when varying N_θ between 20, 24, and 30, as shown in Figure 6.1. This suggests that further analysis is necessary. For this kind of graphs the integrated rates are normalized by the total fission rate for each code independently, as explained in Appendix A. The vertical light gray lines show the material layer boundaries of the UOX fuel assembly, as reported in Figure 5.4.;



(a) Integrated fission rates.

(b) Integrated capture rates.

Figure 6.1: Comparison between integrated fission and capture rates in 30 (reference in Equation (6.1)), 24 (CamiVVER option), and 20 horizontal angle simulations.

[2]Always considering that HPC cluster nodes do not have uniform performance.

- A non-convergent behavior was also observed for the integrated rates when for $\Delta_z \geq 0.6$, when the relative difference with respect to the default results starts to increase in magnitude;
- Similarly, a non-convergent behavior was also observed for the integrated rates for $\Delta_{xy} \geq 0.07$. Indeed, it can be noted a *significant* increase of k_{eff} discrepancy.

Before further investigating the N_θ parameter, it is important to recall that in MOC-3D simulations, parameters are interdependent. For instance, increasing N_θ requires a corresponding increase in Δ_{xy} to ensure proper coverage of the mesh. For this reason, two cross-parameter combinations were analyzed: N_θ–Δ_{xy} and N_φ–Δ_z, as presented in Table 6.2. It was observed that the latter pair achieves convergence even with the parameter values used in Table 6.1. In contrast, the former pair allows for more detailed conclusions. Only three tests with $N_\theta = 20$ were performed, as it was decided not to use coarser configurations than those of CamiVVER.

Regarding $N_\theta = 30$, convergence was observed for $\Delta_{xy} \leq 0.03$, while the value 0.04 did not yield convergence for the integrated rates. In conclusion, the combination of $N_\theta = 30$ and $\Delta_{xy} = 0.03$ appears sufficient to achieve a satisfactory result. It should be noted that the computational time increased from the default 7:13:52 to 9:40:20 with this configuration. Although this represents a significant increase, it remains manageable.

Table 6.2: TDT parameter convergence analysis simultaneously changing different parameters.

N_θ-Δ_{xy}	20-0.01	20-0.015	20-0.02	**24-0.05**
k_{eff}	1.29589 (-17)	1.29586 (-19)	1.29588 (-18)	1.29603 (-6)
time	17:38:06	15:07:01	10:50:45	7:13:52
N_θ-Δ_{xy} **(cont.)**	30-0.015	30-0.02	30-0.03	30-0.04
k_{eff}	1.29611	1.29612 (+1)	1.29611 (0)	1.29610 (-1)
time	15:33:33	12:55:24	9:40:20	7:21:34
N_φ-Δ_z	**4-0.3**	5-0.1	5-0.2	
k_{eff}	1.29603 (-4)	1.29606 (-2)	1.29608	
time	7:13:52	16:21:56	9:58:32	

However, as already explained, two additional parametric studies were carried out separately to ensure convergence with this configuration. These studies tested the combinations of N_θ and Δ_{xy} set to 30 and 0.02 (Table 6.3) and to 30 and 0.03 (Table 6.4), respectively. Fortunately, the configuration with $N_\theta = 30$ and $\Delta_{xy} = 0.03$ was found

to be already convergent, both in terms of k_{eff} and integrated reaction rates. This result is particularly favorable considering the computational time demand; therefore, all subsequent results presented in this master's thesis involving the APOLLO3® code are computed using the configuration $N_\theta = 30$ and $\Delta_{xy} = 0.03$.

Table 6.3: Analysis with $\Delta_{xy} = 0.02$.

N_φ	**4**	5		
k_{eff}	1.29606 (-7)	1.29612		
time	9:40:20	10:22:09		
N_{mesh}	200	**300**	400	500
k_{eff}	1.29609 (+2)	1.29603 (-3)	1.29606 (-1)	1.29607
time	8:52:32	7:13:52	9:39:33	10:32:06
Δ_z	0.2	**0.3**	0.4	
k_{eff}	1.29607	1.29603 (-3)	1.29607 (0)	
time	12:16:19	12:07:05	8:05:46	
N_φ-Δ_z	**4-0.3**	5-0.2		
k_{eff}	1.29603 (-6)	1.29611		
time	7:13:52	13:59:36		

Table 6.4: Analysis with $\Delta_{xy} = 0.03$.

N_φ	**4**	5		
k_{eff}	1.29606 (-5)	1.29612		
time	9:40:20	10:22:09		
N_{mesh}	200	**300**	400	500
k_{eff}	1.29609 (+2)	1.29603 (-3)	1.29606 (-1)	1.29607
time	8:52:32	7:13:52	9:39:33	10:32:06
Δ_z	0.2	**0.3**	0.4	
k_{eff}	1.29607	1.29603 (-3)	1.29607 (0)	
time	12:16:19	7:13:52	8:05:46	
N_φ-Δ_z	**4-0.3**	5-0.2		
k_{eff}	1.29603 (-6)	1.29611		
time	7:13:52	13:59:36		

While a finer discretization using $N_\theta = 30$ and $\Delta_{xy} = 0.02$ could be selected, the resulting increase in computational cost is not justified by the marginal improvement in accuracy.

6.1.2. Radial mesh convergence

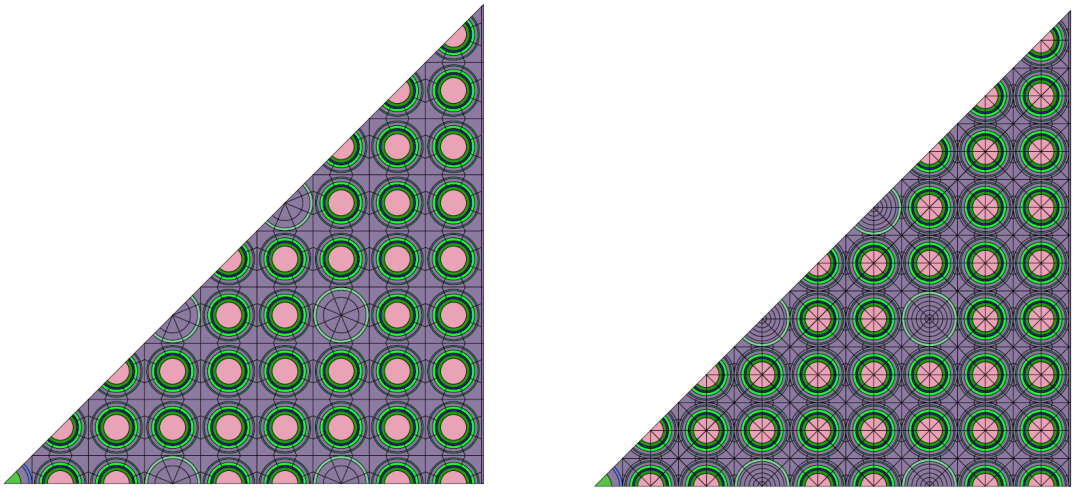
The definitive radial mesh used is shown in Figure 6.2a. Compared to Figure 5.1, two main modifications were introduced:

- The use of 8 sectors also in the guide tube cells;
- The implementation of sector rotation, also referred to as the *windmill* mesh.

The first modification was adopted because the increase in computational time was approximately negligible: +25 min for just 5 guide tube cells. Moreover, using only 4 sectors in combination with sector rotation would result in a mismatch between the fuel cell sectors and those of the guide tubes at the interface. This misalignment could potentially lead to fictitious results.

The second modification was introduced to improve the accuracy of cell reaction rates. In particular, neutron thermalization is highest in the corners of the cells, as these regions are furthest from the fuel. Dividing these corners in half could lead to an inaccurate representation of the neutron flux, since the flux is averaged over each independent zone that is each area in Figure 6.2a delimited by black lines. Notably, this choice has zero

computational cost, as it involves only a geometric rotation and does not require additional characteristic lines, and so additional computational time.



(a) Definitive X-Y layout. (b) Super-refined radial mesh.

Figure 6.2: Comparison between the definitive and super-refined APOLLO3® radial meshes.

The other options tested were:

- Increasing the number of annuli from 1 to 2 inside the guide tube;
- Using 8 angular sectors in the fuel rods up to the outer crown[3];

All results associated with these configurations confirmed that the mesh shown in Figure 6.2a is the convergent one, based on both k_{eff} and integrated reaction rates. Each option was tested independently and compared against the reference mesh in Figure 5.1 and the definitive one in Figure 6.2a.

An attempt was also made to increase the number of fuel annuli by +1 and +2, from the original value of 4 to better model the skin effect (spatial self-shielding) of thermal reactors. This issues regards the fact that in thermal reactors, neutrons are born fast within the fuel rod and are thermalized when exit and approach water. Then, they re-enter in the fuel and fission occurs, but the first portion of the fuel encountered is the border one, thus a higher fuel consumption in the border. For this reason, it is necessary to divide fuel rods in annuli to correctly capture fuel depletion. Since the 4 crown configuration was already convergent, these additional simulations were not further investigated.

[3]It is not evident in Figure 6.2a that the sectors start from the outer crown of the fuel to correctly model the spatial self-shielding of a thermal reactor.

Finally, a super-refined simulation (Figure 6.2b) was tested to definitively confirm the convergence of the radial mesh reported in Figure 6.2a. The result confirmed the assumption: the mesh in Figure 6.2a yielded $k_{eff} = 1.296107$ in 10:16:40, while the super refined mesh produced $k_{eff} = 1.296037$ in 21:12:09, with a relative difference (taking the latter as reference) of 5.4 pcm.

6.2. Serpent

To properly benchmark the TDT MOC-3D solver of APOLLO3® during depletion, a series of simulations were performed by varying the number of axial layers. In principle, increasing the number of axial layers allows Serpent to analyze more distinct burnup zones, which generally leads to a more accurate estimation of the nuclear parameters.

It has to be noted that this test was not performed with APOLLO3® because the work in [5] proved the convergence of the material layer axial mesh.

However, a finer axial mesh also results in a higher number of burnup zones, making the simulation significantly more computationally demanding. As a consequence, given a strict runtime limit of 3 days for the EDF HPC cluster (Cronos) for each simulation, the neutron population must be reduced as the number of burnup zones increases; therefore, the final reference configuration represents a compromise between these two competing factors: achieving sufficient resolution for accurate k_{eff} estimation while maintaining computational feasibility within the allowed time frame.

6.2.1. Axial mesh convergence

The link between the number of the axial layers and the number of burnup zones can be retrieved by observing Figure 5.2: since there are 39 fuel rods (27 full and 12 halves) divided in 4 different radial regions and 1 fission chamber rod (dark green at the bottom left in Figure 5.2) without radial subdivision, the way to link the number of fuel axial layers and the different burnup zones is the following:

$$\begin{aligned} BU\_zones = \; & 39 \times 4 \times Fuel\_Axial\_Layers + 1 \times Fuel\_Axial\_Layers = \\ & 157 \times Fuel\_Axial\_Layers. \end{aligned} \tag{6.2}$$

A total of 4 different axial meshes were tested and are reported in Table 6.5, where the first row represents the material mesh and the fourth one, the CamiVVER stepmesh.

Table 6.5: Parameters of Serpent simulations for axial mesh convergence where n is the number of neutrons per batch and N the number of batches.

BU zones	Fuel layers	n / N ($\times 10^3$)	`set opti`	$\sigma(k_{eff})$ [pcm]
1099 (material mesh)	7	100 / 30	4	1.83
4082	26	200 / 9.5	4	2.29
7065	45	200 / 8	4	2.50
12089 (stepmesh)	77	200 / 6	2	2.89

There was a specific reason behind the choice of the axial meshes:

- *1099 (material mesh)*: it was chosen to start with the material mesh subdivision because it is the coarser one. It has 9 different layers (Figure 5.4): 4 fissile fuel ones without grids (white), 3 fissile fuel layers with the grids (dark gray) and 2 layers for the reflectors (light gray), which, as already pointed out in Chapter 5, have a different material composition;

- *4082*: it is a compromise between high accuracy (closest to material mesh) and computational cost. The idea was to divide the grid layers in two zones each and to progressively subdivide the other layers with finer layers closer to the fuel interfaces and fuel-reflector interface. This can lead to a better representation of interface discontinuities, where the neutron flux has steeper gradients, leading to steeper variations of the total cross section during depletion. Thus, it is better to refine those areas since each layer has a constant cross sections within. Figure 6.3 shows the layer subdivided by dotted light gray lines;

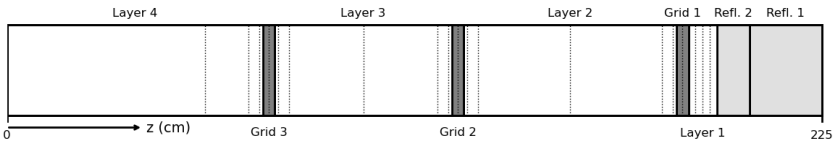


Figure 6.3: Fuel rod axial subdivision in 26 axial layers.

- *7065*: it is the *empirically-found*[4] limit for different burnup zones that Serpent can process in `set opti 4` (maximum optimization that allows variance reduction

[4]For example, 7222 different burnup layers are already too many, which is just one fuel layer more. The problem is linked to an out of memory error, indeed `set opti` 4 allows Serpent to store less information as explained in [36].

techniques). The idea was still to better subdivide grids (3 layers) and set finer layers close to fuel interfaces and fuel-reflector ones, as shown in Figure 6.4;

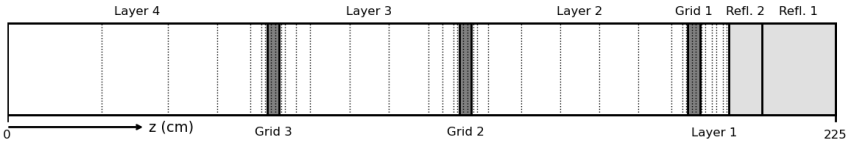


Figure 6.4: Fuel rod axial subdivision in 45 axial layers.

- *12089 (stepmesh)*: it was chosen to use a finer layering to check if the previous subdivision gave a convergent solution. So, the idea was to use the CamiVVER Figure 6.5 stepmesh layering since it was used to compare TRIPOLI-4® and APOLLO3® during depletion.

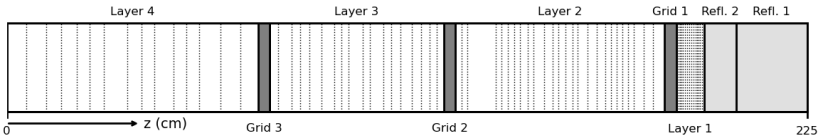


Figure 6.5: Fuel rod axial subdivision in 77 (stepmesh) axial layers.

Quick notes on how to read k_{eff} comparison graphs

Taking Figure 6.6 as a representative example, the plot features a dual y-axis layout to simultaneously illustrate the reactivity evolution and the results of the convergence study. The left y-axis indicates the values of k_{eff}; the corresponding curves, which start at approximately 1.3, represent the fuel depletion for each of the four configurations listed in the legend (Table 6.6 explains how to properly read it). Due to the high degree of agreement between the subdivisions, these four lines appear almost perfectly overlapped. While, the right y-axis refers to the relative difference ($\Delta k/k$, as defined in Equation (6.1)) between each simulation and the reference case. These curves provide a direct measure of the axial mesh convergence, quantifying the discrepancy in pcm as a function of burnup. It should be noted that the x-axis actually represents GWd/tHM; this applies to all graphs of this type.

Table 6.6: Explanation of Figure 6.6 legend symbols.

Label	Fuel layers	BU zones	Line style
S2_ref	77	12089	Solid
S2_1	45	7065	Dashed
S2_2	26	4082	Dotted
S2_3	7	1099	Dash-dotted

After having understood how to properly read the k_{eff} comparison graphs, one can observe the result in Figure 6.6 showing the evolution of k_{eff} for the 4 different axial mesh configurations during burnup. It is clear that the configuration with 7065 burnup zones (45 fissile layers) is the converged case. The criterion adopted to assess convergence was that the relative difference in k_{eff} between two simulations had to fluctuate around zero, and the k_{eff} values at the same burnup point had to lie within the corresponding 3σ statistical error bars. In other words, the k_{eff} value of one configuration must fall within the 3σ uncertainty interval of the other configuration at each burnup step. Where, for relative difference, Equation (6.1) was applied and, here, k_{eff}^{ref} are the stepmesh ones, since it is the axially finest.

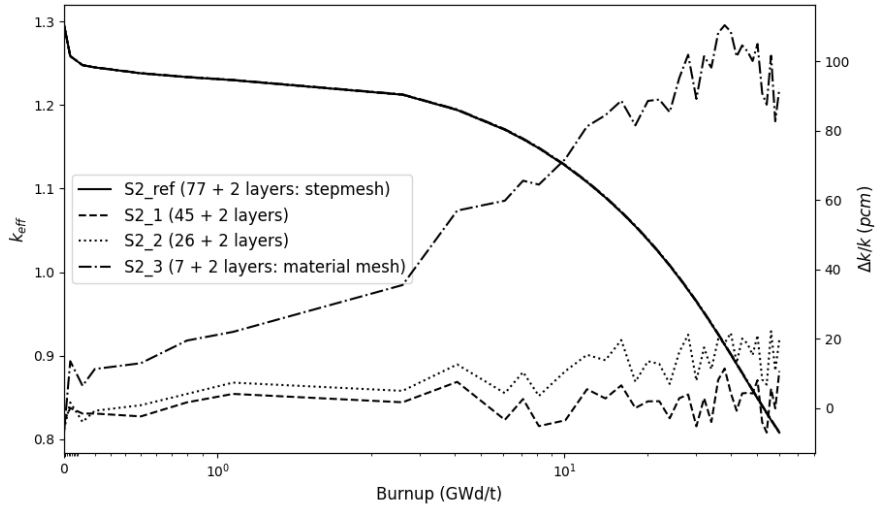


Figure 6.6: Convergence of k_{eff} varying axial mesh.

It is interesting to highlight that already with the first refinement (26 layers) the relative error with the finest one is not too significant. So, in case of computational cost limit, its choice could be anyway well suited.

Furthermore, it should be noted that in Table 6.5 the neutron population, so the uncertainty on k_{eff}, is not the same. As already explained, the precision of the simulations has to be relaxed when the neutron population increases in order to contain the computational cost increase. It was chosen to decrease the number of batches rather than the neutron population because, in order to have a good statistics when the axial layers become finer and finer, it is a wise choice to have a significant number of neutrons per batch. This is also why it was chosen to double it when passing from the material mesh to the axial mesh refinements. The inactive cycles always remained 300.

Also the integrated capture and fission rates at zero burnup and at the end of the calculation (60 100 MWd/tHM) were checked to be sure the result was actually at convergence.

6.2.2. Burnup mesh convergence

In order to obtain reliable results at convergence, a time-dependent study must be performed. Specifically, an analysis of how the outcome varies when the burnup mesh is modified. It should be noted that the results shown in Figure 6.6 adopt a burnup mesh that is significantly different from the one implemented in CamiVVER, which was defined as follows:

Listing 6.1: CamiVVER burnup mesh. Values are in MWd/tHM.

```
bu_mesh = [0.0, 40.0, 120.0, 200.0, 500., 800., 1100., 2600., 4100., 8100.,
    12100., 16100., 20100., 30100., 40100., 50100., 60100.]
```

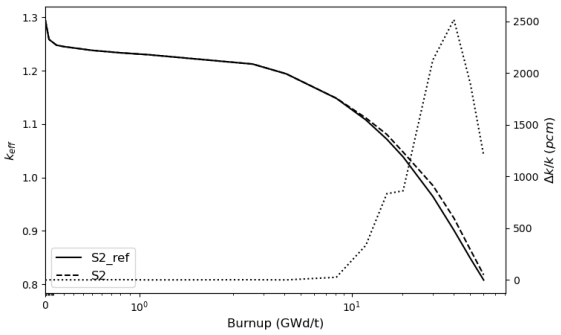
The burnup mesh used in Figure 6.6 is instead:

Listing 6.2: Burnup mesh at convergence. Values are in MWd/tHM.

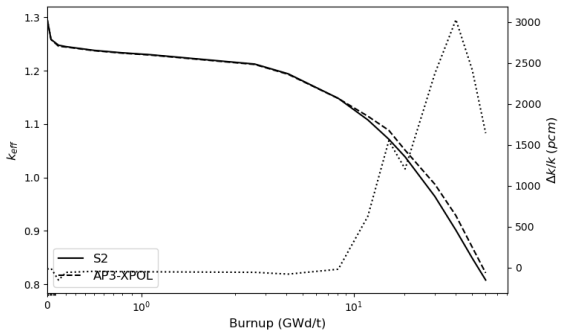
```
bu_mesh = [0.0, 40.0, 120.0, 200., 500., 800., 1100., 2600., 4100., 6100.,
   7100., 8100., 10100., 12100., 14100., 16100., 18100., 20100., 22100.,
   24100., 26100., 28100., 30100., 32100., 34100., 36100., 38100., 40100.,
   42100., 44100., 46100., 48100., 50100., 52100., 54100., 56100., 58100.,
   60100.]
```

This mesh was initially derived from the CamiVVER one following a rule commonly adopted in industry that limits step sizes to no more than 2000 MWd/tHM; however, the inclusion of the point at 7100 MWd/tHM violates this rule. This exception was made after observing that omitting this point caused a spike in the relative difference at 8100 MWd/tHM, with values exceeding the 3σ confidence interval. In the interest of transparency, this observation was later found to be incorrect, but the additional step did not affect the results presented in this master's thesis since the mesh will be further modified.

Comparing the two simulations, Figure 6.7a shows a significant discrepancy, where $S2\_ref$ uses the mesh from Listing 6.2. The divergence clearly begins at 8100 MWd/tHM, reaching an astounding 2516.4 pcm at 40 100 MWd/tHM. This behavior is likely due to error propagation starting at 8100 MWd/tHM. Note that whenever the "AP3-XPOL" label is present in a graph (such as in Figure 6.7b, comparing the Serpent result with the converged mesh against APOLLO3® with the CamiVVER mesh), it highlights that APOLLO3® adopts axial polynomial expansions for both flux and cross sections.



(a) Comparison of converged ($S2_{ref}$) vs. CamiVVER burnup meshes (Listings 6.1 and 6.2).



(b) Comparison of Serpent results (converged mesh, reference) vs. CamiVVER APOLLO3® (Listings 6.1 and 6.2).

Figure 6.7: Analysis of the burnup mesh influence and code-to-code comparison.

To confirm that Listing 6.2 corresponds to a converged case, it must be compared with both a coarser and a finer burnup mesh. For the finer mesh, an additional step was added in between each one of Listing 6.2.

For the coarser mesh, as can be seen in the previous graphs, the largest deviations from the CamiVVER mesh occur after 20 100 MWd/tHM. Before this value, further reduction in step size offers no benefit, as the mesh is already sufficiently coarse. Moreover, retaining this level of refinement at low-burnup is important to accurately capture the behavior of xenon and iodine. This ensures that even in a converged case, enough points are present to allow a meaningful and fair comparison between the two codes.

To reduce computational time, the no-more-than-2000 MWd/tHM rule was relaxed, resulting in the mesh shown in Listing 6.3, which reduces the number of steps after 20 100 MWd/tHM from 4 to 3 each set of 10 000 MWd/tHM:

Listing 6.3: Burnup mesh with less steps. Values are in MWd/tHM.

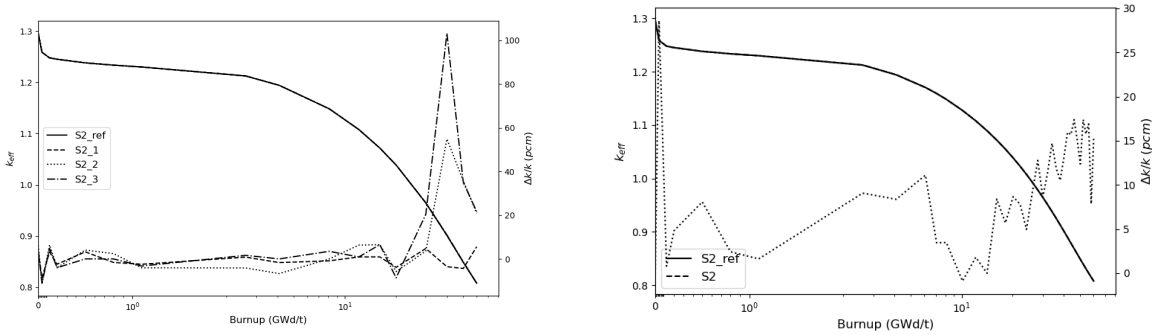
```
bu_mesh = [0.0, 40.0, 120.0, 200., 500., 800., 1100., 2600., 4100., 6100.,
    7100., 8100., 10100., 12100., 14100., 16100., 18100., 20100., 23400.,
    26800., 30100., 33400., 36800., 40100., 43400., 46800., 50100., 53400.,
    56800., 60100.]
```

It should be said that the fine and coarse runs were performed with different batches: 3000 and 2000 batches respectively, both with 200 000 neutrons per batch. Anyway, the impact is just on the uncertainty on the k_{eff} and on the rates.

The coarse run had fewer batches because additional analyses were performed using 2000 batches of 200 000 neutrons each and a lower number of batches was necessary to

speed up the simulations. Specifically, it was tested whether Listing 6.3 could be used by increasing the predictor-corrector substeps from the original 3 and 5 to 10 for both.

Figure 6.8a shows the comparison between the simulation with double the burnup steps (reference), Listing 6.2 (S2_1), Listing 6.3 with 10 predictor-corrector substeps (S2_2), and Listing 6.3 with 3 and 5 substeps (S2_3).



(a) Testing the Listing 6.2 as the converged burnup mesh.

(b) Comparison with (reference) and without predictor-correctors.

Figure 6.8: Burnup mesh refinement benchmark and impact of numerical algorithms on depletion accuracy.

From Figure 6.8a, it can be observed that Listing 6.2 shows a statistically low relative difference compared to the finest mesh. In contrast, the simulations using Listing 6.3 does not converge, regardless of the predictor-corrector substep configuration.

A counterintuitive behavior was observed for simulations (S2_2) and (S2_3): the one with more predictor-corrector substeps (S2_3) was actually farther from the converged reference than the one with fewer substeps (S2_2). This odd result could possibly be explained by the fact that, when the burnup mesh is too coarse, the fluctuation between steps becomes larger than what the predictor-corrector method can effectively smooth out. In such cases, increasing the number of substeps might not help and could even worsen the result because the method is trying to interpolate over a space that is too discontinuous. Moreover, these odd convergence and instability cases were studied in [20–23] where it is shown that the predictor-corrector stability is highly dependent on the system geometry simulated, so no general assumptions can be drawn, as explained in Section 2.2.4.

Even though this is neither a complete nor exhaustive explanation, this very issue was not further investigated since it falls outside the scope of this thesis.

Other predictor-corrector studies were performed, each with 2000 batches and 200 000 neutrons per batch:

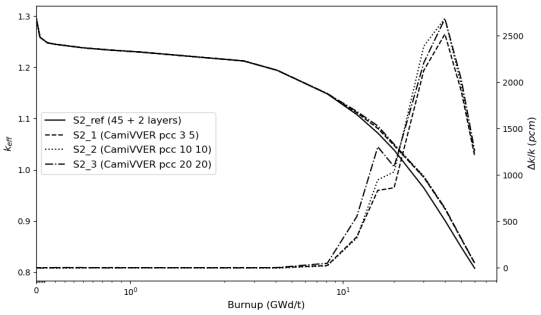
- Using Listing 6.2 without predictor-corrector;
- Using Listing 6.1 with 10 substeps for both predictor and corrector;
- Using Listing 6.1 with 20 substeps for both.

Figure 6.8b demonstrates that predictor-correctors are necessary, as the run using Listing 6.2 without them does not reach convergence.

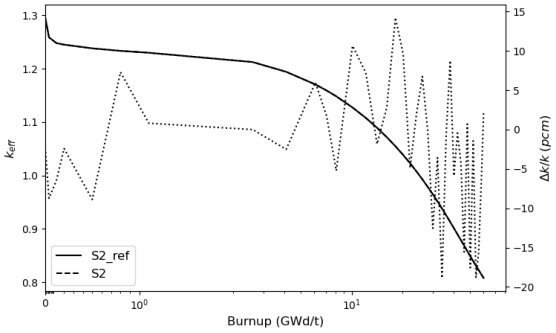
It should be noted that although the relative differences are not extremely high, they are sufficient to fall outside the 3σ confidence interval and are consistently biased toward positive values, indicating that the run without predictor-correctors yields higher k_{eff} values. Moreover, a noticeable spike appears at 40 MWd/tHM, which aligns with findings in [17], where it is shown that substeps are particularly effective for short-lived nuclides such as xenon and iodine. Since the computational cost of implementing the number of substeps is negligible, no further convergence study was performed to reduce them. Additionally, as discussed in [17], using 3 and 5 substeps for predictor and corrector is generally sufficient.

Figure 6.9a shows a comparison between the converged mesh Listing 6.2 and the CamiVVER mesh under different predictor-corrector configurations: without (same curve as in Figure 6.7b), with 10 substeps, and with 20 substeps.

It is evident that increasing the number of substeps improves consistency with Figure 6.7b, except at 30 100 MWd/tHM and 40 100 MWd/tHM, where the simulation with 10 substeps shows a larger relative difference than the one with fewer substeps. This anomaly is likely due to the fact that these points exhibit the highest discrepancies, which are affected by uncertainty propagation and correlation effects. This is a similar behavior of Figure 6.8a and another possible explanation is that when the discrepancy between simulations becomes too large, the assumption of linearity between predictor-corrector steps breaks down. In such cases, increasing the number of substeps may not necessarily improve convergence, as the system's response becomes non-linear and more sensitive to small perturbations. This could lead to unexpected behavior, such as larger deviations despite finer temporal resolution. Further investigation would be needed to confirm this hypothesis.



(a) Comparison between the converged burnup mesh Listing 6.2 and the CamiVVER mesh Listing 6.1 under different predictor-corrector configurations.



(b) Comparison between the converged burnup mesh Listing 6.2 with 5 crowns per fuel rod (reference) and 4 crowns per fuel rod (used in all other simulations).

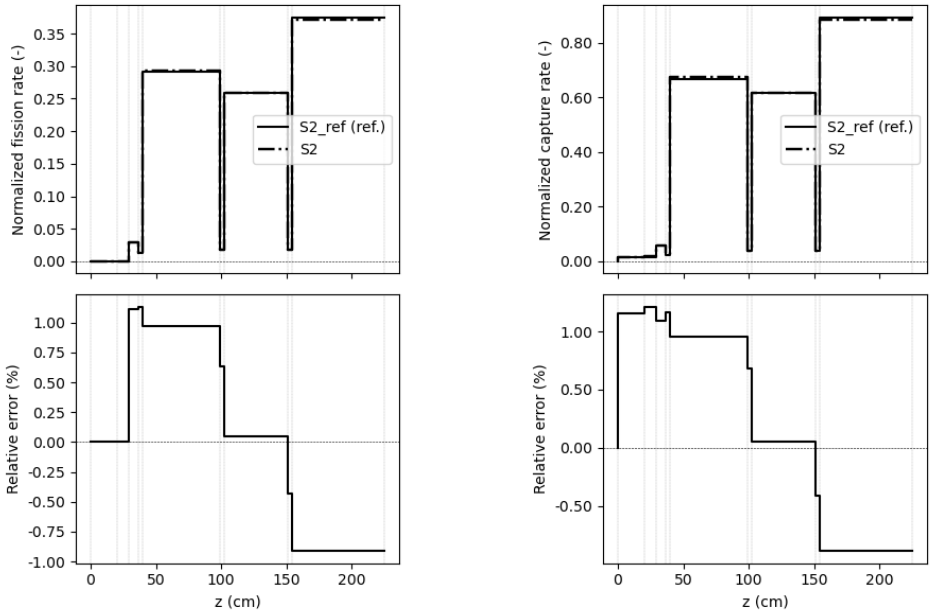
Figure 6.9: Sensitivity analysis of numerical parameters: impact of predictor-corrector configurations and radial mesh refinement.

6.2.3. Radial mesh convergence

Analogously to the APOLLO3® case, the possibility of refining the radial discretization of the fuel rods from 4 to 5 crowns was analyzed. The option of reducing the number of crowns to 3 was not considered, since 4 radial crowns is generally regarded as the minimum acceptable value at industrial level for UOX fuel rod modeling. This ensures sufficient resolution to capture radial gradients in temperature, burnup, and isotopic composition.

Figure 6.9b shows the comparison in k_{eff} between the two configurations. The reference simulation uses 5 crowns and was run with 200 000 neutrons per batch and 2000 batches. It was chosen not to further increase the number of neutrons per batch since it is already a substantial value.

However, since k_{eff} is an integrated quantity, it may be subject to error counterbalance; therefore, to ensure the result's robustness, in this case, both integrated reaction rates and 2D cell-wise rates were analyzed. Figure 6.10 shows the integrated capture and fission rates, respectively. It was chosen 54 000 MWd/tHM because is the closest burnup step to the end with the highest relative difference magnitude: -18.8 pcm.



(a) Integrated fission rates.

(b) Integrated capture rates.

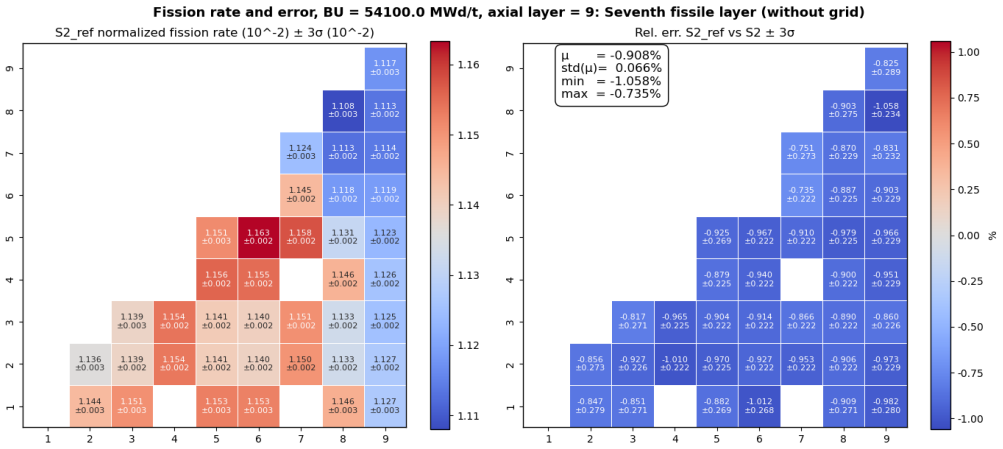
Figure 6.10: Comparison between integrated fission and capture rates for a 5 fuel crown convergent simulation (reference) and a 4 fuel crown one at 54 000 MWd/tHM.

To further assess the impact of increasing the number of crowns, Figure 6.11 presents the 2D cell-by-cell reaction rates for the layer closest to the midplane where the relative error identified in Figure 6.10 is one of the highest. For a detailed explanation on how to properly interpret this kind of graph, see Section A.2.

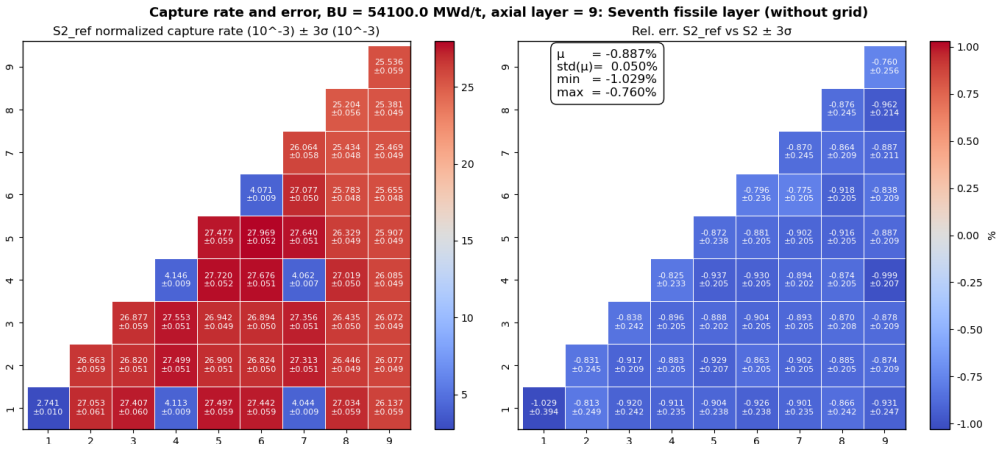
The discrepancies observed have a maximum value of about 1 % (thus, lower ones for the other layers and burnup steps) and, considering all layers, they fluctuate without a clear trend, indicating that the 4-crown configuration is already sufficient for convergence. Consequently, the use of 5 crowns, while slightly more accurate, does not justify the additional computational cost. Indeed, adopting 5 crowns modifies the burnup zone calculation in Equation (6.2) as follows:

$$BU\_zones = 39 \times \mathbf{5} \times Fuel\_Axial\_Layers + 1 \times Fuel\_Axial\_Layers = \mathbf{196} \times Fuel\_Axial\_Layers. \tag{6.3}$$

The bold term highlights the changes compared to Equation (6.2). With the reference axial layer configuration, this results in 8820 burnup zones. Such a high number forces Serpent to disable its best optimization mode, preventing the use of variance reduction techniques and significantly increasing computational cost.



(a) 2D pin-by-pin fission rate.



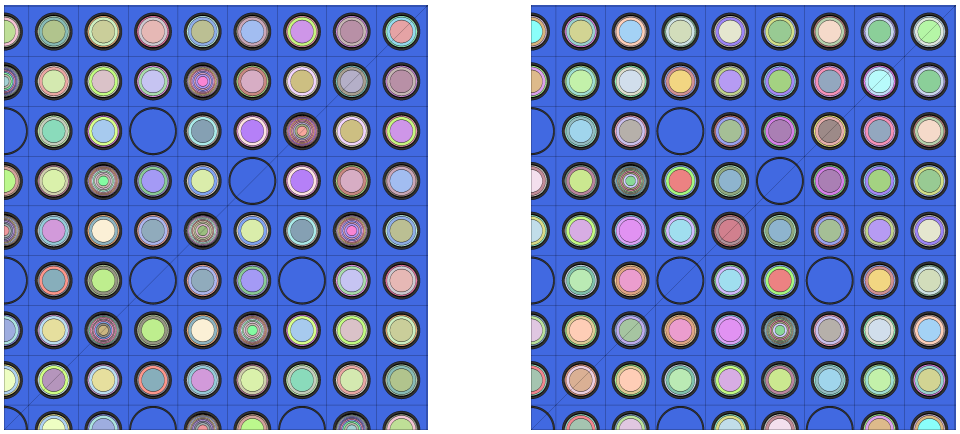
(b) 2D pin-by-pin capture rate.

Figure 6.11: Comparison between 2D pin-by-pin fission and capture rates for a 5 fuel crown convergent simulation (reference) and a 4 fuel crown one for the fissile layers closest to the center, at 54 000 MWd/tHM.

6.3. UOX-Gd assemblies

Since each logical passage to get to the convergent solution was already explained in this chapter, just the major differences will be presented for the UOX-Gd assembly, for the sake of conciseness.

Two major UOX-Gd assembly configurations were studied, one with a total of 36 UOX-Gd rods (Figure 6.12a) and the other with just 8 (Figure 6.12b), as explained in Chapter 5. The UOX-Gd rods shown in the pictures are the ones that have more radial crowns to better model the skin effect. Each different color refers to an independent burnup zone. First, the one with more gadolinium was analyzed.



(a) 36 UOX-Gd assembly. (b) 8 UOX-Gd assembly.

Figure 6.12: 2D cross-sectional geometries of the analyzed UOX-Gd assemblies modeled in Serpent.

6.4. APOLLO3®: 36 UOX-Gd rods assembly

For the 36 UOX-Gd assembly, a dedicated TDT parameter convergence study was omitted, as the optimal settings identified for the UOX case were assumed to remain valid. Further refinement would have resulted in a prohibitive computational cost as reported for the UOX case. Consequently, the analysis proceeded directly to the radial mesh convergence analysis to maintain a balance between precision and execution time.

6.4.1. Radial mesh convergence

The definitive radial mesh retrieved is presented in Figure 6.13, where the UOX-Gd rods are clearly visible. It is interesting that, apart from the further refinement of the rods with gadolinium, the radial mesh is the same as the UOX assembly one, without the grids (Figure 6.2a).

The convergence was confirmed by checking the following refinements:

1. As Figure 6.13, but the angular sectors of the UOX-Gd rods arrive up to the inner crown (instead of just at the outer crown). This choice was made to check if the gadolinium absorption was analogously modeled;

2. The guide tubes have 16 angular sectors. In literature, for a 2D analysis, this choice is usually adopted;

3. The guide tubes have one more radial crown inside the metal guide. Also this choice

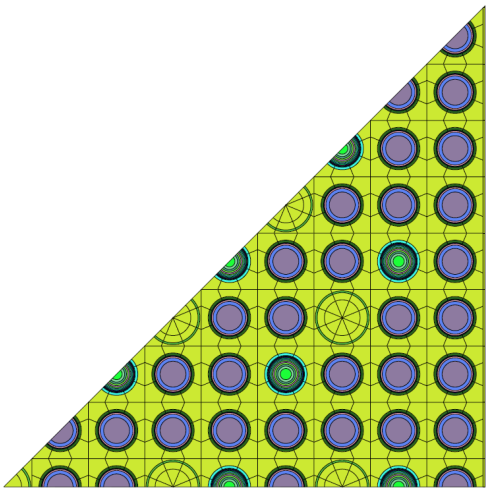


Figure 6.13: Definitive X-Y layout of the UOX-Gd 3D assembly calculation.

was found in literature for 2D analyses.

It has to be said that no other configurations were tested as the computational time was already demanding: initial hints of not being able to perform a depletion simulation up to 60 GWd/tHM were quite clear.

Table 6.7 confirms the convergence for k_{eff}. The comparisons between integrated and 2D rates are omitted for the sake of conciseness.

Table 6.7: Radial mesh convergence analysis for the 36 UOX-Gd assembly. The configuration numbers refer to the previous list.

Configuration	Figure 6.13 (Ref.)	**1**	**2**	**3**
$\boldsymbol{k_{eff}}$	0.91496	0.91493	0.91501	0.91497
Discrepancy [pcm]	/	-3	5	1

6.5. Serpent: 36 UOX-Gd rods assembly

Similarly to the UOX case, axial and burnup mesh convergence analyses were performed.

6.5.1. Axial mesh convergence

The tested axial meshes are presented in Table 6.8. It should be noted a lower precision in terms of $\sigma(k_{eff})$ with respect to Table 6.5 due to the fact that the gadolinium necessitates the use of way more burnup steps than the UOX case to correctly model the steep gradient

of neutron flux due to its high neutron capture cross section. This results in the need of a relaxation of the precision, namely the batches. The neutrons per batch were not changed since the UOX-Gd rods need a finer radial mesh subdivision (11 crowns each, a subdivision used at industrial level), so the fuel crown width decreases with respect to the UOX case (just 4 crowns) and it is better to have a good statistics for each batch in order to be sure not to get results affected by too high uncertainties (for examples few neutrons per batch hitting a small volume of a UOX-Gd crown). It should be remembered that this configuration has an empty instrumentation channel on the bottom left (without fission chamber).

Table 6.8: Parameters of Serpent simulations for the axial mesh convergence analysis (UOX-Gd assembly).

BU zones	**Fuel layers**	n / N ($\times 10^3$)	`set opti`	$\sigma(k_{eff})$ **[pcm]**
2050	10	200 / 2	4	5.00
4100	20	200 / 2	4	5.00
6150	30	200 / 2	4	5.00
8200	40	200 / 1.5	2	5.77

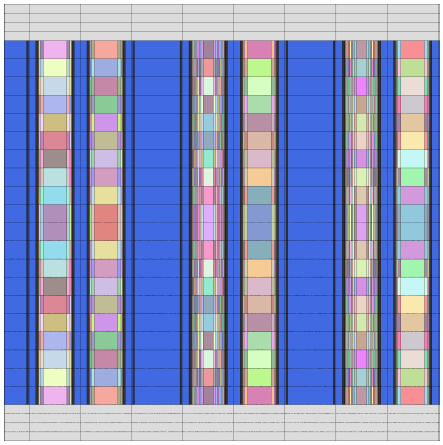
Figure 6.14a shows the YZ cut at z = 0 for the 36 UOX-Gd rods case. The 4 layers of the detector are clearly visible, as the empty instrumentation channel and the UOX-Gd rods.

The choices for the layer subdivision followed the logic to use the material mesh 10 (fissile) and 4 (reflectors) and divide each fissile layer respectively in 2, 3 and 4. The division by 2 and 3 allows to use Serpent in its best optimization mode while the one by 4 needs `set opti 2`. Figure 6.14b, Figure 6.14c, Figure 6.14d and Figure 6.14e show respectively the material mesh, division in 20, 30, and 40 fissile layers. As for the UOX case, the reflector was not further subdivided.

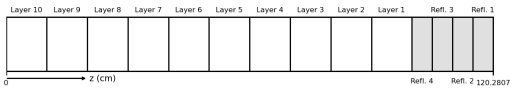
The amount of independent burnable zones is retrieved analogously to Equation (6.2) and reported in Equation (6.4):

$$\begin{aligned} BU\_zones = \; & 32 \times 4 \times Fuel\_Axial\_Layers + 7 \times 11 \times Fuel\_Axial\_Layers = \\ & 205 \times Fuel\_Axial\_Layers. \end{aligned} \tag{6.4}$$

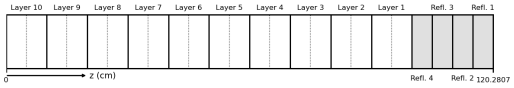
The results are shown in Figure 6.15 where it is evident that the simulations with 20 and 30 fissile layers are already at convergence, while the material one shows significant fluctuations. Anyway, it was not observed a significant difference in computational time between the two simulations (hh:mm:ss): 2d-01:45:01 for the former, 2d-05:06:43 for the



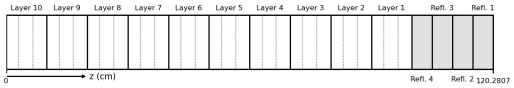
(a) YZ cut of the 36 UOX-Gd assembly.



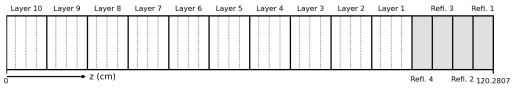
(b) Material mesh.



(c) 20 axial layers.



(d) 30 axial layers.



(e) 40 axial layers.

Figure 6.14: UOX-Gd assembly modeling and mesh refinement: (a) global YZ section, (b) initial material mesh, and (c-e) progressive axial discretization from 20 to 40 layers.

latter. Again, this could be due to the fact that different nodes in the EDF HPC cluster have difference computational capabilities; anyway, the computational time needed by the 30 fissile layer subdivision is manageable. Furthermore, in Chapter 7 it will be also explained the need to reduce the burnup mesh length due to APOLLO3® computational problems in reaching the end of the simulation. Thus, analogously to the UOX assembly case, it was chosen to study the finest Serpent configuration that allows its best optimization mode (`set opti 4`).

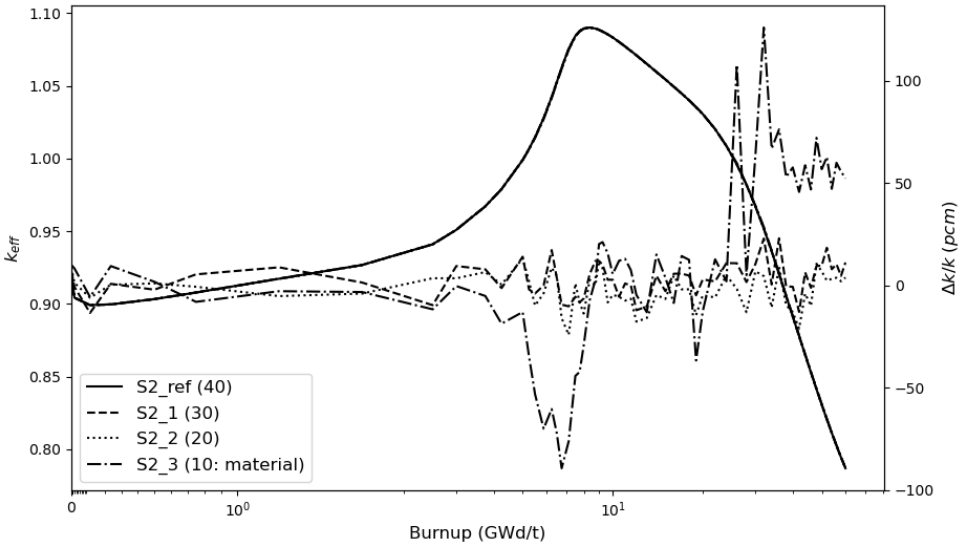


Figure 6.15: Convergence of k_{eff} varying axial mesh for the 36 UOX-Gd assembly.

6.5.2. Burnup mesh convergence

The results in Figure 6.15 are retrieved with the following burnup mesh (Listing 6.4):

Listing 6.4: Burnup mesh at convergence for the UOX-Gd assembly with 61 steps. Values are in MWd/tHM.

```
bu_mesh =[0.0, 18.75, 112.500, 237.500, 500.000, 750.000, 1250.00, 1750.00,
    2500.00, 3000.00, 3750.00, 4250.00, 5000.00, 5250.00, 5500.00, 5875.00,
    6250.00, 6500.00, 6750.00, 7125.00, 7500.00, 7750.00, 8000.00, 8375.00,
    8750.00, 9000.00, 9250.00, 9625.00, 10000.00, 10500.0, 11000.0,
   11500.0, 12000.0, 13000.0, 14000.0, 15000.0, 16000.0, 17000.0, 18000.0,
   19000.0, 20000.0, 22000.0, 24000.0, 26000.0, 28000.0, 30000.0, 32000.0,
   34000.0, 36000.0, 38000.0, 40000.0, 42000.0, 44000.0, 46000.0, 48000.0,
   50000.0, 52000.0, 54000.0, 56000.0, 58000.0, 60000.0]
```

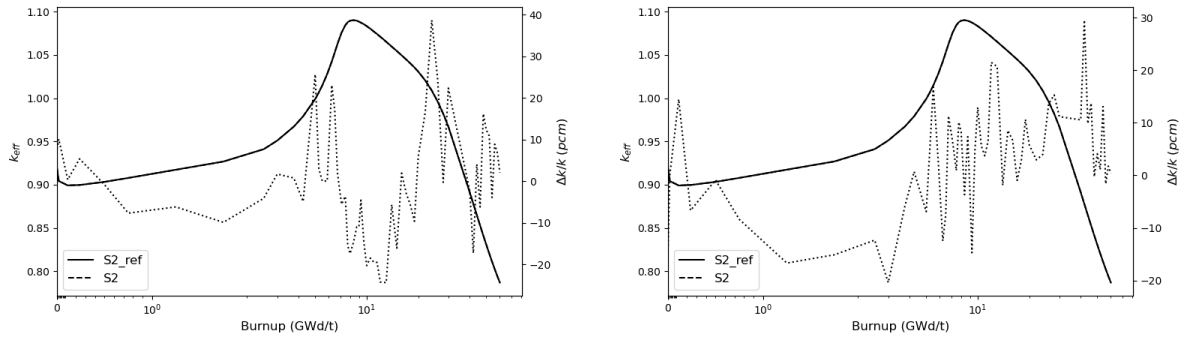
It was derived from the one provided for APOLLO3® by the validation team of CEA, written in Listing 6.5. It should be noted that, as in Listing 6.2, for the steps after 20 000 MWd/tHM there are steps of maximum +2000 MWd/tHM otherwise the convergence is not achieved. Furthermore, with respect to Listing 6.2, there are finer steps around 10 000 MWd/tHM where the gadolinium contribution plays a significant role. Indeed, with respect to Listing 6.5, Listing 6.4 has a burnup step in the middle of each one between 5500 MWd/tHM and 16 000 MWd/tHM.

Listing 6.5: Burnup mesh of the validation team at CEA. Values are in MWd/tHM.

```
bu_mesh =[0.0, 18.75, 112.500, 237.500, 500.000, 750.000, 1250.00, 1750.00,
    2500.00, 3000.00, 3750.00, 4250.00, 5000.00, 5500.00, 6250.00, 6750.00,
    7500.00, 8000.00, 8750.00, 9250.00, 10000.00, 11000.0, 12000.0, 14000.0,
    16000.0, 18000.0, 19000.0, 20000.0, 22000.0, 24000.0, 28000.0, 30000.0,
    35000.0, 36000.0, 37500.0, 40000.0, 42000.0, 44000.0, 46000.0, 48000.0,
    50000.0, 52000.0, 54000.0, 56000.0, 58000.0, 60000.0]
```

To benchmark the convergence of Listing 6.4, another burnup mesh was tested that consisted in Listing 6.5 with the double the steps in between each original one. Figure 6.16a proves that Listing 6.4 is the converged one. Indeed, before 5500 MWd/tHM and after 16 000 MWd/tHM, when there are the double of burnup steps that the ones in Listing 6.4, the convergence holds, while in between these values, when there are half of the burnup steps of Listing 6.4, the convergence fails.

In Figure 6.16a, S2_ref is the at-convergence one (Listing 6.4) while S2 is the one retrieved by Listing 6.5 with the double of steps in between each original one.



(a) Burnup mesh convergence (36 UOX-Gd assembly).

(b) Radial crown convergence for the UOX-Gd rods (36 UOX-Gd assembly).

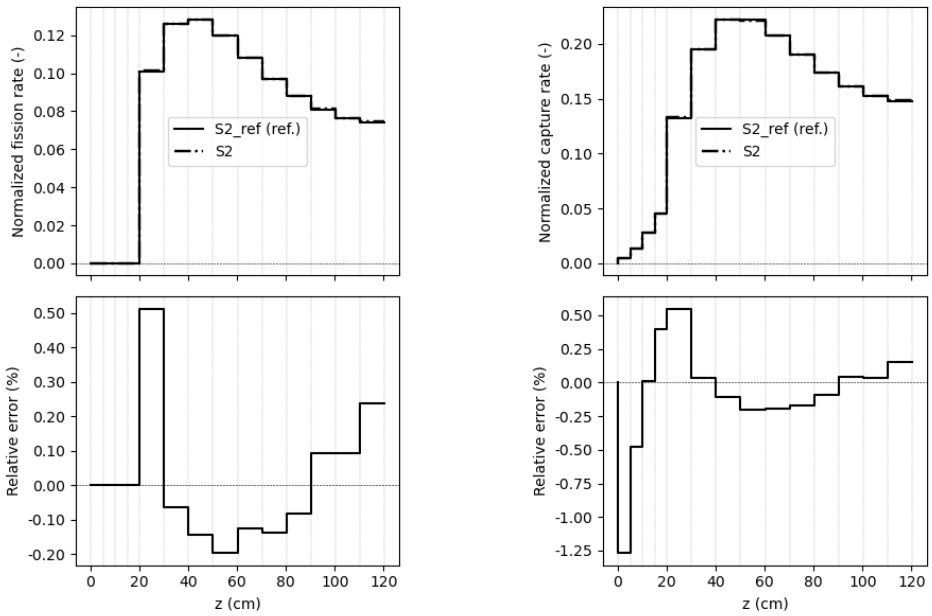
Figure 6.16: Convergence analysis for the 36 UOX-Gd assembly: (a) burnup mesh and (b) impact of radial fuel rod discretization (Listing 6.4).

6.5.3. Radial mesh convergence

It was chosen to test the refinement of the crown subdivision of UOX-Gd rods; not the UOX ones since it has been already proved that there is no need of further refinement in Figure 6.9b, Figure 6.10 and Figure 6.11.

The convergence of k_{eff} is further confirmed by the results illustrated in Figure 6.16b. This figure presents an analysis comparing the reference configuration, which employs 12 radial crowns per UOX-Gd fuel rod, against the standard 11-crown discretization, both utilizing the converged burnup mesh defined in Listing 6.4. For the standard UOX rods, a 4-crown subdivision was maintained, consistent with the previous UOX assembly study.

To further analyze the effect on the rates, both integrated and 2D cell-by-cell were studied. First, Figure 6.17 shows the integrated ones at 42 000 MWd/tHM which is the burnup step with highest discrepancies in Figure 6.16b with a value of 29.5 pcm. It is evident that the relative errors for the capture rate in the outer reflector is the highest one and this trend was observed also at burnup zero where the sign of the discrepancy changed. This trend occurred also for the comparisons between other simulations, namely the ones presented in Figure 6.15. This behavior is likely due to the fact that the capture rate magnitude in the outer reflector layer is really low compared to the others and small absolute difference results in a significantly high relative one. Furthermore, the intrinsic statistical characteristic of a Monte Carlo code results in the presence of fluctuations that are above all visible in high uncertainty areas, namely the ones with a lower neutron population.



(a) Integrated fission rates.

(b) Integrated capture rates.

Figure 6.17: Comparison between integrated fission and capture rates with 12 crowns for the UOX-Gd rods (reference) and with 11 UOX-Gd ones at 42 000 MWd/tHM for the 36 UOX-Gd assembly. The UOX rods have 4 crowns, as in the UOX assembly case.

The study of the 2D cell-by-cell rates focused on the fissile layer closer to the inner reflector because it has the highest relative difference in Figure 6.17. Two major patterns can be observed in Figure 6.18:

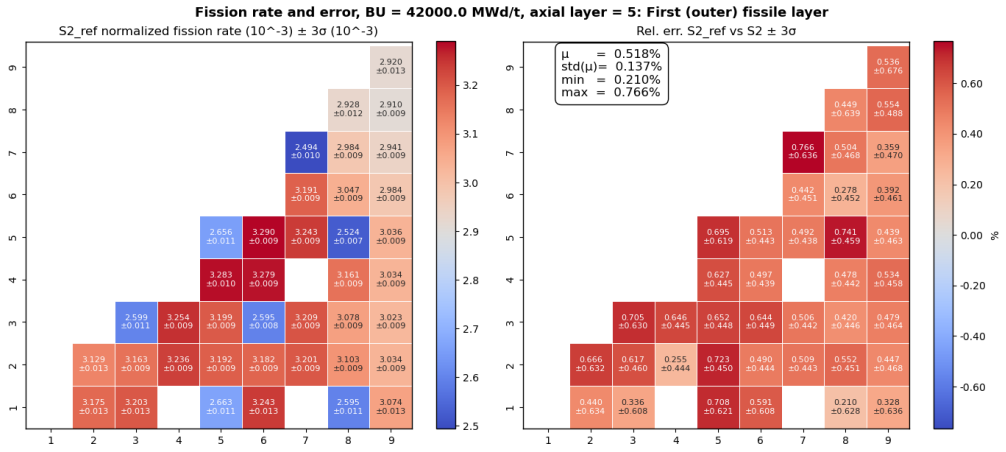
- UOX-Gd rods have a higher capture and lower fission rates thanks to the presence of gadolinium with respect to the UOX ones;
- There is not a visible trend on the UOX-Gd cells, sometimes the relative errors are higher, other times lower, thus it is assumed a convergent situation already with 11 crowns.

Furthermore, the amount of independent burnable regions would become

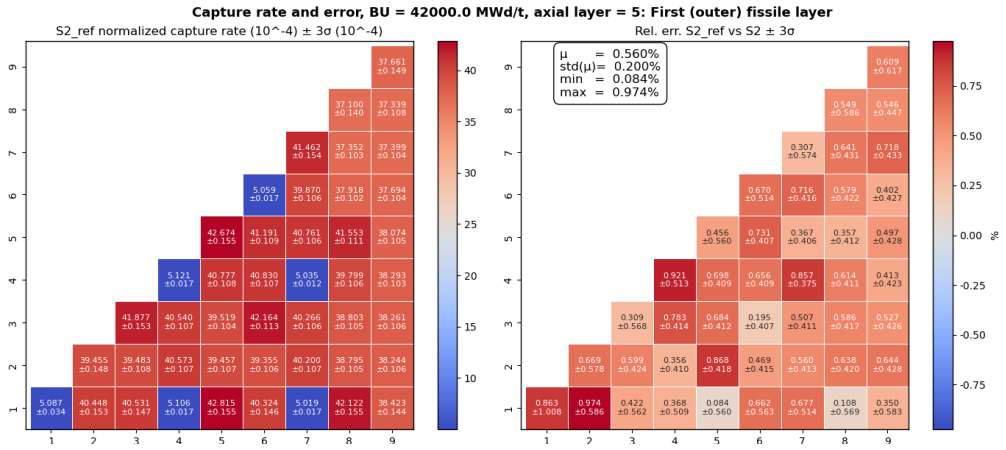
$$BU\_zones = 32 \times 4 \times Fuel\_Axial\_Layers + 7 \times 12 \times Fuel\_Axial\_Layers = \mathbf{212} \times Fuel\_Axial\_Layers. \tag{6.5}$$

This means that, with 30 axial layers, the would be 6360 and still allows Serpent to be used with its best optimization mode (`set opti 4`).

The simulation with 12 crowns for the UOX-Gd rods was run with 200 000 neutrons per batch and 2000 batches as the other UOX-Gd ones.



(a) 2D pin-by-pin fission rate.



(b) 2D pin-by-pin capture rate.

Figure 6.18: Comparison of 2D fission and capture rates at 42 000 MWd/tHM between 12-crown (ref.) and 11-crown UOX-Gd rod discretizations.

6.6. 8 UOX-Gd rods assembly

For what concerns the 8 UOX-Gd fuel rods assembly, the same burnup and radial meshes of the 36 UOX-Gd case were adopted both for APOLLO3® and Serpent.

This choice was related to the poor depletion results obtained for the 36 UOX-Gd rods comparison (as will be explained in Chapter 7). Thus, adopting the same meshes for a case with less gadolinium, so with weaker flux gradients, implies that the results are for sure at convergence and allow a significant amount of time saving, since no time was invested in performing a further convergence analysis.

7 | APOLLO3® - Serpent depletion benchmark

After the convergence analyses for both codes, the first full depletion simulation comparison was performed without using isotopic mixtures in the self-shielding routine, as for CamiVVER. This choice was made because preliminary tests revealed that including mixtures would cause the simulation time to become unreasonably long. This observation led to an optimization of the self-shielding options, which will be detailed later in this chapter.

It has to be said that all depletion simulations of APOLLO3® have been launched on the CEA cluster Orcus due to the 3-day time limit on the EDF one Cronos. Indeed, the computation time is nearly 10 days.

Figure 7.1 presents the first depletion benchmark results obtained in this thesis.

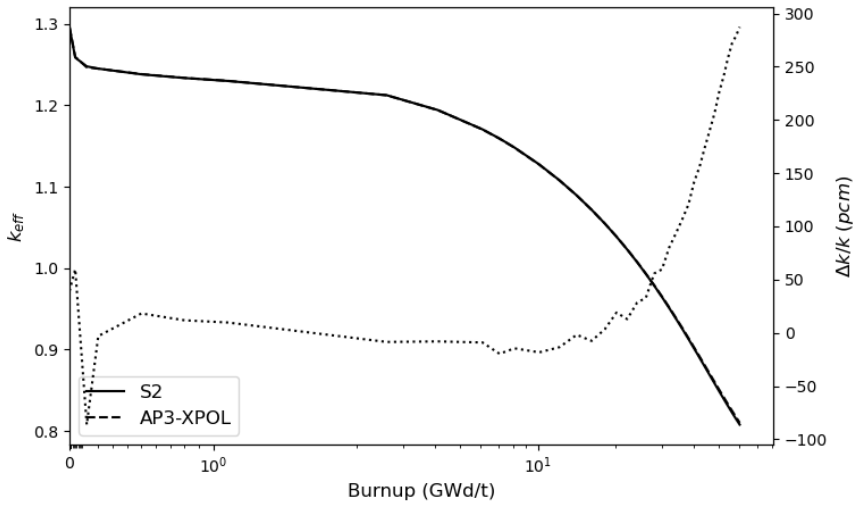
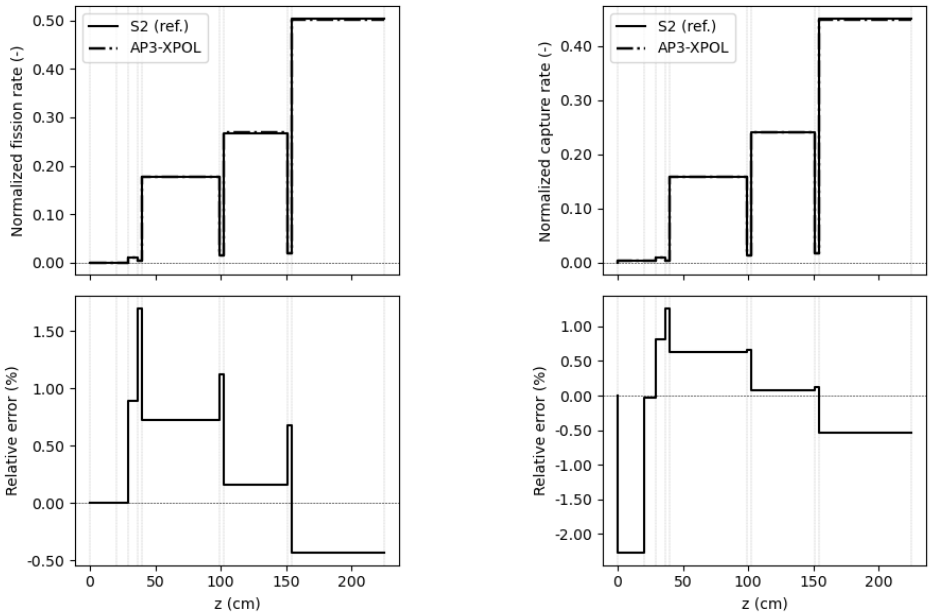


Figure 7.1: First full depletion benchmark result of the APOLLO3® code against Serpent.

The result is substantial: with a maximum relative difference of 287.3 pcm at 60 100 MWd/tHM, the APOLLO3® calculations can be considered benchmarked for the purposes of this work; however, two noteworthy observations can be made from Figure 7.1:

- A local error is visible at the very beginning, specifically at 40 MWd/tHM and 120 MWd/tHM, where the relative differences are 59.7 pcm and -86.3 pcm, respectively. This may suggest that APOLLO3® requires additional intermediate burnup steps in this range[1].
- The relative difference remains below 20 pcm (except for the aforementioned points and the zero-burnup step) up to 20 100 MWd/tHM which shows a relative difference of 39.4 pcm. Beyond this point, it increases progressively. This is hypothesized to be caused by differences between the two codes in the isotopic depletion chains or in their energy deposition models.

Additionally, the integrated reaction rates at zero burnup were analyzed, as shown in Figure 7.2.



(a) Integrated fission rates. (b) Integrated capture rates.

Figure 7.2: Comparison between integrated fission and capture rates of Serpent (reference) and APOLLO3® at zero burnup.

It is clear that the largest discrepancies occur in the outer reflector region, and this difference remains among the highest throughout the depletion. This suggests that the neutron flux in APOLLO3® may not have reached convergence in this region, indicating that a refinement of the reflector layer should be investigated (even if, as pointed out in Chapter 6,

[1]This is certainly not due to Serpent, since Figure 6.8a already demonstrated convergence with twice as many burnup steps.

[5] proved the convergence of APOLLO3® with the axial material mesh, but the TDT parameters were different). In light of these considerations, Serpent and APOLLO3® underwent slight modifications with respect to the parameters chosen according to the analyses reported in Chapter 6.

7.1. APOLLO3®

As previously mentioned, the following aspects were investigated in better detail:

- *Self-shielding optimization:* evaluation of the use of mixtures and which energy group range should be chosen in order to optimize the whole self-shielding treatment;
- *Burnup mesh refinement:* verification of whether a refined burnup mesh could reduce the fluctuations observed for the very first burnup steps;
- *Reflector layer subdivision:* assessment of whether further subdivision of the reflector region could reduce the discrepancies observed in the outer reflector layer;
- *Energy deposition model:* investigation of the possibility of adopting the same energy deposition model in both codes and evaluating whether differences in this aspect could lead to divergent isotopic concentration evolutions.

7.1.1. Self-shielding optimization

First, it was verified that the use of mixtures with the default self-shielding options does not affect the results[2] , as illustrated in Figure 7.3.

Therefore, even though Figure 7.1 presents an APOLLO3® configuration without mixtures (contrary to common practice), its results can still be considered reasonably accurate[3]; nevertheless, an attempt was made to implement mixtures by improving the original self-shielding options to those usually adopted for the verification and validation activities on APOLLO3®.

It has to be said that all APOLLO3® simulations presented in this work differed from the original CamiVVER self-shielding configuration, which consisted of:

- Self-shielding, for fissile layers, only the following isotopes were involved: ^{239}Pu, ^{240}Pu, ^{241}Pu, ^{242}Pu, ^{234}U, ^{235}U, ^{236}U, ^{238}U, and ^{90}Zr (in this exact order), and only ^{90}Zr for reflector layers. Thus, plutonium isotopes were treated before uranium isotopes, a non-optimal choice given uranium's higher abundance in the fuel;

[2]The simulation stops at 46 000 MWd/tHM because it reached the maximum allowed time limit on the CEA cluster Orcus (14 days).
[3]There is no reason to assume a divergent behavior beyond 46 000 MWd/tHM.

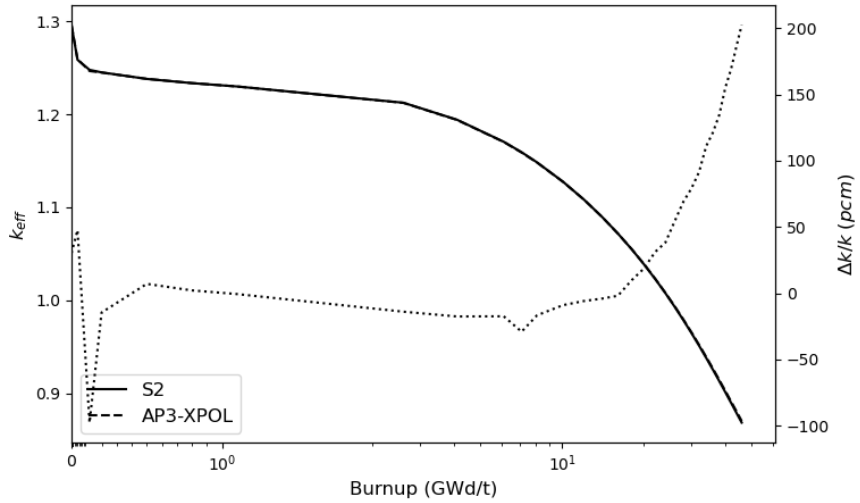


Figure 7.3: Partial (up to 46 000 MWd/tHM) depletion result of the APOLLO3® code without self-shielding mixtures benchmarked against Serpent.

- No isotopic mixtures were implemented;
- A slowing-down model set to ST ("STatistical") was used, which assumes a statistical distribution of narrow resonances within each group.

For this work, ^{238}U was prioritized, followed by plutonium isotopes, as uranium's greater atomic density outweighs the epithermal resonance importance of plutonium.

Then, from now on, the presented results will have an optimized mixture of the four main fissile isotopes: ^{238}U, ^{235}U, ^{239}Pu, ^{240}Pu, as recommended by the literature. Additional nuclides were included for self-shielding in grid and reflector layers: ^{50}Cr, ^{52}Cr, ^{53}Cr, ^{54}Cr, ^{117}Sn, ^{119}Sn, ^{54}Fe, ^{56}Fe, ^{57}Fe, and ^{58}Fe. The slowing-down model was switched to TR ("Toute Résonance") for the mixture, which accounts for all resonance shapes and is considered more accurate, albeit slower, than ST.

While these modifications offer a computational advantage, restricting the self-shielding options to validated energy ranges, following CEA internal validation protocols, resulted in significant time savings. Consequently, the overall cost of the depletion simulation was not substantially affected, remaining well within the 14 day limit.

7.1.2. Burnup mesh refinement

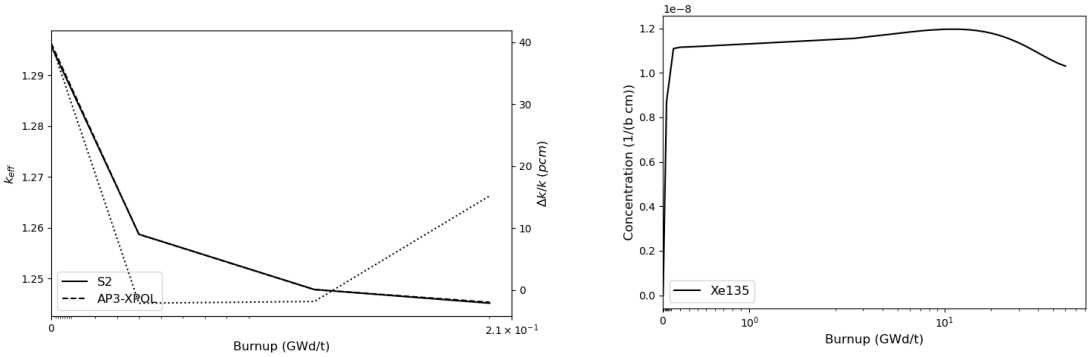
Due to the initial fluctuation noticed in Figure 7.1, an attempt to refine APOLLO3® was carried out. Considering the first three burnup steps (MWd/tHM):

$$0.0 \quad 40 \quad 120 \tag{7.1}$$

It was tried to add:

- 2 steps before 40 MWd/tHM, and 1 within 40 MWd/tHM and 120 MWd/tHM;
- 1 step before 40 MWd/tHM, and 2 within 40 MWd/tHM and 120 MWd/tHM;
- 1 step before 40 MWd/tHM;
- 1 step within 40 MWd/tHM and 120 MWd/tHM;
- 1 step before 40 MWd/tHM, and 1 within 40 MWd/tHM and 120 MWd/tHM;

The last option resulted yet in converged results, bringing the discrepancies with Serpent at 40 MWd/tHM and 120 MWd/tHM from 59.7 pcm and -86.3 pcm to just -2.2 pcm and -1.9 pcm respectively, as shown in Figure 7.4a. So, the final burnup mesh for APOLLO3® has also 20 MWd/tHM and 80 MWd/tHM as further burnup steps.



(a) APOLLO3® mesh refinement zoom (up to 200 MWd/tHM).

(b) ^{135}Xe concentration vs. burnup (2D fuel cell).

Figure 7.4: Analysis of the initial burnup phase: benchmark of the APOLLO3® mesh refinement and evolution of ^{135}Xe.

This is a quite interesting result because it shows that the two codes perform in a different way. This discrepancy is probably related to a different treatment of the xenon isotopes

which are really relevant at the very beginning of a depletion simulation due to their high absorption cross section for thermal neutrons.

For example, for ^{135}Xe, $\sigma_{(n,\gamma)} = 2.778\,\text{Mb}$ at $0.0253\,\text{eV}$ as reported by the Japan Atomic Energy Agency (JAEA). Indeed, even a small change in its composition leads to a steep change in the neutron flux. That is why, usually, the burnup meshes present a finer arrangement at the beginning to correctly capture the ^{135}Xe contribution that at low burnup reaches the equilibrium condition, as shown in Figure 7.4b. This result is obtained from a 2D fissile cell retrieved from the UOX assembly analyzed in this thesis with reflective boundary conditions. The grid was not modeled in this simplified test case. Definitively, if xenon is the primary driver of this discrepancy, it can be said that Serpent outperforms APOLLO3®. Anyway, this aspect was no further studied since the goal was just to find a convergent solution.

7.1.3. Reflector layer subdivision

A series of studies were conducted by varying the subdivision of both the inner and outer reflectors. Each reflector material layer was equally divided into either two or three segments, and the results are reported in Table 7.1.

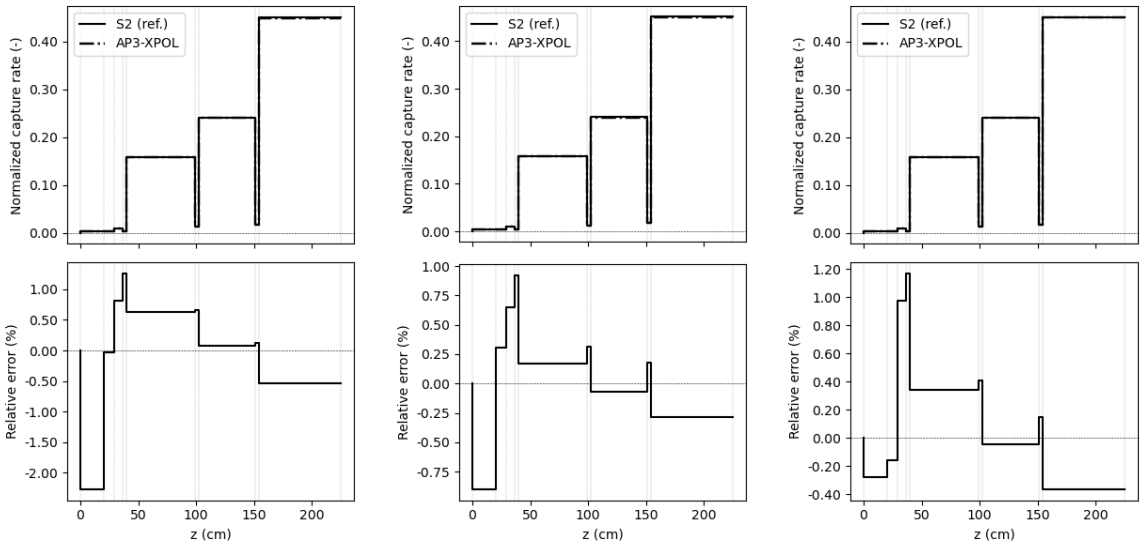
Table 7.1: Refinement of the axial reflector in APOLLO3® with respect to the CamiVVER material mesh choice (zero burnup).

No. of Layers			
Outer Refl.	**Inner Refl.**	k_{eff}	**Time**
1	1	1.296 15	9:59:00
1	2	1.296 24	15:59:16
1	3	1.296 21	11:29:00
2	1	1.296 22	11:04:27
2	2	1.296 25	16:35:20
3	1	1.296 29	1d 22:08:28
3	3	1.296 26	17:06:16

It should be noted that the "no-subdivision" case (first row in Table 7.1) does not employ the same radial mesh as in Figure 7.1 and Figure 7.3, but rather the one in Figure 6.2a. In the former cases, the guide tubes are modeled with only four sectors, an initial choice to reduce computational cost; however, this was later replaced with the Figure 6.2a configuration, because, as explained in Chapter 6, combining just four sectors with the sector

rotation option leads to mismatches between fuel cell and guide tube sectors at their interfaces. These results also incorporate the modifications described in Section 7.1.1.

From Table 7.1 and the integrated rate analysis, the "no-subdivision" case slightly reduces the relative difference with Serpent, lowering it from 39.4 pcm (in Figure 7.1) to 34.2 pcm. In contrast, reflector refinement slightly increases k_{eff} and thus the relative difference. This is probably due to the fact that the reflector is better modeled, which in turn affects the behavior of reflected neutrons and consequently impacts the fissile layers; nevertheless, integrated rate comparisons show a significant improvement when both reflectors are subdivided into two or three layers, as illustrated in Figure 7.5.



(a) Copy of Figure 7.2b. Reported here just for clarity.

(b) Reflectors divided into 2 equally spaced layers.

(c) Reflectors divided into 3 equally spaced layers.

Figure 7.5: Comparison between integrated capture rates of Serpent (reference) and APOLLO3® at zero burnup with different reflector discretizations.

Figure 7.5 clearly shows that refining both reflectors significantly reduces discrepancies with Serpent; however, subdividing only one reflector while leaving the other unchanged consistently led to worse results, regardless of the refinement level. The related graphs are not reported here, for the sake of conciseness. This trend suggests that refining one reflector influences the other, and convergence is achieved only when both are subdivided. This also indicates that previous results, both in this work and in the CamiVVER project, were likely not fully converged with respect to reflector meshing; however, the impact on k_{eff} remained small due to the much lower neutron importance in the reflector compared to fissile layers.

In conclusion, the subdivision into two layers was selected, as it yields a relative difference below 1 %, removing the reflector as the largest source of discrepancy. Although the three-layer option further improves the integrated rate agreement, the increase in computational time is not justified.

7.1.4. Energy deposition model

This topic warrants a dedicated section, as it represents a key turning point in this benchmark study. It is discussed in detail in Section 7.4.

7.2. Serpent

An initial attempt to improve Serpent results, compared to those in Chapter 6, involved changing one parameter. Although this adjustment introduces a slight bias with respect to Chapter 6, the overall reasoning and conclusions of that chapter remain valid. The parameter modified is

- `set ures 1 0.0`

This option enables sampling from the unresolved resonance probability tables regardless of the initial isotopic atomic fractions. By default, Serpent applies a threshold of 10^{-9}, as detailed in the Serpent Wiki, meaning that unresolved resonance treatment is only activated for nuclides with concentrations above this limit. Setting it to zero removes this threshold to ensure maximum theoretical accuracy.

It was verified that including or omitting this correction does not cause statistically significant differences, as shown in Figure 7.6. The simulation with `set ures 1 0.0` used 200 000 neutrons per batch and 3000 batches, yielding a statistical uncertainty of approximately $\sigma \simeq 4.1$ pcm.

In addition, integrated reaction rates were compared to ensure consistency with the results presented in Appendix A. Notably, TRIPOLI-4® does not apply a threshold analogous to Serpent's default unresolved resonance cutoff; therefore, using `set ures 1 0.0` in Serpent better aligns with TRIPOLI-4 treatment. The integrated reaction rates between TRIPOLI-4® and Serpent (with `set ures 1 0.0`) did not show statistically significant discrepancies. The new Serpent simulation was run with 100 000 neutrons per batch and 100 000 batches to reach the same precision as the simulations of Appendix A. The k_{eff} resulted in 1.295 73 with an uncertainty of 0.44 pcm, which is completely within statistical fluctuations with respect to the result in Table A.1.

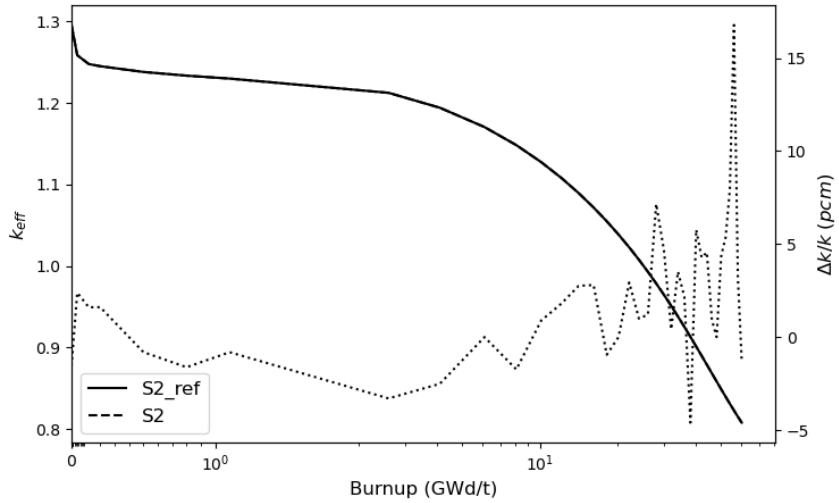


Figure 7.6: Comparison of the burnup mesh convergence Listing 6.2 between Serpent runs with `set ures 1` (reference) and `set ures 1 0.0`.

7.3. New burnup mesh

The subsequent 3D full assembly depletion simulations presented in this thesis use a slightly modified burnup mesh compared to the one defined in Listing 6.2:

Listing 7.1: Final burnup mesh for the UOX assembly (Serpent). Values are in MWd/tHM. A total of 38 steps.

```
bu_mesh = [0., 40., 120., 200., 500., 800., 1000., 2500., 4000., 5500., 7000.,
    8500., 10000., 12000., 14000., 16000., 18000., 20000., 22000., 24000.,
    26000., 28000., 30000., 32000., 34000., 36000., 38000., 40000., 42000.,
    44000., 46000., 48000., 50000., 52000., 54000., 56000., 58000., 60000.]
```

Basically, the +100 MWd/tHM incremental step beyond 20 000 MWd/tHM was removed in favor of an increasing step size, with the exception of a +200 MWd/tHM step between 800 MWd/tHM and 1000 MWd/tHM. This choice avoids breaking the previously established pattern in CamiVVER (since the two steps before 800 MWd/tHM were +300 MWd/tHM, thus a +100 MWd/tHM step here would have resulted in 10 100 MWd/tHM). The steps prior to 800 MWd/tHM were kept unchanged to maintain a strictly increasing step size without significantly altering the burnup mesh in Listing 6.2, thus preserving the validity of the convergence analysis from Chapter 6. Tests confirmed that removing the +100 MWd/tHM steps after 20 000 MWd/tHM does not affect convergence. Consequently, the definitive burnup mesh adopted for APOLLO3® is (with 20 MWd/tHM and 40 MWd/tHM):

Listing 7.2: Final burnup mesh for the UOX assembly (APOLLO3®). Values are in MWd/tHM. A total of 40 steps.

```
bu_mesh = [0., 20., 40., 80., 120., 200., 500., 800., 1000., 2500., 4000.,
    5500., 7000., 8500., 10000., 12000., 14000., 16000., 18000., 20000.,
    22000., 24000., 26000., 28000., 30000., 32000., 34000., 36000., 38000.,
    40000., 42000., 44000., 46000., 48000., 50000., 52000., 54000., 56000.,
    58000., 60000.]
```

Figure 7.7 presents the full depletion benchmark of APOLLO3® against Serpent incorporating all the corrections discussed in this chapter.

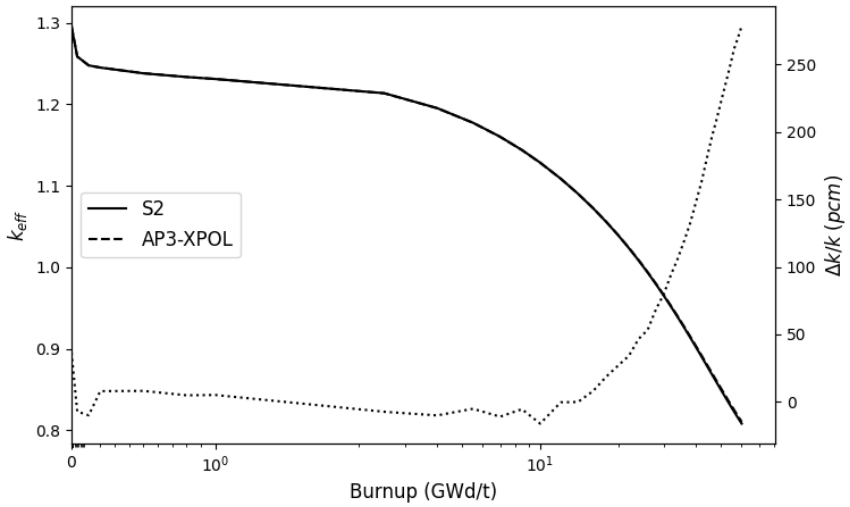


Figure 7.7: Full depletion benchmark of APOLLO3® against Serpent with the optimizations described in this chapter.

Since this represents the nearly final result, Serpent's statistical uncertainty from Monte Carlo sampling was further reduced by repeating the simulation ten times and averaging the outcomes. This approach, using different random seeds for each run, provides a way to control the statistical uncertainty originating from the Monte Carlo neutronics. Indeed, within a single simulation, each batch is not completely independent from the previous one. For example, if by statistical coincidence a cluster of neutrons forms near a fissile region in an isolated, uniform, 3D axially infinite fuel rod, this influences the next batch, which observes a higher neutron flux in that area and consequently a higher fission rate leading to a fictitious result. Conversely, the expected physical outcome is a uniform flux, given the axial infinity of the rod. This correlation between batches becomes even more significant in depletion calculations, where such errors can propagate over time,

potentially leading to completely incorrect results.

With respect to Figure 7.1, Figure 7.7 reports a clear improved result for the early burnup, while there is still an issue for the latter part of the depletion mesh that is addressed in the next section.

7.4. Decay chain and energy deposition model

As introduced at the beginning of this chapter, the discrepancy observed after 20 000 MWd/tHM in Figure 7.1, Figure 7.3, and Figure 7.7 may be attributed to the following factors:

- Differences between the decay chains adopted in APOLLO3® and Serpent;
- Differences in the energy deposition models implemented in the two codes.

A dedicated test was performed: a 2D cell was extracted from a fissile cell without grid from the 3D UOX assembly studied in this thesis[4]. The study is reported in Appendix B and shows that the primary cause of the disagreement is the difference in the energy deposition models between the two codes. Thus, the following results reported with Serpent have the `set U235H 193.7201` card and for APOLLO3® the results were retrieved with a code version that considers only local energy deposition for a fission event. This choice allows the two codes to treat the energy deposition similarly during a fission event.

Since the 2D simulation gave a real improvement, the whole assembly 3D test was carried out as shown in Figure 7.8. The outcome was obtained by averaging Serpent results over 20 independent runs (each with 4000 batches and 200 000 neutrons each and 300 inactive cycles, yielding $\sigma(k_{eff}) = 3.5$ pcm).

The comparison shows a generally low relative error, exceeding only 100 pcm between 20 000 MWd/tHM and 40 000 MWd/tHM with a peak of about 140 pcm at 30 000 MWd/tHM. Figure 7.9 shows the relative discrepancy on the radially integrated fission and capture reaction rates per unit length. Here, gray bars centered at zero relative error represent the 3σ Monte Carlo statistical absolute uncertainty. These slight change in graph representation (rates per unit length and the uncertainty bars) were adopted to present this work results in a conference paper [40] (with the same title of this thesis) for PHYSOR 2026, the International Conference on the Physics of Reactors, in April, in Turin. The largest relative errors occur in the first two active layers near the reflectors (the second being a grid) for both fission and capture, while the remaining layers exhibit much smaller discrepancies.

[4]The results presented in Figure 7.4b originate from this Serpent simulation.

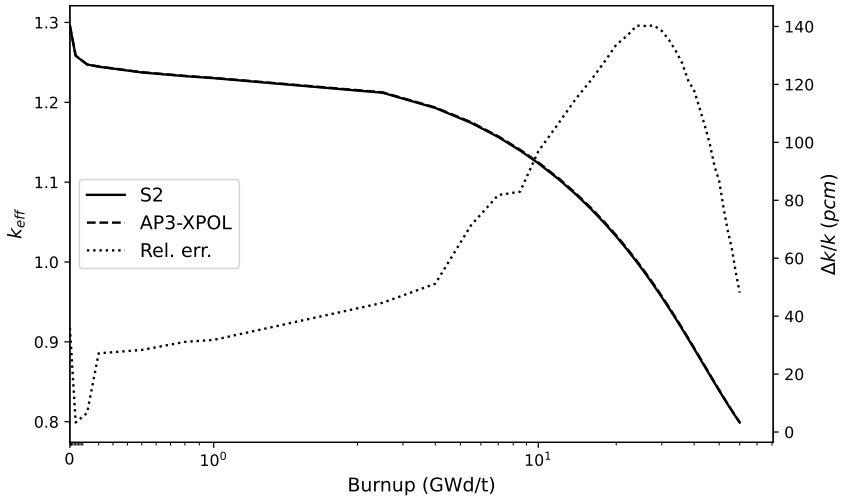
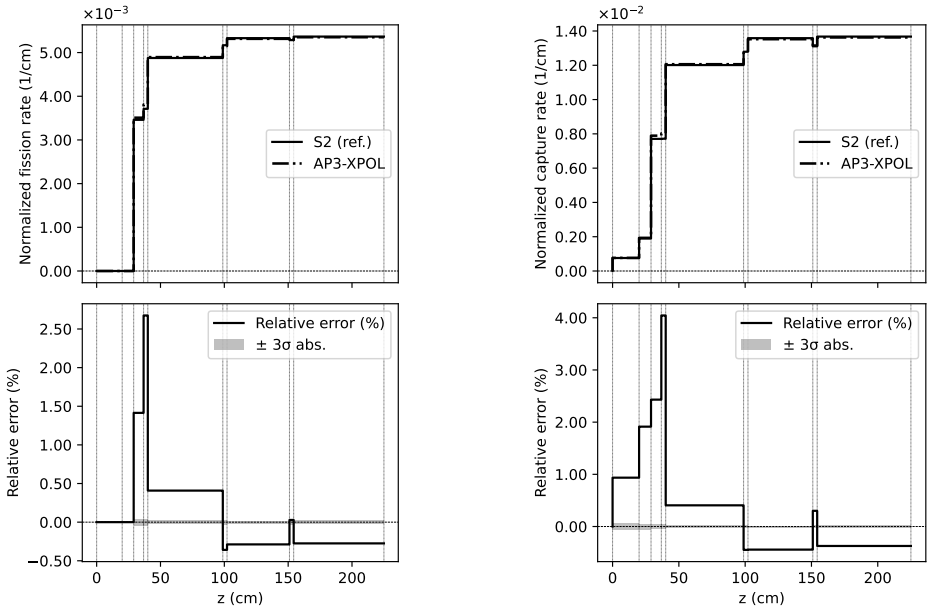


Figure 7.8: Full 3D UOX assembly depletion benchmark of APOLLO3® against Serpent (reference) with the correct energy deposition model.



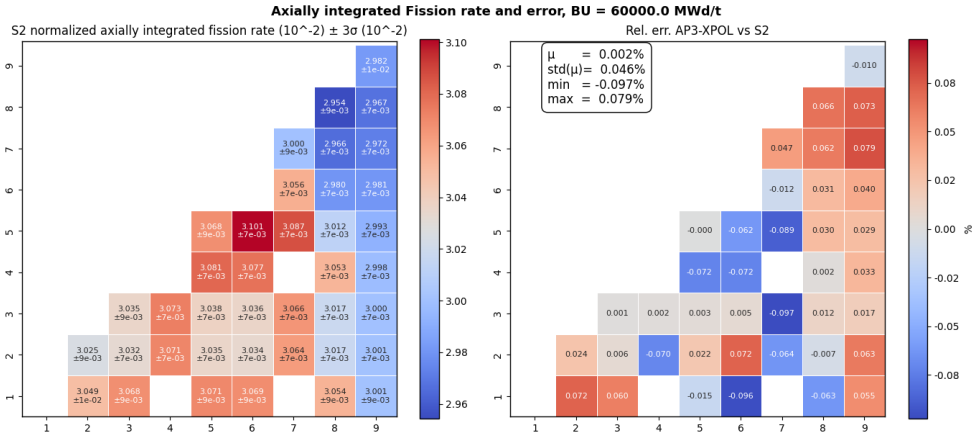
(a) Integrated fission rates.

(b) Integrated capture rates.

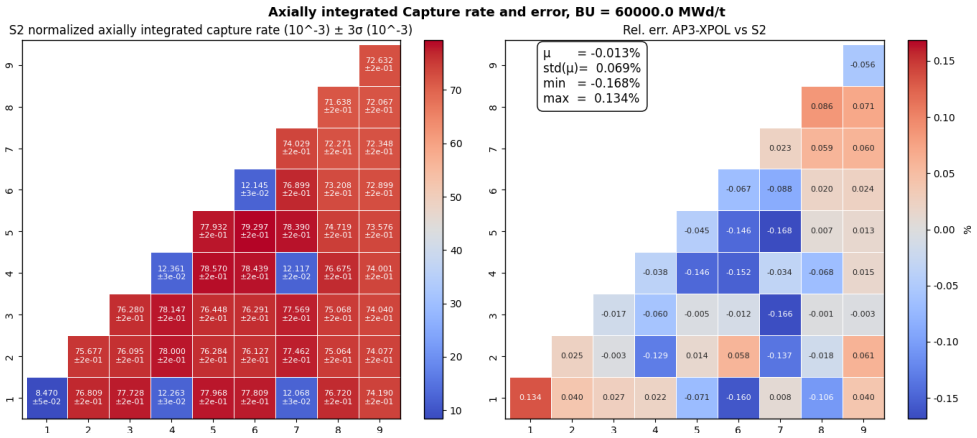
Figure 7.9: Comparison between integrated fission and capture rates of Serpent (reference) and APOLLO3® at 60 000 MWd/tHM with the new energy deposition model.

Pin-wise reactions rates were also analyzed in each layer, but they are not shown here for conciseness. The maximum discrepancies reached 2.9 % for fission and 4.8 % for capture in the first grid layer, consistent with what shown in Figure 7.9. Figure 7.10 reports

instead the pin-wise distribution of the relative error on the axially integrated[5] one-group reaction rates at 60 000 MWd/tHM: the maximum discrepancy is lower than 0.1 % for the fission rate and around 0.15 % for the capture one, while the mean discrepancy is ten times lower for the latter (for the former, it is almost null due to the renormalization explained in Appendix A). For both rates, the statistical uncertainty is $3\sigma \approx 0.2\,\%$.



(a) 2D pin-by-pin axially integrated fission rate.



(b) 2D pin-by-pin axially integrated capture rate.

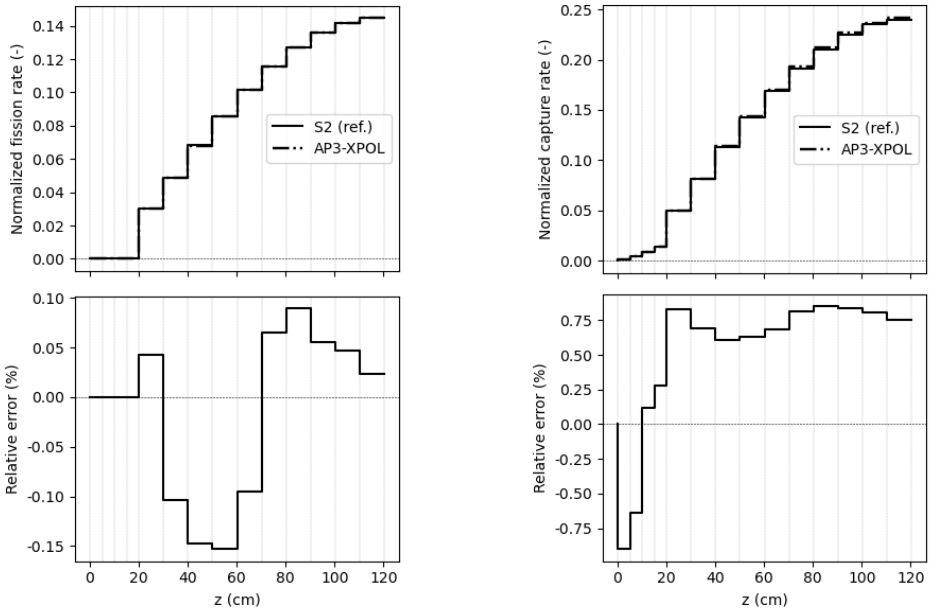
Figure 7.10: Comparison of 2D fission and capture axially integrated rates at 60 000 MWd/tHM.

It is clear that, with these final options, the benchmark of the TDT MOC-3D solver of APOLLO3® against Serpent for a UOX assembly can be considered completed. The last concern regards the remaining k_{eff} deviations that are most likely due to differences in evolution chains implemented in the two codes.

[5] Also the choice of axially integrating the 2D rates was made to present this work at PHYSOR 2026 as in a single graph the whole 3D geometry data are condensed.

7.5. Gadolinium assembly

The comparison between Serpent and APOLLO3® at zero burnup for the 36 UOX-Gd fuel assembly already shows a significant discrepancy in terms of k_{eff} (-467.2 pcm where Serpent (reference) gives 0.919 26 with an uncertainty of 4.1 pcm and APOLLO3® 0.914 96), that is reflected on the rates (Figure 7.11).

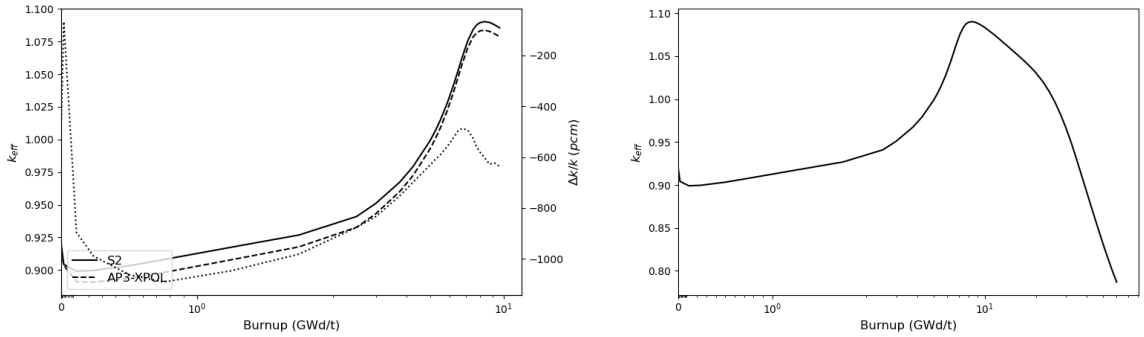


(a) Fission rate. (b) Capture rate.

Figure 7.11: Comparison between integrated fission and capture rates of Serpent (reference) and APOLLO3® at zero burnup for the 36 UOX-Gd fuel assembly.

This really significant difference at zero burnup leads to an even larger relative differences in depletion, as reported in Figure 7.12a. It has to be said that these graphs were retrieved prior to the change in energy deposition model of the two codes.

Furthermore, it can be noticed that the simulation finishes at 9625 MWd/tHM without reaching the normal end at 60 000 MWd/tHM. The problem is linked to the slowness of APOLLO3® that took 14 days (the time limit for Orcus HPC cluster at CEA) to arrive just to 9625 MWd/tHM. Just to have an idea, Figure 7.12b illustrates how Serpent models the full depletion simulation for the 36 UOX-Gd assembly.



(a) APOLLO3® vs. Serpent (up to 9625 MWd/tHM) for the 36 UOX-Gd case.

(b) Full depletion Serpent simulation (up to 60 000 MWd/tHM).

Figure 7.12: Depletion analysis of the 36 UOX-Gd assembly: benchmark comparison up to 9625 MWd/tHM (left) and full reference evolution performed by Serpent (right).

These problems lead to a change of strategy for this study and important choices were made:

- Stop the analysis of the results before 60 000 MWd/tHM for the two codes. This means that the precision of Serpent could be increased; in fact, the new simulations had been run with 200 000 neutrons per batch (as for the previous ones) and 4000 batches. The inactive cycles remain 300 and the uncertainty on k_{eff} becomes 3.54 pcm. The ideal case would be to end at 20 000 MWd/tHM because after that value, all the gadolinium quantity is practically consumed, so the assembly behaves like a UOX one, which was already benchmarked in Section 7.4;
- The 8-rod UOX-Gd assembly is directly analyzed since the results with the 36 one are too poor to expect a great improvement with the new energy deposition model. The hope is to reduce the discrepancies above all at zero burnup and also during depletion, and to reach 20 000 MWd/tHM by reducing the assembly complexity. Moreover, the new energy deposition model is implemented to obtain a better agreement between the two codes.

As already explained in the end of Chapter 6, this change of assembly does not need a new convergence analysis because, since it has less gadolinium, convergence is assured if the 36-rod one converged. In Figure 7.13 the benchmark between APOLLO3® and Serpent for the 8 UOX-Gd assembly is reported up to 5875 MWd/tHM, to better focus on the region with higher discrepancies. Serpent results are averages over 15 independent simulations. It can be noted that the highest relative difference (in absolute value) is at 750 MWd/tHM with a value of -753 pcm. Conversely, in Figure 7.12a, the highest

discrepancy is still at the same step, but it is about -1100 pcm. Furthermore, it can be noted that there is a strong fluctuation concerning the first three burnup steps for both graphs: the second relative error is considerably higher than the first and third ones, while after the first three, the trend seems not fluctuating.

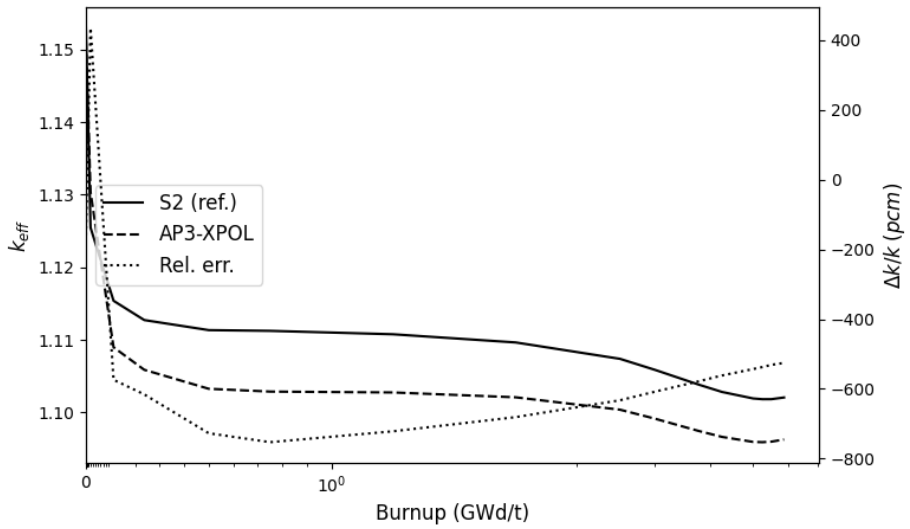
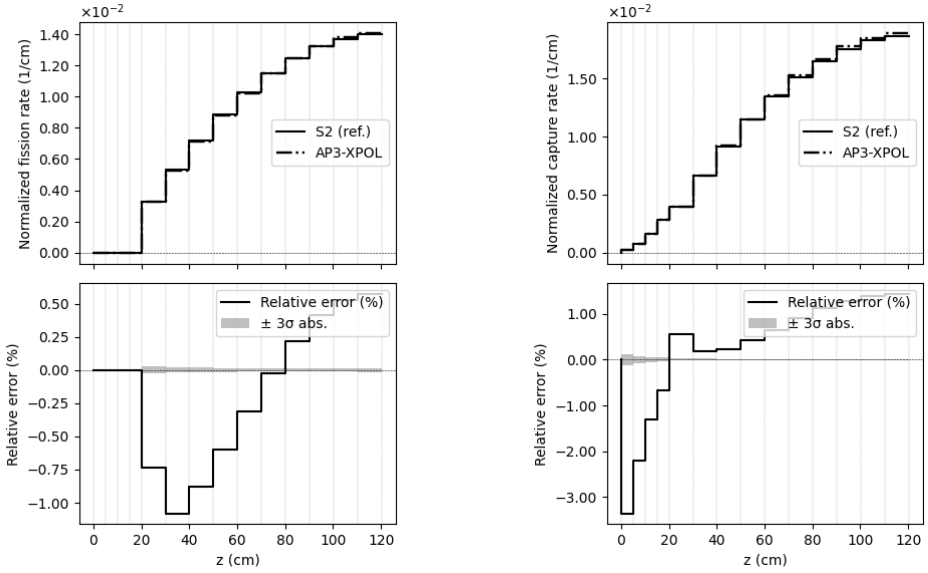


Figure 7.13: APOLLO3® vs. Serpent (up to 5875 MWd/tHM) for the 8 UOX-Gd case with the new energy deposition model.



(a) Fission rate.

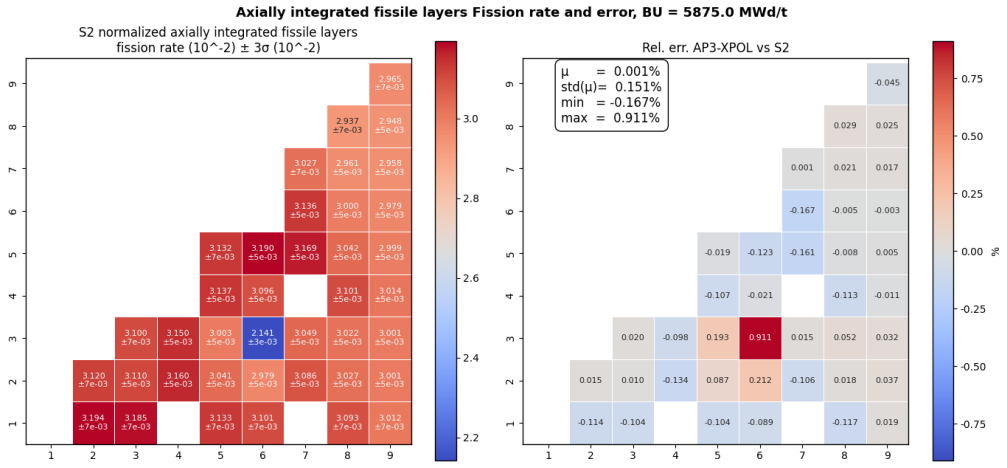
(b) Capture rate.

Figure 7.14: Comparison between integrated fission and capture rates of Serpent (reference) and APOLLO3® at 5875 MWd/tHM for the 8 UOX-Gd fuel assembly.

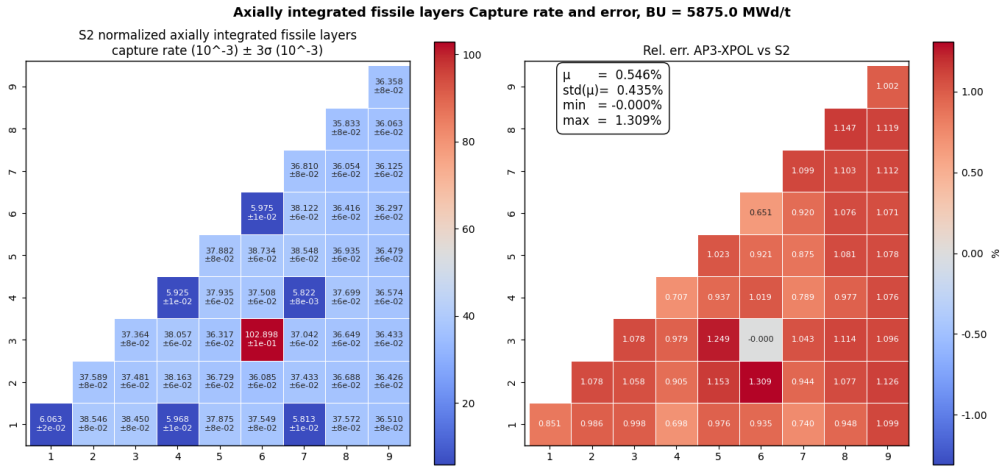
Moreover, Figure 7.14 illustrates the relative discrepancy on the radially integrated fission and capture reaction rates per unit length. The outermost reflector mesh shows a relative error $> 3\,\%$ (in absolute value), but the ones close to the core can be considered satisfactory. Besides, absorption and fission in the fissile region are also acceptable, with just one layer that has values slightly greater than $1\,\%$.

Additionally, Figure 7.15 shows the pin-wise distribution of the relative error on the axially integrated reaction rates at 5875 MWd/tHM. It is clear that the cell with the UOX-Gd rod is the one with a lower fission rate and a higher capture one with respect to the other fissile cells, due to the presence of gadolinium. It is also the one with different relative error trends: the fission discrepancy is clearly higher than the other rods and it can be noted that the nearby cells seem to be affected by it, showing a slightly higher error than the average. Regarding capture, in the figure it seems to have a zero discrepancy but it has a slightly negative value while the other cells show a positive error.

It is clear that, the implementation of the new energy deposition model and decreasing the quantity of gadolinium in the core from 36 to 8 UOX-Gd rods increases the agreement between APOLLO3® and Serpent; nevertheless, the discrepancies are still significant (above all in the UOX-Gd cells), thus the implementation of the LS method in the radial plane is needed.



(a) 2D pin-by-pin axially integrated fission rate.



(b) 2D pin-by-pin axially integrated capture rate.

Figure 7.15: Comparison of 2D fission and capture axially integrated rates at 5875 MWd/tHM for the 8 UOX-Gd assembly.

8 | Conclusions and perspectives

The eventual transition to high-performance, one-step core calculations necessitates ongoing improvements of deterministic neutron transport solvers. The recent evolution of the TDT (Two-Three Dimensional Transport) solver within the APOLLO3® framework represents a significant milestone toward this goal. The Method of Characteristics (MOC) was extended to three dimensions and implemented axial polynomial expansions for both neutron flux and macroscopic cross sections, reducing computational cost while preserving accuracy with respect to more traditional approaches.

This master's thesis is positioned as a continuation of the CamiVVER project that had the objective to benchmark the TDT MOC-3D solver of APOLLO3® against TRIPOLI-4® for a standard PWR UOX assembly. For this reason, this work aims to improve the results by benchmarking APOLLO3® against Serpent that has better depletion capabilities than TRIPOLI-4®. Furthermore, this work was extended to assemblies with 8 and 36 UOX-Gd rods in order to assess the robustness of APOLLO3® regarding gadolinium concentration.

For the UOX assembly study, the result revealed a good agreement between the two codes: the discrepancy in k_{eff} is very small throughout depletion, with a peak of about 140 pcm at 30 000 MWd/tHM. The relative errors on one-group fission and capture rates are excellent as well, remaining within 0.5 % in the most important axial layers, when considering the radially integrated rates, and even below 0.15 % when considering the axially integrated ones. The remaining k_{eff} deviations are most likely due to differences in depletion chains implemented in the two codes; a non-negligible difference was found in the ^{148}Nd concentration, a typical burnup indicator, during depletion for a simplified 2D single fissile cell test case. This discrepancy indicates that there could be the buildup, and later the consumption, of different isotopic amounts. There is also a minor error due to the fact that Serpent cross sections are at 600 K thus an interpolation is needed to study the system at 624 K.

Regarding the gadolinium cases, the poor results revealed the need for the Linear Surface MOC method on the radial plane due to the high neutron capture cross section of the gadolinium isotopes that causes steep flux gradients which are not well represented by the solver. Furthermore, APOLLO3® necessitates a significant amount of time to

process the calculations, preventing the calculation from reaching 60 000 MWd/tHM. Indeed, the comparison between the two codes for the 36 UOX-Gd case arrived at about 10 000 MWd/tHM and shows peaks of relative difference in terms of k_{eff} greater than 1000 pcm, in absolute value. In contrast, for the 8 UOX-Gd assembly, the k_{eff} discrepancy drops with values not greater than -753 pcm. Clearly, higher gadolinium content reduces APOLLO3® precision, making the LS method essential.

Regarding future perspectives, for the UOX case, a further subdivision of the first two fissile layers near the reflectors in APOLLO3® could be explored to reduce the discrepancies observed in the integrated fission and capture rates. Furthermore, the differences in the depletion chains between the two codes should be investigated to assess their impact and determine if the discrepancies can be further reduced. Regarding the gadolinium cases, the focus should be on implementing the LS method within the APOLLO3® framework, both to improve agreement and to reduce computational costs.

In addition, a further benchmark would involve the study of a MOX assembly. Testing the ability of the APOLLO3® TDT MOC-3D solver to model MOX assemblies is essential, as recycling spent nuclear fuel to partially close the nuclear fuel cycle has become a topic of primary interest in the industrial fission reactor field. An analysis varying the MOX fuel quantity, similar to the UOX-Gd approach, could also be investigated. A positive benchmark would push towards a better modeling of MOX assemblies that have a well-known peculiar bias for the axial power distribution near reflector layers [41]. Indeed, deterministic codes fail to accurately estimate the local overpower phenomenon, resulting in a dangerous underestimation of power peaks. Utilizing an axial expansion for both neutron flux and macroscopic cross section is likely to diminish this bias.

Finally, a study adopting different isothermal temperatures, or a more realistic distribution within the assembly materials, could be a valuable further test. The former would assess the agreement between the two codes in cases where APOLLO3® interpolates cross sections while Serpent does not (for example, by imposing 600 K for the whole assembly); this would avoid Serpent interpolation errors encountered in this work. The latter could impose, for instance, 900 K for the fuel and 600 K for other materials, thereby resembling a more realistic scenario.

Once the APOLLO3® TDT MOC-3D solver has been successfully benchmarked, multiparametric libraries for macroscopic cross sections at the lattice level could be generated. This step represents the fundamental link between high-fidelity assembly calculations and the core solvers employed in the EDF industrial code suite.

The findings of this work are summarized in a conference paper [40] that will be presented in PHYSOR 2026, the International Conference on the Physics of Reactors, in April, in Turin.

Bibliography

[1] P. Mosca, L. Bourhrara, A. Calloo, A. Gammicchia, F. Goubioud, L. Mao, F. Madiot, F. Malouch, E. Masiello, F. Moreau, S. Santandrea, D. Sciannandrone, I. Zmijarevic, E. Garcia-Cervantes, G. Valocchi, J.-F. Vidal, F. Damian, P. Laurent, A. Willien, and B. Vezzoni. "APOLLO3®: Overview of the New Code Capabilities for Reactor Physics Analysis." *Nuclear Science and Engineering*, **volume 199**, pp. 1–14 (2024).

[2] S. Santandrea, D. Sciannandrone, R. Sanchez, L. Mao, and L. Graziano. "A Neutron Transport Characteristics Method for 3D Axially Extruded Geometries Coupled with a Fine Group Self-Shielding Environment." *Nuclear Science and Engineering*, **volume 186**(3), pp. 239–276 (2017). URL https://cea.hal.science/cea-02381240.

[3] L. Graziano. *An axial polynomial expansion and acceleration of the characteristics method for the solution of the Neutron Transport Equation.* Ph.D. Thesis, Université Paris Saclay (COmUE) (2018). URL https://theses.hal.science/tel-01923317.

[4] S. Santandrea, L. Graziano, and D. Sciannandrone. "Accelerated polynomial axial expansions for full 3D neutron transport MOC in the APOLLO3® code system as applied to the ASTRID fast breeder reactor." *Annals of Nuclear Energy*, **volume 113**, pp. 194–236 (2018). URL https://www.sciencedirect.com/science/article/pii/S0306454917302402.

[5] A. Gammicchia. *Development and acceleration of a 3D characteristics method including an axial polynomial expansion of cross sections.* Ph.D. Thesis, Université Paris-Saclay (2021). URL https://theses.hal.science/tel-03406149.

[6] S. Santandrea, M. Tiberga, E. Noblat, L. Graziano, B. Vezzoni, and A. Brighenti. "CamiVVER - D4.6 – Assessment of new advanced 2D and 3D models in neutronic multi-parameter library generation." Deliverable D4.6 (31/08/2023). URL https://www.camivver-h2020.eu/results.html.

[7] E. Brun, F. Damian, C. Diop, E. Dumonteil, F. Hugot, C. Jouanne, Y. Lee, F. Malvagi, A. Mazzolo, O. Petit, J. Trama, T. Visonneau, and A. Zoia. "TRIPOLI-4®, CEA, EDF and AREVA reference Monte Carlo code." *Annals of Nuclear Energy*,

volume 82, pp. 151–160 (2015). URL https://www.sciencedirect.com/science/article/pii/S0306454914003843. Joint International Conference on Supercomputing in Nuclear Applications and Monte Carlo 2013, SNA + MC 2013.

[8] J. Leppänen, V. Valtavirta, A. Rintala, and R. Tuominen. "Status of Serpent Monte Carlo code in 2024." *EPJ - Nuclear Sciences & Technologies*, **volume 11** (2025). URL https://www.sciencedirect.com/science/article/pii/S2491929225000032.

[9] R. Vuiart, A. Eustache, S. Eveillard, and G. Prulhière. "PRATIC: A soluble-boron-free, pressurized water cooled, SMR core benchmark." *EPJ - Nuclear Sciences & Technologies*, **volume 10** (2024). URL https://www.sciencedirect.com/science/article/pii/S2491929224000256.

[10] J. J. Duderstadt and L. J. Hamilton. *Nuclear Reactor Analysis*. John Wiley & Sons, New York (1976).

[11] G. I. Bell and S. Glasstone. *Nuclear Reactor Theory*. Van Nostrand Reinhold Ltd., New York (1970).

[12] H. Bateman. "On the numerical solution of linear integral equation." *Proceedings of the Royal Society of London Series A, Containing Papers of a Mathematical and Physical Character*, **volume 100**(705), pp. 441–449 (1922). URL https://doi.org/10.1098/rspa.1922.0009.

[13] A. Isotalo and P. Aarnio. "Comparison of depletion algorithms for large systems of nuclides." *Annals of Nuclear Energy*, **volume 38**(2), pp. 261–268 (2011). URL https://www.sciencedirect.com/science/article/pii/S0306454910003889.

[14] M. Pusa and J. Leppänen. "Computing the Matrix Exponential in Burnup Calculations." *Nuclear Science and Engineering*, **volume 164**(2), pp. 140–150 (2010). URL https://doi.org/10.13182/NSE09-14.

[15] D. J. Rose, R. E. Tarjan, and G. S. Lueker. "Algorithmic Aspects of Vertex Elimination on Graphs." *SIAM Journal on Computing*, **volume 5**(2), pp. 266–283 (1976). URL https://doi.org/10.1137/0205021.

[16] R. E. Tarjan, J. R. Bunch, and D. J. Rose. "Graph Theory and Gaussian Elimination." In *Sparse Matrix Computations*, pp. 3–22. Academic Press (1976). URL https://www.sciencedirect.com/science/article/pii/B9780121410506500064.

[17] A. Isotalo and P. Aarnio. "Substep methods for burnup calculations with Bateman solutions." *Annals of Nuclear Energy*, **volume 38**(11), pp. 2509–2514 (2011). URL https://www.sciencedirect.com/science/article/pii/S0306454911002726.

[18] A. Isotalo. *Computational methods for burnup calculations with Monte Carlo neutronics*. Ph.D. Thesis, Väitöskirja, Helsinki (2013). URL https://www.finna.fi/Record/jykdok.1435916.

[19] A. Isotalo and P. Aarnio. "Higher order methods for burnup calculations with Bateman solutions." *Annals of Nuclear Energy*, **volume 38**(9), pp. 1987–1995 (2011). URL https://www.sciencedirect.com/science/article/pii/S0306454911001708.

[20] P. Cosgrove, E. Shwageraus, and G. Parks. "A simple implicit coupling scheme for Monte Carlo neutronics and isotopic depletion." *Annals of Nuclear Energy*, **volume 141**, p. 107374 (2020). URL https://www.sciencedirect.com/science/article/pii/S0306454920300724.

[21] P. Cosgrove and N. Adamowicz. "Stability Analysis of Spatial Discretisation in Burn-Up Calculations." In *Proceedings of the International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, M and C 2021*, p. 2104 – 2113. American Nuclear Society (2021). URL https://www.scopus.com/inward/record.uri?eid=2-s2.0-85183598364&doi=10.13182%2fM%26C21-33714&partnerID=40&md5=98aa9327b0940b2dfcfdd9b24a2086ad.

[22] P. Cosgrove and E. Shwageraus. "Stability analysis of higher-order neutronics-depletion coupling schemes and Bateman operators." *Journal of Computational Physics*, **volume 448**, p. 110702 (2022). URL https://www.sciencedirect.com/science/article/pii/S0021999121005970.

[23] P. Cosgrove, E. Shwageraus, and G. Parks. "Stability analysis of predictor-corrector schemes for coupling neutronics and depletion." *Annals of Nuclear Energy*, **volume 149**, p. 107781 (2020). URL https://www.sciencedirect.com/science/article/pii/S0306454920304795.

[24] CEA. *Neutronics*. Le Moniteur Editions, Paris (2013).

[25] R. Lenain, E. Masiello, F. Damian, and R. Sanchez. "Domain-decomposition method for 2D and 3D transport calculations using hybrid MPI/OpenMP parallelism." In *Mathematics and Computations, Supercomputing in Nuclear Applications*

and Monte Carlo International Conference (M&C+SNA+MC 2015). Nashville, Tennessee (2015).

[26] S. Lahaye, A. Tsilanizara, P. Bellier, and T. Bittar. "Implementation of a CRAM solver in MENDEL Depletion Code System." In *The International Topical Meeting on Mathematics and Computation, Supercomputing, Reactor Physics and Nuclear and Biological Applications (M&C)*. Korean Nuclear Society - KNS (2017).

[27] N. Hfaiedh and A. Santamarina. "Determination of the optimized SHEM mesh for neutron transport calculations." In *The International Topical Meeting on Mathematics and Computation, Supercomputing, Reactor Physics and Nuclear and Biological Applications (M&C), Avignon (France)* (2005).

[28] J. Leppänen. *Development of a New Monte Carlo Reactor Physics code: Dissertation*. Ph.D. Thesis, Aalto University, Finland (2007).

[29] E. R. Woodcock, T. Murphy, P. J. Hemmings, and T. C. Longworth. "Techniques Used in the GEM Code for Monte Carlo Neutronics Calculations in Reactors and Other Systems of Complex Geometry." Technical Report ANL-7050, Argonne National Laboratory, Argonne, IL (1965).

[30] F. Malvagi, E. Dumonteil, and F.-X. Hugot. "Les bonnes pratiques dans les calculs critiques en Monte Carlo." Rapport technique SERMA/LTSD/RT/12-5330, Service d'Études des Réacteurs et de Mathématiques Appliquées (SERMA), CEA (2012).

[31] A. Santamarina, D. Bernard, P. Blaise, M. Coste, A. Courcelle, T. Huynh, C. Jouanne, P. Leconte, O. Litaize, S. Mengelle, G. Noguere, J. Ruggieri, O. Serot, J. Tommasi, C. Vaglio-Gaudard, and J.-F. Vidal. "The JEFF-3.1.1 Nuclear Data Library. Validation Results from JEF-2.2 to JEFF-3.1.1." *OECD 2009 NEA No 6807, ISBN 978-92-64-99074-6* (2009).

[32] T. Viitanen. *Doppler-Broadening Preprocessor for Monte Carlo Reactor Physics Code Serpent*. Master's thesis, Teknillinen korkeakoulu (Helsinki University of Technology), Espoo, Finland (2009). URL https://urn.fi/URN:NBN:fi:aalto-2020120555739.

[33] J. Solbrig, A. W. "Doppler Effect in Neutron Absorption Resonances." *American Journal of Physics*, **volume 29**(4), pp. 257–261 (1961). URL https://doi.org/10.1119/1.1937737.

[34] T. Viitanen and J. L. and. "Explicit Treatment of Thermal Motion in Continuous-Energy Monte Carlo Tracking Routines." *Nuclear Science and Engineering*, **volume 171**(2), pp. 165–173 (2012). URL https://doi.org/10.13182/NSE11-36.

[35] T. Viitanen and J. Leppänen. "New Interpolation Capabilities for Thermal Scattering Data in Serpent 2." In *PHYSOR 2016: Unifying Theory and Experiments in the 21st Century*. American Nuclear Society (ANS), United States (1 -5 May 2016).

[36] J. Leppänen and A. Isotalo. "Burnup calculation methodology in the serpent 2 Monte Carlo code." In *International Conference on the Physics of Reactors 2012, PHYSOR 2012*, volume 2, pp. 924–935. American Nuclear Society (ANS), United States (15-20 April 2012).

[37] R. Brun and F. Rademakers. "ROOT — An object oriented data analysis framework." *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, **volume 389**(1), pp. 81–86 (1997). URL https://www.sciencedirect.com/science/article/pii/S016890029700048X. New Computing Techniques in Physics Research V.

[38] M. Tiberga and S. Santandrea. "A Novel High-Order Surface Characteristics Scheme for the Neutron Transport Equation in 2D Unstructured Meshes." *Nuclear Science and Engineering*, pp. 1–45 (2023). URL https://cea.hal.science/cea-04477378.

[39] A. Le Bars. *Surface characteristics scheme for the neutron transport equation in extruded and non-conformal 3D geometries*. Ph.D. Thesis, Université Paris-Saclay (2024). URL https://theses.hal.science/tel-04902147.

[40] M. Montanari, M. Tiberga, S. Santandrea, A. Gammicchia, B. Vezzoni, and S. Lorenzi. "Benchmark of the TDT MOC 3D solver of APOLLO3® against Serpent during depletion." In *Proceedings of the International Conference on the Physics of Reactors (PHYSOR 2026)*. Turin, Italy (2026).

[41] A. Rispo. *Modélisation neutronique avancée des interfaces avec la chaîne SCIENCE V2 pour la remontée axiale du flux dans les réacteurs REP*. Ph.D. Thesis, Université Paris-Saclay (2024). URL https://theses.hal.science/tel-04662600.

[42] M. Tiberga, P. Laurent, A. Willien, N. Guler, B. Vezzoni, A. Brighenti, and G. H. Zavala. "CamiVVER - D4.4 – Results of the verification phases on PWR and VVER geometry configurations." Deliverable D4.4 (31/08/2023). URL https://www.camivver-h2020.eu/results.html.

[43] V. Salino and IRSN. "PyNjoy2016: ACE files generated from JEFF-3.1.1 and JEFF-3.3 using the PURR module." https://github.com/IRSN/PyNjoy2016/releases/tag/ACE_JEFF-3.x (2023). Release of ACE-format cross section libraries for SERPENT2 with advanced energy deposition model support.

[44] R. J. Moffat. "Describing the uncertainties in experimental results." *Experimental Thermal and Fluid Science*, **volume 1**(1), pp. 3–17 (1988). URL https://www.sciencedirect.com/science/article/pii/089417778890043X.

[45] T. Sekine and H. Baba. "Reactor-neutron-capture cross sections of 165Dy isomers." *Journal of Inorganic and Nuclear Chemistry*, **volume 43**(6), pp. 1107–1113 (1981). URL https://www.sciencedirect.com/science/article/pii/0022190281800012.

[46] J.-F. Vidal and D. Raynaud. "Combining the Tone's and subgroup models for reference LWR self-shielding calculation in APOLLO3." In *Proceedings of the 4th International Conference on Physics and Technology of Reactors and Applications (PHYTRA4)*. Marrakech, Morocco (2018).

[47] R. Tuominen, V. Valtavirta, and J. Leppänen. "New energy deposition treatment in the Serpent 2 Monte Carlo transport code." *Annals of Nuclear Energy*, **volume 129**, pp. 224–232 (2019). URL https://www.sciencedirect.com/science/article/pii/S0306454919300726.

A | TRIPOLI-4® - Serpent zero-burnup mismatch

All calculations presented in this chapter were performed using version 4.12.1 (December 2024) of TRIPOLI-4®.

The neutron population used for criticality assessments was determined based on the number of neutrons per batch, as already addressed in Section 4.3.1. According to best practices outlined in [30], a population of 10^4 neutrons per batch is generally sufficient for assembly-level simulations; however, a higher value of 10^5 neutrons per batch was selected in this study. This increase was found to be computationally manageable and beneficial for statistical accuracy, especially considering the fuel column is subdivided into four concentric regions. A larger neutron population ensures adequate sampling even in such small geometrical regions. The number of batches was set to 10^5 to achieve a statistical uncertainty of approximately 1 pcm on the effective multiplication factor k_{eff}, as can be estimated with Equation (4.18). To ensure convergence of the neutron source distribution, 300 inactive cycles were used, well above the 100 cycles recommended in [30]. This conservative choice was motivated by the nature of the initial batch, which assumes a uniform neutron distribution across the entire system. As a result, neutrons may initially appear to be born in non-fissile regions such as reflectors, water, or cladding. The early cycles are therefore essential to allow the system to evolve toward a physically meaningful source distribution. Since the computational cost difference between 100 and 300 inactive cycles is negligible, the more robust configuration was adopted, as illustrated in Figure A.1. The simulation configuration described here corresponds to the case reported in Table A.1.

A known discrepancy between TRIPOLI-4® and Serpent, previously reported in [42], was also observed in this work. The values obtained are shown in Table A.1, and consistent with the findings in [42]. Theoretically, this discrepancy should not occur since different neutronics Monte Carlo codes should give the same results, starting from the same nuclear data libraries, within a certain uncertainty, which does not occur here, as reported in Table A.1. It presents the values of k_{eff} and their uncertainty (σ_{T4}) obtained with TRIPOLI-4®, as well as the corresponding uncertainty (σ_{S2})[1] for Serpent.

[1]The 2 in S2 stands for the fact that the Serpent version in use is the **2**.2.1 one.

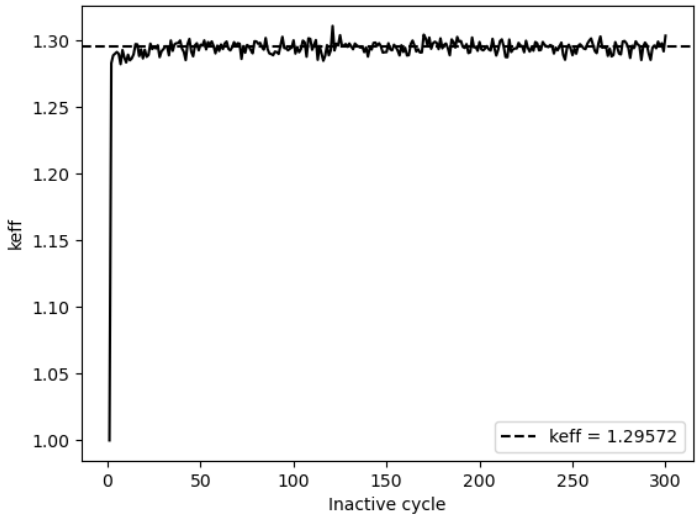


Figure A.1: Convergence of k_{eff} with respect to the number of inactive cycles. The simulation here corresponds to the result presented in Table A.1.

Table A.1: TRIPOLI-4® - Serpent zero-burnup k_{eff}. All the σ results are in pcm.

TRIPOLI-4®	$\sigma_{\mathbf{T4}}$	**Serpent**	$\sigma_{\mathbf{S2}}$	Δ_k	ε_k	ε_ρ
1.297 27	1.2	1.295 72	0.44	154.8	119.5	92.1

For Δ_k, ε_k and ε_ρ, Equation (A.1), Equation (6.1) and Equation (A.2) were used, where k_{eff}^{ref} is the TRIPOLI-4® one and k_{eff} the Serpent one. This means that TRIPOLI-4®'s k_{eff} is consistently higher.

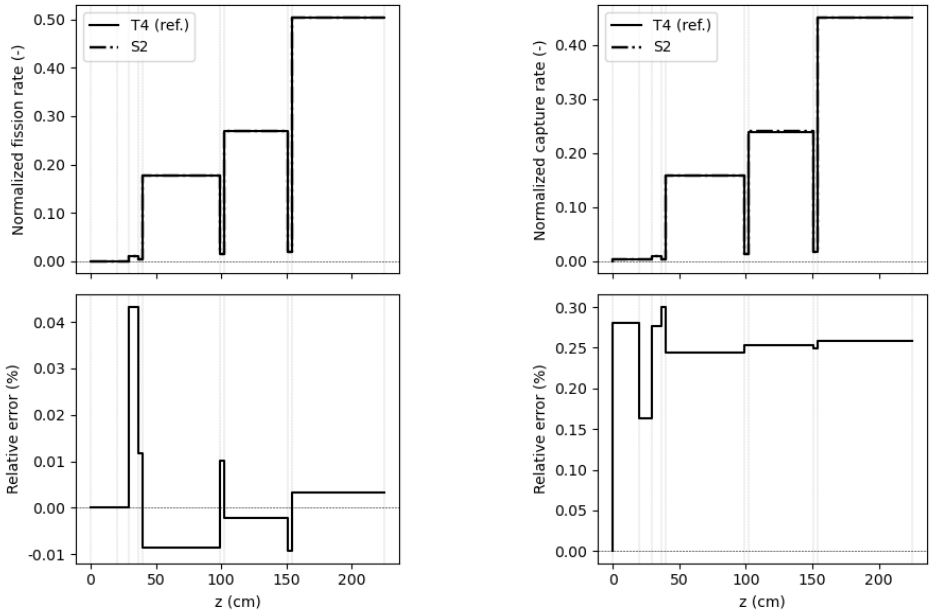
$$\Delta_k = k_{eff} - k_{eff}^{ref}, \tag{A.1}$$

$$\varepsilon_\rho = \frac{1}{k_{eff}^{ref}} - \frac{1}{k_{eff}} = \frac{k_{eff} - k_{eff}^{ref}}{k_{eff}^{ref} \times k_{eff}}. \tag{A.2}$$

The relative differences between the integrated fission and capture rates across the nine material layers were analyzed. To ensure consistency, these rates were normalized with respect to the total fission rate per burnup step over the entire 3D domain. Specifically, for each burnup step, the total fission rate of the fuel assembly was computed independently for each code, and both the fission and capture rates were divided by this value. As a consequence of this normalization, the relative difference between the integrated fission rates over the full 3D domain is, by definition, always zero; however, when the same normalization is applied to the capture rates, the resulting relative difference reflects approximately reflects the discrepancy in k_{eff}, since k_{eff} is an integrated quantity

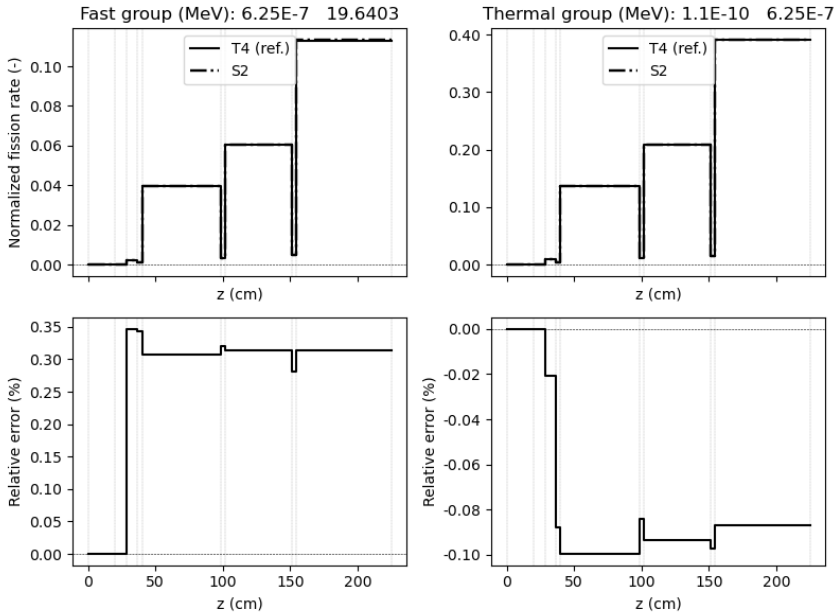
that depends on both neutron production and absorption. It should be noted that this normalization approach does not yield exactly the same result as Equation (6.1), because neutron leakage is also implicitly included in the capture rate normalization. Moreover, the difference between the k_{eff} relative difference and the 3D integrated capture rate discrepancy cannot be directly attributed to neutron leakage alone. This is only valid if both codes treat all other aspects identically, namely neutron production from fission and (n,xn) reactions; therefore, the integrated capture rate discrepancy was used as an indicator to identify potential inconsistencies or modeling issues between the two codes. This approach yielded a discrepancy of 259.9 pcm in the simulations presented in Table A.1.

The comparison of integrated fission and capture rates over the nine material layers indicates that Serpent systematically overestimates the capture rates relative to TRIPOLI-4®, as shown in Figure A.2.

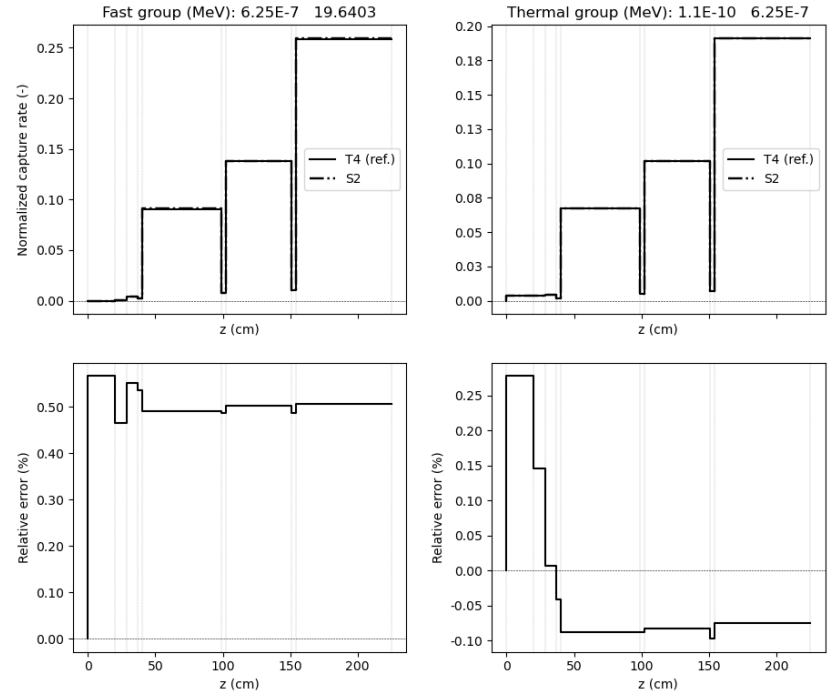


(a) Fission rate (b) Capture rate

Figure A.2: Comparison between TRIPOLI-4® (reference) and Serpent integrated fission and capture rates at zero burnup.



(a) Integrated fission rate



(b) Integrated capture rate

Figure A.3: Comparison between TRIPOLI-4® (reference) and Serpent integrated fission and capture rates at zero burnup with two anergy groups.

To investigate whether the discrepancy in capture rates was linked to a specific neutron energy range, a two-group analysis was carried out, as shown in Figure A.3. The energy groups used in this analysis correspond to thermal and fast neutrons, with boundaries defined as $[1.1 \times 10^{-10},\ 6.25 \times 10^{-7},\ 19.6403]$ MeV. It is important to say that all one-group reaction rates were calculated over the energy range $[1.1 \times 10^{-10},\ 19.6403]$ MeV. In contrast, the CamiVVER study used a slightly different range of $[1 \times 10^{-10},\ 19.6403]$ MeV. This adjustment was made to align with the energy range used in APOLLO3®, which starts at 1.1×10^{-10} MeV. As a result, the TRIPOLI-4® simulations in CamiVVER introduced a small bias due to this discrepancy that was corrected in the present master's thesis.

From Figure A.3, it can be observed that the fast energy group is primarily responsible for the capture rate behavior seen in Figure A.2. In contrast, the thermal group exhibits both a lower magnitude and an opposite sign in fissile layers, particularly those closest to the inner reflector. Regarding fission rates, as previously discussed in this chapter, the 3D integrated fission rate, when normalized by the total fission rate (as done in this thesis), must sum to zero across all energy groups; therefore, if the discrepancies in one energy group are consistently positive, those in the other group must be negative to maintain the overall balance.

A.1. TRIPOLI-4® and Serpent options

In order to better understand this odd behavior, a series of tests were carried out, also combining different option choices. Table A.2 shows the new TRIPOLI-4® results obtained by changing the related options addressed in Chapter 4. Where:

- $k_{eff\,surf}$: obtained using the *surfacique* geometry;
- $k_{eff\,SVT}$: obtained using the SVT technique;
- $k_{eff\,notabprob}$: obtained without unresolved table probability sampling;
- $k_{eff\,both}$: obtained using both SVT and without unresolved table probability sampling.

Table A.2: TRIPOLI-4® zero-burnup k_{eff}. All the σ results are in pcm.

$k_{eff\,surf}$	σ_{surf}	$k_{eff\,SVT}$	σ_{SVT}	$k_{eff\,notabprob}$	$\sigma_{notabprob}$	$k_{eff\,both}$	σ_{both}
1.297 26	2.1	1.296 11	1.2	1.297 12	1.2	1.295 99	1.2

All simulations except the one using the *surfacique* geometry were performed with the ROOT geometry (also the value reported in Table A.1).

From Table A.1 and Table A.2, several conclusions can be drawn. First, the discrepancy between TRIPOLI-4® and Serpent at zero burnup is not caused by the choice of geometry (*surfacique* vs. ROOT). The relative difference between the two geometries (surfacique taken as reference) is

$$\varepsilon_k = -1.3\,\text{pcm}. \tag{A.3}$$

This is well within three times the combined uncertainty:

$$3\sigma_{A-B} = 7.32\,\text{pcm}, \tag{A.4}$$

where the combined uncertainty is calculated as:

$$\sigma_{A-B} = \sqrt{\sigma_A^2 + \sigma_B^2}, \tag{A.5}$$

with A and B being two independent measurements affected by uncertainty.

Therefore, TRIPOLI-4® provides consistent results at zero burnup, within a reasonable uncertainty margin, which is reassuring for the validity of the simulations. Moreover, it can be observed that the application of the SVT method and the removal of unresolved resonance probability sampling both contribute to a reduction in the k_{eff} value. Specifically, the relative differences with respect to the reference value in Table A.1 are approximately -89.1 pcm and -11.3 pcm, respectively. When both modifications are applied simultaneously, the total relative difference becomes

$$\varepsilon_{k_{both}} = -98.7\,\text{pcm}. \tag{A.6}$$

If $k_{eff\,both}$ is compared to the Serpent result (taken as reference), the relative difference is

$$\varepsilon_{k_{S2}} = 20.6\,\text{pcm}, \tag{A.7}$$

which is significantly lower than the initial discrepancy of 119.5 pcm in Table A.1.

As the Serpent result reported in Table A.1 is lower than the TRIPOLI-4® one, it is good practice to investigate which TRIPOLI-4® option reduces k_{eff} and brings it closer to the Serpent value. Similarly, it is useful to identify which Serpent options increases its k_{eff} to better match TRIPOLI-4®. For this reason, an additional set of simulations was performed, varying several Serpent parameters, as presented in Table A.3.

Table A.3: Serpent zero-burnup k_{eff}. All the σ results are in pcm.

$k_{eff\,tms}$	σ_{tms}	$k_{eff\,notabprob}$	$\sigma_{notabprob}$	$k_{eff\,both}$	σ_{both}
1.295 63	0.44	1.295 63	0.46	1.295 62	0.45

It can be observed that both options that involve the removal of the unresolved resonance probability sampling, and adoption of the TMS, lead to a similar reduction in k_{eff}. In fact, their relative difference with respect to the reference value (reported in Table A.1, obtained using the `tmp` card) is approximately:

$$\epsilon_k \simeq -7.0\,\text{pcm}. \tag{A.8}$$

Interestingly, when both options are applied simultaneously, the resulting k_{eff} remains statistically unchanged compared to the individual cases. This suggests that the effects of the two options do not add up, and may even interfere with each other. Although this behavior warrants further investigation, it falls outside the scope of this master's thesis, and no additional studies were conducted.

Regardless, both options result in a decrease in k_{eff}, thereby increasing the discrepancy with the TRIPOLI-4® value. Specifically, when comparing the k_{eff} values obtained without unresolved probability table sampling, the relative difference between Serpent and TRIPOLI-4® (taking the latter as the reference) is:

$$\varepsilon_k = -115.2\,\text{pcm}. \tag{A.9}$$

This value is very close to the initial discrepancy reported in Table A.1, indicating that this option is not responsible for the observed mismatch.

It has to be also specified that the Serpent simulations were performed using the same neutron population per batch (100 000), the same number of batches (100 000), and 300 inactive cycles; however, the reported uncertainties σ are not equal to the expected 0.71 pcm as predicted by Equation (4.18). This deviation is likely due to the variance reduction techniques implemented in Serpent, explained in Section 4.2.1. Moreover, the uncertainty values are not consistent across simulations, showing slight variations likely attributable to statistical fluctuations. Each simulation was run with one node and 48 processors of the high performance computer (HPC) cluster of EDF R&D, Cronos, taking on average 25 hours.

A.1.1. Serpent Vivian Salino's IRSN library

In order to better assess the possibility of reducing discrepancies between the two codes, an enhancement of the previously cited study [42] was conducted. Specifically, the impact of using an updated nuclear data library [43] was evaluated, in comparison to the one used in [42]. Both libraries were generated by Vivian Salino at the Institut de Radioprotection et de Sûreté Nucléaire (IRSN) and are based on the JEFF-3.1.1 evaluation. The key difference between the two libraries lies in the correction of an indexing issue for metastable isotopes in the Serpent `.xsdata` file. Specifically, the last digit of each isotope identifier should be "1" if the isotope is metastable. This correction is crucial, as absorbers such as ^{148m}Pm, which play a significant role during depletion, were not properly accounted for in the first library due to incorrect indexing; however, the updated IRSN library still contains only 306 isotopes, whereas the modified Serpent library includes 76 additional isotopes[2]. For instance, ^{152}Gd, which is part of the burnable absorber composition in PRATIC's fuel assembly and has a significant impact on depletion, is missing. For this reason, the IRSN library was not tested in depletion simulations nor in the UOX-Gd case. Additionally, the cross section and thermal scattering data for hydrogen in water are provided at different temperatures in the IRSN library (Table A.4) compared to the default Serpent one (Table 4.1). Since the simulations are performed at an isothermal temperature of 624 K, using the IRSN library introduces interpolation errors in both the $S(\alpha,\beta)_{H_2O}$ thermal scattering data and the isotopic cross sections.

Table A.4: Available temperatures for nuclide cross sections and $S(\alpha,\beta)_{H_2O}$ thermal scattering law data in Vivian Salino's IRSN library.

Data Type	**Temperatures [K]**
Cross Sections	293, 550, 900, 1000, 1200, 1999
$S(\alpha,\beta)$	294, 324, 374, 424, 474, 524, 574, 647, 800

Using the IRSN library, the Serpent simulation yielded the following result:

$$k_{eff\,IRSN} = 1.296\,90, \quad \text{with} \quad \sigma_{\text{IRSN}} = 0.44\,\text{pcm}. \tag{A.10}$$

This corresponds to a relative difference of 91.1 pcm compared to the original Serpent result and -28.4 pcm compared to the TRIPOLI-4® one both in Table A.1, representing

[2]This number can also be inferred by considering that the default JEFF-3.1.1 library for Serpent includes 433 isotopes, checking Table 4.2, and recalling that ^{253}Cf is retained.

a significant improvement. Furthermore, when compared to $k_{eff\,notabprob}$ in Table A.2, the relative difference is reduced to -17.1 pcm.

For this reason, the same set of Serpent options explored in Table A.3 was re-evaluated using the IRSN library, as presented in Table A.5.

Table A.5: Serpent zero-burnup k_{eff} with Vivian Salino's IRSN library. All the σ results are in pcm. Also the k_{eff} with its uncertainty presented in Equation (A.10) was included.

$k_{eff\,IRSN}$	$\sigma_{IRSN+tms}$	$k_{eff\,IRSN+tms}$	$\sigma_{IRSN+tms}$
1.296 90	0.44	1.296 74	0.46
$k_{eff\,IRSN+notabprob}$	$\sigma_{IRSN+notabprob}$	$k_{eff\,IRSN+both}$	$\sigma_{IRSN+both}$
1.296 75	0.44	1.296 74	0.45

First of all, a similar trend can be observed when using TMS or disabling the unresolved resonance probability table sampling: both lead to a decrease in k_{eff}, approximately 12 pcm each; however, this reduction is slightly larger in magnitude compared to the ≈ -7.0 pcm previously observed. This indicates that the behavior of reducing k_{eff} is consistent across both libraries and aligns with the TRIPOLI-4® case (≈ -11 pcm); therefore, the use of TMS consistently increases the discrepancy between the two codes. For this reason[3], TMS was no longer used in this document.

Regarding the unresolved probability table sampling, since its effect is again a consistent decrease in k_{eff} of similar magnitude, there is no justification for disabling this feature, which is physically more accurate.

Furthermore, comparing Table A.3 and Table A.5, a similar trend is observed when both options are applied simultaneously: the effects do not sum up but result in statistically equivalent values.

What is more, an interesting result emerges when comparing $k_{eff\,SVT}$ in Table A.2 or the TRIPOLI-4® k_{eff} in Table A.1 with $k_{eff\,IRSN}$ in Table A.5. The relative differences are 60.8 pcm and -28.4 pcm respectively (TRIPOLI-4® is the reference in Equation (6.1)). Unfortunately, these discrepancies remain statistically significant, and Serpent does not support the SVT technique, preventing a direct comparison; nevertheless, it is evident that the use of the IRSN Vivian Salino's library significantly improves Serpent's results.

In conclusion, a 2D cell-by-cell rates analysis was also performed, comparing the individual fission and capture rates of the fuel assembly in the XY plane between the two

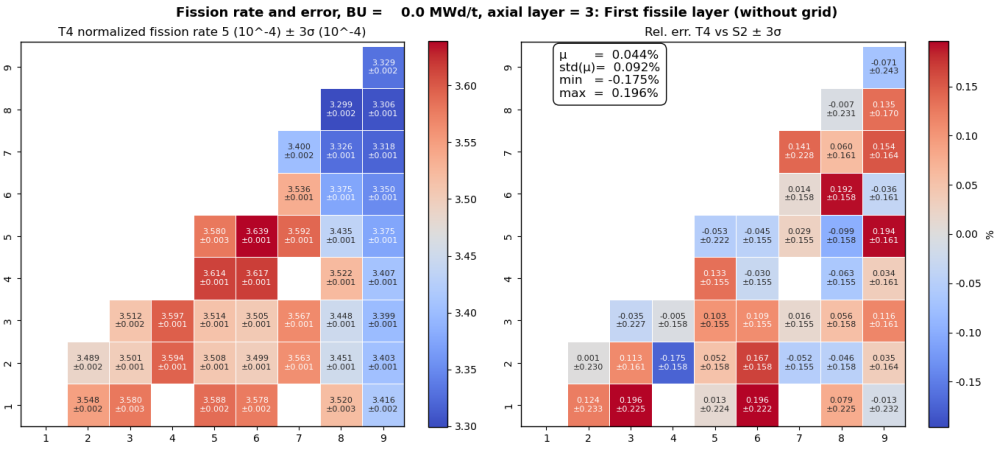
[3] Actually, as explained in [35], the use of TMS increases the computational time, so it was another reason to use the Serpent Doppler-broadening preprocessor with the `tmp` card in depletion simulations.

codes. Figure A.4 presents both comparisons for the first fissile layer (the one closest to the inner reflector), which, as shown in Figure A.2, exhibited some of the highest discrepancies. While no clear pattern (a trend consistent across all layers) could be identified in the fission rate comparisons, a more evident tendency emerged in the capture case. As illustrated in Figure A.5 and similarly observed in the other fissile layers, which are not reported here, it appears that the relative errors in the guide tubes and the instrumentation channel are lower and of opposite sign compared to those in the fuel rods. The other layer comparisons are omitted for the sake of conciseness.

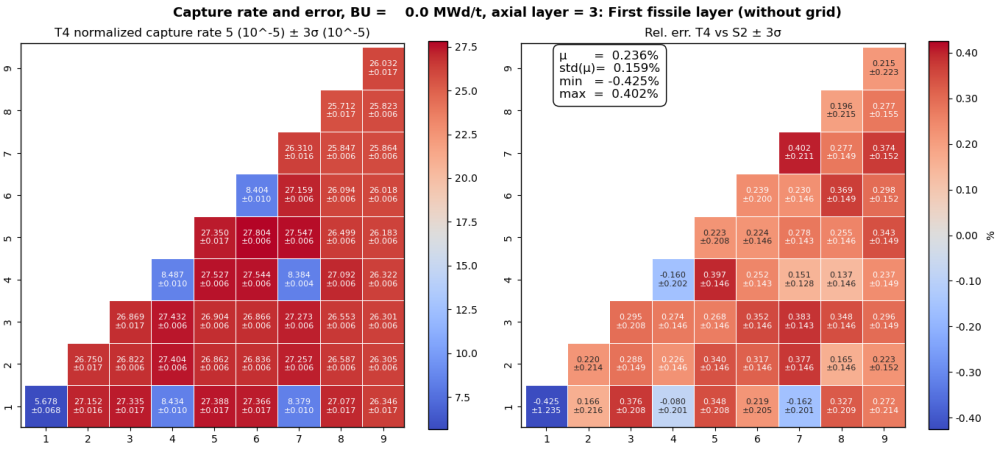
Anyway, since the topic of this chapter was not the main focus of this master's thesis, it was not further investigated, even though the notable TRIPOLI-4®–Serpent mismatch at zero burnup deserves a deeper analysis. Even if this analysis or further ones would still lead to an open question, the main culprit could be the difference in the two codes at an intrinsic level, namely different choices of algorithms, approximations, numerical cancellations, and so on.

An important disclaimer that must be provided regards the 2D cell-by-cell graphs concerning TRIPOLI-4® results with the ROOT geometry. Indeed, this choice radially simulates the whole assembly, not just 1/8. This means that to retrieve these graphs with values for each cell, an average was performed. In principle, one should consider the correlation between cells, since their values are correlated (if one neutron interacts in one cell then does not interact in the other). Of course, this correlation increases the uncertainty on the average values but is hard, if not impossible, to retrieve mathematically. For this reason, the correlation was not considered and the formulas shown in this work did not change for the sake of simplicity. One should consider that the uncertainties reported are actually slightly higher.

Finally, one should remember that Serpent does not have nuclear cross section data at 624 K for the JEFF-3.1.1 library, contrary to TRIPOLI-4® that adopts CEAV-514 library data. For this reason, Serpent has to utilize Doppler-broadening preprocessor from 600 K to 624 K, thus introducing interpolation errors. One possible way to avoid it is to adopt ENDF/B7.1 nuclear data library, thanks to which both Serpent and TRIPOLI-4® have cross sections data at 624 K.



(a) 2D pin-by-pin fission rate



(b) 2D pin-by-pin capture rate

Figure A.4: Comparison between 2D pin-by-pin fission and capture rates for the fissile layer closest to the inner reflector between Serpent and TRIPOLI-4® (reference) at zero burnup.

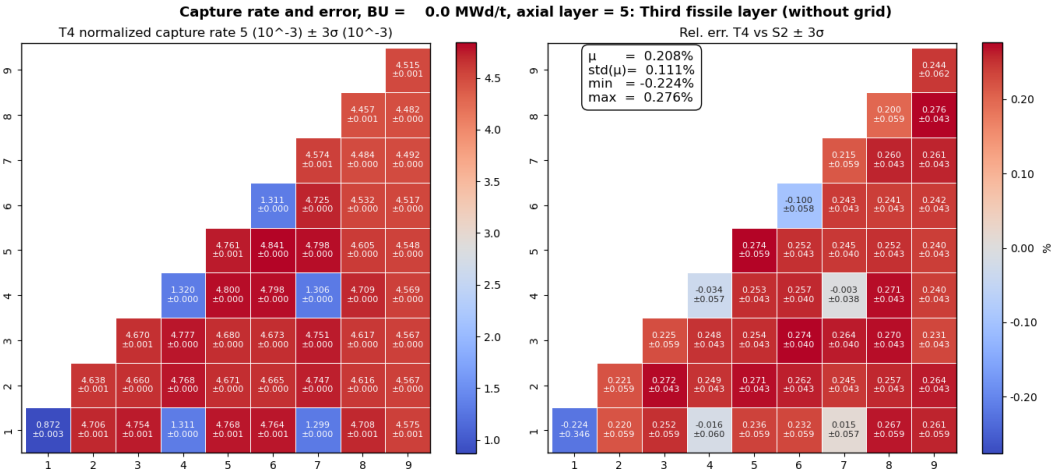


Figure A.5: Comparison between 2D pin-by-pin capture rates for the third fissile layer (it has not the grid) between Serpent and TRIPOLI-4® (reference) at zero burnup.

A.2. Interpretation of 2D cell-by-cell graphs

The 2D cell-by-cell graphs necessitate more comments on how they should be read and the information that could be retrieved from them. For this explanation, Figure A.4 is taken as reference:

- The left graph contains the normalized rates for the reference code used in the simulation with three times its absolute uncertainty right below each value. Both order of magnitudes are written in the title of each graph;
- The one on the right presents the relative errors between the two codes with three times their uncertainty right below the values, evaluated with Equation (A.11), where A and A_{ref} represent the rates.

$$\frac{1}{A_{ref}^2}\sqrt{A_{ref}^2 \times \sigma_A^2 + A^2 \times \sigma_{A_{ref}}^2}; \tag{A.11}$$

- The one on the right presents also the average relative error μ, its standard deviation std(μ) and the minimum and maximum relative errors for the layer;
- It can be noted from the graphs on the left that the more the cells are far from the center of the assembly (instrumentation channel) the more the rates diminish. This is easily observable thanks to the fission rates graph since they change color from a red tonality (greater than the average) to a blue tonality (lower than the average). It is also visible in the capture rates graph where the red vividness diminishes;
- The cells closest to the guide tubes have higher rates, both for capture and fission. This is due to the fact that the guide tubes contain more water, thus the thermalization is the highest and thermal neutrons are better captured and better induce fission by the closest fuel cells;
- The missing cells in the fission graph are the guide tube (no fissile material) and the instrumentation channel. The choice to exclude the latter was linked to the reason that the fissile quantity is negligible with respect to the one in the fuel cells (see Section A.3), thus the fission rate would be too low (so it would affect the graph colormap visualization) and too uncertain. This is an analogous reason for the choice not to display the water blade cells and the water corner.

Equation (A.11) is derived from the general expression [44] reported in Equation (A.12), where $f = f(x_1, x_2, \ldots, x_N)$ and $x_1, x_2, \ldots, x_N$ are uncorrelated quantities.

$$\sigma_f = \sqrt{\sum_{i=1}^{N} \left(\frac{\partial f}{\partial x_i} \sigma_i \right)^2}. \tag{A.12}$$

Knowing that the relative error between two quantities is evaluated as Equation (6.1), where k_{eff} and $k_{eff\,ref}$ could be generally seen as A and A_{ref}, the derivation follows as[4]:

$$\begin{aligned} \sigma_{rel.\,err} &= \sqrt{\left(\frac{\partial f}{\partial A} \sigma_A \right)^2 + \left(\frac{\partial f}{\partial A_{ref}} \sigma_{A_{ref}} \right)^2} = \sqrt{\left(\frac{1}{A_{ref}} \sigma_A \right)^2 + \left(-\frac{A}{A_{ref}^2} \sigma_{A_{ref}} \right)^2} \\ &= \sqrt{\frac{1}{A_{ref}^2} \sigma_A^2 + \frac{A^2}{A_{ref}^4} \sigma_{A_{ref}}^2} = \sqrt{\frac{A_{ref}^2 \times \sigma_A^2 + A^2 \times \sigma_{A_{ref}}^2}{A_{ref}^4}} \\ &= \frac{1}{A_{ref}^2} \sqrt{A_{ref}^2 \times \sigma_A^2 + A^2 \times \sigma_{A_{ref}}^2}. \end{aligned} \tag{A.13}$$

[4]Of course the uncertainty on Equation (6.1) is the same as $\frac{A}{A_{ref}}$ since -1 has not uncertainty.

A.3. Assembly characteristics and material compositions

If the reader is interested in the specifications of the assemblies studied and their relative isotopic compositions, in the following tables the main parameters are reported.

Table A.6: UOX analyzed configuration characteristics (1/8 radially).

Parameter	**Value**
Fuel rods per assembly	39
Guide tubes	5
Instrumentation channel	1 (with fission chamber)
Axial reflector layers	2 (different material)
Fuel axial layers (with grid)	3
Fuel axial layers (without grid)	4
Fuel rod crown subdivision	4
Cell pitch	1.26 cm
Fuel radius	0.4103 cm
Cladding and Gap thickness	647 µm
Grid thickness	300 µm
Guide tube thickness	646 µm
Fission Chamber radius	0.3805 cm
Water Blade thickness	400 µm
Temperature of all materials	624 K
Uranium mass enrichment	3.7 %
Power density	5.4×10^{-2} kW/g
Final burnup step	60 100 MWd/tHM

Table A.7: 36 and 8 UOX-GD configuration characteristics (1/8 radially).

Parameter	**36 UOX-Gd**	**8 UOX-Gd**
UOX / UOX-Gd rods	32 / 7	31 / 1
Uranium mass enrichment	3.5 wt%	2.5 wt%
Guide tubes / Instr. channel	5 / 1	
Axial refl. layers / Fuel axial layers	4 / 10	
UOX / UOX-Gd crown subdivision	4 / 11	
Cell pitch / Fuel radius	1.2618 cm / 0.4107 cm	
Gap / Cladding thickness	79 µm / 571 µm	
GT thickness (instr. / standard)	460 µm / 410 µm	
Water Blade thickness	403.5 µm	
Material Temperature / Power density	624 K / 2.473×10^{-2} kW/g	
Gd_2O_3 content / Final burnup	8 wt% / 60 100 MWd/tHM	

Table A.8: UOX assembly material composition [10^{24}/cm^3].

Iso	Dens	Iso	Dens	Iso	Dens	Iso	Dens
Fuel (Total $\approx 6.8570 \times 10^{-2}$)							
^{16}O	4.5713×10^{-2}	^{234}U	7.3767×10^{-6}	^{235}U	8.5611×10^{-4}	^{236}U	1.3699×10^{-6}
^{238}U	2.1992×10^{-2}	^{239}Pu	2.2750×10^{-8}	^{240}Pu	2.2655×10^{-9}	^{241}Pu	2.2561×10^{-9}
^{242}Pu	2.2468×10^{-9}						
Cladding & Gap (Total $\approx 3.8082 \times 10^{-2}$)							
^{90}Zr	3.7606×10^{-2}	^{117}Sn	1.7621×10^{-4}	^{119}Sn	1.9604×10^{-4}	^{54}Fe	2.7105×10^{-6}
^{50}Cr	1.6634×10^{-6}	^{52}Cr	3.2078×10^{-5}	^{53}Cr	3.6369×10^{-6}	^{54}Cr	9.0540×10^{-7}
^{56}Fe	4.2136×10^{-5}	^{57}Fe	9.6474×10^{-7}	^{58}Fe	1.2863×10^{-7}	^{14}N	1.9142×10^{-5}
Metal (Guide Tubes) (Total $\approx 4.2853 \times 10^{-2}$)							
^{90}Zr	4.2318×10^{-2}	^{117}Sn	1.9829×10^{-4}	^{119}Sn	2.2060×10^{-4}	^{54}Fe	3.0501×10^{-6}
^{50}Cr	1.8718×10^{-6}	^{52}Cr	3.6097×10^{-5}	^{53}Cr	4.0926×10^{-6}	^{54}Cr	1.0188×10^{-6}
^{56}Fe	4.7416×10^{-5}	^{57}Fe	1.0856×10^{-6}	^{58}Fe	1.4475×10^{-7}	^{14}N	2.1540×10^{-5}
Thermowell (Total $\approx 2.3470 \times 10^{-2}$)							
^{90}Zr	2.3177×10^{-2}	^{117}Sn	1.0860×10^{-4}	^{119}Sn	1.2082×10^{-4}	^{14}N	1.1797×10^{-5}
^{54}Fe	1.6705×10^{-6}	^{50}Cr	1.0252×10^{-6}	^{52}Cr	1.9770×10^{-5}	^{53}Cr	2.2415×10^{-6}
^{54}Cr	5.5802×10^{-7}	^{56}Fe	2.5969×10^{-5}	^{57}Fe	5.9459×10^{-7}	^{58}Fe	7.9278×10^{-8}
^{235}U	4.6118×10^{-8}						
Water (Total $\approx 6.6311 \times 10^{-2}$)							
^{1}H	4.419×10^{-2}	^{16}O	2.2095×10^{-2}	^{10}B	5.2061×10^{-6}	^{11}B	2.0567×10^{-5}
Inner Reflector (Total $\approx 1.432\,33 \times 10^{-1}$)							
^{1}H	7.1806×10^{-2}	^{16}O	3.5903×10^{-2}	^{90}Zr	3.4632×10^{-2}	^{117}Sn	3.3051×10^{-4}
^{119}Sn	3.6770×10^{-4}	^{14}N	3.5831×10^{-5}	^{50}Cr	3.1200×10^{-6}	^{52}Cr	6.0166×10^{-5}
^{53}Cr	6.8216×10^{-6}	^{54}Cr	1.6982×10^{-6}	^{54}Fe	5.0839×10^{-6}	^{56}Fe	7.9033×10^{-5}
^{57}Fe	1.8095×10^{-6}	^{58}Fe	2.4127×10^{-7}				
Outer Reflector (Total $\approx 9.0399 \times 10^{-2}$)							
^{1}H	3.4012×10^{-2}	^{16}O	1.7006×10^{-2}	^{90}Zr	3.8677×10^{-2}	^{117}Sn	2.6092×10^{-4}
^{119}Sn	2.9028×10^{-4}	^{14}N	2.8287×10^{-5}	^{50}Cr	2.4630×10^{-6}	^{52}Cr	4.7498×10^{-5}
^{53}Cr	5.3852×10^{-6}	^{54}Cr	1.3406×10^{-6}	^{54}Fe	4.0134×10^{-6}	^{56}Fe	6.2391×10^{-5}
^{57}Fe	1.4285×10^{-6}	^{58}Fe	1.9047×10^{-7}				

Table A.9: 36 UOX-Gd assembly material composition [10^{24}/cm^3].

Iso	Dens	Iso	Dens	Iso	Dens	Iso	Dens
UOX Fuel [$\rho = 10.1711\,\mathrm{g/cm^3}$] (*left*) & **UOX-Gd Fuel** [$\rho = 9.8003\,\mathrm{g/cm^3}$] (*right*)							
^{16}O	4.5377×10^{-2}	^{234}U	4.8143×10^{-6}	^{16}O	4.4133×10^{-2}	^{234}U	4.2677×10^{-6}
^{235}U	5.7429×10^{-4}	^{236}U	2.6305×10^{-6}	^{235}U	5.0908×10^{-4}	^{236}U	2.3319×10^{-6}
^{238}U	2.2107×10^{-2}			^{238}U	1.9597×10^{-2}	^{152}Gd	5.2100×10^{-6}
				^{154}Gd	5.6790×10^{-5}	^{155}Gd	3.8554×10^{-4}
				^{156}Gd	5.3325×10^{-4}	^{157}Gd	4.0769×10^{-4}
				^{158}Gd	6.4709×10^{-4}	^{160}Gd	5.6946×10^{-4}
Metal (Cladding and Guide Tubes) [$\rho = 6.5215\,\mathrm{g/cm^3}$]							
^{90}Zr	2.1733×10^{-2}	^{96}Zr	1.1827×10^{-3}	^{116}Sn	6.9750×10^{-5}	^{50}Cr	3.2818×10^{-6}
^{91}Zr	4.7393×10^{-3}	^{120}Sn	1.5629×10^{-4}	^{117}Sn	3.6842×10^{-5}	^{52}Cr	6.3287×10^{-5}
^{92}Zr	7.2442×10^{-3}	^{122}Sn	2.2210×10^{-5}	^{118}Sn	1.1619×10^{-4}	^{53}Cr	7.1762×10^{-6}
^{94}Zr	7.3413×10^{-3}	^{124}Sn	2.7775×10^{-5}	^{119}Sn	4.1207×10^{-5}	^{54}Cr	1.7863×10^{-6}
^{50}Sn	4.6532×10^{-6}	^{54}Fe	8.6321×10^{-6}	^{57}Fe	3.1294×10^{-6}	^{16}O	3.0692×10^{-4}
^{114}Sn	3.1661×10^{-6}	^{56}Fe	1.3551×10^{-4}	^{58}Fe	4.1647×10^{-7}	^{115}Sn	1.6310×10^{-6}
Gap [$\rho = 1.6 \times 10^{-3}\,\mathrm{g/cm^3}$] (*first 2 iso.*) & **Water** [$\rho = 0.726\,83\,\mathrm{g/cm^3}$] (*others*)							
^{4}He	2.4044×10^{-4}	^{3}He	4.8089×10^{-10}	^{1}H	4.8606×10^{-2}	^{16}O	2.4303×10^{-2}
Reflector Mix [$\rho = 7.9181\,\mathrm{g/cm^3}$]							
^{16}O	1.3332×10^{-2}	^{56}Fe	2.4191×10^{-2}	^{58}Ni	2.4968×10^{-3}	^{50}Cr	3.4177×10^{-4}
^{57}Fe	5.5869×10^{-4}	^{60}Ni	9.6175×10^{-4}	^{1}H	2.6663×10^{-2}	^{52}Cr	6.5908×10^{-3}
^{58}Fe	7.4351×10^{-5}	^{61}Ni	4.1806×10^{-5}	^{53}Cr	7.4734×10^{-4}	^{54}Fe	1.5411×10^{-3}
^{55}Mn	7.8365×10^{-4}	^{62}Ni	1.3330×10^{-4}	^{54}Cr	1.8603×10^{-4}	^{28}Si	4.2414×10^{-4}
^{29}Si	2.1536×10^{-5}	^{64}Ni	3.3947×10^{-5}	^{30}Si	1.4196×10^{-5}		

B | 2D fissile cell analysis

B.1. Decay chain

Regarding the decay chain, the APOLLO3® code always employed in this thesis a reduced decay chain called `ch_CEAV2005_V3c`; however, it is also possible to perform calculations using a more complete one, referred to as `DecayData_CEAV2005_V3`. The primary difference between these two datasets lies in their isotope content: `DecayData_CEAV2005_V3` includes 157 isotopes in the `Chain` folder, whereas `ch_CEAV2005_V3c` contains 154. The three additional isotopes are ^{153}Gd, ^{165}Dy, and ^{131m}Te, which are not tracked by Serpent.

Moreover, `DecayData_CEAV2005_V3` provides fission yield data for 64 fissile isotopes, compared to 26 in `ch_CEAV2005_V3c`; however, the 38 additional isotopes for which fission yields are included do not appear in the `Chain` dataset. Since these isotopes are neither present in the initial composition nor produced through decay or transmutation in the considered chains, they do not manifest during simulations. Consequently, their fission yield data are irrelevant for this work. These isotopes are also absent from JEFF-3.1.1, further confirming their lack of contribution to results.

Therefore, the only genuine isotope differences between the two libraries are the presence of ^{153}Gd, ^{165}Dy, and ^{131m}Te; however, it is peculiar that ^{153}Gd appears since it is not part of JEFF-3.1.1.

The choice to avoid a full 3D assembly depletion study was driven by the computational cost of additional depletion calculations, including isotopic concentration studies. It was observed that using Serpent's `set depout 3` option in the 3D case dramatically increased computational time[1]. This is because `set depout 3` instructs Serpent to write burnable material compositions to depletion output files separated by the `div` card (for 4 crowns here), significantly increasing output volume and file printing time (see Serpent Wiki).

Given this, and that `set depout 2` (used in the 3D simulations here) writes isotopic concentrations without division into crowns, `set depout 3` could increase computational time up to nearly eightfold in the worst case; therefore, it may be useful to disable this option (`set depout 0`) when isotopic concentration output is not essential.

[1] "Dramatically" refers to the near doubling of computational time each increase in axial refinement. Hence, this option was abandoned for 3D simulations.

Alternatively, the `set deppara` card can be used to limit depletion output to specific variables (e.g., atomic densities only), reducing output size and computation time (see Serpent Wiki).

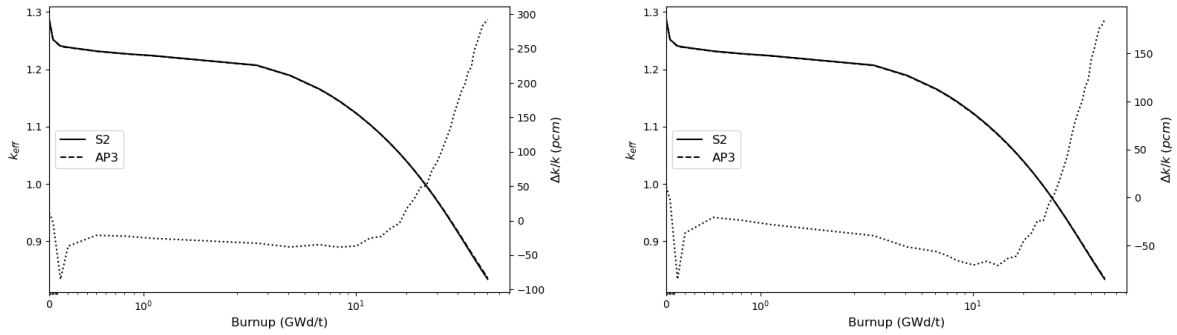
Three depletion simulations were run using the burnup mesh in Listing 6.2[2] on a 2D fissile cell without grid of the UOX assembly:

- Serpent;
- APOLLO3® with `ch_CEAV2005_V3c`;
- APOLLO3® with `DecayData_CEAV2005_V3`.

First, it was confirmed that ^{153}Gd concentration remained zero throughout the simulation, since precursor isotopes necessary for its formation are absent from the decay chain.

Thus, the only meaningful difference between the two chains lies in ^{165}Dy and ^{131m}Te, unless there are differences in fission yields, branching ratios, or decay constants data which were not verified due to limited accessibility within EDF and because the accessible data are stored in a non-human-readable format without comprehensive documentation. No published comparison of these parameters for the two datasets was found.

Next, the k_{eff} comparison between Serpent and APOLLO3® for both decay chains is shown in Figure B.1. Interestingly, the APOLLO3® results with `DecayData_CEAV2005_V3` align more closely with Serpent, despite including the two additional isotopes. Although this may seem advantageous, it actually represents a bias, since Serpent does not track ^{165}Dy and ^{131m}Te; therefore, results with `ch_CEAV2005_V3c` remain the more consistent choice.



(a) `ch_CEAV2005_V3c`. (b) `DecayData_CEAV2005_V3`.

Figure B.1: Comparison of k_{eff} between Serpent and APOLLO3® using two different decay data libraries.

[2]This burnup mesh differs from Listing 7.1 and Listing 7.2 since these tests were performed before the final burnup mesh was established.

Furthermore, at the end of the simulation (60 100 MWd/tHM), the discrepancy with Serpent for APOLLO3 using `ch_CEAV2005_V3c` is about 300 pcm, similar to that observed in Figure 7.1, Figure 7.3, and Figure 7.7. This consistency further confirms the correctness of the 3D simulations.

An analysis of isotopic concentration evolution was also performed for all isotopes present and produced during fission, capture, and transmutation processes (a total of 143) in the whole active fuel (not crown by crown). Some relevant isotopes (^{135}I, ^{135}Xe, ^{235}U, ^{239}Pu are shown in Figure B.2.

The selected isotopes were chosen due to their influence on k_{eff} (^{235}U and ^{239}Pu), their abundance (^{238}U), and their impact on reactivity early in the simulation (^{135}I and ^{135}Xe).

From this set of images, it is evident that APOLLO3® results with `DecayData_CEAV2005_V3` generally show better agreement with Serpent than those with `ch_CEAV2005_V3c`, consistent with previous observations for k_{eff}. This confirms the effect of including or excluding ^{165}Dy and ^{131m}Te in the decay chain. Notably, ^{165}Dy has a high thermal neutron absorption cross section (on the order of 10^3 barn [45]), which could reduce k_{eff} and explain the improved agreement. By contrast, ^{131m}Te has negligible capture cross sections and likely does not significantly influence results. Anyway, it is reported [45] that the influence on reactivity by these two isotopes is below 0.01 % at 60 000 MWd/tHM. This is a suggestion that probably the two decay chains are different also for fission yields, branching ratios, or decay constants and should be checked in a future work.

In any case, regardless of the APOLLO3® decay chain used, a significant discrepancy with Serpent remains. This difference is further examined and substantially reduced for the 2D case in the following section.

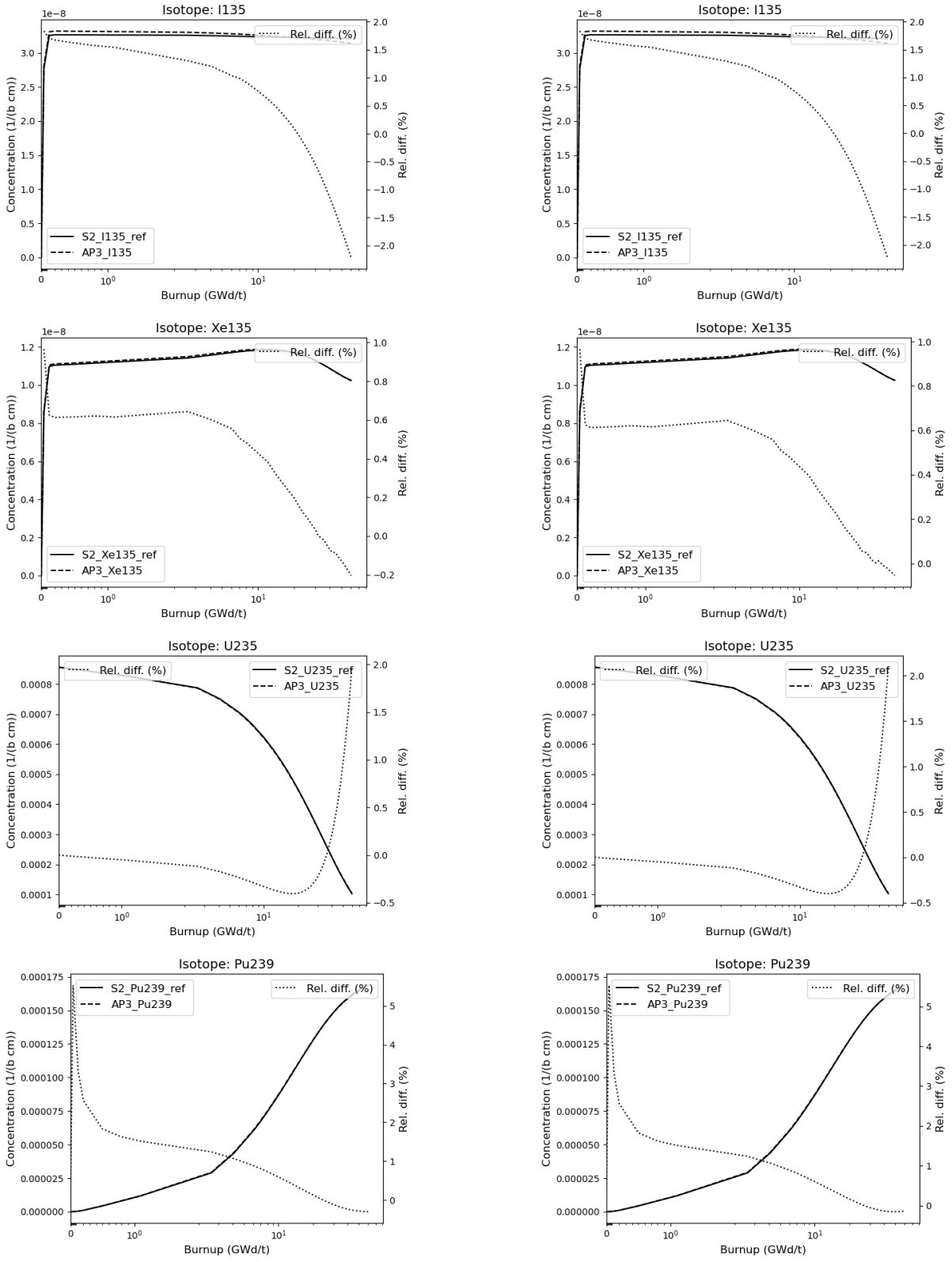


Figure B.2: Comparison of isotopic concentrations between Serpent and APOLLO3® using `ch_CEAV2005_V3c` (left) and `DecayData_CEAV2005_V3` (right).

B.2. Energy deposition model

First of all, a possible explanation for the observed discrepancy after 20 000 MWd/tHM could be linked to the energy deposition model. This hypothesis arises from the relative errors observed in the capture rates within the guide tubes. Indeed, choosing for example the values at zero burnup for the results in Figure 7.1, a straightforward pattern can be identified by observing the capture rates for the first fissile layers, the closest to the inner reflector, shown in Figure B.3, as already observed in Appendix A for Figure A.5. It is clear that the relative errors in the guide tubes are again lower in absolute value than the other ones.

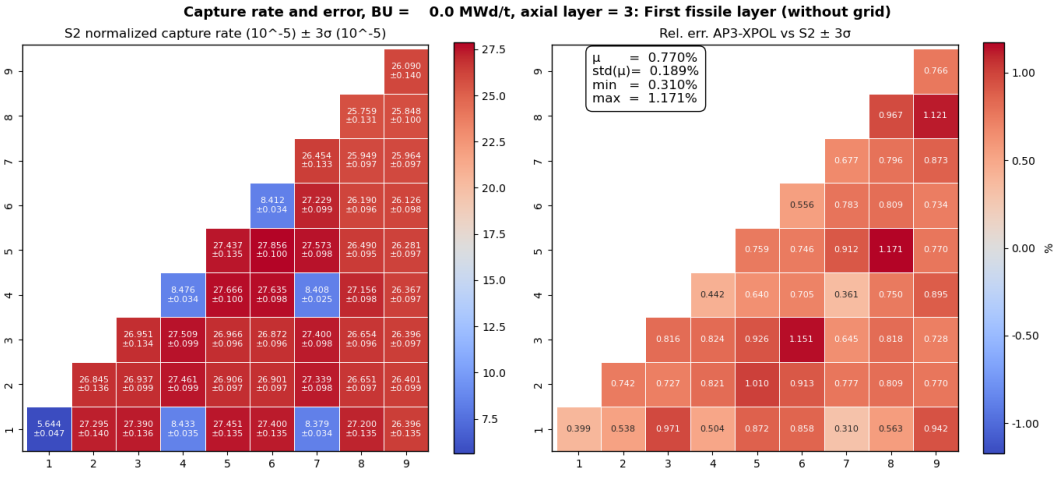


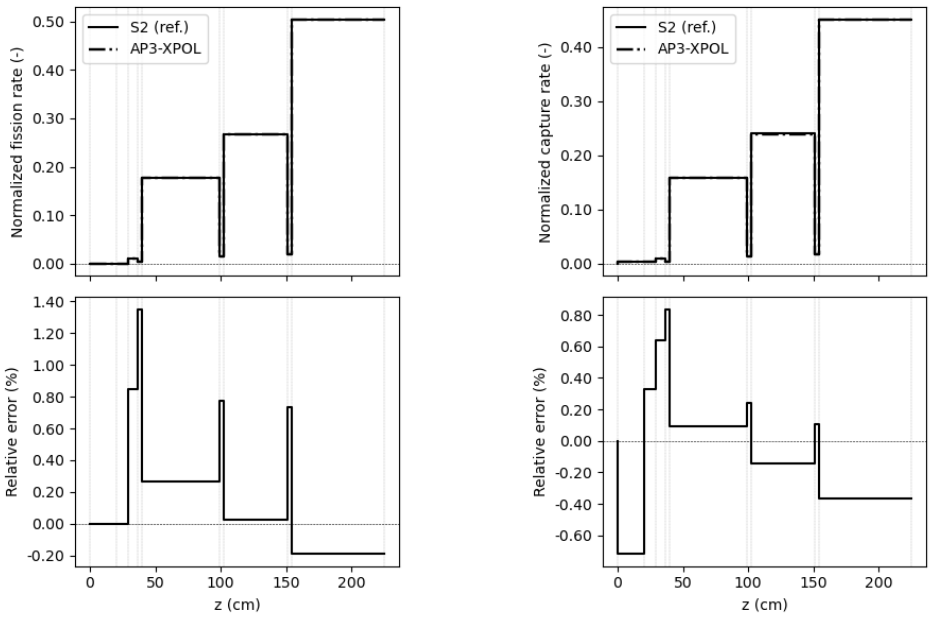
Figure B.3: Pin-by-pin fission and capture rates for the first fissile layer at zero burnup correspondent to the results in Figure 7.1.

This pattern does not occur for the layers with grids, probably the discrepancies are dominated by another effect that is greater in magnitude since the integrated rates are higher for those layers. The explanation could be due to the fact that the grid layers have self-shielded isotopes which cause, even if correctly self-shielded, further spatial heterogeneity with respect to Serpent. Moreover, the layers with grids are thinner than the ones without; thus, the reaction rates are smaller and the relative differences are greater.

The pattern was further analyzed by refining the guide tubes and observing that the relative discrepancies do not change. The cases analyzed were:

- The use of 8 angular sectors in the guide tubes;
- The same of the previous point plus having within the metal 1 more annulus each 0.1 cm to further capture the thermalization capability of water;
- The implementation of 16 sectors in the guide tubes and 1 more annulus each 0.1 cm.

The most refined case still gives similar relative errors than the simulation without further refinements. Moreover, it was also tested to check the possibility to decrease the relative errors in the grid layers by using REL383 energy groups [46] with APOLLO3® instead of the standard SHEM281 groups. Figure B.4 shows that the relative errors improve but still the grid errors are the highest. The k_{eff} at zero burnup becomes 1.29673 with a relative difference of 79.1 pcm with respect to 1.29571 in Figure 7.1.



(a) Fission rate comparison (b) Capture rate comparison

Figure B.4: Comparison between integrated fission and capture rates for Serpent (reference) and APOLLO3® with 383 groups at zero burnup.

These results imply that, probably, the energy deposition models used by the two codes must be taken into account, rather than the discretization of radial crowns for guide tubes. In particular, the default option for Serpent, which is the one used in this thesis, considers a constant energy deposition per fission, which means

$$E_{f,i} = \frac{Q_i}{Q_{235}} \times H_{235}, \tag{B.1}$$

where $E_{f,i}$ is the energy deposited by a fission event of fissile isotope i, Q_i is the Q-value found in the cross section libraries of the i-th isotope, Q_{235} is the ^{235}U one and H_{235} is the fission heating value of ^{235}U which default value is 202.27 MeV and can be set with the `set U235H` Serpent card.

The modelling of energy deposition in Monte Carlo neutron and photon transport calculations is a complex task. Particles can deposit their energy in a large number

of reactions and some reactions such as fission and radiative capture release additional energy. To further complicate the process, reactions can produce secondary particles which transport energy away from the reaction site and deposit it elsewhere; however, often the transport of some secondary particles is not simulated and their energy is instead deposited locally at the reaction site. In particular, in [47] it is written: "*These sort of approximations affect the spatial accuracy of the energy deposition calculations.*" This is a hint toward the guide tube discrepancies observed in Figure A.4, Figure A.5 and Figure B.3.

This constant energy deposition is the most simplified treatment where only neutrons are transported and it is assumed that a user defined constant amount of energy is released per fission. In addition, all energy is deposited locally at the fission site and only fission reactions deposit energy.

In Serpent, the energy deposition model can be set with the `set edepmode` card and is better described in [47]: nevertheless, this option was not changed because of two reasons reported in the Wiki:

- *"Energy deposition modes "1", "2" and "3" require data which is not available in the standard ACE format cross section files used by Serpent. Separately distributed ACE-files (file `endfb71_edep.tar.gz`) containing additional data are required to use these modes;"* therefore, data from ENDF/B-VII.1 would have to be used instead of JEFF-3.1.1, which is not feasible in this thesis because, as already explained, JEFF-3.1.1 is compliant with the APOLLO3® CEAV-514 library.
- *"The energy deposition modes "1", "2" and "3" have been tested mainly in criticality source simulations and one should proceed with caution when using them in other types of calculations such as transient and burnup calculations."* Thus, these modes are not yet fully tested by the Serpent developers in depletion calculations at the moment of writing.

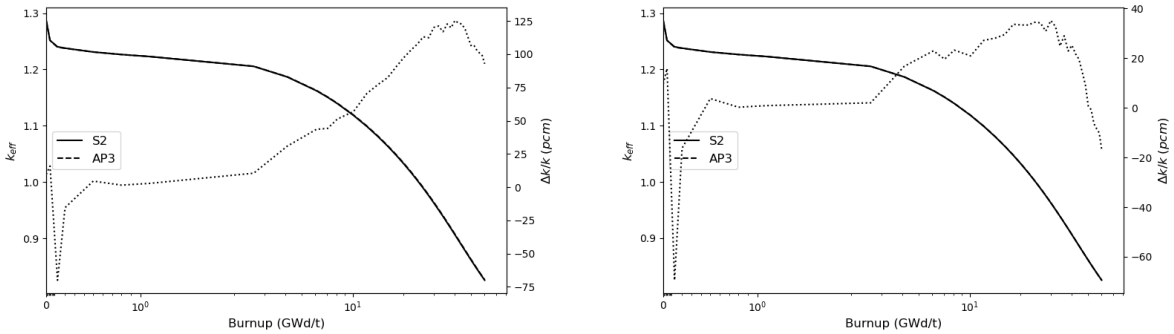
In conclusion, it was chosen to still use the constant energy deposition (default choice: `set edepmode 0`) and to change H_{235} in Equation (B.1) by using the card

```
set  U235H  193.7201,
```

because it was set at the same value as in the CEAV-514 library used by APOLLO3®: 193.7201 MeV. This last number considers only the energy released by the fission event of ^{235}U without the one released by gamma captures. Analogously, it was adopted an APOLLO3® version that considers only local energy deposition for a fission event. This choice was the key to get the best match between the two codes.

The study comparing Serpent and APOLLO3® was performed on the 2D cell with both decay chains and the results are shown in Figure B.5.

It must be said that this test was performed before the introduction of Listing 7.2 as APOLLO3® definitive burnup mesh, but this does not influence the results after 20 000 MWd/tHM. Indeed, it can be straightforwardly seen that the discrepancies on the k_{eff} between the two codes are evidently reduced.



(a) `ch_CEAV2005_V3c`. (b) `DecayData_CEAV2005_V3`.

Figure B.5: Full depletion 2D fissile cell simulations of APOLLO3® against Serpent (with `set U235H 193.7201`) comparing different decay chains.

C | Notes on Serpent boundary conditions and convergence analysis

This appendix contains two interesting findings of this master's thesis work:

- One regarding a particular feature of Serpent boundary conditions;
- The other concerning the convergence analysis.

C.1. Notes on Serpent boundary conditions

In reactor physics, assembly-level simulations are a fundamental prerequisite for full-core analysis; however, modeling a complete assembly brings a significant computational burden, which can be mitigated by exploiting geometric symmetries. A common approach involves reducing the domain to a 1/8 radial sector and halving the axial height. This reduction dictates specific boundary conditions along the vertical axis: a reflective condition at the axial mid-plane to enforce symmetry, and a vacuum (or 'black') condition at the top to simulate neutron leakage. Consequently, the simulation requires the application of heterogeneous boundary conditions along the z-axis, while maintaining reflective conditions on the radial boundaries

Unfortunately, at the time of writing and to the author's knowledge, Serpent does not allow different boundary conditions along the same axis; therefore, a strategy is required. A possibility found in literature is to apply reflective boundary conditions everywhere and add an *infinite* absorber layer on top of the upper reflector; however, this approach introduces a series of shortcomings:

- *Neutron shortage:* when an initially homogeneous neutron source is simulated in the system, the larger and more effective the absorber layer is, the more neutrons are born within it, making them likely to be immediately absorbed. This causes a severe reduction in the neutron population that occupies the actual region of interest: the

active one. It was chosen not to set the first neutron generation starting from a particular zone of the reactor (usually the active region) because, even if it is physically correct (the neutrons are only born in the fuel), it introduces a bias with respect to the uniform case, leading to possible neutron clusters that produce fictitious results;

- *Absorber dimension:* the bigger the absorber the more effective it is. But, to determine its effectiveness, a further study (that requires time) must be performed. It can be considered perfect when *zero* neutrons are reflected back to the upper reflector layer, otherwise wrong results are retrieved;

- *Time cost:* in order to have an effective reflector it must be not only big enough but also dense enough. Actually, a sweet spot between dimension and density must be found which means an optimization study which is time-consuming. Anyway, the time cost is not just about this study but also related to the simulations themselves. Indeed, a denser and highly absorbing material causes the increase of the majorant cross section, thus a worsening of the delta-tracking technique. As explained in Section 4.2.1, if the majorant cross section becomes too high, there is a continuous sample-rejection when using the delta-tracking variance reduction technique causing a useless waste of time. It has to be said that Serpent allows to set delta tracking options with several cards:

  - set dt: Sets the probability threshold for switching from delta-tracking to surface-tracking. The switch occurs when the probability of sampling virtual collisions, defined as the ratio between the material's total cross section and the majorant cross section, exceeds the specified threshold. By default, this threshold is set to 0.9 which means that delta-tracking is used when the ratio of the total cross section to the majorant is greater than 0.1;
  - set forcedt: defines the list of materials where delta-tracking is always applied, regardless of the probability threshold set by the `set dt` option. This allows overriding the automatic tracking mode selection material-wise.
  - set blockdt: Analogous of `set forcedt` but performs a material-wise turn off of delta-tracking.

  Unfortunately, no Serpent option allows to remove a material from the majorant cross section computation; therefore, the implementation of a dense absorber has an impact on the whole system thus in the whole simulation cost. Furthermore, also the absorber material choice is impacting: the better the absorber (so the higher

the capture cross section) the higher the majorant cross section. For this reason, choosing a *foreign* isotope (in the sense not a constituent of the simulated system) could lead to a significant increase in computational time. In conclusion, the best choice not to affect the computational time is to adopt an absorber with the same density and the same isotopic composition of the one already present in the reactor, namely boron in a borated water configuration or gadolinium in a UOX-Gd core. But, since their density is not really high, the need for a big absorber is obvious, leading to the previous problem.

Hence, due to these strictly computational issues, this strategy was abandoned in favor of another developed in this master's thesis project which is as counterintuitive as more efficient. Basically, the idea is to define 3 different *outside* region cells in Serpent, where *outside* is the keyword for the card that indicated where the boundary conditions are applied. After defining a 3D cuboid that hosts the assembly, an *outside* cell is defined out of the cuboid such that the reflective boundary conditions are applied along each axis. Then two planes are defined: the first is the top of the upper reflector and the other is the bottom of the reflected upper reflector. In this way, all the infinitely reflected system geometry is cut, as shown in Figure C.1.

In Figure C.1, if a neutron reaches the black area is instantaneously terminated, respecting the black boundary condition.

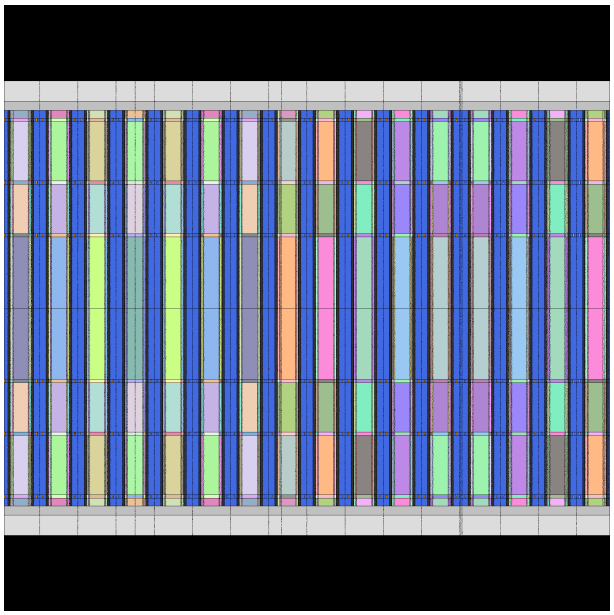


Figure C.1: YZ cut of a fuel assembly simulated in Serpent. The black part on top and on bottom is the "void" region. It is visible the reflective boundary condition at the z axis midplane.

C.2. Notes on convergence analysis

It should be noted that the convergence of integrated and 2D reaction rates was not explicitly verified for every burnup step or individual axial layer in the preliminary convergence analysis simulations (Chapter 6). This approach was adopted to optimize computational efficiency and avoid excessive data processing during the initial convergence study. Instead, the analysis primarily relied on k_{eff} stability as the governing metric for convergence. A more rigorous, detailed comparison of local reaction rates was reserved for the final, comprehensive benchmark between APOLLO3® and Serpent, where high-fidelity spatial precision is required (Chapter 7).

It must be emphasized that determining the appropriate trade-off between the number of burnup steps and the axial mesh refinement was far from straightforward. In fact, a peculiar behavior was observed during the initial phase of the study. Initially, it was assumed that the CamiVVER burnup mesh defined in Listing 6.1 was suitable for Serpent. Based on this assumption, a series of simulations with varying axial mesh refinements were carried out and are presented in Table C.1.

Table C.1: Parameters of Serpent simulations for the axial mesh convergence analysis with the CamiVVER burnup mesh (Listing 6.1). Here, n is the number of neutrons per batch and N the number of batches.

BU zones	**Fuel layers**	n / N $(\times 10^3)$	`set opti`	$\sigma(k_{eff})$ **[pcm]**
1099 (material mesh)	7	100 / 30	4	1.83
4082	26	200 / 9.5	4	2.29
7065	45	200 / 8	4	2.50
12 089 (stepmesh)	77	200 / 6	2	2.89
13 031	83	200 / 6	2	2.89
31 243 (1 cm)	199	200 / 6	2	2.89
62 015 (0.5 cm)	395	200 / 5	2	3.16
102 678 (0.3 cm)	654	200 / 3.5	2	3.78

The values from the fifth row onward were selected based on the following ideas:

- 13 031: this configuration provides a slightly finer refinement than the stepmesh case, allowing verification of whether the stepmesh layering was appropriately chosen;
- 31 243 (1 cm), and 62 015 (0.5 cm), 102 678 (0.3 cm): these configurations were generated by adding one layer per centimeter, 0.5 cm, and 0.3 cm respectively. It is important to note that the second case does not result in exactly double the number of layers of the first, and more importantly, the first case does not consist of 205 layers (which would correspond to the height of the fissile region). This happens because the original material layers were not modified, and the layering restarts from the material layer origin each time a new material layer is encountered. For example, a layer of 48.9 cm contains $48+1$ layers (since each width is not an integer: $int(width)+1$, where int denotes the integer part, approximated by defect), with the last layer having a thickness of 0.9 cm. Indeed, according to the assembly dimensions presented in Chapter 5, the number of layers for the 1 cm subdivision is the following (starting from the inner region to the closest one to the inner reflector):

$$(70+1)+(3+1)+(48+1)+(3+1)+(58+1)+(3+1)+(7+1)=199, \tag{C.1}$$

and, for the 0.5 cm one:

$$\begin{aligned}(70\times 2+2)+(3\times 2+1)+(48\times 2+2)+(3\times 2+1)\\ +(58\times 2+2)+(3\times 2+1)+(7\times 2+2)=395.\end{aligned} \tag{C.2}$$

The +2 in this case is justified by the fact that the fissile layers without the grid have a decimal part > 0.5, which means they host an additional layer.

The derivation for the 0.3 cm case is analogous.

Figure C.2 illustrates a rather peculiar behavior: regardless of the axial mesh refinement, the simulation does not seem to reach a converged result, and it suggests that the burnup mesh and the axial mesh are strongly interdependent. This is a completely different behavior with respect to deterministic codes. In Monte Carlo codes, it is not possible to perform a nearly-continuous refinement of one without having the other already close to convergence. This behavior is likely intrinsic to probabilistic methods, or at least to Serpent. In contrast, deterministic methods behave differently: if one refines the axial mesh while keeping the burnup mesh constant (or vice versa), the solution typically stabilizes. This occurs because deterministic solvers rely on discretized equations and

numerical schemes that converge independently in each dimension; therefore, refining one mesh usually improves accuracy without requiring simultaneous refinement of the other. Monte Carlo methods, however, rely on statistical sampling, and the interaction between spatial and depletion meshes can introduce noise or bias if not balanced properly.

This result provides a useful insight and supports the interpretation given in Figure 6.8a and Figure 6.9a.

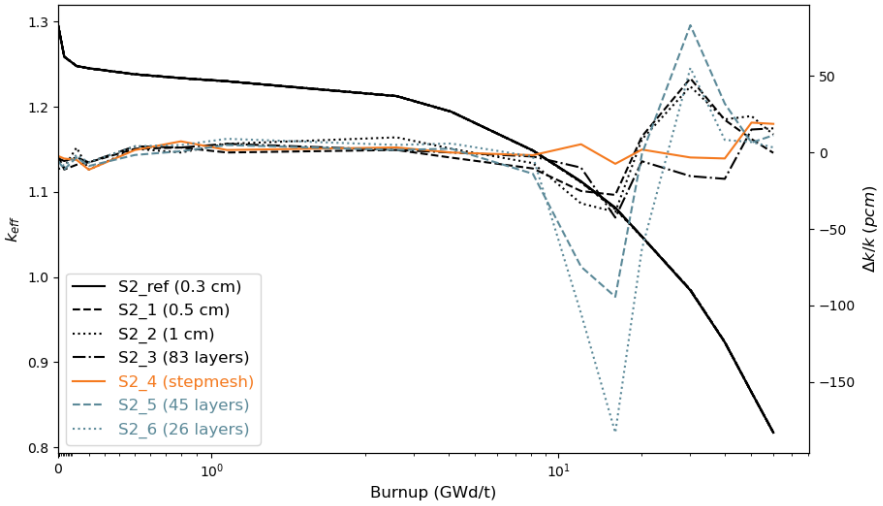


Figure C.2: Axial mesh refinement using the CamiVVER burnup mesh (Listing 6.1).

Since this issue was not the main focus of the thesis, it was not investigated further, but it deserved to be presented nonetheless.

To conclude with, it was observed that the CamiVVER APOLLO3®-TRIPOLI-4® comparison reported in [6] has a similar fluctuation pattern of the ones in Figure C.2: a further hint that the CamiVVER original burnup mesh Listing 6.1 is not the converged one, as already explained in previous chapters.

Dediche

Ed ora il capitolo più difficile. Il capitolo che non tratta di numeri ma di sentimenti ed emozioni: le dediche.

Innanzitutto, ci tengo a ringraziare le persone senza le quali non sarei mai riuscito a portare a termine questa tesi. Grazie al mio relatore, il Prof. Lorenzi, per avermi dato un preziosissimo consiglio iniziale sull'uso di Serpent e per avermi aiutato ogni qualvolta ne avessi bisogno. Grazie a Simone Santandrea e ad Andrea Gammicchia per i feedback e gli scambi di informazioni costanti e fondamentali. Grazie per avermi insegnato il lavoro di squadra e come portare a termine un progetto impegnativo che richiede molta concentrazione. Grazie a Barbara Vezzoni per i suggerimenti puntuali su come approcciare il problema. Grazie per avermi aiutato nonostante gli impegni, l'ho apprezzato moltissimo. Ed infine, grazie al mio supervisore, alla mia guida, Marco. Grazie per avermi insegnato a fare ricerca, per aver ascoltato costantemente le mie domande e per aver sempre risposto a tutte. Grazie per tutto il tempo che mi hai dedicato, senza mai dirmi un no. Ho avuto davvero tanta fortuna a trovare una così bella persona non solo professionalmente, ma soprattutto dal punto di vista umano. Grazie per avermi fatto arrivare ogni giorno a lavoro col sorriso, sapendo che ci saresti stato sempre, senza se e senza ma, ad aiutarmi.

Vorrei ora ringraziare tutte quelle persone che mi sono state vicine in questi anni di università, ma non solo. Anche quelle con le quali sono cresciuto e che hanno formato in me gli strumenti necessari per arrivarci e portarla a termine.

Ringrazio i miei compagni di una vita, gli amici del mare, coi quali sono cresciuto e che hanno accompagnato ogni estate, rendendole una dopo l'altra, speciali. Grazie Schürrle, Tommy, Lori, Nic, Andre, Fra, El, Michi, Cami, Gre, Giulia, Bob, Luca, Daddo, Max, Mathieu, Giacomo, Marco, Popoz, Giacomino, Mirko, Sve, Zampa per la vostra leggerezza. Sapere di avere un porto sicuro, per quanto temporalmente breve, mi ha permesso di dimenticare tutte le sfide e difficoltà affrontate a scuola ed all'università. Grazie per la vostra spensieratezza, autenticità e sincerità. Grazie per non avere pretese, grazie per regalarmi costantemente un sorriso. Spesso, nei momenti complicati, penso a voi come un'ancora di salvezza. Confesso che molte volte ho guardato al calendario non per trovare il giorno, ma per contare il numero dei mesi che mi separavano da luglio e agosto. Grazie.

Grazie a Sandy, amico d'infanzia, adolescenza ed età adulta. Il primo amico del mare, dai tubi al calcetto al Beccaccino. Grazie per tutti quanti gli anni in cui abbiamo inaugurato la stagione, per poi chiuderla, e per quanti ancora lo faremo, sempre. Abbiamo chiacchierato di tutto e di niente, senza mai intoppi ed esitazioni. Parlarci risulta naturale, anche se per tanti mesi si rimane a distanza. Grazie per i viaggi in auto, per il ditino sulle parole crociate, per la calma delle tue battute.

Un grazie speciale a Lollo, la nostra guida coinvolgente, il nostro collante. Colui che si nota quando è presente, grazie alla sua mente, e anche quando è assente, perché ci si sente un po' più soli, come persi. Seppur così diversi, la nostra somiglianza fa di te una guida che ho sempre seguito con ammirazione. Sei per me come un faro da seguire, è sufficiente ascoltarti un istante per avvertire le infinite qualità che possiedi, grazie.

Ringrazio gli amici di Ravenna. Le persone che ho incontrato all'asilo, chi alle superiori, chi grazie alla pallavolo. Grazie Fede, Vale, Ridolf. Ogni volta, rivederci per me ha significato non solo tornare a casa, ma essere a casa. Grazie a Edo per tutte le sciocchezze e stupidaggini continue. Anche senza parlare, la nostra intesa divertente coinvolge chi ci circonda portando allegria. Grazie a Conti, il mio compagno di banco, compagno di risate, prese in giro, scherzi. Non posso non essere felice quando ripenso alle superiori e questo è soprattutto merito tuo. Grazie Mancio per tutti i momenti trascorsi insieme, le feste di compleanno, la pallavolo fino alle uscite a Milano. Ogni capitolo della mia vita ha un bel ricordo insieme. Grazie Ges per la tua mente, sempre piena di curiosità, nuovi spunti di ragionamento e nuove idee. Grazie per le risate fatte e per quelle che faremo. Grazie per avere dato un taglio di profondità al mio pensiero. Grazie ad Ale Ghini, l'amico ritrovato, che avevo perso di vista ma di cui avevo bisogno e che mi ha insegnato come migliorarmi. Grazie a Filli, grazie per i tuoi sorrisi contagiosi, la tua allegria costante, la tua felicità concreta. Spesso avrei voluto chiacchierare con te in momenti difficili, perché sai tramutare i guai in inezie, la preoccupazione in spensieratezza. Grazie a Gio per le chiamate di ogni fine settimana. Grazie per le chiacchiere che alleggeriscono ogni carico. Grazie per la continua vicinanza.

Ringrazio i miei amici dell'università. Gli amici con i quali non abbiamo mai mollato, ma abbiamo resistito di semestre in semestre fino ad arrivare a questo capitolo finale. Gli amici grazie ai quali sono maturato di più, che mi hanno insegnato a pensare, riflettere, ragionare. Grazie a Edo, Andre, Fabri, Ali, Vale, Dani, Matte Poli, Matte Beltra, Marco, Giulia, Diego. Grazie per avermi fatto crescere fino a diventare pienamente adulto. Grazie per avermi insegnato cosa vuol dire avventura, grazie per le vacanze ed i viaggi insieme e per avermi insegnato cos'è sia la profondità che la spensieratezza. Grazie per rendere ogni incontro naturale, anche se dopo lungo tempo, come se il tempo trascorso lontani fosse solo apparente. Grazie a Rick e Casals per essere entrati nel gruppo come se ci foste

sempre stati e fossimo amici da sempre.

Grazie a Ferr per aver affinato la mia mente. Grazie per le lunghe chiacchierate notturne. Grazie per parlare del più e del meno sempre comprendendo e spiegandomi ogni dettaglio. Grazie per la tua invidiabile rapidità di ragionamento. Grazie per avermi insegnato come capire le persone. Grazie per quell'abbraccio. Grazie per essere un bro.

Grazie a Fede per avermi insegnato cosa significa essere veri amici. Grazie per l'empatia nel comprendere le mie esigenze senza chiedere nulla in cambio. Grazie per avermi fatto capire il valore della condivisione. Grazie per ascoltare sempre i miei sproloqui senza battere ciglio. Grazie per avermi ricordato che ci si può emozionare. Grazie per avermi fatto riscoprire il cinema. In una parola, grazie.

Ringrazio i miei amici dell'Erasmus del Poli e non solo. Grazie per avermi fatto sentire sempre a casa, seppure a 2000 km di distanza. Grazie per avermi mostrato la rilassatezza e come uscire dalla zona di comfort. Grazie Cristian, Christian, Francesca, Andrea, Matteo, Chiara, Giorgia, Mario, Lucia, Andrea, Veronica. Avete reso quei sei mesi così densi di esperienze, uscite, passeggiate e cene che non ho fatto in tempo ad accorgermi che fossero iniziati, quando oramai erano già finiti. Grazie a Sofia per i lunghi viaggi in bus e poi in metro, per il tempo trascorso insieme alla fermata del 50 e per le lunghe chiacchierate insieme. Grazie per aver reso Lappis una seconda casa.

Ringrazio i miei amici dell'Erasmus del KTH. Grazie ad Alberto, Alessandro, Anita, Luca per essere stati i miei compagni di corso per 6 mesi come se fossimo stati insieme dal primo anno. Grazie per avermi accolto tra di voi come se fossimo amici da sempre. Grazie a Luca per essere l'amico che tutti vogliono, ad Anita per fare sempre la scelta giusta, ad Alessandro per farsi sempre in quattro per gli altri, cercando di capire ed empatizzare. Grazie ad Alberto per le lunghe chiacchierate, discussioni e dibattiti. Grazie per avermi mostrato sempre un punto di vista diverso con estrema sincerità e chiarezza. E grazie anche a Samuel, un pezzo del KTH conosciuto in Francia. Grazie per la simpatia ed i bei momenti parigini passati in gruppo.

Grazie a Pulce, il mio cagnolino, per avermi rallegrato l'infanzia e l'adolescenza, per avermi fatto giocare e divertire sempre.

Ringrazio i miei familiari che mi hanno fornito i mattoni fondamentali per la costruzione delle fondamenta del mio percorso. Grazie per avermi insegnato l'educazione, l'impegno e la dedizione. Grazie alla nonna che non c'è più per avermi cresciuto con amore e con affetto. Grazie alla nonna che mi ha sempre accolto a casa per le festività, con tutta la famiglia riunita. Grazie ai nonni che non ci sono più e ho conosciuto poco, grazie per avermi regalato questa bellissima famiglia con la quale sono cresciuto. Grazie allo zio ed alla zia per avermi sempre supportato durante il mio percorso di vita e per avermi sempre consigliato saggiamente. Grazie al babbo per avermi insegnato che lavorando sodo si

possono ottenere risultati che da principio sembrano lontani. Grazie per avermi dato il buon esempio, senza il quale non sarei arrivato dove sono adesso, con i risultati ottenuti. Grazie alla mamma per il suo amore viscerale ed incondizionato che offre senza se e senza ma. Grazie per amarmi più di ogni cosa, annullandosi pur di soddisfare ogni mio piccolo capriccio, per offrirsi di risolvere ogni mio problema. Grazie perché so che potrò sempre contare su di te.

Un *grazie* sincero a tutti.